LinuxQMISDK-Lite

SLQS04.00.27

Generated by Doxygen 1.8.6

Thu Aug 13 2020 08:48:39

Contents

1 Welcome to the Sierra Wireless Linux QMI SDK API Reference Guide **1**

1.1 Important Notice . . . 1

1.2 Limitation of Liability . . . 1

1.3 Patents . . . 1

1.4 Copyright . . . 2

1.5 Trademarks . . . 2

1.6 Contact Information . . . 2

2 Module Index **3**

2.1 Modules . . . 3

3 Namespace Index **5**

3.1 Namespace List . . . 5

4 Data Structure Index **7**

4.1 Data Structures . . . 7

5 File Index **31**

5.1 File List . . . 31

6 Module Documentation **33**

6.1 QMI pack/unpack (lite-qmi) . . . 33

6.1.1 Detailed Description . . . 33

6.2 Streaming Download Protocol (lite-fw) . . . 34

6.2.1 Detailed Description . . . 34

6.3 QMUX layer (lite-qmux) . . . 35

6.3.1 Detailed Description . . . 35

6.4 MBIM layer (lite-mbim) . . . 36

6.4.1 Detailed Description . . . 36

7 Namespace Documentation **37**

7.1 Tables Namespace Reference . . . 37

7.1.1 Detailed Description . . . 37

8 Data Structure Documentation **39**

8.1 _litefw_FirmwareFileInfo Struct Reference . . . 39
8.1.1 Detailed Description . . . 39
8.1.2 Field Documentation . . . 40
8.1.2.1 carrierStr . . . 40
8.1.2.2 fullPath . . . 40
8.1.2.3 headerType . . . 40
8.1.2.4 imageMask . . . 40
8.1.2.5 imageType . . . 40
8.1.2.6 modelIdStr . . . 40
8.1.2.7 packageIdStr . . . 40
8.1.2.8 partNoStr . . . 40
8.1.2.9 priVersionStr . . . 40
8.1.2.10 releaseDate . . . 40
8.1.2.11 skuStr . . . 40
8.1.2.12 versionStr . . . 40
8.2 _litefw_FirmwareInfo_ Struct Reference . . . 40
8.2.1 Detailed Description . . . 41
8.2.2 Field Documentation . . . 41
8.2.2.1 szCarrier_str . . . 41
8.2.2.2 szCarrierPriversion_str . . . 41
8.2.2.3 szFwversion_str . . . 41
8.2.2.4 szModelid_str . . . 41
8.2.2.5 szPackageid_str . . . 41
8.2.2.6 szSku_str . . . 41
8.3 _litefw_FirmwarePartNo_ Struct Reference . . . 41
8.3.1 Detailed Description . . . 42
8.3.2 Field Documentation . . . 42
8.3.2.1 szPartno_str . . . 42
8.4 altSrcInfo_t Struct Reference . . . 42
8.4.1 Detailed Description . . . 42
8.4.2 Field Documentation . . . 43
8.4.2.1 coverage . . . 43
8.4.2.2 linkage . . . 43
8.4.2.3 source . . . 43
8.5 appStats Struct Reference . . . 43
8.5.1 Detailed Description . . . 43
8.5.2 Field Documentation . . . 45
8.5.2.1 aidLength . . . 45
8.5.2.2 aidVal . . . 45

8.5.2.3 appState . . . 45
8.5.2.4 appType . . . 45
8.5.2.5 persoFeature . . . 45
8.5.2.6 persoRetries . . . 45
8.5.2.7 persoState . . . 45
8.5.2.8 persoUnblockRetries . . . 46
8.5.2.9 pin1Retries . . . 46
8.5.2.10 pin1State . . . 46
8.5.2.11 pin2Retries . . . 46
8.5.2.12 pin2State . . . 46
8.5.2.13 puk1Retries . . . 46
8.5.2.14 puk2Retries . . . 46
8.5.2.15 univPin . . . 46
8.6 audio_RXAGCList Struct Reference . . . 46
8.6.1 Detailed Description . . . 46
8.6.2 Field Documentation . . . 46
8.6.2.1 pRXAIG . . . 47
8.6.2.2 pRXComprSlope . . . 47
8.6.2.3 pRXComprThres . . . 47
8.6.2.4 pRXExpSlope . . . 47
8.6.2.5 pRXExpThres . . . 47
8.6.2.6 pRXStaticGain . . . 47
8.7 audio_RXAVCList Struct Reference . . . 47
8.7.1 Detailed Description . . . 47
8.7.2 Field Documentation . . . 47
8.7.2.1 pAVRXAVCHeadroom . . . 47
8.7.2.2 pAVRXAVCSens . . . 47
8.8 audio_RXPCMIIRFltr Struct Reference . . . 47
8.8.1 Detailed Description . . . 47
8.8.2 Field Documentation . . . 49
8.8.2.1 pFlag . . . 49
8.8.2.2 pStage0Val . . . 49
8.8.2.3 pStage1Val . . . 49
8.8.2.4 pStage2Val . . . 49
8.8.2.5 pStage3Val . . . 49
8.8.2.6 pStage4Val . . . 49
8.8.2.7 pStageCnt . . . 49
8.9 audio_TXAGCList Struct Reference . . . 49
8.9.1 Detailed Description . . . 49
8.9.2 Field Documentation . . . 49

8.9.2.1 pTXAIG . . . 50
8.9.2.2 pTXComprSlope . . . 50
8.9.2.3 pTXComprThres . . . 50
8.9.2.4 pTXExpSlope . . . 50
8.9.2.5 pTXExpThres . . . 50
8.9.2.6 pTXStaticGain . . . 50
8.10 audio_TXPCMIIRFltr Struct Reference . . . 50
8.10.1 Detailed Description . . . 50
8.10.2 Field Documentation . . . 51
8.10.2.1 pFlag . . . 51
8.10.2.2 pStage0Val . . . 51
8.10.2.3 pStage1Val . . . 51
8.10.2.4 pStage2Val . . . 51
8.10.2.5 pStage3Val . . . 51
8.10.2.6 pStage4Val . . . 51
8.10.2.7 pStageCnt . . . 51
8.11 CarrierImage_t Struct Reference . . . 51
8.11.1 Detailed Description . . . 52
8.11.2 Field Documentation . . . 52
8.11.2.1 m_FwBuildId . . . 52
8.11.2.2 m_FwImageId . . . 52
8.11.2.3 m_nCarrierId . . . 52
8.11.2.4 m_nFolderId . . . 52
8.11.2.5 m_nStorage . . . 52
8.11.2.6 m_PriBuildId . . . 53
8.11.2.7 m_PriImageId . . . 53
8.12 cat_AlPhaIdentifierTlv Struct Reference . . . 53
8.12.1 Detailed Description . . . 53
8.12.2 Field Documentation . . . 53
8.12.2.1 AlphaID . . . 53
8.12.2.2 AlphaIDLength . . . 53
8.12.2.3 ReferenceID . . . 53
8.13 cat_commonEventTlv Struct Reference . . . 53
8.13.1 Detailed Description . . . 53
8.13.2 Field Documentation . . . 54
8.13.2.1 CatEvent . . . 54
8.13.2.2 EventID . . . 54
8.13.2.3 EventLength . . . 54
8.13.2.4 TlvPresent . . . 54
8.14 cat_currentCatEvent Union Reference . . . 54

8.14.1 Detailed Description . . . 54
8.14.2 Field Documentation . . . 54
8.14.2.1 CatAlphaIdtfr . . . 54
8.14.2.2 CatEndPS . . . 54
8.14.2.3 CatEventLst . . . 54
8.14.2.4 CatEvIDData . . . 54
8.14.2.5 CatRefresh . . . 54
8.15 cat_EndProactiveSessionTlv Struct Reference . . . 55
8.15.1 Detailed Description . . . 55
8.15.2 Field Documentation . . . 55
8.15.2.1 EndProactiveSession . . . 55
8.16 cat_EventIDDataTlv Struct Reference . . . 55
8.16.1 Detailed Description . . . 55
8.16.2 Field Documentation . . . 55
8.16.2.1 Data . . . 55
8.16.2.2 DataLength . . . 55
8.16.2.3 ReferenceID . . . 55
8.17 cat_EventListTlv Struct Reference . . . 55
8.17.1 Detailed Description . . . 56
8.17.2 Field Documentation . . . 56
8.17.2.1 SetupEventList . . . 56
8.18 cat_RefreshTlv Struct Reference . . . 56
8.18.1 Detailed Description . . . 56
8.18.2 Field Documentation . . . 56
8.18.2.1 RefreshMode . . . 56
8.18.2.2 RefreshStage . . . 56
8.19 cdmaSSInfo Struct Reference . . . 56
8.19.1 Detailed Description . . . 57
8.19.2 Field Documentation . . . 57
8.19.2.1 ecio . . . 57
8.19.2.2 rssi . . . 57
8.20 connectionStatus Struct Reference . . . 57
8.20.1 Detailed Description . . . 57
8.20.2 Field Documentation . . . 57
8.20.2.1 MDMCallDuration . . . 57
8.20.2.2 MDMConnStatus . . . 58
8.21 crashInfoParams Struct Reference . . . 58
8.21.1 Detailed Description . . . 58
8.21.2 Field Documentation . . . 58
8.21.2.1 crashInfo . . . 58

8.21.2.2 crashStatus . . . 58
8.22 crashInformation Struct Reference . . . 58
8.22.1 Detailed Description . . . 58
8.22.2 Field Documentation . . . 59
8.22.2.1 crashData . . . 59
8.22.2.2 crashId . . . 59
8.22.2.3 crashString . . . 59
8.22.2.4 crashStrlen . . . 59
8.22.2.5 gcdumpString . . . 59
8.22.2.6 gcdumpStrlen . . . 59
8.22.2.7 numCrashes . . . 59
8.23 CtlService Struct Reference . . . 59
8.23.1 Detailed Description . . . 59
8.23.2 Field Documentation . . . 59
8.23.2.1 service . . . 60
8.24 CtlTransactionHdr Struct Reference . . . 60
8.24.1 Field Documentation . . . 60
8.24.1.1 flags . . . 60
8.24.1.2 transactionId . . . 60
8.25 currNetworkInfo Struct Reference . . . 60
8.25.1 Detailed Description . . . 60
8.25.2 Field Documentation . . . 60
8.25.2.1 NetworkType . . . 60
8.25.2.2 RATMask . . . 60
8.25.2.3 SOMask . . . 60
8.26 dms_ActivationStatusTlv Struct Reference . . . 60
8.26.1 Detailed Description . . . 61
8.26.2 Field Documentation . . . 61
8.26.2.1 activationStatus . . . 61
8.26.2.2 TlvPresent . . . 61
8.27 dms_devCaps Struct Reference . . . 61
8.27.1 Detailed Description . . . 61
8.27.2 Field Documentation . . . 62
8.27.2.1 DataServiceCapability . . . 62
8.27.2.2 MaxRXChannelRate . . . 62
8.27.2.3 MaxTXChannelRate . . . 62
8.27.2.4 RadioIfaces . . . 62
8.27.2.5 RadioIfacesSize . . . 62
8.27.2.6 SimCapability . . . 62
8.28 dms_devCurSubsCaps Struct Reference . . . 62

8.28.1 Detailed Description . . . 62
8.28.2 Field Documentation . . . 63
8.28.2.1 CurSubsCapsLen . . . 63
8.28.2.2 SubsCapList . . . 63
8.29 dms_devMaxCfgListCaps Struct Reference . . . 63
8.29.1 Detailed Description . . . 63
8.29.2 Field Documentation . . . 64
8.29.2.1 CurIndex . . . 64
8.29.2.2 DevCfgListLen . . . 64
8.29.2.3 MaxActive . . . 64
8.29.2.4 MaxSubs . . . 64
8.29.2.5 SubsDevList . . . 64
8.30 dms_devMaxSubsCaps Struct Reference . . . 64
8.30.1 Detailed Description . . . 64
8.30.2 Field Documentation . . . 64
8.30.2.1 MaxSubsCapLen . . . 64
8.30.2.2 MaxSubsList . . . 64
8.31 dms_devMultiSimCaps Struct Reference . . . 64
8.31.1 Detailed Description . . . 65
8.31.2 Field Documentation . . . 65
8.31.2.1 MaxSubs . . . 65
8.31.2.2 SubsCfgList . . . 65
8.31.2.3 SubsCfgListLen . . . 65
8.32 dms_devMultiSimVoiceDataCaps Struct Reference . . . 65
8.32.1 Detailed Description . . . 65
8.32.2 Field Documentation . . . 65
8.32.2.1 MaxActive . . . 65
8.32.2.2 MaxSubs . . . 66
8.33 dms_devSubsCfgList Struct Reference . . . 66
8.33.1 Detailed Description . . . 66
8.33.2 Field Documentation . . . 66
8.33.2.1 MaxActive . . . 66
8.33.2.2 SubsList . . . 66
8.33.2.3 SubsListLen . . . 66
8.34 dms_devSubsFeatureModeCaps Struct Reference . . . 66
8.34.1 Detailed Description . . . 66
8.34.2 Field Documentation . . . 67
8.34.2.1 SubsFeatureLen . . . 67
8.34.2.2 SubsFeatureList . . . 67
8.35 dms_devSubsList Struct Reference . . . 67

8.35.1 Detailed Description . . . 67
8.35.2 Field Documentation . . . 67
8.35.2.1 SubsList . . . 67
8.35.2.2 SubsListLen . . . 67
8.36 dms_devSubsVoiceDataCaps Struct Reference . . . 68
8.36.1 Detailed Description . . . 68
8.36.2 Field Documentation . . . 68
8.36.2.1 SubsVoiceDataCapLen . . . 68
8.36.2.2 SubsVoiceDataList . . . 68
8.37 dms_devSubsVoiceDataList Struct Reference . . . 68
8.37.1 Detailed Description . . . 68
8.37.2 Field Documentation . . . 69
8.37.2.1 SimVoiceDataCap . . . 69
8.37.2.2 SubsVoiceDataCap . . . 69
8.38 dms_LteBandsSupport Struct Reference . . . 69
8.38.1 Detailed Description . . . 69
8.38.2 Field Documentation . . . 69
8.38.2.1 lteBands . . . 69
8.38.2.2 supportedLteBandLen . . . 69
8.38.2.3 TLVPresent . . . 69
8.39 dms_OperatingModeTlv Struct Reference . . . 69
8.39.1 Detailed Description . . . 69
8.39.2 Field Documentation . . . 70
8.39.2.1 operatingMode . . . 70
8.39.2.2 TlvPresent . . . 70
8.40 dms_PSMActiveTimerIndTlv Struct Reference . . . 70
8.40.1 Detailed Description . . . 70
8.40.2 Field Documentation . . . 70
8.40.2.1 ActiveTimerInd . . . 70
8.40.2.2 TlvPresent . . . 70
8.41 dms_PSMActiveTimerTlv Struct Reference . . . 70
8.41.1 Detailed Description . . . 71
8.41.2 Field Documentation . . . 71
8.41.2.1 activeTimer . . . 71
8.41.2.2 TlvPresent . . . 71
8.42 dms_PSMDurationDueToOOSTlv Struct Reference . . . 71
8.42.1 Detailed Description . . . 71
8.42.2 Field Documentation . . . 71
8.42.2.1 durationDueToOOS . . . 71
8.42.2.2 TlvPresent . . . 71

8.43 dms_PSMDurationThresholdTlv Struct Reference . . . 71
8.43.1 Detailed Description . . . 72
8.43.2 Field Documentation . . . 72
8.43.2.1 durationThreshold . . . 72
8.43.2.2 TlvPresent . . . 72
8.44 dms_PSMEarlyWakeupTimeTlv Struct Reference . . . 72
8.44.1 Detailed Description . . . 72
8.44.2 Field Documentation . . . 72
8.44.2.1 earlyWakeupTime . . . 72
8.44.2.2 TlvPresent . . . 72
8.45 dms_PSMEnableStateIndTlv Struct Reference . . . 73
8.45.1 Detailed Description . . . 73
8.45.2 Field Documentation . . . 73
8.45.2.1 EnableStateInd . . . 73
8.45.2.2 TlvPresent . . . 73
8.46 dms_PSMEnableStateTlv Struct Reference . . . 73
8.46.1 Detailed Description . . . 73
8.46.2 Field Documentation . . . 73
8.46.2.1 enableState . . . 74
8.46.2.2 TlvPresent . . . 74
8.47 dms_PSMPeriodicUpdateTimerIndTlv Struct Reference . . . 74
8.47.1 Detailed Description . . . 74
8.47.2 Field Documentation . . . 74
8.47.2.1 PeriodicUpdateTimerInd . . . 74
8.47.2.2 TlvPresent . . . 74
8.48 dms_PSMPeriodicUpdateTimerTlv Struct Reference . . . 74
8.48.1 Detailed Description . . . 74
8.48.2 Field Documentation . . . 75
8.48.2.1 periodicUpdateTimer . . . 75
8.48.2.2 TlvPresent . . . 75
8.49 dms_PSMRandomizationWindowTlv Struct Reference . . . 75
8.49.1 Detailed Description . . . 75
8.49.2 Field Documentation . . . 75
8.49.2.1 randomizationWindow . . . 75
8.49.2.2 TlvPresent . . . 75
8.50 dms_TemperatureTlv Struct Reference . . . 75
8.50.1 Detailed Description . . . 75
8.50.2 Field Documentation . . . 76
8.50.2.1 Temperature . . . 76
8.50.2.2 TempStat . . . 76

8.50.2.3 TlvPresent . . . 76
8.51 dms_UimAutoSwitchActSlotTlv Struct Reference . . . 76
8.51.1 Detailed Description . . . 76
8.51.2 Field Documentation . . . 76
8.51.2.1 TlvPresent . . . 76
8.51.2.2 uimAutoSwitchActSlot . . . 77
8.52 dms_UimStatusTlv Struct Reference . . . 77
8.52.1 Detailed Description . . . 77
8.52.2 Field Documentation . . . 77
8.52.2.1 event . . . 77
8.52.2.2 intf . . . 77
8.52.2.3 TlvPresent . . . 77
8.53 dms_VoltageTlv Struct Reference . . . 77
8.53.1 Detailed Description . . . 77
8.53.2 Field Documentation . . . 78
8.53.2.1 TlvPresent . . . 78
8.53.2.2 Voltage . . . 78
8.53.2.3 VoltStat . . . 78
8.54 DMScustSettingInfo Struct Reference . . . 78
8.54.1 Detailed Description . . . 78
8.54.2 Field Documentation . . . 79
8.54.2.1 cust_attr . . . 79
8.54.2.2 cust_id . . . 79
8.54.2.3 cust_value . . . 79
8.54.2.4 id_length . . . 79
8.54.2.5 value_length . . . 79
8.55 DMScustSettingList Struct Reference . . . 79
8.55.1 Detailed Description . . . 79
8.55.2 Field Documentation . . . 79
8.55.2.1 custSetting . . . 79
8.55.2.2 list_type . . . 79
8.55.2.3 num_instances . . . 79
8.56 DMSgetCustomFeatureV2 Struct Reference . . . 80
8.56.1 Detailed Description . . . 80
8.56.2 Field Documentation . . . 80
8.56.2.1 pCustSettingInfo . . . 80
8.56.2.2 pCustSettingList . . . 80
8.56.2.3 pGetCustomInput . . . 80
8.57 DMSgetCustomInput Struct Reference . . . 80
8.57.1 Detailed Description . . . 80

8.57.2 Field Documentation . . . 81
8.57.2.1 cust_id . . . 81
8.57.2.2 list_type . . . 81
8.58 DsdApnAvailableSystemInfo_t Struct Reference . . . 81
8.58.1 Detailed Description . . . 81
8.58.2 Field Documentation . . . 81
8.58.2.1 apn_avail_sys_len . . . 81
8.58.2.2 apn_name . . . 81
8.58.2.3 apn_name_len . . . 81
8.58.2.4 dsdGlobalAvailableSystem . . . 81
8.59 DsdApnCurRecAvailableSystemInfo_t Struct Reference . . . 81
8.59.1 Detailed Description . . . 82
8.59.2 Field Documentation . . . 82
8.59.2.1 dsdCurrentPreferdSystem . . . 82
8.59.2.2 dsdRecommendedPreferdSystem . . . 82
8.60 DsdApnCurRecPreferredSystemInfo_t Struct Reference . . . 82
8.60.1 Detailed Description . . . 82
8.60.2 Field Documentation . . . 83
8.60.2.1 apn_name . . . 83
8.60.2.2 apn_name_len . . . 83
8.60.2.3 dsdApnCurRecPreferredSystemInfo . . . 83
8.61 DsdGlobalUIInformation_t Struct Reference . . . 83
8.61.1 Detailed Description . . . 83
8.61.2 Field Documentation . . . 83
8.61.2.1 dsdAvailableSystem . . . 83
8.61.2.2 ui_mask . . . 84
8.62 DsdNullBearerReason_t Struct Reference . . . 84
8.62.1 Detailed Description . . . 84
8.62.2 Field Documentation . . . 84
8.62.2.1 null_bearer_reason . . . 84
8.63 DsdSystemInfo_t Struct Reference . . . 84
8.63.1 Detailed Description . . . 84
8.63.2 Field Documentation . . . 87
8.63.2.1 rat_value . . . 87
8.63.2.2 so_mask . . . 87
8.63.2.3 technology . . . 87
8.64 DsdUIApnAvailableSystemInfo_t Struct Reference . . . 87
8.64.1 Detailed Description . . . 87
8.64.2 Field Documentation . . . 88
8.64.2.1 apn_name . . . 88

8.64.2.2 apn_name_len . . . 88
8.64.2.3 apn_type_mask . . . 88
8.64.2.4 is_internet . . . 89
8.64.2.5 rat_value . . . 89
8.64.2.6 so_mask . . . 89
8.64.2.7 technology . . . 89
8.64.2.8 ui_mask . . . 89
8.65 dunchannelRate Struct Reference . . . 89
8.65.1 Detailed Description . . . 89
8.65.2 Field Documentation . . . 89
8.65.2.1 CurrChanRxRate . . . 89
8.65.2.2 CurrChanTxRate . . . 89
8.65.2.3 MaxChanRxRate . . . 89
8.65.2.4 MaxChanTxRate . . . 89
8.66 eriDataparams Struct Reference . . . 90
8.66.1 Detailed Description . . . 90
8.66.2 Field Documentation . . . 90
8.66.2.1 eriData . . . 90
8.66.2.2 eriDataLen . . . 90
8.67 eTWSPLMNInfoTlv Struct Reference . . . 90
8.67.1 Detailed Description . . . 90
8.67.2 Field Documentation . . . 90
8.67.2.1 ETWSPLMNInfo . . . 90
8.67.2.2 TlvPresent . . . 91
8.68 FMSImageElement Struct Reference . . . 91
8.68.1 Detailed Description . . . 91
8.68.2 Field Documentation . . . 91
8.68.2.1 buildId . . . 91
8.68.2.2 buildIdLength . . . 91
8.68.2.3 imageId . . . 91
8.68.2.4 imageType . . . 91
8.69 FMSImageIdElement Struct Reference . . . 91
8.69.1 Detailed Description . . . 91
8.69.2 Field Documentation . . . 92
8.69.2.1 buildID . . . 92
8.69.2.2 buildIDLength . . . 92
8.69.2.3 failureCount . . . 92
8.69.2.4 imageID . . . 92
8.69.2.5 storageIndex . . . 92
8.70 FMSImageIDEntries Struct Reference . . . 92

8.70.1 Detailed Description . . . 92
8.70.2 Field Documentation . . . 93
8.70.2.1 executingImage . . . 93
8.70.2.2 imageIDElement . . . 93
8.70.2.3 imageIDSize . . . 93
8.70.2.4 imageType . . . 93
8.70.2.5 maxImages . . . 93
8.71 FMSImageList Struct Reference . . . 93
8.71.1 Detailed Description . . . 93
8.71.2 Field Documentation . . . 93
8.71.2.1 imageIDEntries . . . 93
8.71.2.2 listSize . . . 93
8.72 FMSPrefImageList Struct Reference . . . 94
8.72.1 Detailed Description . . . 94
8.72.2 Field Documentation . . . 94
8.72.2.1 listEntries . . . 94
8.72.2.2 listSize . . . 94
8.73 GetClientIdReqMsg Struct Reference . . . 94
8.73.1 Field Documentation . . . 94
8.73.1.1 length . . . 94
8.73.1.2 messageId . . . 94
8.73.1.3 serviceType . . . 94
8.73.1.4 tlvLength . . . 94
8.73.1.5 tlvType . . . 94
8.74 GetClientIdRspMsg Struct Reference . . . 94
8.74.1 Field Documentation . . . 95
8.74.1.1 clientId . . . 95
8.74.1.2 error . . . 95
8.74.1.3 length . . . 95
8.74.1.4 messageId . . . 95
8.74.1.5 result . . . 95
8.74.1.6 serviceType . . . 95
8.74.1.7 tlv2Length . . . 95
8.74.1.8 tlv2Type . . . 95
8.74.1.9 tlvLength . . . 95
8.74.1.10 tlvType . . . 95
8.75 hdrSSInfo Struct Reference . . . 95
8.75.1 Detailed Description . . . 95
8.75.2 Field Documentation . . . 96
8.75.2.1 ecio . . . 96

8.75.2.2 io . . . 96
8.75.2.3 rssi . . . 96
8.75.2.4 sinr . . . 96
8.76 image_info_t Struct Reference . . . 96
8.76.1 Detailed Description . . . 96
8.76.2 Field Documentation . . . 97
8.76.2.1 buildID . . . 97
8.76.2.2 buildIDLen . . . 97
8.76.2.3 imageType . . . 97
8.76.2.4 uniqueID . . . 97
8.77 ims_AMRModeInfo Struct Reference . . . 97
8.77.1 Detailed Description . . . 97
8.77.2 Field Documentation . . . 98
8.77.2.1 amrMode . . . 98
8.77.2.2 TlvPresent . . . 98
8.78 ims_AMROctAlgnInfo Struct Reference . . . 98
8.78.1 Detailed Description . . . 98
8.78.2 Field Documentation . . . 98
8.78.2.1 amrOctAlgn . . . 98
8.78.2.2 TlvPresent . . . 98
8.79 ims_AMRWBModeInfo Struct Reference . . . 98
8.79.1 Detailed Description . . . 99
8.79.2 Field Documentation . . . 99
8.79.2.1 amrWBMode . . . 99
8.79.2.2 TlvPresent . . . 99
8.80 ims_AMRWBOctAlgnInfo Struct Reference . . . 99
8.80.1 Detailed Description . . . 99
8.80.2 Field Documentation . . . 99
8.80.2.1 amrWBOctAlgn . . . 99
8.80.2.2 TlvPresent . . . 99
8.81 ims_CSCFPortNameInfo Struct Reference . . . 99
8.81.1 Detailed Description . . . 100
8.81.2 Field Documentation . . . 100
8.81.2.1 cscfPortName . . . 100
8.81.2.2 TlvPresent . . . 100
8.82 ims_EnabAMRWBInfo Struct Reference . . . 100
8.82.1 Detailed Description . . . 100
8.82.2 Field Documentation . . . 100
8.82.2.1 amrWBEnable . . . 100
8.82.2.2 TlvPresent . . . 100

8.83 ims_EnabSCRAMRInfo Struct Reference . . . 100
8.83.1 Detailed Description . . . 101
8.83.2 Field Documentation . . . 101
8.83.2.1 scrAmrEnable . . . 101
8.83.2.2 TlvPresent . . . 101
8.84 ims_EnabSCRAMRWBInfo Struct Reference . . . 101
8.84.1 Detailed Description . . . 101
8.84.2 Field Documentation . . . 101
8.84.2.1 scrAmrWBEnable . . . 101
8.84.2.2 TlvPresent . . . 101
8.85 ims_IMSDomainInfo Struct Reference . . . 101
8.85.1 Detailed Description . . . 101
8.85.2 Field Documentation . . . 102
8.85.2.1 imsDomainName . . . 102
8.85.2.2 TlvPresent . . . 102
8.86 ims_IMSTestModeInfo Struct Reference . . . 102
8.86.1 Detailed Description . . . 102
8.86.2 Field Documentation . . . 102
8.86.2.1 imsTestMode . . . 102
8.86.2.2 TlvPresent . . . 102
8.87 ims_MinSessExpInfo Struct Reference . . . 102
8.87.1 Detailed Description . . . 102
8.87.2 Field Documentation . . . 103
8.87.2.1 minSessExp . . . 103
8.87.2.2 TlvPresent . . . 103
8.88 ims_PCSCFPortInfo Struct Reference . . . 103
8.88.1 Detailed Description . . . 103
8.88.2 Field Documentation . . . 103
8.88.2.1 priCSCFPort . . . 103
8.88.2.2 TlvPresent . . . 103
8.89 ims_PhCtxtURIInfo Struct Reference . . . 103
8.89.1 Detailed Description . . . 103
8.89.2 Field Documentation . . . 104
8.89.2.1 PhCtxtURI . . . 104
8.89.2.2 TlvPresent . . . 104
8.90 ims_RngBkTmrInfo Struct Reference . . . 104
8.90.1 Detailed Description . . . 104
8.90.2 Field Documentation . . . 104
8.90.2.1 RingBkTmr . . . 104
8.90.2.2 TlvPresent . . . 104

8.91 ims_RngTmrInfo Struct Reference . . . 104
8.91.1 Detailed Description . . . 104
8.91.2 Field Documentation . . . 105
8.91.2.1 RingTmr . . . 105
8.91.2.2 TlvPresent . . . 105
8.92 ims_RTPRTCPInactTmrDurInfo Struct Reference . . . 105
8.92.1 Detailed Description . . . 105
8.92.2 Field Documentation . . . 105
8.92.2.1 InactTmr . . . 105
8.92.2.2 TlvPresent . . . 105
8.93 ims_SessDurInfo Struct Reference . . . 105
8.93.1 Detailed Description . . . 105
8.93.2 Field Documentation . . . 106
8.93.2.1 sessExp . . . 106
8.93.2.2 TlvPresent . . . 106
8.94 ims_SigCompEnInfo Struct Reference . . . 106
8.94.1 Detailed Description . . . 106
8.94.2 Field Documentation . . . 106
8.94.2.1 SigCompEn . . . 106
8.94.2.2 TlvPresent . . . 106
8.95 ims_SIPPortInfo Struct Reference . . . 106
8.95.1 Detailed Description . . . 106
8.95.2 Field Documentation . . . 107
8.95.2.1 SIPLocalPort . . . 107
8.95.2.2 TlvPresent . . . 107
8.96 ims_SIPRegnTmrInfo Struct Reference . . . 107
8.96.1 Detailed Description . . . 107
8.96.2 Field Documentation . . . 107
8.96.2.1 TlvPresent . . . 107
8.96.2.2 tmrSIPRegn . . . 107
8.97 ims_SMSFmtInfo Struct Reference . . . 107
8.97.1 Detailed Description . . . 107
8.97.2 Field Documentation . . . 107
8.97.2.1 smsFormat . . . 108
8.97.2.2 TlvPresent . . . 108
8.98 ims_SMSoIPNwInfo Struct Reference . . . 108
8.98.1 Detailed Description . . . 108
8.98.2 Field Documentation . . . 108
8.98.2.1 smsoIPNW . . . 108
8.98.2.2 TlvPresent . . . 108

8.99 ims_SubscrTmrInfo Struct Reference . . . 108
8.99.1 Detailed Description . . . 108
8.99.2 Field Documentation . . . 108
8.99.2.1 subscrTmr . . . 108
8.99.2.2 TlvPresent . . . 108
8.100 ims_TmrT1Info Struct Reference . . . 109
8.100.1 Detailed Description . . . 109
8.100.2 Field Documentation . . . 109
8.100.2.1 TlvPresent . . . 109
8.100.2.2 tmrT1 . . . 109
8.101 ims_TmrT2Info Struct Reference . . . 109
8.101.1 Detailed Description . . . 109
8.101.2 Field Documentation . . . 109
8.101.2.1 TlvPresent . . . 109
8.101.2.2 tmrT2 . . . 109
8.102 ims_TmrTfInfo Struct Reference . . . 109
8.102.1 Detailed Description . . . 110
8.102.2 Field Documentation . . . 110
8.102.2.1 TlvPresent . . . 110
8.102.2.2 tmrTf . . . 110
8.103 imsa_IMSFailErrCodeTlv Struct Reference . . . 110
8.103.1 Detailed Description . . . 110
8.103.2 Field Documentation . . . 110
8.103.2.1 ImsFailErrCode . . . 110
8.103.2.2 TlvPresent . . . 110
8.104 imsa_IMSRegStatusErrorCodeInfo Struct Reference . . . 110
8.104.1 Detailed Description . . . 111
8.104.2 Field Documentation . . . 111
8.104.2.1 ErrorCode . . . 111
8.104.2.2 TlvPresent . . . 111
8.105 imsa_IMSRegStatusInfo Struct Reference . . . 111
8.105.1 Detailed Description . . . 111
8.105.2 Field Documentation . . . 111
8.105.2.1 ImsRegistered . . . 112
8.105.2.2 TlvPresent . . . 112
8.106 imsa_NewIMSRegStatusInfo Struct Reference . . . 112
8.106.1 Detailed Description . . . 112
8.106.2 Field Documentation . . . 112
8.106.2.1 ImsRegStatus . . . 112
8.106.2.2 TlvPresent . . . 112

8.107imsa_RatHandoverStatusInfo Struct Reference . . . 112
8.107.1 Detailed Description . . . 112
8.107.2 Field Documentation . . . 113
8.107.2.1 ErrorCodeData . . . 113
8.107.2.2 ErrorCodeLen . . . 113
8.107.2.3 RatHandoverStatus . . . 113
8.107.2.4 SourceRAT . . . 113
8.107.2.5 TargetRAT . . . 113
8.107.2.6 TlvPresent . . . 113
8.108imsa_SmsRatInfo Struct Reference . . . 113
8.108.1 Detailed Description . . . 113
8.108.2 Field Documentation . . . 114
8.108.2.1 SmsRatVal . . . 114
8.108.2.2 TlvPresent . . . 114
8.109imsa_SmsSvcStatusInfo Struct Reference . . . 114
8.109.1 Detailed Description . . . 114
8.109.2 Field Documentation . . . 114
8.109.2.1 SmsSvcStatus . . . 114
8.109.2.2 TlvPresent . . . 114
8.110imsa_UtRatInfo Struct Reference . . . 114
8.110.1 Detailed Description . . . 114
8.110.2 Field Documentation . . . 115
8.110.2.1 TlvPresent . . . 115
8.110.2.2 UtRatVal . . . 115
8.111imsa_UtSvcStatusInfo Struct Reference . . . 115
8.111.1 Detailed Description . . . 115
8.111.2 Field Documentation . . . 115
8.111.2.1 TlvPresent . . . 115
8.111.2.2 UtSvcStatus . . . 115
8.112imsa_VoipRatInfo Struct Reference . . . 115
8.112.1 Detailed Description . . . 116
8.112.2 Field Documentation . . . 116
8.112.2.1 TlvPresent . . . 116
8.112.2.2 VoipRatVal . . . 116
8.113imsa_VoipSvcStatusInfo Struct Reference . . . 116
8.113.1 Detailed Description . . . 116
8.113.2 Field Documentation . . . 116
8.113.2.1 TlvPresent . . . 116
8.113.2.2 VoipSvcStatus . . . 116
8.114imsa_VtRatInfo Struct Reference . . . 116

8.114.1 Detailed Description . . . 117
8.114.2 Field Documentation . . . 117
8.114.2.1 TlvPresent . . . 117
8.114.2.2 VtRatVal . . . 117
8.115imsa_VtSvcStatusInfo Struct Reference . . . 117
8.115.1 Detailed Description . . . 117
8.115.2 Field Documentation . . . 117
8.115.2.1 TlvPresent . . . 117
8.115.2.2 VtSvcStatus . . . 117
8.116ipv6AddressInfo Struct Reference . . . 117
8.116.1 Detailed Description . . . 118
8.116.2 Field Documentation . . . 118
8.116.2.1 IPAddressV6 . . . 118
8.116.2.2 IPV6PrefixLen . . . 118
8.117LibPackGPRSRequestedQoS Struct Reference . . . 118
8.117.1 Detailed Description . . . 118
8.117.2 Field Documentation . . . 119
8.117.2.1 delayClass . . . 119
8.117.2.2 meanThroughputClass . . . 119
8.117.2.3 peakThroughputClass . . . 119
8.117.2.4 precedenceClass . . . 119
8.117.2.5 reliabilityClass . . . 119
8.118LibPackPCOIDList Struct Reference . . . 119
8.118.1 Detailed Description . . . 119
8.118.2 Field Documentation . . . 119
8.118.2.1 PcoList . . . 119
8.119LibPackPDNThrottleTimer Struct Reference . . . 119
8.119.1 Detailed Description . . . 119
8.119.2 Field Documentation . . . 119
8.119.2.1 ThrottleTimer . . . 119
8.120LibpackProfile3GPP Struct Reference . . . 119
8.120.1 Detailed Description . . . 120
8.120.2 Field Documentation . . . 125
8.120.2.1 pAddrAllocPref . . . 125
8.120.2.2 pAPNClass . . . 125
8.120.2.3 pAPNDisabledFlag . . . 125
8.120.2.4 pAPNName . . . 125
8.120.2.5 pAPNnameSize . . . 125
8.120.2.6 pAuthenticationPref . . . 125
8.120.2.7 pGPRSMinimumQoS . . . 125

8.120.2.8 pGPRSRequestedQos . . . 126
8.120.2.9 pImCnFlag . . . 126
8.120.2.10 pIPv4AddrPref . . . 126
8.120.2.11 pIPv6AddPref . . . 126
8.120.2.12 pPassword . . . 126
8.120.2.13 pPasswordSize . . . 126
8.120.2.14 pPcscfAddrUsingDhcp . . . 126
8.120.2.15 pPcscfAddrUsingPCO . . . 126
8.120.2.16 pPDNInactivTimeout . . . 126
8.120.2.17 pPdpAccessConFlag . . . 126
8.120.2.18 pPdpContext . . . 126
8.120.2.19 pPdpDataCompType . . . 126
8.120.2.20 pPdpHdrCompType . . . 126
8.120.2.21 pPDPtype . . . 126
8.120.2.22 pPriDNSIPv4AddPref . . . 126
8.120.2.23 pPriDNSIPv6addpref . . . 126
8.120.2.24 pPrimaryID . . . 126
8.120.2.25 pProfilename . . . 126
8.120.2.26 pProfilenameSize . . . 126
8.120.2.27 pQosClassID . . . 126
8.120.2.28 pSecDNSIPv4AddPref . . . 126
8.120.2.29 pSecDNSIPv6addpref . . . 126
8.120.2.30 pSecondaryFlag . . . 126
8.120.2.31 pSupportEmergencyCalls . . . 126
8.120.2.32 pTFTID1Params . . . 126
8.120.2.33 pTFTID2Params . . . 126
8.120.2.34 pUMTSMinQoS . . . 126
8.120.2.35 pUMTSMinQosSigInd . . . 126
8.120.2.36 pUMTSReqQoS . . . 127
8.120.2.37 pUMTSReqQoSSigInd . . . 127
8.120.2.38 pUsername . . . 127
8.120.2.39 pUsernameSize . . . 127
8.121 LibpackProfile3GPP2 Struct Reference . . . 127
8.121.1 Detailed Description . . . 127
8.121.2 Field Documentation . . . 131
8.121.2.1 pAllowLinger . . . 131
8.121.2.2 pAPNClass3GPP2 . . . 132
8.121.2.3 pAPNEnabled3GPP2 . . . 132
8.121.2.4 pApnString . . . 132
8.121.2.5 pApnStringSize . . . 132

8.121.2.6 pAppPriority . . . 132
8.121.2.7 pAppType . . . 132
8.121.2.8 pAuthPassword . . . 132
8.121.2.9 pAuthPasswordSize . . . 132
8.121.2.10 pAuthProtocol . . . 132
8.121.2.11 pAuthRetryCount . . . 132
8.121.2.12 pAuthTimeout . . . 132
8.121.2.13 pDataMode . . . 132
8.121.2.14 pDataRate . . . 132
8.121.2.15 pIpcpAckTimeout . . . 132
8.121.2.16 pIpcpCreqRetryCount . . . 132
8.121.2.17 pIsPcscfAddressNedded . . . 132
8.121.2.18 pLcpAckTimeout . . . 132
8.121.2.19 pLcpCreqRetryCount . . . 132
8.121.2.20 pNegoDnsSrvrPref . . . 132
8.121.2.21 pPDNInactivTimeout3GPP2 . . . 132
8.121.2.22 pPdnType . . . 132
8.121.2.23 pPppSessCloseTimer1x . . . 132
8.121.2.24 pPppSessCloseTimerDO . . . 132
8.121.2.25 pPrimaryV4DnsAddress . . . 132
8.121.2.26 pPriV6DnsAddress . . . 132
8.121.2.27 pRATType . . . 132
8.121.2.28 pSecondaryV4DnsAddress . . . 132
8.121.2.29 pSecV6DnsAddress . . . 132
8.121.2.30 pUserId . . . 133
8.121.2.31 pUserIdSize . . . 133
8.122 LibpackProfile3GPPV2 Struct Reference . . . 133
8.122.1 Detailed Description . . . 134
8.122.2 Field Documentation . . . 141
8.122.2.1 pAddrAllocPref . . . 141
8.122.2.2 pAPNBearer . . . 141
8.122.2.3 pAPNClass . . . 141
8.122.2.4 pAPNDisabledFlag . . . 141
8.122.2.5 pAPNName . . . 141
8.122.2.6 pAPNnameSize . . . 141
8.122.2.7 pAppUserData . . . 141
8.122.2.8 pAuthenticationPref . . . 141
8.122.2.9 pClatFlag . . . 141
8.122.2.10 pDnsWithDHCPFlag . . . 141
8.122.2.11 pGPRSMinimumQoS . . . 141

8.122.2.12pGPRSRequestedQos . . . 142
8.122.2.13pImCnFlag . . . 142
8.122.2.14pIPv4AddrPref . . . 142
8.122.2.15pIPv6AddPref . . . 142
8.122.2.16pIPV6DelegFlag . . . 142
8.122.2.17pIWLANtoLTEHandoverFlag . . . 142
8.122.2.18pLteRoamPDPType . . . 142
8.122.2.19pLTEtoIWLANHandoverFlag . . . 142
8.122.2.20pMaxPDN . . . 142
8.122.2.21pMaxPDNTimer . . . 142
8.122.2.22pMcc . . . 142
8.122.2.23pMnc . . . 142
8.122.2.24pMsisdnFlag . . . 142
8.122.2.25pOperatorPCOID . . . 142
8.122.2.26pOverridePDPType . . . 142
8.122.2.27pPassword . . . 142
8.122.2.28pPasswordSize . . . 142
8.122.2.29pPCOIDList . . . 142
8.122.2.30pPcscfAddrUsingDhcp . . . 142
8.122.2.31pPcscfAddrUsingPCO . . . 142
8.122.2.32pPDNDisconnectWaitTimer . . . 142
8.122.2.33pPDNInactivTimeout . . . 142
8.122.2.34pPDNThrottleTimer . . . 142
8.122.2.35pPDNWaitTimer . . . 142
8.122.2.36pPdpAccessConFlag . . . 142
8.122.2.37pPdpContext . . . 142
8.122.2.38pPdpDataCompType . . . 142
8.122.2.39pPdpHdrCompType . . . 142
8.122.2.40pPDPtype . . . 143
8.122.2.41pPersistFlag . . . 143
8.122.2.42pPriDNSIPv4AddPref . . . 143
8.122.2.43pPriDNSIPv6addpref . . . 143
8.122.2.44pPrimaryID . . . 143
8.122.2.45pProfilename . . . 143
8.122.2.46pProfilenameSize . . . 143
8.122.2.47pQosClassID . . . 143
8.122.2.48pRoamDisallowFlag . . . 143
8.122.2.49pSecDNSIPv4AddPref . . . 143
8.122.2.50pSecDNSIPv6addpref . . . 143
8.122.2.51pSecondaryFlag . . . 143

8.122.2.52 pSupportEmergencyCalls . . . 143
8.122.2.53 pTFTID1Params . . . 143
8.122.2.54 pTFTID2Params . . . 143
8.122.2.55 pUMTSMinQoS . . . 143
8.122.2.56 pUMTSMinQosSigInd . . . 143
8.122.2.57 pUMTSReqQoS . . . 143
8.122.2.58 pUMTSReqQoSSigInd . . . 143
8.122.2.59 pUmtsRoamPDPType . . . 143
8.122.2.60 pUsername . . . 143
8.122.2.61 pUsernameSize . . . 143
8.123 LibPackprofile_3GPP Struct Reference . . . 143
8.123.1 Detailed Description . . . 144
8.123.2 Field Documentation . . . 148
8.123.2.1 pAddrAllocPref . . . 148
8.123.2.2 pAPNClass . . . 148
8.123.2.3 pAPNDisabledFlag . . . 148
8.123.2.4 pAPNName . . . 148
8.123.2.5 pAPNnameSize . . . 148
8.123.2.6 pAuthenticationPref . . . 148
8.123.2.7 pGPRSMinimumQoS . . . 148
8.123.2.8 pGPRSRequestedQos . . . 148
8.123.2.9 pImCnFlag . . . 148
8.123.2.10 pIPv4AddrPref . . . 148
8.123.2.11 pIPv6AddPref . . . 148
8.123.2.12 pPassword . . . 148
8.123.2.13 pPasswordSize . . . 148
8.123.2.14 pPcscfAddrUsingDhcp . . . 148
8.123.2.15 pPcscfAddrUsingPCO . . . 148
8.123.2.16 pPDNInactivTimeout . . . 148
8.123.2.17 pPdpAccessConFlag . . . 148
8.123.2.18 pPdpContext . . . 148
8.123.2.19 pPdpDataCompType . . . 148
8.123.2.20 pPdpHdrCompType . . . 149
8.123.2.21 pPDPtype . . . 149
8.123.2.22 pPriDNSIPv4AddPref . . . 149
8.123.2.23 pPriDNSIPv6addpref . . . 149
8.123.2.24 pPrimaryID . . . 149
8.123.2.25 pProfilename . . . 149
8.123.2.26 pProfilenameSize . . . 149
8.123.2.27 pQosClassID . . . 149

8.123.2.28 pSecDNSIPv4AddPref . . . 149
8.123.2.29 pSecDNSIPv6addpref . . . 149
8.123.2.30 pSecondaryFlag . . . 149
8.123.2.31 pSupportEmergencyCalls . . . 149
8.123.2.32 pTFTID1Params . . . 149
8.123.2.33 pTFTID2Params . . . 149
8.123.2.34 pUMTSMinQoS . . . 149
8.123.2.35 pUMTSMinQosSigInd . . . 149
8.123.2.36 pUMTSReqQoS . . . 149
8.123.2.37 pUMTSReqQoSSigInd . . . 149
8.123.2.38 pUsername . . . 149
8.123.2.39 pUsernameSize . . . 149
8.124 LibPackprofile_3GPP2 Struct Reference . . . 149
8.124.1 Detailed Description . . . 150
8.124.2 Field Documentation . . . 153
8.124.2.1 pAllowLinger . . . 153
8.124.2.2 pAPNClass3GPP2 . . . 153
8.124.2.3 pAPNEnabled3GPP2 . . . 153
8.124.2.4 pApnString . . . 153
8.124.2.5 pApnStringSize . . . 154
8.124.2.6 pAppPriority . . . 154
8.124.2.7 pAppType . . . 154
8.124.2.8 pAuthPassword . . . 154
8.124.2.9 pAuthPassword_tSize . . . 154
8.124.2.10 pAuthProtocol . . . 154
8.124.2.11 pAuthRetryCount . . . 154
8.124.2.12 pAuthTimeout . . . 154
8.124.2.13 pDataMode . . . 154
8.124.2.14 pDataRate . . . 154
8.124.2.15 pIpcpAckTimeout . . . 154
8.124.2.16 pIpcpCreqRetryCount . . . 154
8.124.2.17 pIsPcscfAddressNedded . . . 154
8.124.2.18 pLcpAckTimeout . . . 154
8.124.2.19 pLcpCreqRetryCount . . . 154
8.124.2.20 pNegoDnsSrvrPref . . . 154
8.124.2.21 pPDNInactivTimeout3GPP2 . . . 154
8.124.2.22 pPdnType . . . 154
8.124.2.23 pPppSessCloseTimer1x . . . 154
8.124.2.24 pPppSessCloseTimerDO . . . 154
8.124.2.25 pPrimaryV4DnsAddress . . . 154

8.124.2.26 pPriV6DnsAddress . . . 154
8.124.2.27 pRATType . . . 154
8.124.2.28 pSecondaryV4DnsAddress . . . 154
8.124.2.29 pSecV6DnsAddress . . . 154
8.124.2.30 pUserId . . . 154
8.124.2.31 pUserIdSize . . . 154
8.125 LibPackProfileMnc Struct Reference . . . 154
8.125.1 Detailed Description . . . 155
8.125.2 Field Documentation . . . 155
8.125.2.1 MNC . . . 155
8.125.2.2 PCSFlag . . . 155
8.126 LibPackQosClassID Struct Reference . . . 155
8.126.1 Detailed Description . . . 155
8.126.2 Field Documentation . . . 156
8.126.2.1 gDlBitRate . . . 156
8.126.2.2 gUlBitRate . . . 156
8.126.2.3 maxDlBitRate . . . 156
8.126.2.4 maxUlBitRate . . . 156
8.126.2.5 QCI . . . 156
8.127 LibPackTFTIDParams Struct Reference . . . 156
8.127.1 Detailed Description . . . 156
8.127.2 Field Documentation . . . 157
8.127.2.1 destPortRangeEnd . . . 157
8.127.2.2 destPortRangeStart . . . 157
8.127.2.3 eValid . . . 157
8.127.2.4 filterId . . . 157
8.127.2.5 flowLabel . . . 157
8.127.2.6 IPSECSPI . . . 158
8.127.2.7 ipVersion . . . 158
8.127.2.8 nextHeader . . . 158
8.127.2.9 pSourceIP . . . 158
8.127.2.10 sourceIPMask . . . 158
8.127.2.11 srcPortRangeEnd . . . 158
8.127.2.12 srcPortRangeStart . . . 158
8.127.2.13 tosMask . . . 158
8.128 LibPackUMTSQoS Struct Reference . . . 158
8.128.1 Detailed Description . . . 158
8.128.2 Field Documentation . . . 160
8.128.2.1 deliveryErrSDU . . . 160
8.128.2.2 grntDownlinkBitrate . . . 160

8.128.2.3 grntUplinkBitrate . . . 160
8.128.2.4 maxDownlinkBitrate . . . 160
8.128.2.5 maxSDUSize . . . 160
8.128.2.6 maxUplinkBitrate . . . 160
8.128.2.7 qosDeliveryOrder . . . 160
8.128.2.8 resBerRatio . . . 160
8.128.2.9 sduErrorRatio . . . 160
8.128.2.10 trafficClass . . . 160
8.128.2.11 trafficPriority . . . 160
8.128.2.12 transferDelay . . . 160
8.129 LibPackUMTSReqQoSSigInd Struct Reference . . . 161
8.129.1 Detailed Description . . . 161
8.129.2 Field Documentation . . . 161
8.129.2.1 SigInd . . . 161
8.129.2.2 UMTSReqQoS . . . 161
8.130 loc_accelAcceptReady Struct Reference . . . 161
8.130.1 Detailed Description . . . 161
8.130.2 Field Documentation . . . 162
8.130.2.1 batchPerSec . . . 162
8.130.2.2 injectEnable . . . 162
8.130.2.3 samplesPerBatch . . . 162
8.131 loc_accelTempAcceptReady Struct Reference . . . 162
8.131.1 Detailed Description . . . 162
8.131.2 Field Documentation . . . 163
8.131.2.1 batchPerSec . . . 163
8.131.2.2 injectEnable . . . 163
8.131.2.3 samplesPerBatch . . . 163
8.132 loc_AppProviderInfoTlv Struct Reference . . . 163
8.132.1 Detailed Description . . . 163
8.132.2 Field Documentation . . . 164
8.132.2.1 name . . . 164
8.132.2.2 nameLen . . . 164
8.132.2.3 provider . . . 164
8.132.2.4 providerLen . . . 164
8.132.2.5 TlvPresent . . . 164
8.132.2.6 version . . . 164
8.132.2.7 versionLen . . . 164
8.132.2.8 verValid . . . 164
8.133 loc_BdsSV Struct Reference . . . 164
8.133.1 Detailed Description . . . 164

8.133.2 Field Documentation . . . 164
8.133.2.1 id . . . 164
8.133.2.2 mask . . . 164
8.134loc_BdsSVInfo Struct Reference . . . 165
8.134.1 Detailed Description . . . 165
8.134.2 Field Documentation . . . 165
8.134.2.1 len . . . 165
8.134.2.2 pSV . . . 165
8.135loc_CellDb Struct Reference . . . 165
8.135.1 Detailed Description . . . 165
8.135.2 Field Documentation . . . 166
8.135.2.1 mask . . . 166
8.136loc_ClkInfo Struct Reference . . . 166
8.136.1 Detailed Description . . . 166
8.136.2 Field Documentation . . . 167
8.136.2.1 mask . . . 167
8.137loc_FixCriteriaStatusTlv Struct Reference . . . 167
8.137.1 Detailed Description . . . 168
8.137.2 Field Documentation . . . 168
8.137.2.1 status . . . 168
8.137.2.2 TlvPresent . . . 168
8.138loc_GnssData Struct Reference . . . 168
8.138.1 Detailed Description . . . 168
8.138.2 Field Documentation . . . 169
8.138.2.1 mask . . . 169
8.139loc_gpsTime Struct Reference . . . 169
8.139.1 Detailed Description . . . 170
8.139.2 Field Documentation . . . 170
8.139.2.1 gpsTimeOfWeekMs . . . 170
8.139.2.2 gpsWeek . . . 170
8.140loc_gyroAcceptReady Struct Reference . . . 170
8.140.1 Detailed Description . . . 170
8.140.2 Field Documentation . . . 171
8.140.2.1 batchPerSec . . . 171
8.140.2.2 injectEnable . . . 171
8.140.2.3 samplesPerBatch . . . 171
8.141loc_gyroTempAcceptReady Struct Reference . . . 171
8.141.1 Detailed Description . . . 171
8.141.2 Field Documentation . . . 171
8.141.2.1 batchPerSec . . . 171

8.141.2.2 injectEnable . . . 171
8.141.2.3 samplesPerBatch . . . 172
8.142loc_HorAccuracyLvlTlv Struct Reference . . . 172
8.142.1 Detailed Description . . . 172
8.142.2 Field Documentation . . . 172
8.142.2.1 accuracy . . . 172
8.142.2.2 TlvPresent . . . 172
8.143loc_IntermediateRptStateTlv Struct Reference . . . 172
8.143.1 Detailed Description . . . 172
8.143.2 Field Documentation . . . 173
8.143.2.1 intermediate . . . 173
8.143.2.2 TlvPresent . . . 173
8.144loc_IPv4Config Struct Reference . . . 173
8.144.1 Detailed Description . . . 173
8.144.2 Field Documentation . . . 173
8.144.2.1 IPv4Addr . . . 173
8.144.2.2 IPv4Port . . . 173
8.145loc_IPv4Info Struct Reference . . . 173
8.145.1 Detailed Description . . . 173
8.145.2 Field Documentation . . . 174
8.145.2.1 address . . . 174
8.145.2.2 port . . . 174
8.145.2.3 TlvPresent . . . 174
8.146loc_IPv6Config Struct Reference . . . 174
8.146.1 Detailed Description . . . 174
8.146.2 Field Documentation . . . 174
8.146.2.1 IPv6Addr . . . 174
8.146.2.2 IPv6Port . . . 174
8.147loc_IPv6Info Struct Reference . . . 174
8.147.1 Detailed Description . . . 175
8.147.2 Field Documentation . . . 175
8.147.2.1 address . . . 175
8.147.2.2 port . . . 175
8.147.2.3 TlvPresent . . . 175
8.148loc_LocApplicationInfo Struct Reference . . . 175
8.148.1 Detailed Description . . . 175
8.148.2 Field Documentation . . . 176
8.148.2.1 appNameLength . . . 176
8.148.2.2 appProviderLength . . . 176
8.148.2.3 appVersionLength . . . 176

8.148.2.4 appVersionValid . . . 176
8.148.2.5 pAppName . . . 176
8.148.2.6 pAppProvider . . . 176
8.148.2.7 pAppVersion . . . 176
8.149loc_MinIntervalTlv Struct Reference . . . 176
8.149.1 Detailed Description . . . 176
8.149.2 Field Documentation . . . 177
8.149.2.1 interval . . . 177
8.149.2.2 TlvPresent . . . 177
8.150loc_precisionDilution Struct Reference . . . 177
8.150.1 Detailed Description . . . 177
8.150.2 Field Documentation . . . 177
8.150.2.1 HDOP . . . 177
8.150.2.2 PDOP . . . 177
8.150.2.3 VDOP . . . 177
8.151loc_satelliteInfo Struct Reference . . . 177
8.151.1 Detailed Description . . . 178
8.151.2 Field Documentation . . . 179
8.151.2.1 azimuth . . . 179
8.151.2.2 elevation . . . 179
8.151.2.3 gnssSvId . . . 179
8.151.2.4 healthStatus . . . 179
8.151.2.5 snr . . . 179
8.151.2.6 svInfoMask . . . 180
8.151.2.7 svListLen . . . 180
8.151.2.8 svStatus . . . 180
8.151.2.9 system . . . 180
8.151.2.10validMask . . . 180
8.152loc_sensorDataUsage Struct Reference . . . 180
8.152.1 Detailed Description . . . 180
8.152.2 Field Documentation . . . 180
8.152.2.1 aidingIndicatorMask . . . 180
8.152.2.2 usageMask . . . 180
8.153loc_SV Struct Reference . . . 180
8.153.1 Detailed Description . . . 181
8.153.2 Field Documentation . . . 181
8.153.2.1 id . . . 181
8.153.2.2 mask . . . 181
8.153.2.3 system . . . 181
8.154loc_SVInfo Struct Reference . . . 181

8.154.1 Detailed Description . . . 181
8.154.2 Field Documentation . . . 182
8.154.2.1 len . . . 182
8.154.2.2 pSV . . . 182
8.155loc_svUsedforFix Struct Reference . . . 182
8.155.1 Detailed Description . . . 182
8.155.2 Field Documentation . . . 182
8.155.2.1 gnssSvUsedList . . . 182
8.155.2.2 gnssSvUsedList_len . . . 182
8.156loc_urlAddr Struct Reference . . . 182
8.156.1 Detailed Description . . . 183
8.156.2 Field Documentation . . . 183
8.156.2.1 address . . . 183
8.156.2.2 TlvPresent . . . 183
8.157loc_URLAddrInfo Struct Reference . . . 183
8.157.1 Detailed Description . . . 183
8.157.2 Field Documentation . . . 183
8.157.2.1 urlAddr . . . 183
8.158lteSSInfo Struct Reference . . . 183
8.158.1 Detailed Description . . . 184
8.158.2 Field Documentation . . . 184
8.158.2.1 rsrp . . . 184
8.158.2.2 rsrq . . . 184
8.158.2.3 rssi . . . 184
8.158.2.4 snr . . . 184
8.159MBIM_CLOSE_DONE_MSG Struct Reference . . . 184
8.159.1 Field Documentation . . . 184
8.159.1.1 MessageHeader . . . 184
8.159.1.2 Status . . . 185
8.160MBIM_CLOSE_MSG Struct Reference . . . 185
8.160.1 Field Documentation . . . 185
8.160.1.1 MessageHeader . . . 185
8.161MBIM_COMMAND_DONE_MSG Struct Reference . . . 185
8.161.1 Field Documentation . . . 185
8.161.1.1 CID . . . 185
8.161.1.2 DeviceServiceId . . . 185
8.161.1.3 FragmentHeader . . . 185
8.161.1.4 InformationBufferLength . . . 185
8.161.1.5 MessageHeader . . . 185
8.161.1.6 Status . . . 185

8.162MBIM_COMMAND_MSG Struct Reference . . . 185
8.162.1 Field Documentation . . . 186
8.162.1.1 CID . . . 186
8.162.1.2 CommandType . . . 186
8.162.1.3 DeviceServiceId . . . 186
8.162.1.4 FragmentHeader . . . 186
8.162.1.5 InformationBufferLength . . . 186
8.162.1.6 MessageHeader . . . 186
8.163MBIM_FRAGMENT_HEADER Struct Reference . . . 186
8.163.1 Field Documentation . . . 186
8.163.1.1 CurrentFragment . . . 186
8.163.1.2 TotalFragments . . . 186
8.164MBIM_FRAGMENT_MSG Struct Reference . . . 186
8.164.1 Field Documentation . . . 186
8.164.1.1 FragmentHeader . . . 186
8.164.1.2 MessageHeader . . . 186
8.165MBIM_INDICATE_STATUS_MSG Struct Reference . . . 186
8.165.1 Field Documentation . . . 187
8.165.1.1 CID . . . 187
8.165.1.2 DeviceServiceId . . . 187
8.165.1.3 FragmentHeader . . . 187
8.165.1.4 InformationBufferLength . . . 187
8.165.1.5 MessageHeader . . . 187
8.166MBIM_MESSAGE_HEADER Struct Reference . . . 187
8.166.1 Field Documentation . . . 187
8.166.1.1 MessageLength . . . 187
8.166.1.2 MessageType . . . 187
8.166.1.3 TransactionId . . . 187
8.167MBIM_OPEN_DONE_MSG Struct Reference . . . 187
8.167.1 Field Documentation . . . 187
8.167.1.1 MessageHeader . . . 187
8.167.1.2 Status . . . 187
8.168MBIM_OPEN_MSG Struct Reference . . . 187
8.168.1 Field Documentation . . . 187
8.168.1.1 MaxControlTransfer . . . 187
8.168.1.2 MessageHeader . . . 188
8.169MbimEventEntry Struct Reference . . . 188
8.169.1 Detailed Description . . . 188
8.169.2 Field Documentation . . . 188
8.169.2.1 CidCount . . . 188

8.169.2.2 Cids . . . 188
8.169.2.3 DeviceServiceId . . . 188
8.170 MbimIndicator Struct Reference . . . 188
8.170.1 Detailed Description . . . 188
8.170.2 Field Documentation . . . 189
8.170.2.1 pIndicationCallback . . . 189
8.170.2.2 pIndicationCallbackContext . . . 189
8.170.2.3 pNext . . . 189
8.170.2.4 pPrev . . . 189
8.171 MbimSyncObject Struct Reference . . . 189
8.171.1 Detailed Description . . . 189
8.171.2 Field Documentation . . . 190
8.171.2.1 condition . . . 190
8.171.2.2 informationBuffer . . . 190
8.171.2.3 informationBufferLength . . . 190
8.171.2.4 mutex . . . 190
8.171.2.5 status . . . 190
8.172 MbimTransaction Struct Reference . . . 190
8.172.1 Detailed Description . . . 190
8.172.2 Field Documentation . . . 191
8.172.2.1 pDoneCallback . . . 191
8.172.2.2 pDoneCallbackContext . . . 191
8.172.2.3 pNext . . . 191
8.172.2.4 pPrev . . . 191
8.172.2.5 status . . . 191
8.172.2.6 transactionId . . . 191
8.173 MbimTransport Struct Reference . . . 191
8.173.1 Detailed Description . . . 191
8.173.2 Field Documentation . . . 192
8.173.2.1 deviceFd . . . 192
8.173.2.2 indicatorList . . . 192
8.173.2.3 indicatorListLock . . . 192
8.173.2.4 pIncomingMessage . . . 192
8.173.2.5 readThread . . . 192
8.173.2.6 shutdownFd . . . 192
8.173.2.7 transactionId . . . 192
8.173.2.8 transactionIdLock . . . 192
8.173.2.9 transactionList . . . 192
8.173.2.10 transactionListLock . . . 192
8.173.2.11 writeLock . . . 192

8.174 messageModeTlv Struct Reference . . . 192
8.174.1 Detailed Description . . . 193
8.174.2 Field Documentation . . . 193
8.174.2.1 MessageModeInfo . . . 193
8.174.2.2 TlvPresent . . . 193
8.175 nas_acqOrderPref Struct Reference . . . 193
8.175.1 Detailed Description . . . 193
8.175.2 Field Documentation . . . 193
8.175.2.1 acqOrdeLen . . . 194
8.175.2.2 pAcqOrder . . . 194
8.176 nas_AcqOrderPrefTlv Struct Reference . . . 194
8.176.1 Detailed Description . . . 194
8.176.2 Field Documentation . . . 194
8.176.2.1 acqOrdeLen . . . 194
8.176.2.2 pAcqOrder . . . 194
8.176.2.3 TlvPresent . . . 194
8.177 nas_ActPilotPNElement Struct Reference . . . 194
8.177.1 Detailed Description . . . 195
8.177.2 Field Documentation . . . 195
8.177.2.1 ActSetPilotPN . . . 195
8.177.2.2 ActSetPilotPNStrength . . . 195
8.178 nas_AddCDMASysInfo Struct Reference . . . 195
8.178.1 Detailed Description . . . 195
8.178.2 Field Documentation . . . 195
8.178.2.1 geoSysIdx . . . 195
8.178.2.2 regPrd . . . 195
8.179 nas_AddSysInfo Struct Reference . . . 195
8.179.1 Detailed Description . . . 196
8.179.2 Field Documentation . . . 196
8.179.2.1 cellBroadcastCap . . . 196
8.179.2.2 geoSysIdx . . . 196
8.180 nas_BandPrefInfoTlv Struct Reference . . . 196
8.180.1 Detailed Description . . . 196
8.180.2 Field Documentation . . . 197
8.180.2.1 bits_129_192 . . . 197
8.180.2.2 bits_193_256 . . . 197
8.180.2.3 bits_1_64 . . . 197
8.180.2.4 bits_65_128 . . . 197
8.180.2.5 TlvPresent . . . 197
8.181 nas_BandPrefTlv Struct Reference . . . 197

8.181.1 Detailed Description . . . 197
8.181.2 Field Documentation . . . 199
8.181.2.1 BandPref . . . 199
8.181.2.2 TlvPresent . . . 199
8.182 nas_CallBarringSysInfo Struct Reference . . . 199
8.182.1 Detailed Description . . . 199
8.182.2 Field Documentation . . . 199
8.182.2.1 csBarStatus . . . 199
8.182.2.2 psBarStatus . . . 199
8.183 nas_callBarStatus Struct Reference . . . 199
8.183.1 Detailed Description . . . 200
8.183.2 Field Documentation . . . 200
8.183.2.1 csBarStatus . . . 200
8.183.2.2 psBarStatus . . . 200
8.184 nas_CDMAChannel Struct Reference . . . 200
8.184.1 Detailed Description . . . 200
8.184.2 Field Documentation . . . 201
8.184.2.1 priChA . . . 201
8.184.2.2 priChB . . . 201
8.184.2.3 secChA . . . 201
8.184.2.4 secChB . . . 201
8.185 nas_CDMAECIOThresh Struct Reference . . . 201
8.185.1 Detailed Description . . . 201
8.185.2 Field Documentation . . . 201
8.185.2.1 CDMAECIOThreshListLen . . . 201
8.185.2.2 pCDMAECIOThreshList . . . 202
8.186 nas_CDMAInfo Struct Reference . . . 202
8.186.1 Detailed Description . . . 202
8.186.2 Field Documentation . . . 202
8.186.2.1 baseId . . . 202
8.186.2.2 baseLat . . . 202
8.186.2.3 baseLong . . . 202
8.186.2.4 nid . . . 203
8.186.2.5 refpn . . . 203
8.186.2.6 sid . . . 203
8.187 nas_CDMARSSIThresh Struct Reference . . . 203
8.187.1 Detailed Description . . . 203
8.187.2 Field Documentation . . . 203
8.187.2.1 CDMARSSIThreshListLen . . . 203
8.187.2.2 pCDMARSSIThreshList . . . 203

8.188 nas_CDMASysInfo Struct Reference . . . 203
8.188.1 Detailed Description . . . 204
8.188.2 Field Documentation . . . 206
8.188.2.1 baseId . . . 206
8.188.2.2 baseLat . . . 206
8.188.2.3 baseLong . . . 206
8.188.2.4 bsInfoValid . . . 206
8.188.2.5 bsPRev . . . 206
8.188.2.6 bsPRevValid . . . 206
8.188.2.7 ccsSupported . . . 206
8.188.2.8 ccsSupportedValid . . . 206
8.188.2.9 cdmaSysIdValid . . . 206
8.188.2.10 isSysPrlMatch . . . 206
8.188.2.11 isSysPrlMatchValid . . . 206
8.188.2.12 MCC . . . 206
8.188.2.13 MNC . . . 206
8.188.2.14 networkID . . . 206
8.188.2.15 networkIdValid . . . 207
8.188.2.16 packetZone . . . 207
8.188.2.17 packetZoneValid . . . 207
8.188.2.18 pRevInUse . . . 207
8.188.2.19 pRevInUseValid . . . 207
8.188.2.20 sysInfoCDMA . . . 207
8.188.2.21 systemID . . . 207
8.189 nas_CDMASysInfoExt Struct Reference . . . 207
8.189.1 Detailed Description . . . 207
8.189.2 Field Documentation . . . 207
8.189.2.1 imsi_11_12 . . . 207
8.189.2.2 MCC . . . 207
8.190 nas_cellParams Struct Reference . . . 207
8.190.1 Detailed Description . . . 208
8.190.2 Field Documentation . . . 208
8.190.2.1 pci . . . 208
8.190.2.2 rsrp . . . 208
8.190.2.3 rsrq . . . 208
8.190.2.4 rssi . . . 208
8.190.2.5 srxlev . . . 208
8.191 nas_ciotAcqOrderPref Struct Reference . . . 208
8.191.1 Detailed Description . . . 208
8.191.2 Field Documentation . . . 209

8.191.2.1 ciotAcqOrderLen . . . 209
8.191.2.2 pCiotAcqOrder . . . 209
8.192 nas_CiotAcqOrderPrefTlv Struct Reference . . . 209
8.192.1 Detailed Description . . . 209
8.192.2 Field Documentation . . . 210
8.192.2.1 ciotAcqOrderLen . . . 210
8.192.2.2 pCiotAcqOrder . . . 210
8.192.2.3 TlvPresent . . . 210
8.193 nas_CiotLteOpModePrefTlv Struct Reference . . . 210
8.193.1 Detailed Description . . . 210
8.193.2 Field Documentation . . . 210
8.193.2.1 ciotLteOpModePref . . . 210
8.193.2.2 TlvPresent . . . 210
8.194 nas_CommInfo Struct Reference . . . 210
8.194.1 Detailed Description . . . 211
8.194.2 Field Documentation . . . 212
8.194.2.1 imsRegState . . . 212
8.194.2.2 modemMode . . . 212
8.194.2.3 psState . . . 212
8.194.2.4 systemMode . . . 212
8.194.2.5 temperature . . . 212
8.195 nas_CsgId Struct Reference . . . 212
8.195.1 Detailed Description . . . 212
8.195.2 Field Documentation . . . 212
8.195.2.1 csgId . . . 212
8.195.2.2 TlvPresent . . . 212
8.196 nas_CSGID Struct Reference . . . 212
8.196.1 Detailed Description . . . 213
8.196.2 Field Documentation . . . 213
8.196.2.1 id . . . 213
8.196.2.2 mcc . . . 213
8.196.2.3 mnc . . . 213
8.196.2.4 mncPcsDigits . . . 213
8.196.2.5 rat . . . 213
8.197 nas_CSGInfo Struct Reference . . . 213
8.197.1 Detailed Description . . . 214
8.197.2 Field Documentation . . . 214
8.197.2.1 csg_list_cat . . . 214
8.197.2.2 id . . . 214
8.197.2.3 mcc . . . 214

8.197.2.4 mnc . . . 214
8.197.2.5 name . . . 214
8.197.2.6 name_len . . . 214
8.198 nas_CSGInfoTlv Struct Reference . . . 214
8.198.1 Detailed Description . . . 215
8.198.2 Field Documentation . . . 215
8.198.2.1 csg_info_len . . . 215
8.198.2.2 csgInfo . . . 215
8.198.2.3 TlvPresent . . . 215
8.199 nas_currentPLMN Struct Reference . . . 215
8.199.1 Detailed Description . . . 215
8.199.2 Field Documentation . . . 216
8.199.2.1 MCC . . . 216
8.199.2.2 MNC . . . 216
8.199.2.3 netDescr . . . 216
8.199.2.4 netDescrLength . . . 216
8.200 nas_dataSrvCapabilities Struct Reference . . . 216
8.200.1 Detailed Description . . . 216
8.200.2 Field Documentation . . . 217
8.200.2.1 dataCapabilities . . . 217
8.200.2.2 dataCapabilitiesLen . . . 217
8.201 nas_DataStatusDetail Struct Reference . . . 217
8.201.1 Detailed Description . . . 217
8.201.2 Field Documentation . . . 218
8.201.2.1 IPAddress . . . 218
8.201.2.2 LastErrCode . . . 218
8.202 nas_DDTMSettings Struct Reference . . . 219
8.202.1 Detailed Description . . . 219
8.202.2 Field Documentation . . . 220
8.202.2.1 CurrentDDTMStatus . . . 220
8.202.2.2 DDTMAction . . . 220
8.202.2.3 DDTMPrefSetting . . . 220
8.202.2.4 ServiceOption . . . 220
8.202.2.5 SOLen . . . 220
8.202.2.6 SOListAction . . . 220
8.203 nas_detailSvcInfo Struct Reference . . . 220
8.203.1 Detailed Description . . . 220
8.203.2 Field Documentation . . . 221
8.203.2.1 hdrHybrid . . . 221
8.203.2.2 hdrSrvStatus . . . 221

8.203.2.3 isSysForbidden . . . 221
8.203.2.4 srvCapability . . . 221
8.203.2.5 srvStatus . . . 221
8.204 nas_DeviceConfigDetail Struct Reference . . . 221
8.204.1 Detailed Description . . . 221
8.204.2 Field Documentation . . . 222
8.204.2.1 Chipset . . . 222
8.204.2.2 HWVersion . . . 222
8.204.2.3 QLIC . . . 222
8.204.2.4 Technology . . . 222
8.205 nas_dirNum Struct Reference . . . 222
8.205.1 Detailed Description . . . 222
8.205.2 Field Documentation . . . 223
8.205.2.1 dirNum . . . 223
8.205.2.2 dirNumLen . . . 223
8.206 nas_DRCParams Struct Reference . . . 223
8.206.1 Detailed Description . . . 223
8.206.2 Field Documentation . . . 223
8.206.2.1 DRCCover . . . 223
8.206.2.2 DRCValue . . . 223
8.207 nas_ecioListElement Struct Reference . . . 223
8.207.1 Detailed Description . . . 223
8.207.2 Field Documentation . . . 224
8.207.2.1 ecio . . . 224
8.207.2.2 radioIf . . . 224
8.208 nas_ECIOThresh Struct Reference . . . 224
8.208.1 Detailed Description . . . 224
8.208.2 Field Documentation . . . 224
8.208.2.1 ECIOThresListLen . . . 224
8.208.2.2 pECIOThresList . . . 225
8.209 nas_EdrxCiotLteMode Struct Reference . . . 225
8.209.1 Detailed Description . . . 225
8.209.2 Field Documentation . . . 225
8.209.2.1 lteOpMode . . . 225
8.209.2.2 TlvPresent . . . 225
8.210 nas_EdrxCycleLength Struct Reference . . . 225
8.210.1 Detailed Description . . . 225
8.210.2 Field Documentation . . . 226
8.210.2.1 cycleLength . . . 226
8.210.2.2 TlvPresent . . . 226

8.211 nas_EdrxEnableType Struct Reference . . . 226
8.211.1 Detailed Description . . . 226
8.211.2 Field Documentation . . . 226
8.211.2.1 edrxEnabled . . . 226
8.211.2.2 TlvPresent . . . 226
8.212 nas_EdrxPagingTimeWindow Struct Reference . . . 226
8.212.1 Detailed Description . . . 226
8.212.2 Field Documentation . . . 227
8.212.2.1 edrxPtw . . . 227
8.212.2.2 TlvPresent . . . 227
8.213 nas_EdrxRatType Struct Reference . . . 227
8.213.1 Detailed Description . . . 227
8.213.2 Field Documentation . . . 227
8.213.2.1 edrxRatType . . . 227
8.213.2.2 TlvPresent . . . 227
8.214 nas_EmerModeTlv Struct Reference . . . 227
8.214.1 Detailed Description . . . 228
8.214.2 Field Documentation . . . 228
8.214.2.1 EmerMode . . . 228
8.214.2.2 TlvPresent . . . 228
8.215 nas_errorRateListElement Struct Reference . . . 228
8.215.1 Detailed Description . . . 228
8.215.2 Field Documentation . . . 229
8.215.2.1 errorRate . . . 229
8.215.2.2 radioIf . . . 229
8.216 nas_ForbiddenNetworks3GPP Struct Reference . . . 229
8.216.1 Detailed Description . . . 229
8.216.2 Field Documentation . . . 230
8.216.2.1 forbiddenNwInstLen . . . 230
8.216.2.2 MCC . . . 230
8.216.2.3 MNC . . . 230
8.216.2.4 TlvPresent . . . 230
8.217 nas_GERANInfo Struct Reference . . . 230
8.217.1 Detailed Description . . . 230
8.217.2 Field Documentation . . . 231
8.217.2.1 arfcn . . . 231
8.217.2.2 bsic . . . 231
8.217.2.3 cellID . . . 231
8.217.2.4 insNmrCellInfo . . . 232
8.217.2.5 lac . . . 232

8.217.2.6 nmrInst . . . 232
8.217.2.7 plmn . . . 232
8.217.2.8 rxLev . . . 232
8.217.2.9 timingAdvance . . . 232
8.218 nas_geranInstArr Struct Reference . . . 232
8.218.1 Detailed Description . . . 232
8.218.2 Field Documentation . . . 232
8.218.2.1 geranArfcn . . . 232
8.218.2.2 geranBsicBcc . . . 232
8.218.2.3 geranBsicNcc . . . 232
8.218.2.4 geranRank . . . 233
8.218.2.5 geranRssi . . . 233
8.219 nas_geranInstInfo Struct Reference . . . 233
8.219.1 Detailed Description . . . 233
8.219.2 Field Documentation . . . 233
8.219.2.1 geranArfcn . . . 233
8.219.2.2 geranBsicBcc . . . 233
8.219.2.3 geranBsicNcc . . . 233
8.219.2.4 geranRssi . . . 233
8.220 nas_gsmCellInfo Struct Reference . . . 233
8.220.1 Detailed Description . . . 234
8.220.2 Field Documentation . . . 234
8.220.2.1 arfcn . . . 234
8.220.2.2 band1900 . . . 234
8.220.2.3 bsicId . . . 234
8.220.2.4 cellIdValid . . . 234
8.220.2.5 rssi . . . 234
8.220.2.6 srxlev . . . 234
8.221 nas_GSMRSSIThresh Struct Reference . . . 234
8.221.1 Detailed Description . . . 235
8.221.2 Field Documentation . . . 235
8.221.2.1 GSMRSSIThreshListLen . . . 235
8.221.2.2 pGSMRSSIThreshList . . . 235
8.222 nas_GSMSrvStatusInfo Struct Reference . . . 235
8.222.1 Detailed Description . . . 235
8.222.2 Field Documentation . . . 236
8.222.2.1 isPrefDataPath . . . 236
8.222.2.2 srvStatus . . . 236
8.222.2.3 trueSrvStatus . . . 236
8.223 nas_GSMSysInfo Struct Reference . . . 236

8.223.1 Detailed Description . . . 236
8.223.2 Field Documentation . . . 238
8.223.2.1 cellId . . . 238
8.223.2.2 cellIdValid . . . 238
8.223.2.3 dtmSupp . . . 238
8.223.2.4 dtmSuppValid . . . 238
8.223.2.5 egprsSupp . . . 238
8.223.2.6 egprsSuppValid . . . 238
8.223.2.7 lac . . . 239
8.223.2.8 lacValid . . . 239
8.223.2.9 MCC . . . 239
8.223.2.10 MNC . . . 239
8.223.2.11 networkIdValid . . . 239
8.223.2.12 regRejectInfoValid . . . 239
8.223.2.13 rejCause . . . 239
8.223.2.14 rejectSrvDomain . . . 239
8.223.2.15 sysInfoGSM . . . 239
8.224 nas_GWAcqOrderPrefTlv Struct Reference . . . 239
8.224.1 Detailed Description . . . 239
8.224.2 Field Documentation . . . 239
8.224.2.1 GWAcqOrderPref . . . 239
8.224.2.2 TlvPresent . . . 239
8.225 nas_HDRECIOThresh Struct Reference . . . 239
8.225.1 Detailed Description . . . 240
8.225.2 Field Documentation . . . 240
8.225.2.1 HDRECIOThreshListLen . . . 240
8.225.2.2 pHDRECIOThreshList . . . 240
8.226 nas_HDRIOThresh Struct Reference . . . 240
8.226.1 Detailed Description . . . 240
8.226.2 Field Documentation . . . 240
8.226.2.1 HDRIOThreshListLen . . . 240
8.226.2.2 pHDRIOThreshList . . . 240
8.227 nas_HDRPersonality_Ind_Data Struct Reference . . . 241
8.227.1 Detailed Description . . . 241
8.227.2 Field Documentation . . . 241
8.227.2.1 pCurrentPersonality . . . 241
8.227.2.2 pPersonalityListLength . . . 241
8.227.2.3 pProtocolSubtypeElement . . . 241
8.228 nas_HDRRSSIThresh Struct Reference . . . 241
8.228.1 Detailed Description . . . 241

8.228.2 Field Documentation . . . 242
8.228.2.1 HDRRSSIThreshListLen . . . 242
8.228.2.2 pHDRRSSIThreshList . . . 242
8.229 nas_HDRSINRThresh Struct Reference . . . 242
8.229.1 Detailed Description . . . 242
8.229.2 Field Documentation . . . 242
8.229.2.1 HDRSINRThresListLen . . . 242
8.229.2.2 pHDRSINRThresList . . . 242
8.230 nas_HDRSINRThreshold Struct Reference . . . 242
8.230.1 Detailed Description . . . 243
8.230.2 Field Documentation . . . 243
8.230.2.1 HDRSINRThreshListLen . . . 243
8.230.2.2 pHDRSINRThreshList . . . 243
8.231 nas_HDRSysInfo Struct Reference . . . 243
8.231.1 Detailed Description . . . 243
8.231.2 Field Documentation . . . 245
8.231.2.1 hdrActiveProt . . . 245
8.231.2.2 hdrActiveProtValid . . . 245
8.231.2.3 hdrPersonality . . . 245
8.231.2.4 hdrPersonalityValid . . . 245
8.231.2.5 is856SysId . . . 245
8.231.2.6 is856SysIdValid . . . 245
8.231.2.7 isSysPrlMatch . . . 245
8.231.2.8 isSysPrlMatchValid . . . 245
8.231.2.9 sysInfoHDR . . . 245
8.232 nas_homeNwMNC3GppTlv Struct Reference . . . 245
8.232.1 Detailed Description . . . 245
8.232.2 Field Documentation . . . 246
8.232.2.1 is3GppNw . . . 246
8.232.2.2 mccIncPcsDigit . . . 246
8.232.2.3 TlvPresent . . . 246
8.233 nas_homeSIDNID Struct Reference . . . 246
8.233.1 Detailed Description . . . 246
8.233.2 Field Documentation . . . 246
8.233.2.1 numInstances . . . 246
8.233.2.2 SidNid . . . 246
8.234 nas_ImsVoiceSupportLteTlv Struct Reference . . . 246
8.234.1 Detailed Description . . . 247
8.234.2 Field Documentation . . . 247
8.234.2.1 ImsVoiceSupportLte . . . 247

8.234.2.2 TLVPresent . . . 247
8.235 nas_infoInterFreq Struct Reference . . . 247
8.235.1 Detailed Description . . . 247
8.235.2 Field Documentation . . . 248
8.235.2.1 cell_resel_priority . . . 248
8.235.2.2 cellInterFreqParams . . . 248
8.235.2.3 cells_len . . . 248
8.235.2.4 earfcn . . . 248
8.235.2.5 threshXHigh . . . 248
8.235.2.6 threshXLow . . . 248
8.236 nas_IOThresh Struct Reference . . . 248
8.236.1 Detailed Description . . . 248
8.236.2 Field Documentation . . . 249
8.236.2.1 IOThresListLen . . . 249
8.236.2.2 pIOThresList . . . 249
8.237 nas_LTEARFCNTlv Struct Reference . . . 249
8.237.1 Detailed Description . . . 249
8.237.2 Field Documentation . . . 249
8.237.2.1 lteArfcn . . . 249
8.237.2.2 numlteArfcn . . . 249
8.238 nas_lteBandPrefExt Struct Reference . . . 249
8.238.1 Detailed Description . . . 249
8.238.2 Field Documentation . . . 250
8.238.2.1 bits_129_192 . . . 250
8.238.2.2 bits_193_256 . . . 250
8.238.2.3 bits_1_64 . . . 250
8.238.2.4 bits_65_128 . . . 250
8.239 nas_LTEBandPrefTlv Struct Reference . . . 250
8.239.1 Detailed Description . . . 250
8.239.2 Field Documentation . . . 251
8.239.2.1 LTEBandPref . . . 252
8.239.2.2 TlvPresent . . . 252
8.240 nas_LteCiotOpModeTlv Struct Reference . . . 252
8.240.1 Detailed Description . . . 252
8.240.2 Field Documentation . . . 252
8.240.2.1 campedCiotLteOpMode . . . 252
8.240.2.2 TlvPresent . . . 252
8.241 nas_lteEARFCN Struct Reference . . . 252
8.241.1 Detailed Description . . . 252
8.241.2 Field Documentation . . . 253

8.241.2.1 earfcn0 . . . 253
8.241.2.2 earfcn1 . . . 253
8.241.2.3 status . . . 253
8.242 nas_LteEarfcnInfo Struct Reference . . . 253
8.242.1 Detailed Description . . . 253
8.242.2 Field Documentation . . . 253
8.242.2.1 lteEarfcn . . . 253
8.242.2.2 lteInterEarfcnlen . . . 253
8.242.2.3 TlvPresent . . . 253
8.243 nas_LteEmbmsCoverageTlv Struct Reference . . . 253
8.243.1 Detailed Description . . . 254
8.243.2 Field Documentation . . . 254
8.243.2.1 LteEmbmsCoverage . . . 254
8.243.2.2 TLVPresent . . . 254
8.244 nas_LteEmbmsTraceIdTlv Struct Reference . . . 254
8.244.1 Detailed Description . . . 254
8.244.2 Field Documentation . . . 254
8.244.2.1 LteEmbmsTraceId . . . 254
8.244.2.2 TLVPresent . . . 255
8.245 nas_lteGsmCellInfo Struct Reference . . . 255
8.245.1 Detailed Description . . . 255
8.245.2 Field Documentation . . . 255
8.245.2.1 cellReselPriority . . . 255
8.245.2.2 cells_len . . . 256
8.245.2.3 GsmCellInfo . . . 256
8.245.2.4 nccPermitted . . . 256
8.245.2.5 threshGsmHigh . . . 256
8.245.2.6 threshGsmLow . . . 256
8.246 nas_LTEInfo Struct Reference . . . 256
8.246.1 Detailed Description . . . 256
8.246.2 Field Documentation . . . 257
8.246.2.1 band . . . 257
8.246.2.2 bandwidth . . . 258
8.246.2.3 emmConnState . . . 258
8.246.2.4 emmState . . . 258
8.246.2.5 emmSubState . . . 258
8.246.2.6 RXChan . . . 258
8.246.2.7 TXChan . . . 258
8.247 nas_LTEInfoInterfreq Struct Reference . . . 258
8.247.1 Detailed Description . . . 258

8.247.2 Field Documentation . . . 258
8.247.2.1 freqsLen . . . 258
8.247.2.2 InfoInterfreq . . . 258
8.247.2.3 ueInIdle . . . 258
8.248 nas_LTEInfoIntrafreq Struct Reference . . . 258
8.248.1 Detailed Description . . . 259
8.248.2 Field Documentation . . . 260
8.248.2.1 CellParams . . . 260
8.248.2.2 cellReselPriority . . . 260
8.248.2.3 cellsLen . . . 260
8.248.2.4 earfcn . . . 260
8.248.2.5 globalCellId . . . 260
8.248.2.6 plmn . . . 260
8.248.2.7 servingCellId . . . 260
8.248.2.8 sIntraSearch . . . 260
8.248.2.9 sNonIntraSearch . . . 260
8.248.2.10 tac . . . 260
8.248.2.11 threshServingLow . . . 260
8.248.2.12 ueInIdle . . . 260
8.249 nas_LTEInfoNeighboringGSM Struct Reference . . . 260
8.249.1 Detailed Description . . . 261
8.249.2 Field Documentation . . . 261
8.249.2.1 freqsLen . . . 261
8.249.2.2 LteGsmCellInfo . . . 261
8.249.2.3 ueInIdle . . . 261
8.250 nas_LTEInfoNeighboringWCDMA Struct Reference . . . 261
8.250.1 Detailed Description . . . 261
8.250.2 Field Documentation . . . 262
8.250.2.1 freqsLen . . . 262
8.250.2.2 LTEWCDMACellInfo . . . 262
8.250.2.3 ueInIdle . . . 262
8.251 nas_LteM1BandPrefTlv Struct Reference . . . 262
8.251.1 Detailed Description . . . 262
8.251.2 Field Documentation . . . 263
8.251.2.1 lteM1BandPref . . . 264
8.251.2.2 TlvPresent . . . 264
8.252 nas_LteNb1BandPrefTlv Struct Reference . . . 264
8.252.1 Detailed Description . . . 264
8.252.2 Field Documentation . . . 265
8.252.2.1 lteNb1BandPref . . . 266

8.252.2.2 TlvPresent . . . 266
8.253 nas_LTEOperationalModeTlv Struct Reference . . . 266
8.253.1 Detailed Description . . . 266
8.253.2 Field Documentation . . . 266
8.253.2.1 ciotLteOpMode . . . 266
8.253.2.2 TlvPresent . . . 266
8.254 nas_LTEOperationMode Struct Reference . . . 266
8.254.1 Detailed Description . . . 266
8.254.2 Field Documentation . . . 267
8.254.2.1 pLTEOperationMode . . . 267
8.254.2.2 TlvPresent . . . 267
8.255 nas_LteOpMode Struct Reference . . . 267
8.255.1 Detailed Description . . . 267
8.255.2 Field Documentation . . . 267
8.255.2.1 lteOpMode . . . 267
8.255.2.2 TlvPresent . . . 267
8.256 nas_lteOpModeTlv Struct Reference . . . 267
8.256.1 Detailed Description . . . 268
8.256.2 Field Documentation . . . 268
8.256.2.1 lteOpMode . . . 268
8.256.2.2 lteOpModeLen . . . 268
8.256.2.3 MCC . . . 268
8.256.2.4 MNC . . . 268
8.256.2.5 TlvPresent . . . 268
8.257 nas_ltePCI Struct Reference . . . 268
8.257.1 Detailed Description . . . 269
8.257.2 Field Documentation . . . 269
8.257.2.1 earfcn . . . 269
8.257.2.2 pci . . . 269
8.257.2.3 status . . . 269
8.258 nas_LteRegDomainTlv Struct Reference . . . 269
8.258.1 Detailed Description . . . 269
8.258.2 Field Documentation . . . 270
8.258.2.1 LteRegDomain . . . 270
8.258.2.2 TLVPresent . . . 270
8.259 nas_lteRsrpinformation Struct Reference . . . 270
8.259.1 Detailed Description . . . 270
8.259.2 Field Documentation . . . 270
8.259.2.1 rsrplevel . . . 270
8.260 nas_LTERSRPThresh Struct Reference . . . 270

8.260.1 Detailed Description . . . 270
8.260.2 Field Documentation . . . 271
8.260.2.1 LTERSRPThreshListLen . . . 271
8.260.2.2 pLTERSRPThreshList . . . 271
8.261 nas_LTERSRQThresh Struct Reference . . . 271
8.261.1 Detailed Description . . . 271
8.261.2 Field Documentation . . . 271
8.261.2.1 LTERSRQThreshListLen . . . 271
8.261.2.2 pLTERSRQThreshList . . . 271
8.262 nas_LTERSSIThresh Struct Reference . . . 271
8.262.1 Detailed Description . . . 271
8.262.2 Field Documentation . . . 272
8.262.2.1 LTERSSIThreshListLen . . . 272
8.262.2.2 pLTERSSIThreshList . . . 272
8.263 nas_LTESigRptCfg Struct Reference . . . 272
8.263.1 Detailed Description . . . 272
8.263.2 Field Documentation . . . 273
8.263.2.1 avgPeriod . . . 273
8.263.2.2 rptRate . . . 273
8.264 nas_LTESigRptConfig Struct Reference . . . 273
8.264.1 Detailed Description . . . 273
8.264.2 Field Documentation . . . 274
8.264.2.1 avgPeriod . . . 274
8.264.2.2 rptRate . . . 274
8.265 nas_lteSnrinformation Struct Reference . . . 274
8.265.1 Detailed Description . . . 274
8.265.2 Field Documentation . . . 274
8.265.2.1 snrlevel . . . 274
8.266 nas_LTESNRThresh Struct Reference . . . 274
8.266.1 Detailed Description . . . 275
8.266.2 Field Documentation . . . 275
8.266.2.1 LTESNRThresListLen . . . 275
8.266.2.2 pLTESNRThresList . . . 275
8.267 nas_LTESNRThreshold Struct Reference . . . 275
8.267.1 Detailed Description . . . 275
8.267.2 Field Documentation . . . 275
8.267.2.1 LTESNRThreshListLen . . . 275
8.267.2.2 pLTESNRThreshList . . . 276
8.268 nas_LTESysInfo Struct Reference . . . 276
8.268.1 Detailed Description . . . 276

8.268.2 Field Documentation . . . 277
8.268.2.1 cellId . . . 278
8.268.2.2 cellIdValid . . . 278
8.268.2.3 lac . . . 278
8.268.2.4 lacValid . . . 278
8.268.2.5 MCC . . . 278
8.268.2.6 MNC . . . 278
8.268.2.7 networkIdValid . . . 278
8.268.2.8 regRejectInfoValid . . . 278
8.268.2.9 rejCause . . . 278
8.268.2.10 rejectSrvDomain . . . 278
8.268.2.11 sysInfoLTE . . . 278
8.268.2.12 tac . . . 278
8.268.2.13 tacValid . . . 278
8.269 nas_LteVoiceDomainTlv Struct Reference . . . 278
8.269.1 Detailed Description . . . 278
8.269.2 Field Documentation . . . 278
8.269.2.1 LteVoiceDomain . . . 279
8.269.2.2 TLVPresent . . . 279
8.270 nas_lteWcdmaCellInfo Struct Reference . . . 279
8.270.1 Detailed Description . . . 279
8.270.2 Field Documentation . . . 279
8.270.2.1 cellReselPriority . . . 279
8.270.2.2 cellsLen . . . 279
8.270.2.3 threshXhigh . . . 280
8.270.2.4 threshXlow . . . 280
8.270.2.5 uarfcn . . . 280
8.270.2.6 WCDMACellInfo . . . 280
8.271 nas_Mdn Struct Reference . . . 280
8.271.1 Detailed Description . . . 280
8.271.2 Field Documentation . . . 280
8.271.2.1 mdn . . . 280
8.271.2.2 mdnLen . . . 280
8.272 nas_minBasedIMSI Struct Reference . . . 280
8.272.1 Detailed Description . . . 280
8.272.2 Field Documentation . . . 281
8.272.2.1 imsiM1112 . . . 281
8.272.2.2 imsiMS1 . . . 281
8.272.2.3 imsiMS2 . . . 281
8.272.2.4 mccM . . . 281

8.273nas_MNCPCSDigitStatus Struct Reference . . . 281
8.273.1 Detailed Description . . . 281
8.273.2 Field Documentation . . . 282
8.273.2.1 MCC . . . 282
8.273.2.2 MNC . . . 282
8.273.2.3 MNCIncPCSDigit . . . 282
8.273.2.4 TLVPresent . . . 282
8.274nas_MNRInfo Struct Reference . . . 282
8.274.1 Detailed Description . . . 282
8.274.2 Field Documentation . . . 282
8.274.2.1 mcc . . . 282
8.274.2.2 mnc . . . 282
8.274.2.3 rat . . . 282
8.275nas_ModePrefTlv Struct Reference . . . 282
8.275.1 Detailed Description . . . 283
8.275.2 Field Documentation . . . 283
8.275.2.1 ModePref . . . 283
8.275.2.2 TlvPresent . . . 283
8.276nas_namName Struct Reference . . . 283
8.276.1 Detailed Description . . . 283
8.276.2 Field Documentation . . . 284
8.276.2.1 namName . . . 284
8.276.2.2 namNameLen . . . 284
8.277nas_netSelectionPref Struct Reference . . . 284
8.277.1 Detailed Description . . . 284
8.277.2 Field Documentation . . . 284
8.277.2.1 mcc . . . 284
8.277.2.2 mnc . . . 284
8.277.2.3 netReg . . . 284
8.278nas_NetSelPrefTlv Struct Reference . . . 284
8.278.1 Detailed Description . . . 284
8.278.2 Field Documentation . . . 285
8.278.2.1 NetSelPref . . . 285
8.278.2.2 TlvPresent . . . 285
8.279nas_NetworkNameSrcTlv Struct Reference . . . 285
8.279.1 Detailed Description . . . 285
8.279.2 Field Documentation . . . 286
8.279.2.1 nw_name_src . . . 286
8.279.2.2 nw_name_src_len . . . 286
8.279.2.3 TlvPresent . . . 286

8.280 nas_networkNameSrcTlv Struct Reference . . . 286
8.280.1 Detailed Description . . . 286
8.280.2 Field Documentation . . . 286
8.280.2.1 nwNameSrc . . . 286
8.280.2.2 nwNameSrcLen . . . 286
8.280.2.3 TlvPresent . . . 286
8.281 nas_NetworkScan3GPPInfo Struct Reference . . . 287
8.281.1 Detailed Description . . . 287
8.281.2 Field Documentation . . . 288
8.281.2.1 mcc . . . 288
8.281.2.2 mnc . . . 288
8.281.2.3 mnc_includes_pcs_digit . . . 288
8.281.2.4 network_desc . . . 288
8.281.2.5 network_desc_len . . . 288
8.281.2.6 network_status . . . 288
8.281.2.7 rat . . . 288
8.282 nas_NetworkScan3GPPInfoTlv Struct Reference . . . 288
8.282.1 Detailed Description . . . 288
8.282.2 Field Documentation . . . 289
8.282.2.1 netScan3GPPInfo . . . 289
8.282.2.2 num_inst . . . 289
8.282.2.3 TlvPresent . . . 289
8.283 nas_NetworkScanStatusTlv Struct Reference . . . 289
8.283.1 Detailed Description . . . 289
8.283.2 Field Documentation . . . 290
8.283.2.1 scanStatus . . . 290
8.283.2.2 TlvPresent . . . 290
8.284 nas_NetworkStat1x Struct Reference . . . 290
8.284.1 Detailed Description . . . 290
8.284.2 Field Documentation . . . 292
8.284.2.1 ActSetCnt . . . 292
8.284.2.2 NeighborSetCnt . . . 292
8.284.2.3 pActPilotPNElements . . . 292
8.284.2.4 pNeighborSetPilotPN . . . 292
8.284.2.5 RX_EC_IO . . . 292
8.284.2.6 RX_PWR . . . 292
8.284.2.7 SO . . . 292
8.284.2.8 State . . . 292
8.284.2.9 TX_PWR . . . 292
8.285 nas_NetworkStatEVDO Struct Reference . . . 292

8.285.1 Detailed Description . . . 293
8.285.2 Field Documentation . . . 294
8.285.2.1 MACIndex . . . 294
8.285.2.2 PER . . . 294
8.285.2.3 PilotEnergy . . . 294
8.285.2.4 pSectorID . . . 294
8.285.2.5 RX_PWR . . . 294
8.285.2.6 SectorIDLen . . . 294
8.285.2.7 SNR . . . 294
8.285.2.8 State . . . 294
8.286 nas_nmrCellInfo Struct Reference . . . 294
8.286.1 Detailed Description . . . 294
8.286.2 Field Documentation . . . 295
8.286.2.1 nmrArfcn . . . 295
8.286.2.2 nmrBsic . . . 295
8.286.2.3 nmrCellID . . . 295
8.286.2.4 nmrLac . . . 295
8.286.2.5 nmrPlmn . . . 295
8.286.2.6 nmrRxLev . . . 295
8.287 nas_nr5gBandPref Struct Reference . . . 295
8.287.1 Detailed Description . . . 295
8.287.2 Field Documentation . . . 296
8.287.2.1 bits_129_192 . . . 296
8.287.2.2 bits_193_256 . . . 296
8.287.2.3 bits_1_64 . . . 296
8.287.2.4 bits_65_128 . . . 296
8.288 nas_NR5GCellStatusInfoTlv Struct Reference . . . 296
8.288.1 Detailed Description . . . 296
8.288.2 Field Documentation . . . 296
8.288.2.1 nr5gCellStatus . . . 296
8.288.2.2 TLVPresent . . . 296
8.289 nas_NR5GCellStatusTlv Struct Reference . . . 296
8.289.1 Detailed Description . . . 297
8.289.2 Field Documentation . . . 297
8.289.2.1 nr5gCellStatus . . . 297
8.289.2.2 TlvPresent . . . 297
8.290 nas_NR5GSerStatTlv Struct Reference . . . 297
8.290.1 Detailed Description . . . 297
8.290.2 Field Documentation . . . 298
8.290.2.1 isPrefDataPath . . . 298

8.290.2.2 srvStatus . . . 298
8.290.2.3 TlvPresent . . . 298
8.290.2.4 trueSrvStatus . . . 298
8.291 nas_NR5GSrvStatusTlv Struct Reference . . . 298
8.291.1 Detailed Description . . . 298
8.291.2 Field Documentation . . . 299
8.291.2.1 isPrefDataPath . . . 299
8.291.2.2 srvStatus . . . 299
8.291.2.3 TLVPresent . . . 299
8.291.2.4 trueSrvStatus . . . 299
8.292 nas_NR5GSysInfoTlv Struct Reference . . . 299
8.292.1 Detailed Description . . . 300
8.292.2 Field Documentation . . . 302
8.292.2.1 cellId . . . 302
8.292.2.2 cellIdValid . . . 302
8.292.2.3 isSysForbidden . . . 302
8.292.2.4 isSysForbiddenValid . . . 302
8.292.2.5 lac . . . 302
8.292.2.6 lacValid . . . 302
8.292.2.7 mcc . . . 302
8.292.2.8 mnc . . . 302
8.292.2.9 nwIdValid . . . 302
8.292.2.10 regRejectInfoValid . . . 302
8.292.2.11 rejectCause . . . 302
8.292.2.12 rejectSrvDomain . . . 303
8.292.2.13 roamStatus . . . 303
8.292.2.14 roamStatusValid . . . 303
8.292.2.15 srvCapability . . . 303
8.292.2.16 srvCapabilityValid . . . 303
8.292.2.17 srvDomain . . . 303
8.292.2.18 srvDomainValid . . . 303
8.292.2.19 tac . . . 303
8.292.2.20 tacValid . . . 303
8.292.2.21 TLVPresent . . . 303
8.293 nas_NR5GSystemInfoTlv Struct Reference . . . 303
8.293.1 Detailed Description . . . 303
8.293.2 Field Documentation . . . 306
8.293.2.1 cellId . . . 306
8.293.2.2 cellIdValid . . . 306
8.293.2.3 lac . . . 306

8.293.2.4 lacValid . . . 306
8.293.2.5 MCC . . . 306
8.293.2.6 MNC . . . 306
8.293.2.7 nwIdValid . . . 306
8.293.2.8 regRejectInfoValid . . . 306
8.293.2.9 rejCause . . . 306
8.293.2.10 rejectSrvDomain . . . 306
8.293.2.11 roamStatus . . . 306
8.293.2.12 roamStatusValid . . . 306
8.293.2.13 srvcapability . . . 306
8.293.2.14 srvCapValid . . . 306
8.293.2.15 srvDomain . . . 306
8.293.2.16 srvDomainValid . . . 306
8.293.2.17 sysForbidden . . . 306
8.293.2.18 sysForbiddenValid . . . 306
8.293.2.19 tac . . . 306
8.293.2.20 tacValid . . . 306
8.293.2.21 TlvPresent . . . 306
8.294 nas_NumScellsConfig Struct Reference . . . 306
8.294.1 Detailed Description . . . 307
8.294.2 Field Documentation . . . 307
8.294.2.1 numScellsCfg . . . 307
8.294.2.2 TlvPresent . . . 307
8.295 nas_nwNameSrc3GppTlv Struct Reference . . . 307
8.295.1 Detailed Description . . . 307
8.295.2 Field Documentation . . . 307
8.295.2.1 NwNameSrc3Gpp . . . 307
8.295.2.2 TlvPresent . . . 307
8.296 nas_operatorNameString Struct Reference . . . 308
8.296.1 Detailed Description . . . 308
8.296.2 Field Documentation . . . 308
8.296.2.1 PLMNName . . . 308
8.297 nas_OperatorPLMNData Struct Reference . . . 308
8.297.1 Detailed Description . . . 308
8.297.2 Field Documentation . . . 309
8.297.2.1 lac1 . . . 309
8.297.2.2 lac2 . . . 309
8.297.2.3 mcc . . . 309
8.297.2.4 mnc . . . 309
8.297.2.5 PLMNRecID . . . 309

8.298 nas_operatorPLMNList Struct Reference . . . 309
8.298.1 Detailed Description . . . 309
8.298.2 Field Documentation . . . 309
8.298.2.1 numInstance . . . 309
8.298.2.2 PLMNData . . . 309
8.299 nas_PhyCaAggDlBW Struct Reference . . . 309
8.299.1 Detailed Description . . . 309
8.299.2 Field Documentation . . . 310
8.299.2.1 aggDlBW . . . 310
8.299.2.2 TlvPresent . . . 310
8.300 nas_PhyCaAggPcellInfo Struct Reference . . . 310
8.300.1 Detailed Description . . . 310
8.300.2 Field Documentation . . . 311
8.300.2.1 dl_bw_value . . . 311
8.300.2.2 freq . . . 311
8.300.2.3 iLTEbandValue . . . 311
8.300.2.4 pci . . . 311
8.300.2.5 TlvPresent . . . 311
8.301 nas_PhyCaAggScellDlBw Struct Reference . . . 311
8.301.1 Detailed Description . . . 311
8.301.2 Field Documentation . . . 311
8.301.2.1 dl_bw_value . . . 311
8.301.2.2 TlvPresent . . . 311
8.302 nas_PhyCaAggScellIndex Struct Reference . . . 311
8.302.1 Detailed Description . . . 311
8.302.2 Field Documentation . . . 312
8.302.2.1 scell_idx . . . 312
8.302.2.2 TlvPresent . . . 312
8.303 nas_PhyCaAggScellIndType Struct Reference . . . 312
8.303.1 Detailed Description . . . 312
8.303.2 Field Documentation . . . 312
8.303.2.1 freq . . . 312
8.303.2.2 pci . . . 312
8.303.2.3 scell_state . . . 312
8.303.2.4 TlvPresent . . . 312
8.304 nas_PhyCaAggScellInfo Struct Reference . . . 313
8.304.1 Detailed Description . . . 313
8.304.2 Field Documentation . . . 315
8.304.2.1 dl_bw_value . . . 315
8.304.2.2 freq . . . 315

8.304.2.3 iLTEbandValue . . . 315
8.304.2.4 pci . . . 315
8.304.2.5 scell_state . . . 315
8.304.2.6 TlvPresent . . . 315
8.305 nas_PilotSetData Struct Reference . . . 315
8.305.1 Detailed Description . . . 315
8.305.2 Field Documentation . . . 315
8.305.2.1 NumPilots . . . 315
8.305.2.2 pPilotSetInfo . . . 315
8.306 nas_PilotSetParams Struct Reference . . . 316
8.306.1 Detailed Description . . . 316
8.306.2 Field Documentation . . . 316
8.306.2.1 PilotPN . . . 316
8.306.2.2 PilotStrength . . . 316
8.306.2.3 PilotType . . . 316
8.307 nas_PlmnID Struct Reference . . . 316
8.307.1 Detailed Description . . . 316
8.307.2 Field Documentation . . . 317
8.307.2.1 mcc . . . 317
8.307.2.2 mnc . . . 317
8.307.2.3 pcsDigit . . . 317
8.307.2.4 TlvPresent . . . 317
8.308 nas_PLMNNetworkName Struct Reference . . . 317
8.308.1 Detailed Description . . . 317
8.308.2 Field Documentation . . . 317
8.308.2.1 numInstance . . . 317
8.308.2.2 PLMNNetName . . . 317
8.309 nas_PLMNNetworkNameData Struct Reference . . . 317
8.309.1 Detailed Description . . . 318
8.309.2 Field Documentation . . . 319
8.309.2.1 codingScheme . . . 319
8.309.2.2 countryInitials . . . 319
8.309.2.3 longName . . . 319
8.309.2.4 longNameLen . . . 319
8.309.2.5 longNameSpareBits . . . 319
8.309.2.6 shortName . . . 319
8.309.2.7 shortNameLen . . . 319
8.309.2.8 shortNameSpareBits . . . 319
8.310 nas_PrefPLMNListTlv Struct Reference . . . 319
8.310.1 Detailed Description . . . 319

8.310.2 Field Documentation . . . 320
8.310.2.1 mcc . . . 320
8.310.2.2 mnc . . . 320
8.310.2.3 mnc_pcs_digit . . . 320
8.310.2.4 plmnListLen . . . 320
8.311 nas_PRLPrefTlv Struct Reference . . . 320
8.311.1 Detailed Description . . . 320
8.311.2 Field Documentation . . . 321
8.311.2.1 PRLPref . . . 321
8.311.2.2 TlvPresent . . . 321
8.312 nas_protocolSubtypeElement Struct Reference . . . 321
8.312.1 Detailed Description . . . 321
8.312.2 Field Documentation . . . 322
8.312.2.1 AccessMac . . . 322
8.312.2.2 AuthProt . . . 322
8.312.2.3 ControlMac . . . 322
8.312.2.4 EncryptProt . . . 322
8.312.2.5 ForwardMac . . . 322
8.312.2.6 IdleState . . . 322
8.312.2.7 KeyExchange . . . 322
8.312.2.8 MultDisc . . . 322
8.312.2.9 PhysicalLayer . . . 322
8.312.2.10 ReverseMac . . . 322
8.312.2.11 SecProt . . . 322
8.312.2.12 VirtStream . . . 322
8.313 nas_qaQmi3Gpp2TimeZone Struct Reference . . . 322
8.313.1 Detailed Description . . . 322
8.313.2 Field Documentation . . . 323
8.313.2.1 daylightSavings . . . 323
8.313.2.2 leapSeconds . . . 323
8.313.2.3 localTimeOffset . . . 323
8.314 nas_QmiNas3GppNetworkInfo Struct Reference . . . 323
8.314.1 Detailed Description . . . 323
8.314.2 Field Documentation . . . 324
8.314.2.1 Desription . . . 324
8.314.2.2 Forbidden . . . 324
8.314.2.3 InUse . . . 324
8.314.2.4 MCC . . . 324
8.314.2.5 MNC . . . 324
8.314.2.6 Preferred . . . 324

8.314.2.7 Roaming . . . 324
8.315 nas_QmiNas3GppNetworkRAT Struct Reference . . . 324
8.315.1 Detailed Description . . . 324
8.315.2 Field Documentation . . . 325
8.315.2.1 MCC . . . 325
8.315.2.2 MNC . . . 325
8.315.2.3 RAT . . . 325
8.316 nas_QmisNasPcsDigit Struct Reference . . . 325
8.316.1 Detailed Description . . . 325
8.316.2 Field Documentation . . . 325
8.316.2.1 includes_pcs_digit . . . 326
8.316.2.2 MCC . . . 326
8.316.2.3 MNC . . . 326
8.317 nas_QmisNasSlqsNasPCICellInfo Struct Reference . . . 326
8.317.1 Detailed Description . . . 326
8.317.2 Field Documentation . . . 326
8.317.2.1 cellID . . . 326
8.317.2.2 freq . . . 326
8.317.2.3 GlobalCellID . . . 326
8.317.2.4 nasQmisNasPcsDigit . . . 326
8.317.2.5 PlmnLen . . . 326
8.318 nas_QmisNasSlqsNasPCIInfo Struct Reference . . . 326
8.318.1 Detailed Description . . . 327
8.318.2 Field Documentation . . . 327
8.318.2.1 nasQmisNasSlqsNasPCICellInfo . . . 327
8.318.2.2 PCICellInfoLen . . . 327
8.318.2.3 rsrp . . . 327
8.318.2.4 rsrpRx0 . . . 327
8.318.2.5 rsrpRx1 . . . 327
8.318.2.6 rsrq . . . 328
8.318.2.7 rsrqRx0 . . . 328
8.318.2.8 rsrqRx1 . . . 328
8.319 nas_RankIndicatorTlv Struct Reference . . . 328
8.319.1 Detailed Description . . . 328
8.319.2 Field Documentation . . . 328
8.319.2.1 count0 . . . 328
8.319.2.2 count1 . . . 328
8.319.2.3 TlvPresent . . . 328
8.320 nas_RatDisabledMaskTlv Struct Reference . . . 328
8.320.1 Detailed Description . . . 328

8.320.2 Field Documentation . . . 329
8.320.2.1 ratDisabledMask . . . 329
8.320.2.2 TlvPresent . . . 329
8.321 nas_RejectReasonTlv Struct Reference . . . 329
8.321.1 Detailed Description . . . 329
8.321.2 Field Documentation . . . 329
8.321.2.1 rejectCause . . . 329
8.321.2.2 serviceDomain . . . 329
8.321.2.3 TlvPresent . . . 329
8.322 nas_RFBandInfoElements Struct Reference . . . 329
8.322.1 Detailed Description . . . 330
8.322.2 Field Documentation . . . 330
8.322.2.1 activeBandClass . . . 330
8.322.2.2 activeChannel . . . 330
8.322.2.3 radioInterface . . . 330
8.323 nas_RfBandInfoExtFormat Struct Reference . . . 330
8.323.1 Detailed Description . . . 330
8.323.2 Field Documentation . . . 331
8.323.2.1 pInstancesSize . . . 331
8.323.2.2 pRfBandInfoExtFormatParam . . . 331
8.323.2.3 TlvPresent . . . 331
8.324 nas_RfBandInfoExtFormatElements Struct Reference . . . 331
8.324.1 Detailed Description . . . 331
8.324.2 Field Documentation . . . 331
8.324.2.1 activeBand . . . 331
8.324.2.2 activeChannel . . . 331
8.324.2.3 radioInterface . . . 332
8.325 nas_RFBandInfoExtTlv Struct Reference . . . 332
8.325.1 Detailed Description . . . 332
8.325.2 Field Documentation . . . 332
8.325.2.1 activeBand . . . 333
8.325.2.2 activeChannel . . . 333
8.325.2.3 radioIf . . . 333
8.325.2.4 TlvPresent . . . 333
8.326 nas_RfBandInfoList Struct Reference . . . 333
8.326.1 Detailed Description . . . 333
8.326.2 Field Documentation . . . 333
8.326.2.1 pInstanceSize . . . 333
8.326.2.2 pRFBandInfoParam . . . 333
8.327 nas_RFBandInfoTlv Struct Reference . . . 333

8.327.1 Detailed Description . . . 333
8.327.2 Field Documentation . . . 334
8.327.2.1 activeBand . . . 334
8.327.2.2 activeChannel . . . 334
8.327.2.3 radioIf . . . 334
8.327.2.4 TlvPresent . . . 334
8.328 nas_RfBandwidthInfo Struct Reference . . . 334
8.328.1 Detailed Description . . . 335
8.328.2 Field Documentation . . . 335
8.328.2.1 pInstancesSize . . . 335
8.328.2.2 pRfBandwidthInfoParam . . . 335
8.328.2.3 TlvPresent . . . 335
8.329 nas_RfBandwidthInfoElements Struct Reference . . . 335
8.329.1 Detailed Description . . . 335
8.329.2 Field Documentation . . . 336
8.329.2.1 bandwidth . . . 336
8.329.2.2 radioInterface . . . 336
8.330 nas_RFBandwidthInfoTlv Struct Reference . . . 336
8.330.1 Detailed Description . . . 336
8.330.2 Field Documentation . . . 337
8.330.2.1 bandwidth . . . 337
8.330.2.2 radioIf . . . 337
8.330.2.3 TlvPresent . . . 337
8.331 nas_RfDedicatedBandInfo Struct Reference . . . 337
8.331.1 Detailed Description . . . 338
8.331.2 Field Documentation . . . 338
8.331.2.1 pInstancesSize . . . 338
8.331.2.2 pRfDedicatedBandInfoParam . . . 338
8.331.2.3 TlvPresent . . . 338
8.332 nas_RfDedicatedBandInfoElements Struct Reference . . . 338
8.332.1 Detailed Description . . . 338
8.332.2 Field Documentation . . . 339
8.332.2.1 dedicatedBand . . . 339
8.332.2.2 radioInterface . . . 339
8.333 nas_RFDedicatedBandInfoTlv Struct Reference . . . 339
8.333.1 Detailed Description . . . 339
8.333.2 Field Documentation . . . 339
8.333.2.1 dedicatedBand . . . 339
8.333.2.2 radioIf . . . 340
8.333.2.3 TlvPresent . . . 340

8.334 nas_RFInfoTlv Struct Reference . . . 340
8.334.1 Detailed Description . . . 340
8.334.2 Field Documentation . . . 340
8.334.2.1 activeBandClass . . . 340
8.334.2.2 activeChannel . . . 340
8.334.2.3 radioInterface . . . 341
8.334.2.4 radioInterfaceSize . . . 341
8.334.2.5 TlvPresent . . . 341
8.335 nas_roamIndList Struct Reference . . . 341
8.335.1 Detailed Description . . . 341
8.335.2 Field Documentation . . . 341
8.335.2.1 numInstances . . . 341
8.335.2.2 radioInterface . . . 341
8.335.2.3 roamIndicator . . . 341
8.336 nas_RoamPrefTlv Struct Reference . . . 342
8.336.1 Detailed Description . . . 342
8.336.2 Field Documentation . . . 342
8.336.2.1 RoamPref . . . 342
8.336.2.2 TlvPresent . . . 342
8.337 nas_RSRPThresh Struct Reference . . . 342
8.337.1 Detailed Description . . . 342
8.337.2 Field Documentation . . . 343
8.337.2.1 pRSRPThresList . . . 343
8.337.2.2 RSRPThresListLen . . . 343
8.338 nas_rsrqInformation Struct Reference . . . 343
8.338.1 Detailed Description . . . 343
8.338.2 Field Documentation . . . 343
8.338.2.1 radioIf . . . 343
8.338.2.2 rsrq . . . 343
8.339 nas_RSRQThresh Struct Reference . . . 343
8.339.1 Detailed Description . . . 344
8.339.2 Field Documentation . . . 344
8.339.2.1 pRSRQThresList . . . 344
8.339.2.2 RSRQThresListLen . . . 344
8.340 nas_RSSIThresh Struct Reference . . . 344
8.340.1 Detailed Description . . . 344
8.340.2 Field Documentation . . . 345
8.340.2.1 pRSSIThresList . . . 345
8.340.2.2 RSSIThresListLen . . . 345
8.341 nas_rxInfo Struct Reference . . . 345

8.341.1 Detailed Description . . . 345
8.341.2 Field Documentation . . . 346
8.341.2.1 ecio . . . 346
8.341.2.2 isRadioTuned . . . 346
8.341.2.3 phase . . . 346
8.341.2.4 rscp . . . 346
8.341.2.5 rsrp . . . 346
8.341.2.6 rxPower . . . 346
8.342 nas_RxSigInfo Struct Reference . . . 346
8.342.1 Detailed Description . . . 346
8.342.2 Field Documentation . . . 347
8.342.2.1 isRadioTuned . . . 347
8.342.2.2 rsrp . . . 347
8.342.2.3 rxChainIndex . . . 347
8.342.2.4 rxPower . . . 347
8.343 nas_rxSignalStrengthListElement Struct Reference . . . 347
8.343.1 Detailed Description . . . 347
8.343.2 Field Documentation . . . 348
8.343.2.1 radioIf . . . 348
8.343.2.2 rxSignalStrength . . . 348
8.344 nas_SccRxInfo Struct Reference . . . 348
8.344.1 Detailed Description . . . 348
8.344.2 Field Documentation . . . 349
8.344.2.1 numInstances . . . 349
8.344.2.2 rsrq . . . 349
8.344.2.3 sigInfo . . . 349
8.344.2.4 snr . . . 349
8.344.2.5 TlvPresent . . . 349
8.345 nas_serviceProviderName Struct Reference . . . 349
8.345.1 Detailed Description . . . 349
8.345.2 Field Documentation . . . 349
8.345.2.1 displayCondition . . . 349
8.345.2.2 spn . . . 349
8.345.2.3 spnLength . . . 349
8.346 nas_servSystem Struct Reference . . . 350
8.346.1 Detailed Description . . . 350
8.346.2 Field Documentation . . . 351
8.346.2.1 csAttachState . . . 351
8.346.2.2 numRadioInterfaces . . . 351
8.346.2.3 psAttachState . . . 351

8.346.2.4 radioInterface . . . 351
8.346.2.5 regState . . . 351
8.346.2.6 selNetwork . . . 351
8.347 nas_sidNid Struct Reference . . . 351
8.347.1 Detailed Description . . . 351
8.347.2 Field Documentation . . . 351
8.347.2.1 nid . . . 351
8.347.2.2 sid . . . 351
8.348 nas_SignalQualityTlv Struct Reference . . . 352
8.348.1 Detailed Description . . . 352
8.348.2 Field Documentation . . . 352
8.348.2.1 signal_quality . . . 352
8.348.2.2 signal_quality_len . . . 352
8.348.2.3 TlvPresent . . . 352
8.349 nas_SignalStrengthTlv Struct Reference . . . 352
8.349.1 Detailed Description . . . 352
8.349.2 Field Documentation . . . 353
8.349.2.1 radioInterface . . . 353
8.349.2.2 signalStrength . . . 353
8.349.2.3 TlvPresent . . . 353
8.350 nas_SignalStrngthTlv Struct Reference . . . 353
8.350.1 Detailed Description . . . 353
8.350.2 Field Documentation . . . 353
8.350.2.1 signal_strength . . . 353
8.350.2.2 signal_strength_len . . . 353
8.350.2.3 TlvPresent . . . 353
8.351 nas_SimRejInfoTlv Struct Reference . . . 353
8.351.1 Detailed Description . . . 354
8.351.2 Field Documentation . . . 354
8.351.2.1 SimRejInfo . . . 354
8.351.2.2 TLVPresent . . . 354
8.352 nas_SLQSSignalStrengthsIndReq Struct Reference . . . 354
8.352.1 Detailed Description . . . 354
8.352.2 Field Documentation . . . 355
8.352.2.1 ecioDelta . . . 356
8.352.2.2 ecioThresholdList . . . 356
8.352.2.3 ecioThresholdListLen . . . 356
8.352.2.4 ioDelta . . . 356
8.352.2.5 lteRsrpDelta . . . 356
8.352.2.6 lteSnrDelta . . . 356

8.352.2.7 rsrqDelta . . . 356
8.352.2.8 rxSignalStrengthDelta . . . 356
8.352.2.9 sinrDelta . . . 356
8.352.2.10 sinrThresholdList . . . 356
8.352.2.11 sinrThresholdListLen . . . 356
8.353 nas_SLQSSignalStrengthsInformation Struct Reference . . . 356
8.353.1 Detailed Description . . . 356
8.353.2 Field Documentation . . . 357
8.353.2.1 ecioInfo . . . 357
8.353.2.2 errorRateInfo . . . 357
8.353.2.3 io . . . 357
8.353.2.4 lteRsrpinfo . . . 357
8.353.2.5 lteSnrinfo . . . 357
8.353.2.6 rsrqInfo . . . 357
8.353.2.7 rxSignalStrengthInfo . . . 357
8.353.2.8 sinr . . . 357
8.354 nas_SLQSSignalStrengthsTlv Struct Reference . . . 357
8.354.1 Detailed Description . . . 357
8.354.2 Field Documentation . . . 358
8.354.2.1 sSLQSSignalStrengthsInfo . . . 358
8.354.2.2 TlvPresent . . . 358
8.355 nas_SrvDomainPrefTlv Struct Reference . . . 358
8.355.1 Detailed Description . . . 358
8.355.2 Field Documentation . . . 358
8.355.2.1 SrvDomainPref . . . 358
8.355.2.2 TlvPresent . . . 358
8.356 nas_SrvRegRestrictionTlv Struct Reference . . . 358
8.356.1 Detailed Description . . . 359
8.356.2 Field Documentation . . . 359
8.356.2.1 SrvRegRestriction . . . 359
8.356.2.2 TLVPresent . . . 359
8.357 nas_SrvRegRestricTlv Struct Reference . . . 359
8.357.1 Detailed Description . . . 359
8.357.2 Field Documentation . . . 359
8.357.2.1 SrvRegRestriction . . . 359
8.357.2.2 TlvPresent . . . 360
8.358 nas_SrvStatusInfo Struct Reference . . . 360
8.358.1 Detailed Description . . . 360
8.358.2 Field Documentation . . . 360
8.358.2.1 isPrefDataPath . . . 360

8.358.2.2 srvStatus . . . 360
8.359 nas_sysInfoCommon Struct Reference . . . 360
8.359.1 Detailed Description . . . 361
8.359.2 Field Documentation . . . 362
8.359.2.1 isSysForbidden . . . 362
8.359.2.2 isSysForbiddenValid . . . 362
8.359.2.3 roamStatus . . . 362
8.359.2.4 roamStatusValid . . . 362
8.359.2.5 srvCapability . . . 362
8.359.2.6 srvCapabilityValid . . . 362
8.359.2.7 srvDomain . . . 362
8.359.2.8 srvDomainValid . . . 362
8.360 nas_TDSCDMABandPrefTlv Struct Reference . . . 363
8.360.1 Detailed Description . . . 363
8.360.2 Field Documentation . . . 363
8.360.2.1 TdscdmaBandPref . . . 363
8.360.2.2 TlvPresent . . . 363
8.361 nas_TDSCDMAECIOThresh Struct Reference . . . 363
8.361.1 Detailed Description . . . 363
8.361.2 Field Documentation . . . 364
8.361.2.1 pTDSCDMAECIOThreshList . . . 364
8.361.2.2 TDSCDMAECIOThreshListLen . . . 364
8.362 nas_TDSCDMARSCPThresh Struct Reference . . . 364
8.362.1 Detailed Description . . . 364
8.362.2 Field Documentation . . . 364
8.362.2.1 pTDSCDMARSCPThreshList . . . 364
8.362.2.2 TDSCDMARSCPThreshListLen . . . 364
8.363 nas_TDSCDMARSSIThresh Struct Reference . . . 364
8.363.1 Detailed Description . . . 364
8.363.2 Field Documentation . . . 365
8.363.2.1 pTDSCDMARSSIThreshList . . . 365
8.363.2.2 TDSCDMARSSIThreshListLen . . . 365
8.364 nas_TDSCDMASINRCONFThresh Struct Reference . . . 365
8.364.1 Detailed Description . . . 365
8.364.2 Field Documentation . . . 365
8.364.2.1 pTDSCDMASINRCONFThreshList . . . 365
8.364.2.2 TDSCDMASINRCONFThreshListLen . . . 365
8.365 nas_TDSCDMASINRThresh Struct Reference . . . 365
8.365.1 Detailed Description . . . 365
8.365.2 Field Documentation . . . 366

8.365.2.1 pTDSCDMASINRThreshList . . . 366
8.365.2.2 TDSCDMASINRThreshListLen . . . 366
8.366 nas_timeInfo Struct Reference . . . 366
8.366.1 Detailed Description . . . 366
8.366.2 Field Documentation . . . 367
8.366.2.1 day . . . 367
8.366.2.2 dayLtSavingAdj . . . 367
8.366.2.3 dayOfWeek . . . 367
8.366.2.4 hour . . . 367
8.366.2.5 minute . . . 367
8.366.2.6 month . . . 367
8.366.2.7 radioInterface . . . 367
8.366.2.8 second . . . 368
8.366.2.9 timeZone . . . 368
8.366.2.10 TlvPresent . . . 368
8.366.2.11 year . . . 368
8.367 nas_trueIMSI Struct Reference . . . 368
8.367.1 Detailed Description . . . 368
8.367.2 Field Documentation . . . 368
8.367.2.1 imsiT1112 . . . 368
8.367.2.2 imsiTaddrNum . . . 368
8.367.2.3 imsiTS1 . . . 368
8.367.2.4 imsiTS2 . . . 368
8.367.2.5 mccT . . . 368
8.368 nas_txInfo Struct Reference . . . 369
8.368.1 Detailed Description . . . 369
8.368.2 Field Documentation . . . 369
8.368.2.1 isInTraffic . . . 369
8.368.2.2 txPower . . . 369
8.369 nas_UMTSExtInfo Struct Reference . . . 369
8.369.1 Detailed Description . . . 369
8.369.2 Field Documentation . . . 370
8.369.2.1 cellId . . . 370
8.369.2.2 ecio . . . 370
8.369.2.3 geranInst . . . 371
8.369.2.4 geranInstElement . . . 371
8.369.2.5 lac . . . 371
8.369.2.6 plmn . . . 371
8.369.2.7 psc . . . 371
8.369.2.8 rscp . . . 371

8.369.2.9 squal . . . 371
8.369.2.10 srxLvl . . . 371
8.369.2.11 tlvPresent . . . 371
8.369.2.12 uarfcn . . . 371
8.369.2.13 umtsInst . . . 371
8.369.2.14 umtsInstElement . . . 371
8.370 nas_UMTSInfo Struct Reference . . . 371
8.370.1 Detailed Description . . . 371
8.370.2 Field Documentation . . . 372
8.370.2.1 cellID . . . 372
8.370.2.2 ecio . . . 372
8.370.2.3 geranInst . . . 372
8.370.2.4 GeranInstInfo . . . 372
8.370.2.5 lac . . . 372
8.370.2.6 plmn . . . 372
8.370.2.7 psc . . . 373
8.370.2.8 rscp . . . 373
8.370.2.9 uarfcn . . . 373
8.370.2.10 umtsInst . . . 373
8.370.2.11 UMTSInstInfo . . . 373
8.371 nas_umtsInstArr Struct Reference . . . 373
8.371.1 Detailed Description . . . 373
8.371.2 Field Documentation . . . 374
8.371.2.1 umtsEcio . . . 374
8.371.2.2 umtsPsc . . . 374
8.371.2.3 umtsRank . . . 374
8.371.2.4 umtsRscp . . . 374
8.371.2.5 umtsSet . . . 374
8.371.2.6 umtsSqual . . . 374
8.371.2.7 umtsSrxLvl . . . 374
8.371.2.8 umtsUarfcn . . . 374
8.372 nas_UMTSinstInfo Struct Reference . . . 374
8.372.1 Detailed Description . . . 374
8.372.2 Field Documentation . . . 375
8.372.2.1 umtsEcio . . . 375
8.372.2.2 umtsPsc . . . 375
8.372.2.3 umtsRscp . . . 375
8.372.2.4 umtsUarfcn . . . 375
8.373 nas_umtsLTENbrCell Struct Reference . . . 375
8.373.1 Detailed Description . . . 375

8.373.2 Field Documentation . . . 375
8.373.2.1 cellIsTDD . . . 375
8.373.2.2 earfcn . . . 376
8.373.2.3 pci . . . 376
8.373.2.4 rsrp . . . 376
8.373.2.5 rsrq . . . 376
8.373.2.6 srxlev . . . 376
8.374 nas_UniversalTime Struct Reference . . . 376
8.374.1 Detailed Description . . . 376
8.374.2 Field Documentation . . . 377
8.374.2.1 day . . . 377
8.374.2.2 dayOfWeek . . . 377
8.374.2.3 hour . . . 377
8.374.2.4 minute . . . 377
8.374.2.5 month . . . 377
8.374.2.6 second . . . 377
8.374.2.7 year . . . 377
8.375 nas_UsageSettingTlv Struct Reference . . . 377
8.375.1 Detailed Description . . . 377
8.375.2 Field Documentation . . . 377
8.375.2.1 TlvPresent . . . 377
8.375.2.2 UsageSetting . . . 377
8.376 nas_VoiceDomainPrefTlv Struct Reference . . . 377
8.376.1 Detailed Description . . . 378
8.376.2 Field Documentation . . . 378
8.376.2.1 TlvPresent . . . 378
8.376.2.2 VoiceDomainPref . . . 378
8.377 nas_WCDMAARFCNTlv Struct Reference . . . 378
8.377.1 Detailed Description . . . 378
8.377.2 Field Documentation . . . 378
8.377.2.1 numWcdmaArfcn . . . 378
8.377.2.2 wcdmaArfcn . . . 378
8.378 nas_wcdmaCellInfo Struct Reference . . . 379
8.378.1 Detailed Description . . . 379
8.378.2 Field Documentation . . . 379
8.378.2.1 cpich_ecno . . . 379
8.378.2.2 cpich_rscp . . . 379
8.378.2.3 psc . . . 379
8.378.2.4 srxlev . . . 379
8.379 nas_WCDMACellInfoExt Struct Reference . . . 379

8.379.1 Detailed Description . . . 380
8.379.2 Field Documentation . . . 380
8.379.2.1 TlvPresent . . . 380
8.379.2.2 wAgc . . . 380
8.379.2.3 wDlBler . . . 380
8.379.2.4 wTxAgc . . . 380
8.380 nas_WCDMAECIOThresh Struct Reference . . . 380
8.380.1 Detailed Description . . . 380
8.380.2 Field Documentation . . . 380
8.380.2.1 pWCDMAECIOThreshList . . . 381
8.380.2.2 WCDMAECIOThreshListLen . . . 381
8.381 nas_WCDMAInfoLTENeighborCell Struct Reference . . . 381
8.381.1 Detailed Description . . . 381
8.381.2 Field Documentation . . . 381
8.381.2.1 UMTSLTENbrCell . . . 381
8.381.2.2 umtsLTENbrCellLen . . . 381
8.381.2.3 wcdmaRRCState . . . 381
8.382 nas_WCDMARSSIThresh Struct Reference . . . 381
8.382.1 Detailed Description . . . 382
8.382.2 Field Documentation . . . 382
8.382.2.1 pWCDMARSSIThreshList . . . 382
8.382.2.2 WCDMARSSIThreshListLen . . . 382
8.383 nas_WCDMASysInfo Struct Reference . . . 382
8.383.1 Detailed Description . . . 382
8.383.2 Field Documentation . . . 385
8.383.2.1 cellId . . . 385
8.383.2.2 cellIdValid . . . 385
8.383.2.3 hsCallStatus . . . 385
8.383.2.4 hsCallStatusValid . . . 385
8.383.2.5 hsInd . . . 385
8.383.2.6 hsIndValid . . . 385
8.383.2.7 lac . . . 385
8.383.2.8 lacValid . . . 385
8.383.2.9 MCC . . . 385
8.383.2.10 MNC . . . 385
8.383.2.11 networkIdValid . . . 385
8.383.2.12 psc . . . 385
8.383.2.13 pscValid . . . 385
8.383.2.14 regRejectInfoValid . . . 385
8.383.2.15 rejCause . . . 385

8.383.2.16 rejectSrvDomain . . . 385
8.383.2.17 sysInfoWCDMA . . . 385
8.384 nas_wcdmaUARFCN Struct Reference . . . 385
8.384.1 Detailed Description . . . 386
8.384.2 Field Documentation . . . 386
8.384.2.1 status . . . 386
8.384.2.2 uarfcn . . . 386
8.385 NASAcqOrderPrefTlv Struct Reference . . . 386
8.385.1 Detailed Description . . . 386
8.385.2 Field Documentation . . . 386
8.385.2.1 AcqOrderLen . . . 386
8.385.2.2 AcqOrderPref . . . 386
8.385.2.3 TlvPresent . . . 386
8.386 NASBandPreferenceTlv Struct Reference . . . 386
8.386.1 Detailed Description . . . 387
8.386.2 Field Documentation . . . 387
8.386.2.1 band_pref . . . 387
8.386.2.2 TlvPresent . . . 387
8.387 NASCiotAcqOrderPrefTlv Struct Reference . . . 387
8.387.1 Detailed Description . . . 387
8.387.2 Field Documentation . . . 387
8.387.2.1 CiotAcqOrderLen . . . 387
8.387.2.2 CiotAcqOrderPref . . . 387
8.387.2.3 TlvPresent . . . 387
8.388 NASCiotLteOpModePrefTlv Struct Reference . . . 387
8.388.1 Detailed Description . . . 387
8.388.2 Field Documentation . . . 388
8.388.2.1 CiotLteOpModePref . . . 388
8.388.2.2 TlvPresent . . . 388
8.389 NASEmergencyModeTlv Struct Reference . . . 388
8.389.1 Detailed Description . . . 388
8.389.2 Field Documentation . . . 388
8.389.2.1 EmerMode . . . 388
8.389.2.2 TlvPresent . . . 388
8.390 NasGetLTECphyCaInfo Struct Reference . . . 388
8.390.1 Detailed Description . . . 388
8.390.2 Field Documentation . . . 389
8.390.2.1 PhyCaAggPcellInfo . . . 389
8.390.2.2 PhyCaAggScellArray . . . 389
8.390.2.3 PhyCaAggScellDlBw . . . 389

8.390.2.4 PhyCaAggScellIndex . . . 389
8.390.2.5 PhyCaAggScellIndType . . . 389
8.390.2.6 PhyCaAggScellInfo . . . 389
8.391 NASGWAcqOrderPrefTlv Struct Reference . . . 389
8.391.1 Detailed Description . . . 389
8.391.2 Field Documentation . . . 390
8.391.2.1 GWAcqOrderPref . . . 390
8.391.2.2 TlvPresent . . . 390
8.392 NASLTEBandPreferenceTlv Struct Reference . . . 390
8.392.1 Detailed Description . . . 390
8.392.2 Field Documentation . . . 390
8.392.2.1 LTEBandPref . . . 390
8.392.2.2 TlvPresent . . . 390
8.393 NASLteM1BandPrefTlv Struct Reference . . . 390
8.393.1 Detailed Description . . . 390
8.393.2 Field Documentation . . . 390
8.393.2.1 LteM1BandPref . . . 391
8.393.2.2 TlvPresent . . . 391
8.394 NASLteNasReleaseInfoTlv Struct Reference . . . 391
8.394.1 Detailed Description . . . 391
8.394.2 Field Documentation . . . 391
8.394.2.1 nas_major . . . 391
8.394.2.2 nas_minor . . . 391
8.394.2.3 nas_release . . . 391
8.394.2.4 TlvPresent . . . 391
8.395 NASLteNB1BandPrefTlv Struct Reference . . . 391
8.395.1 Detailed Description . . . 391
8.395.2 Field Documentation . . . 392
8.395.2.1 LteNB1BandPref . . . 392
8.395.2.2 TlvPresent . . . 392
8.396 NASModePreferenceTlv Struct Reference . . . 392
8.396.1 Detailed Description . . . 392
8.396.2 Field Documentation . . . 392
8.396.2.1 ModePref . . . 392
8.396.2.2 TlvPresent . . . 392
8.397 NASNetSelPreferenceTlv Struct Reference . . . 392
8.397.1 Detailed Description . . . 392
8.397.2 Field Documentation . . . 392
8.397.2.1 NetSelPref . . . 392
8.397.2.2 TlvPresent . . . 393

8.398 NASNr5gBandPrefTlv Struct Reference . . . 393
8.398.1 Detailed Description . . . 393
8.398.2 Field Documentation . . . 393
8.398.2.1 Nr5gBandPrefbits1_64 . . . 393
8.398.2.2 Nr5gBandPrefbits_129_192 . . . 393
8.398.2.3 Nr5gBandPrefbits_193_256 . . . 393
8.398.2.4 Nr5gBandPrefbits_65_128 . . . 393
8.398.2.5 TlvPresent . . . 393
8.399 NASOTAMessageTlv Struct Reference . . . 393
8.399.1 Detailed Description . . . 393
8.399.2 Field Documentation . . . 394
8.399.2.1 data_buf . . . 394
8.399.2.2 data_len . . . 394
8.399.2.3 message_type . . . 394
8.399.2.4 TlvPresent . . . 394
8.400 NASPhyCaAggPcellInfo Struct Reference . . . 394
8.400.1 Detailed Description . . . 394
8.400.2 Field Documentation . . . 395
8.400.2.1 dl_bw_value . . . 395
8.400.2.2 freq . . . 395
8.400.2.3 iLTEbandValue . . . 395
8.400.2.4 pci . . . 395
8.400.2.5 TlvPresent . . . 395
8.401 NASPhyCaAggScellArray Struct Reference . . . 395
8.401.1 Detailed Description . . . 395
8.401.2 Field Documentation . . . 396
8.401.2.1 band . . . 396
8.401.2.2 cphy_ca_dl_bandwidth . . . 396
8.401.2.3 cphy_scell_info_list_len . . . 396
8.401.2.4 freq . . . 396
8.401.2.5 pci . . . 396
8.401.2.6 scell_idx . . . 396
8.401.2.7 scell_state . . . 396
8.401.2.8 TlvPresent . . . 396
8.402 NASPhyCaAggScellDlBw Struct Reference . . . 396
8.402.1 Detailed Description . . . 397
8.402.2 Field Documentation . . . 397
8.402.2.1 dl_bw_value . . . 397
8.402.2.2 TlvPresent . . . 397
8.403 NASPhyCaAggScellIndex Struct Reference . . . 397

8.403.1 Detailed Description . . . 397
8.403.2 Field Documentation . . . 397
8.403.2.1 scell_idx . . . 397
8.403.2.2 TlvPresent . . . 397
8.404 NASPhyCaAggScellIndType Struct Reference . . . 397
8.404.1 Detailed Description . . . 398
8.404.2 Field Documentation . . . 398
8.404.2.1 freq . . . 398
8.404.2.2 pci . . . 398
8.404.2.3 scell_state . . . 398
8.404.2.4 TlvPresent . . . 398
8.405 NASPhyCaAggScellInfo Struct Reference . . . 398
8.405.1 Detailed Description . . . 398
8.405.2 Field Documentation . . . 399
8.405.2.1 dl_bw_value . . . 399
8.405.2.2 freq . . . 399
8.405.2.3 iLTEbandValue . . . 399
8.405.2.4 pci . . . 399
8.405.2.5 scell_state . . . 399
8.405.2.6 TlvPresent . . . 399
8.406 NASPRLPreferenceTlv Struct Reference . . . 399
8.406.1 Detailed Description . . . 399
8.406.2 Field Documentation . . . 399
8.406.2.1 PRLPref . . . 400
8.406.2.2 TlvPresent . . . 400
8.407 NASQmiCbkNasSwiOTAMessageInd Struct Reference . . . 400
8.407.1 Detailed Description . . . 400
8.407.2 Field Documentation . . . 400
8.407.2.1 nasRelInfoTlv . . . 400
8.407.2.2 otaMsgTlv . . . 400
8.407.2.3 timeTlv . . . 400
8.408 NASQmiCbkNasSystemSelPrefInd Struct Reference . . . 400
8.408.1 Detailed Description . . . 401
8.408.2 Field Documentation . . . 402
8.408.2.1 AOPTlv . . . 402
8.408.2.2 BPTlv . . . 402
8.408.2.3 CiotAOPTlv . . . 402
8.408.2.4 CiotOpMPTlv . . . 402
8.408.2.5 EMTlv . . . 402
8.408.2.6 GWAOPTlv . . . 402

8.408.2.7 LBPTlv . . . 402
8.408.2.8 M1BandPTlv . . . 402
8.408.2.9 MPTlv . . . 402
8.408.2.10 NB1BandPTlv . . . 402
8.408.2.11 NR5gBandPTlv . . . 402
8.408.2.12 NSPTlv . . . 402
8.408.2.13 PRLPTlv . . . 403
8.408.2.14 RatDMTlv . . . 403
8.408.2.15 RPTlv . . . 403
8.408.2.16 SDPTlv . . . 403
8.409 NASRatDisabledMaskTlv Struct Reference . . . 403
8.409.1 Detailed Description . . . 403
8.409.2 Field Documentation . . . 403
8.409.2.1 RatDisabledMask . . . 403
8.409.2.2 TlvPresent . . . 403
8.410 NASRoamPreferenceTlv Struct Reference . . . 403
8.410.1 Detailed Description . . . 403
8.410.2 Field Documentation . . . 403
8.410.2.1 RoamPref . . . 403
8.410.2.2 TlvPresent . . . 403
8.411 NASServDomainPrefTlv Struct Reference . . . 404
8.411.1 Detailed Description . . . 404
8.411.2 Field Documentation . . . 404
8.411.2.1 SrvDomainPref . . . 404
8.411.2.2 TlvPresent . . . 404
8.412 NASServingSystemInfo Struct Reference . . . 404
8.412.1 Detailed Description . . . 404
8.412.2 Field Documentation . . . 405
8.412.2.1 csAttachState . . . 405
8.412.2.2 hdrPersonality . . . 405
8.412.2.3 psAttachState . . . 405
8.412.2.4 radioInterfaceList . . . 405
8.412.2.5 radioInterfaceNo . . . 405
8.412.2.6 registrationState . . . 405
8.412.2.7 selectedNetwork . . . 405
8.413 NASTimeInfoTlv Struct Reference . . . 405
8.413.1 Detailed Description . . . 406
8.413.2 Field Documentation . . . 406
8.413.2.1 time . . . 406
8.413.2.2 TlvPresent . . . 406

8.414 newMTMessageTlv Struct Reference . . . 406
8.414.1 Detailed Description . . . 406
8.414.2 Field Documentation . . . 406
8.414.2.1 MTMessageInfo . . . 406
8.414.2.2 TlvPresent . . . 406
8.415 pack_audio_SLQSGetAudioPathConfig_t Struct Reference . . . 406
8.415.1 Detailed Description . . . 407
8.415.2 Field Documentation . . . 407
8.415.2.1 Item . . . 407
8.415.2.2 Profile . . . 407
8.416 pack_audio_SLQSGetAudioProfile_t Struct Reference . . . 407
8.416.1 Detailed Description . . . 407
8.416.2 Field Documentation . . . 408
8.416.2.1 Generator . . . 408
8.417 pack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference . . . 408
8.417.1 Detailed Description . . . 408
8.417.2 Field Documentation . . . 408
8.417.2.1 Generator . . . 408
8.417.2.2 Item . . . 408
8.417.2.3 Profile . . . 408
8.417.2.4 Volume . . . 408
8.418 pack_audio_SLQSSetAudioPathConfig_t Struct Reference . . . 408
8.418.1 Detailed Description . . . 409
8.418.2 Field Documentation . . . 410
8.418.2.1 pCodecSTGain . . . 410
8.418.2.2 pDTMFTXGain . . . 410
8.418.2.3 pECMode . . . 410
8.418.2.4 pNSEnable . . . 410
8.418.2.5 Profile . . . 410
8.418.2.6 pRXAGCList . . . 410
8.418.2.7 pRXAVCAGCSwitch . . . 410
8.418.2.8 pRXAVCList . . . 410
8.418.2.9 pRXPCMIIRFltr . . . 410
8.418.2.10 pTXAGCList . . . 410
8.418.2.11 pTXAVCSwitch . . . 410
8.418.2.12 pTXGain . . . 410
8.418.2.13 pTXPCMIIRFltr . . . 410
8.419 pack_audio_SLQSSetAudioProfile_t Struct Reference . . . 410
8.419.1 Detailed Description . . . 411
8.419.2 Field Documentation . . . 411

8.419.2.1 EarMute . . . 411
8.419.2.2 Generator . . . 411
8.419.2.3 MicMute . . . 411
8.419.2.4 Profile . . . 411
8.419.2.5 Volume . . . 411
8.420 pack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference . . . 412
8.420.1 Detailed Description . . . 412
8.420.2 Field Documentation . . . 412
8.420.2.1 Generator . . . 412
8.420.2.2 Item . . . 412
8.420.2.3 Profile . . . 412
8.420.2.4 Volume . . . 412
8.420.2.5 VolValue . . . 412
8.421 pack_cat_CATSendEnvelopeCommand_t Struct Reference . . . 412
8.421.1 Detailed Description . . . 413
8.421.2 Field Documentation . . . 413
8.421.2.1 cmdID . . . 413
8.421.2.2 dataLen . . . 413
8.421.2.3 pData . . . 413
8.422 pack_cat_CATSendTerminalResponse_t Struct Reference . . . 413
8.422.1 Detailed Description . . . 413
8.422.2 Field Documentation . . . 414
8.422.2.1 dataLen . . . 414
8.422.2.2 pData . . . 414
8.422.2.3 refID . . . 414
8.423 pack_cat_SetCATEventCallback_t Struct Reference . . . 414
8.423.1 Detailed Description . . . 414
8.423.2 Field Documentation . . . 414
8.423.2.1 eventMask . . . 414
8.424 pack_ctl_GetClientId_t Struct Reference . . . 415
8.424.1 Detailed Description . . . 415
8.424.2 Field Documentation . . . 415
8.424.2.1 serviceType . . . 415
8.425 pack_ctl_ReleaseClientId_t Struct Reference . . . 415
8.425.1 Detailed Description . . . 415
8.425.2 Field Documentation . . . 415
8.425.2.1 clientId . . . 415
8.425.2.2 serviceType . . . 415
8.426 pack_dms_ActivateAutomatic_t Struct Reference . . . 415
8.426.1 Detailed Description . . . 416

8.426.2 Field Documentation . . . 416
8.426.2.1 actCode . . . 416
8.427 pack_dms_GetCustFeaturesV2_t Struct Reference . . . 416
8.427.1 Detailed Description . . . 416
8.427.2 Field Documentation . . . 416
8.427.2.1 cust_id . . . 416
8.427.2.2 list_type . . . 416
8.427.2.3 Tlvresult . . . 416
8.428 pack_dms_ResetToFactoryDefaults_t Struct Reference . . . 416
8.428.1 Detailed Description . . . 417
8.428.2 Field Documentation . . . 417
8.428.2.1 spc . . . 417
8.429 pack_dms_SetActivationStatusCallback_t Struct Reference . . . 417
8.429.1 Detailed Description . . . 417
8.429.2 Field Documentation . . . 417
8.429.2.1 activationState . . . 417
8.430 pack_dms_SetCrashAction_t Struct Reference . . . 417
8.430.1 Detailed Description . . . 417
8.430.2 Field Documentation . . . 418
8.430.2.1 crashAction . . . 418
8.431 pack_dms_SetCustFeature_t Struct Reference . . . 418
8.431.1 Detailed Description . . . 418
8.431.2 Field Documentation . . . 420
8.431.2.1 DHCPRelayEnabled . . . 420
8.431.2.2 DisableIMSI . . . 420
8.431.2.3 GpsEnable . . . 420
8.431.2.4 GPSLPM . . . 420
8.431.2.5 GPSSel . . . 420
8.431.2.6 IPFamSupport . . . 420
8.431.2.7 IsVoiceEnabled . . . 420
8.431.2.8 RMAutoConnect . . . 420
8.431.2.9 SMSSupport . . . 420
8.432 pack_dms_SetCustFeaturesV2_t Struct Reference . . . 420
8.432.1 Detailed Description . . . 420
8.432.2 Field Documentation . . . 421
8.432.2.1 cust_id . . . 421
8.432.2.2 cust_value . . . 421
8.432.2.3 Tlvresult . . . 421
8.432.2.4 value_length . . . 421
8.433 pack_dms_SetEventReport_t Struct Reference . . . 421

8.433.1 Detailed Description . . . 421
8.433.2 Field Documentation . . . 421
8.433.2.1 mode . . . 421
8.434 pack_dms_SetIndicationRegister_t Struct Reference . . . 421
8.434.1 Detailed Description . . . 421
8.434.2 Field Documentation . . . 422
8.434.2.1 PSMCfgChangeInfo . . . 422
8.434.2.2 PSMStatus . . . 422
8.434.2.3 RptIMSCapability . . . 422
8.435 pack_dms_SetPower_t Struct Reference . . . 422
8.435.1 Detailed Description . . . 422
8.435.2 Field Documentation . . . 423
8.435.2.1 mode . . . 423
8.435.2.2 Tlvresult . . . 423
8.436 pack_dms_SetUSBComp_t Struct Reference . . . 423
8.436.1 Detailed Description . . . 423
8.436.2 Field Documentation . . . 423
8.436.2.1 Tlvresult . . . 423
8.436.2.2 USBComp . . . 424
8.437 pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 424
8.437.1 Detailed Description . . . 424
8.437.2 Field Documentation . . . 424
8.437.2.1 resetInfoInd . . . 424
8.438 pack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference . . . 424
8.438.1 Detailed Description . . . 424
8.438.2 Field Documentation . . . 425
8.438.2.1 pActiveTimer . . . 425
8.438.2.2 pPeriodicUpdateTimer . . . 425
8.438.2.3 pPsmEnableState . . . 425
8.439 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 425
8.439.1 Detailed Description . . . 425
8.439.2 Field Documentation . . . 425
8.439.2.1 clear . . . 425
8.440 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 425
8.440.1 Detailed Description . . . 425
8.440.2 Field Documentation . . . 426
8.440.2.1 pDestSMSContent . . . 426
8.440.2.2 pDestSMSNum . . . 426
8.441 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 426
8.441.1 Detailed Description . . . 426

8.441.2 Field Documentation . . . 427
8.441.2.1 hostID . . . 427
8.441.2.2 manString . . . 427
8.441.2.3 modelString . . . 427
8.441.2.4 plasmaIDString . . . 427
8.441.2.5 swVerString . . . 427
8.442pack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 427
8.442.1 Detailed Description . . . 427
8.442.2 Field Documentation . . . 427
8.442.2.1 nameString . . . 427
8.442.2.2 versionString . . . 427
8.443pack_dms_SwiSetEventReport_t Struct Reference . . . 427
8.443.1 Detailed Description . . . 427
8.443.2 Field Documentation . . . 428
8.443.2.1 pTempReport . . . 428
8.443.2.2 pUIMStatusReport . . . 428
8.443.2.3 pVoltReport . . . 428
8.444pack_dms_SwiUimSelect_t Struct Reference . . . 428
8.444.1 Detailed Description . . . 428
8.444.2 Field Documentation . . . 428
8.444.2.1 uim_select . . . 429
8.445pack_dms_UIMChangePIN_t Struct Reference . . . 429
8.445.1 Detailed Description . . . 429
8.445.2 Field Documentation . . . 429
8.445.2.1 id . . . 429
8.445.2.2 newValue . . . 429
8.445.2.3 oldValue . . . 429
8.446pack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 429
8.446.1 Detailed Description . . . 429
8.446.2 Field Documentation . . . 430
8.446.2.1 facility . . . 430
8.447pack_dms_UIMGetICCID_t Struct Reference . . . 430
8.447.1 Detailed Description . . . 430
8.447.2 Field Documentation . . . 430
8.447.2.1 ParamPresenceMask . . . 430
8.447.2.2 Tlvresult . . . 430
8.448pack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 430
8.448.1 Detailed Description . . . 430
8.448.2 Field Documentation . . . 431
8.448.2.1 facility . . . 431

8.448.2.2 facilityCk . . . 431
8.448.2.3 facilityState . . . 431
8.449 pack_dms_UIMSetPINProtection_t Struct Reference . . . 431
8.449.1 Detailed Description . . . 431
8.449.2 Field Documentation . . . 432
8.449.2.1 bEnable . . . 432
8.449.2.2 id . . . 432
8.449.2.3 value . . . 432
8.450 pack_dms_UIMUnblockControlKey_t Struct Reference . . . 432
8.450.1 Detailed Description . . . 432
8.450.2 Field Documentation . . . 432
8.450.2.1 facility . . . 432
8.450.2.2 facilityCk . . . 432
8.451 pack_dms_UIMUnblockPIN_t Struct Reference . . . 432
8.451.1 Detailed Description . . . 432
8.451.2 Field Documentation . . . 433
8.451.2.1 id . . . 433
8.451.2.2 newPin . . . 433
8.451.2.3 pukValue . . . 433
8.452 pack_dms_UIMVerifyPIN_t Struct Reference . . . 433
8.452.1 Detailed Description . . . 433
8.452.2 Field Documentation . . . 433
8.452.2.1 id . . . 433
8.452.2.2 value . . . 433
8.453 pack_dsd_register_ui_info_changes_t Struct Reference . . . 433
8.453.1 Detailed Description . . . 434
8.453.2 Field Documentation . . . 434
8.453.2.1 pReport_ui_changes . . . 434
8.453.2.2 pSuppress_null_bearer_reason . . . 434
8.453.2.3 pSuppress_so_change . . . 434
8.454 pack_dsd_system_status_changes_t Struct Reference . . . 434
8.454.1 Detailed Description . . . 434
8.454.2 Field Documentation . . . 435
8.454.2.1 pLimit_so_mask_change_ind . . . 435
8.454.2.2 pReport_data_system_status_changes . . . 435
8.454.2.3 pReport_null_bearer_reason . . . 435
8.454.2.4 pReport_pref_tech_change_only . . . 435
8.455 pack_fms_GetImagesPreference_t Struct Reference . . . 435
8.455.1 Detailed Description . . . 435
8.455.2 Field Documentation . . . 435

8.455.2.1 TlvResult . . . 436
8.456 pack_fms_GetStoredImages_t Struct Reference . . . 436
8.456.1 Detailed Description . . . 436
8.456.2 Field Documentation . . . 436
8.456.2.1 TlvResult . . . 436
8.457 pack_fms_SetImagesPreference_t Struct Reference . . . 436
8.457.1 Detailed Description . . . 436
8.457.2 Field Documentation . . . 437
8.457.2.1 bForceDownload . . . 437
8.457.2.2 imageListSize . . . 437
8.457.2.3 modemIndex . . . 437
8.457.2.4 pImageList . . . 437
8.457.2.5 TlvResult . . . 437
8.458 pack_ims_SLQSImsConfigIndicationRegister_t Struct Reference . . . 437
8.458.1 Detailed Description . . . 437
8.458.2 Field Documentation . . . 438
8.458.2.1 pRegMgrConfigEvents . . . 438
8.458.2.2 pSIPConfigEvents . . . 438
8.458.2.3 pSMSConfigEvents . . . 438
8.458.2.4 pUserConfigEvents . . . 438
8.458.2.5 pVoIPConfigEvents . . . 438
8.459 pack_ims_SLQSSetIMSSMSConfig_t Struct Reference . . . 438
8.459.1 Detailed Description . . . 438
8.459.2 Field Documentation . . . 439
8.459.2.1 pPhoneCtxtURI . . . 439
8.459.2.2 pPhoneCtxtURILen . . . 439
8.459.2.3 pSMSFormat . . . 439
8.459.2.4 pSMSOverIPNwInd . . . 439
8.460 pack_ims_SLQSSetIMSUserConfig_t Struct Reference . . . 439
8.460.1 Detailed Description . . . 439
8.460.2 Field Documentation . . . 439
8.460.2.1 pIMSDomain . . . 439
8.460.2.2 pIMSDomainLen . . . 439
8.461 pack_ims_SLQSSetIMSVoIPConfig_t Struct Reference . . . 439
8.461.1 Detailed Description . . . 440
8.461.2 Field Documentation . . . 442
8.461.2.1 pAmrMode . . . 442
8.461.2.2 pAmrOctetAligned . . . 442
8.461.2.3 pAmrWbEnable . . . 442
8.461.2.4 pAmrWBMode . . . 442

8.461.2.5 pAmrWBOctetAligned . . . 442
8.461.2.6 pMinSessionExpiryTimer . . . 442
8.461.2.7 pRingBackTimer . . . 442
8.461.2.8 pRingingTimer . . . 442
8.461.2.9 pRTPRTCPInactTimer . . . 442
8.461.2.10 pScrAmrEnable . . . 442
8.461.2.11 pScrAmrWbEnable . . . 442
8.461.2.12 pSessionExpiryTimer . . . 442
8.462 pack_ims_SLQSSetRegMgrConfig_t Struct Reference . . . 442
8.462.1 Detailed Description . . . 442
8.462.2 Field Documentation . . . 443
8.462.2.1 pCSCFPortName . . . 443
8.462.2.2 pCSCFPortNameLen . . . 443
8.462.2.3 pIMSTestMode . . . 443
8.462.2.4 pPriCSCFPort . . . 443
8.463 pack_ims_SLQSSetSIPConfig_t Struct Reference . . . 443
8.463.1 Detailed Description . . . 443
8.463.2 Field Documentation . . . 443
8.463.2.1 pSigCompEnabled . . . 444
8.463.2.2 pSIPLocalPort . . . 444
8.463.2.3 pSubscribeTimer . . . 444
8.463.2.4 pTimerSIPReg . . . 444
8.463.2.5 pTimerT1 . . . 444
8.463.2.6 pTimerT2 . . . 444
8.463.2.7 pTimerTf . . . 444
8.464 pack_imsa_SLQSRegisterIMSAIndication_t Struct Reference . . . 444
8.464.1 Detailed Description . . . 444
8.464.2 Field Documentation . . . 445
8.464.2.1 has_PdpStatusConfig . . . 445
8.464.2.2 has_RatHandoverStatusConfig . . . 445
8.464.2.3 has_RegStatusConfig . . . 445
8.464.2.4 has_ServiceStatusConfig . . . 445
8.464.2.5 PdpStatusConfig . . . 445
8.464.2.6 RatHandoverStatusConfig . . . 445
8.464.2.7 RegStatusConfig . . . 445
8.464.2.8 ServiceStatusConfig . . . 445
8.465 pack_loc_Delete_Assist_Data_t Struct Reference . . . 445
8.465.1 Detailed Description . . . 445
8.465.2 Field Documentation . . . 446
8.465.2.1 pBdsSVInfo . . . 446

8.465.2.2 pCellDb . . . 446
8.465.2.3 pClkInfo . . . 446
8.465.2.4 pGnssData . . . 446
8.465.2.5 pSVInfo . . . 446
8.465.2.6 TlvResult . . . 446
8.466 pack_loc_EventRegister_t Struct Reference . . . 446
8.466.1 Detailed Description . . . 446
8.466.2 Field Documentation . . . 448
8.466.2.1 eventRegister . . . 448
8.466.2.2 TlvResult . . . 448
8.467 pack_loc_SetExtPowerState_t Struct Reference . . . 448
8.467.1 Detailed Description . . . 448
8.467.2 Field Documentation . . . 448
8.467.2.1 extPowerState . . . 448
8.467.2.2 TlvResult . . . 448
8.468 pack_loc_SetOperationMode_t Struct Reference . . . 448
8.468.1 Detailed Description . . . 449
8.468.2 Field Documentation . . . 449
8.468.2.1 mode . . . 449
8.468.2.2 TlvResult . . . 449
8.469 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 449
8.469.1 Detailed Description . . . 449
8.469.2 Field Documentation . . . 449
8.469.2.1 TlvResult . . . 449
8.469.2.2 xid . . . 450
8.470 pack_loc_SLQSLOCGetServer_t Struct Reference . . . 450
8.470.1 Detailed Description . . . 450
8.470.2 Field Documentation . . . 450
8.470.2.1 has_serverAddrTypeMask . . . 450
8.470.2.2 serverAddrTypeMask . . . 450
8.470.2.3 serverType . . . 450
8.471 pack_loc_SLQSLOCInjectPosition_t Struct Reference . . . 450
8.471.1 Detailed Description . . . 451
8.471.2 Field Documentation . . . 454
8.471.2.1 altitudeSrcInfo . . . 454
8.471.2.2 altitudeWrtEllipsoid . . . 454
8.471.2.3 altitudeWrtMeanSeaLevel . . . 454
8.471.2.4 has_altitudeSrcInfo . . . 454
8.471.2.5 has_altitudeWrtEllipsoid . . . 454
8.471.2.6 has_altitudeWrtMeanSeaLevel . . . 454

8.471.2.7 has_horConfidence . . . 454
8.471.2.8 has_horReliability . . . 454
8.471.2.9 has_horUncCircular . . . 454
8.471.2.10 has_latitude . . . 454
8.471.2.11 has_longitude . . . 454
8.471.2.12 has_positionSrc . . . 454
8.471.2.13 has_rawHorConfidence . . . 454
8.471.2.14 has_rawHorUncCircular . . . 454
8.471.2.15 has_timestampAge . . . 454
8.471.2.16 has_timestampUtc . . . 454
8.471.2.17 has_vertConfidence . . . 454
8.471.2.18 has_vertRelicability . . . 455
8.471.2.19 has_vertUnc . . . 455
8.471.2.20 horConfidence . . . 455
8.471.2.21 horReliability . . . 455
8.471.2.22 horUncCircular . . . 455
8.471.2.23 latitude . . . 455
8.471.2.24 longitude . . . 455
8.471.2.25 positionSrc . . . 455
8.471.2.26 rawHorConfidence . . . 455
8.471.2.27 rawHorUncCircular . . . 455
8.471.2.28 timestampAge . . . 455
8.471.2.29 timestampUtc . . . 455
8.471.2.30 vertConfidence . . . 455
8.471.2.31 vertReliability . . . 455
8.471.2.32 vertUnc . . . 455
8.472 pack_loc_SLQSLOCInjectSensorData_t Struct Reference . . . 455
8.472.1 Detailed Description . . . 455
8.472.2 Field Documentation . . . 456
8.472.2.1 acceleroData . . . 456
8.472.2.2 acceleroTimeSrc . . . 456
8.472.2.3 accelTemp . . . 456
8.472.2.4 gyroData . . . 456
8.472.2.5 gyroTemp . . . 457
8.472.2.6 gyroTimeSrc . . . 457
8.472.2.7 has_acceleroTimeSrc . . . 457
8.472.2.8 has_accelTemp . . . 457
8.472.2.9 has_accleroData . . . 457
8.472.2.10 has_gyroData . . . 457
8.472.2.11 has_gyroTemp . . . 457

8.472.2.12 has_gyroTimeSrc . . . 457
8.472.2.13 has_opaqueId . . . 457
8.472.2.14 opaqueId . . . 457
8.473 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference . . . 457
8.473.1 Detailed Description . . . 457
8.473.2 Field Documentation . . . 457
8.473.2.1 timeMsec . . . 457
8.473.2.2 timeUncMsec . . . 457
8.474 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference . . . 457
8.474.1 Detailed Description . . . 458
8.474.2 Field Documentation . . . 458
8.474.2.1 confidence . . . 458
8.474.2.2 has_confidence . . . 458
8.474.2.3 state . . . 458
8.475 pack_loc_SLQSLOCSetServer_t Struct Reference . . . 458
8.475.1 Detailed Description . . . 458
8.475.2 Field Documentation . . . 459
8.475.2.1 pIPv4Config . . . 459
8.475.2.2 pIPv6Config . . . 459
8.475.2.3 pURLAddr . . . 459
8.475.2.4 serverType . . . 459
8.476 pack_loc_Start_t Struct Reference . . . 459
8.476.1 Detailed Description . . . 459
8.476.2 Field Documentation . . . 460
8.476.2.1 pApplicationInfo . . . 460
8.476.2.2 pConfigAltitudeAssumed . . . 460
8.476.2.3 pHorizontalAccuracyLvl . . . 460
8.476.2.4 pIntermediateReportState . . . 460
8.476.2.5 pMinIntervalTime . . . 460
8.476.2.6 pRecurrenceType . . . 461
8.476.2.7 SessionId . . . 461
8.476.2.8 Tlvresult . . . 461
8.477 pack_loc_Stop_t Struct Reference . . . 461
8.477.1 Detailed Description . . . 461
8.477.2 Field Documentation . . . 461
8.477.2.1 SessionId . . . 461
8.477.2.2 Tlvresult . . . 461
8.478 pack_nas_InitiateDomainAttach_t Struct Reference . . . 461
8.478.1 Detailed Description . . . 461
8.478.2 Field Documentation . . . 461

8.478.2.1 action . . . 462
8.479 pack_nas_PerformNetworkScanPCI_t Struct Reference . . . 462
8.479.1 Detailed Description . . . 462
8.479.2 Field Documentation . . . 464
8.479.2.1 pCiotOpModePref . . . 464
8.479.2.2 pLteM1BandPref . . . 465
8.479.2.3 pLteNB1BandPref . . . 465
8.479.2.4 pNetworkType . . . 465
8.479.2.5 pScanType . . . 465
8.480 pack_nas_SetACCOLC_t Struct Reference . . . 465
8.480.1 Detailed Description . . . 465
8.480.2 Field Documentation . . . 465
8.480.2.1 accolc . . . 465
8.480.2.2 spc . . . 465
8.481 pack_nas_SetCDMANetworkParameters_t Struct Reference . . . 465
8.481.1 Detailed Description . . . 465
8.481.2 Field Documentation . . . 466
8.481.2.1 pApplication . . . 466
8.481.2.2 pBroadcast . . . 466
8.481.2.3 pCustomSCP . . . 467
8.481.2.4 pForceRev0 . . . 467
8.481.2.5 pProtocol . . . 467
8.481.2.6 pRoaming . . . 467
8.481.2.7 pSPC . . . 467
8.482 pack_nas_SetNetworkPreference_t Struct Reference . . . 467
8.482.1 Detailed Description . . . 467
8.482.2 Field Documentation . . . 467
8.482.2.1 Duration . . . 467
8.482.2.2 TechnologyPref . . . 468
8.482.2.3 Tlvresult . . . 468
8.483 pack_nas_SLQSConfigSigInfo_t Struct Reference . . . 468
8.483.1 Detailed Description . . . 468
8.483.2 Field Documentation . . . 469
8.483.2.1 pECIOThresh . . . 469
8.483.2.2 pHDRSINRThresh . . . 469
8.483.2.3 pIOThresh . . . 469
8.483.2.4 pLTESigRptCfg . . . 469
8.483.2.5 pLTESNRThresh . . . 469
8.483.2.6 pRSRPThresh . . . 469
8.483.2.7 pRSRQThresh . . . 469

8.483.2.8 pRSSIThresh . . . 469
8.483.2.9 pTDSCDMASINRCONFThresh . . . 469
8.484 pack_nas_SLQSGetPLMNName_t Struct Reference . . . 469
8.484.1 Detailed Description . . . 469
8.484.2 Field Documentation . . . 470
8.484.2.1 mcc . . . 470
8.484.2.2 mnc . . . 470
8.484.2.3 pMncPcsStatus . . . 470
8.485 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference . . . 470
8.485.1 Detailed Description . . . 470
8.485.2 Field Documentation . . . 471
8.485.2.1 pChangeDuration . . . 471
8.485.2.2 pMncPcsDigitStatus . . . 471
8.485.2.3 pMNRInfo . . . 471
8.485.2.4 regAction . . . 471
8.486 pack_nas_SLQSNasConfigSigInfo2_t Struct Reference . . . 471
8.486.1 Detailed Description . . . 472
8.486.2 Field Documentation . . . 474
8.486.2.1 pCDMAECIODelta . . . 474
8.486.2.2 pCDMAECIOThresh . . . 474
8.486.2.3 pCDMARSSIDelta . . . 474
8.486.2.4 pCDMARSSIThresh . . . 474
8.486.2.5 pGSMRSSIDelta . . . 474
8.486.2.6 pGSMRSSIThresh . . . 474
8.486.2.7 pHDRECIODelta . . . 474
8.486.2.8 pHDRECIOThresh . . . 475
8.486.2.9 pHDRIODelta . . . 475
8.486.2.10 pHDRIOThresh . . . 475
8.486.2.11 pHDRRSSIDelta . . . 475
8.486.2.12 pHDRRSSIThresh . . . 475
8.486.2.13 pHDRSINRDelta . . . 475
8.486.2.14 pHDRSINRThresh . . . 475
8.486.2.15 pLTERSRPDelta . . . 475
8.486.2.16 pLTERSRPThresh . . . 475
8.486.2.17 pLTERSRQDelta . . . 475
8.486.2.18 pLTERSRQThresh . . . 475
8.486.2.19 pLTERSSIDelta . . . 475
8.486.2.20 pLTERSSIThresh . . . 475
8.486.2.21 pLTESigRptConfig . . . 475
8.486.2.22 pLTESNRDelta . . . 475

8.486.2.23 pLTESNRThresh . . . 475
8.486.2.24 pTDSCDMAECIODelta . . . 475
8.486.2.25 pTDSCDMAECIOThresh . . . 475
8.486.2.26 pTDSCDMARSCPDelta . . . 475
8.486.2.27 pTDSCDMARSCPThresh . . . 475
8.486.2.28 pTDSCDMARSSIDelta . . . 475
8.486.2.29 pTDSCDMARSSIThresh . . . 475
8.486.2.30 pTDSCDMASINRDelta . . . 475
8.486.2.31 pTDSCDMASINRThresh . . . 475
8.486.2.32 pWCDMAECIODelta . . . 475
8.486.2.33 pWCDMAECIOThresh . . . 475
8.486.2.34 pWCDMARSSIDelta . . . 475
8.486.2.35 pWCDMARSSIThresh . . . 475
8.487 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 476
8.487.1 Detailed Description . . . 476
8.487.2 Field Documentation . . . 476
8.487.2.1 namID . . . 476
8.488 pack_nas_SLQSNASGeteDRXParamsExt_t Struct Reference . . . 476
8.488.1 Detailed Description . . . 476
8.488.2 Field Documentation . . . 477
8.488.2.1 pEdrxRAT . . . 477
8.488.2.2 pLteOpMode . . . 477
8.489 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 477
8.489.1 Detailed Description . . . 477
8.489.2 Field Documentation . . . 477
8.489.2.1 radio_if . . . 477
8.490 pack_nas_SLQSNasIndicationRegisterExt_t Struct Reference . . . 477
8.490.1 Detailed Description . . . 478
8.490.2 Field Documentation . . . 480
8.490.2.1 pDDTMInd . . . 480
8.490.2.2 pDualStandByPrefInd . . . 480
8.490.2.3 pEdrxChangeInfoInd . . . 480
8.490.2.4 pErrorRateInd . . . 480
8.490.2.5 pHDRNewUATIAssInd . . . 480
8.490.2.6 pHDRSessionCloseInd . . . 480
8.490.2.7 pLTECphyCa . . . 480
8.490.2.8 pManagedRoamingInd . . . 480
8.490.2.9 pNetworkRejectInd . . . 480
8.490.2.10 pNetworkTimeInd . . . 480
8.490.2.11 pServingSystemInd . . . 480

8.490.2.12 pSignalStrengthInd . . . 480
8.490.2.13 pSubscriptionInfoInd . . . 480
8.490.2.14 pSuppressSysInfoInd . . . 480
8.490.2.15 pSysInfoInd . . . 480
8.490.2.16 pSystemSelectionInd . . . 480
8.491 pack_nas_SLQSNasIndicationRegisterV2_t Struct Reference . . . 480
8.491.1 Detailed Description . . . 481
8.491.2 Field Documentation . . . 487
8.491.2.1 pAccessClassBarringInd . . . 487
8.491.2.2 pCallModeStatusInd . . . 487
8.491.2.3 pCSPPLMNModeInd . . . 487
8.491.2.4 pCurrentPLMNNameInd . . . 487
8.491.2.5 pDataSubscriptionPriorityInd . . . 487
8.491.2.6 pDDTMInd . . . 487
8.491.2.7 pDualStandByPrefInd . . . 487
8.491.2.8 pE911StateReadyStatusInd . . . 487
8.491.2.9 pEdrxChangeInfoInd . . . 487
8.491.2.10 pEMBMSStatusInd . . . 487
8.491.2.11 pEmergencyModeStatusInd . . . 487
8.491.2.12 pErrorRateInd . . . 487
8.491.2.13 pGcellInfoInd . . . 487
8.491.2.14 pHDRNewUATIAssInd . . . 488
8.491.2.15 pHDRSessionCloseInd . . . 488
8.491.2.16 pIMSPrefStatusInd . . . 488
8.491.2.17 pLTECphyCa . . . 488
8.491.2.18 pLTERACHFailInd . . . 488
8.491.2.19 pLTERRCTxInfoInd . . . 488
8.491.2.20 pLTESIB16NetworkTimeInd . . . 488
8.491.2.21 pManagedRoamingInd . . . 488
8.491.2.22 pManualNWScanFailureInd . . . 488
8.491.2.23 pNetworkRejectInd . . . 488
8.491.2.24 pNetworkTimeInd . . . 488
8.491.2.25 pOperatorNameDataInd . . . 488
8.491.2.26 pRFBandInfoInd . . . 488
8.491.2.27 pRTREConfigurationInd . . . 488
8.491.2.28 pServingSystemInd . . . 488
8.491.2.29 pSignalStrengthInd . . . 488
8.491.2.30 pSSACBarringExtInd . . . 488
8.491.2.31 pSSACBarringInd . . . 488
8.491.2.32 pSubscriptionChangeInd . . . 488

8.491.2.33 pSubscriptionInfoInd . . . 488
8.491.2.34 pSuppressSysInfoInd . . . 488
8.491.2.35 pSysInfoInd . . . 488
8.491.2.36 pSystemSelectionInd . . . 488
8.491.2.37 pT3346TimerStatusChangeInd . . . 488
8.491.2.38 pT3402TimerValueInd . . . 488
8.491.2.39 pTimerExpiryInd . . . 488
8.492 pack_nas_SLQSNASSeteDRXParams_t Struct Reference . . . 488
8.492.1 Detailed Description . . . 489
8.492.2 Field Documentation . . . 490
8.492.2.1 pCycleLen . . . 490
8.492.2.2 pEdrxCiotLteMode . . . 490
8.492.2.3 pEdrxEnable . . . 490
8.492.2.4 pEdrxRatType . . . 490
8.492.2.5 pPagingTimeWindow . . . 490
8.493 pack_nas_SLQSNasSwiIndicationRegister_t Struct Reference . . . 490
8.493.1 Detailed Description . . . 490
8.493.2 Field Documentation . . . 491
8.493.2.1 gsmUmtsDl . . . 491
8.493.2.2 gsmUmtsUl . . . 491
8.493.2.3 lteEmmDl . . . 491
8.493.2.4 lteEmmUl . . . 491
8.493.2.5 lteEsmDl . . . 491
8.493.2.6 lteEsmUl . . . 491
8.493.2.7 pRankIndicatorInd . . . 491
8.493.2.8 pTimer . . . 491
8.494 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference . . . 491
8.494.1 Detailed Description . . . 492
8.494.2 Field Documentation . . . 492
8.494.2.1 pLteEARFCN . . . 492
8.494.2.2 pLtePCI . . . 492
8.494.2.3 pWcdmaUARFCN . . . 492
8.495 pack_nas_SLQSPerformIncNetworkScanReq_t Struct Reference . . . 492
8.495.1 Detailed Description . . . 492
8.495.2 Field Documentation . . . 497
8.495.2.1 pBandPref . . . 497
8.495.2.2 pIncrementalResults . . . 497
8.495.2.3 pIncResultPeriodicty . . . 497
8.495.2.4 pLTEArfcn . . . 497
8.495.2.5 pLTEBandPref . . . 497

8.495.2.6 pMaxSearchTime . . . 497
8.495.2.7 pNetworkType . . . 497
8.495.2.8 pPrefPLMNList . . . 497
8.495.2.9 pScanType . . . 497
8.495.2.10 pSearchPeriodicity . . . 497
8.495.2.11 pTDSCDMABandPref . . . 497
8.495.2.12 pWCDMAArfcn . . . 497
8.496 pack_nas_SLQSSet3GPP2SubscriptionInfo_t Struct Reference . . . 497
8.496.1 Detailed Description . . . 497
8.496.2 Field Documentation . . . 498
8.496.2.1 namID . . . 498
8.496.2.2 pMdn . . . 498
8.497 pack_nas_SLQSSetBandPreference_t Struct Reference . . . 498
8.497.1 Detailed Description . . . 498
8.497.2 Field Documentation . . . 499
8.497.2.1 bandPref . . . 500
8.498 pack_nas_SLQSSetDDTMPref_t Struct Reference . . . 500
8.498.1 Field Documentation . . . 500
8.498.1.1 DDTMAction . . . 500
8.498.1.2 DDTMPrefSetting . . . 500
8.498.1.3 ServiceOption . . . 500
8.498.1.4 SOLen . . . 500
8.498.1.5 SOListAction . . . 500
8.499 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference . . . 500
8.499.1 Detailed Description . . . 500
8.499.2 Field Documentation . . . 500
8.499.2.1 bEnable . . . 500
8.499.2.2 pSigIndReq . . . 500
8.500 pack_nas_SLQSSetSysSelectionPref_t Struct Reference . . . 500
8.500.1 Detailed Description . . . 501
8.500.2 Field Documentation . . . 505
8.500.2.1 pAcqOrderPref . . . 505
8.500.2.2 pBandPref . . . 505
8.500.2.3 pChgDuration . . . 505
8.500.2.4 pCSGID . . . 505
8.500.2.5 pEmerMode . . . 505
8.500.2.6 pGWAcqOrderPref . . . 505
8.500.2.7 pLTEBandPref . . . 505
8.500.2.8 pMNCIncPCSDigStat . . . 505
8.500.2.9 pModePref . . . 505

8.500.2.10 pNetSelPref . . . 505
8.500.2.11 pPRLPref . . . 505
8.500.2.12 pRAT . . . 505
8.500.2.13 pRoamPref . . . 505
8.500.2.14 pSrvDomainPref . . . 505
8.500.2.15 pSrvRegRestriction . . . 506
8.500.2.16 pTdscdmaBandPref . . . 506
8.501 pack_nas_SLQSSetSysSelectionPrefExt_t Struct Reference . . . 506
8.501.1 Detailed Description . . . 506
8.501.2 Field Documentation . . . 515
8.501.2.1 pAcqOrderPref . . . 515
8.501.2.2 pBandPref . . . 515
8.501.2.3 pChgDuration . . . 515
8.501.2.4 pCiotAcqOrderPref . . . 515
8.501.2.5 pCiotLteOpMode . . . 515
8.501.2.6 pCiotLteOpModePref . . . 515
8.501.2.7 pCSGID . . . 515
8.501.2.8 pEmerMode . . . 515
8.501.2.9 pGWAcqOrderPref . . . 515
8.501.2.10 pLTEBandPref . . . 515
8.501.2.11 pLTEBandPrefExt . . . 515
8.501.2.12 pLteM1BandPref . . . 515
8.501.2.13 pLteNb1BandPref . . . 515
8.501.2.14 pMNCInclPCSDigStat . . . 515
8.501.2.15 pModePref . . . 515
8.501.2.16 pNetSelPref . . . 515
8.501.2.17 pNr5gBandPref . . . 515
8.501.2.18 pPRLPref . . . 516
8.501.2.19 pRAT . . . 516
8.501.2.20 pRoamPref . . . 516
8.501.2.21 pSrvDomainPref . . . 516
8.501.2.22 pSrvRegRestriction . . . 516
8.501.2.23 pTdscdmaBandPref . . . 516
8.502 pack_nas_SLQSSwiPSDetach_t Struct Reference . . . 516
8.502.1 Detailed Description . . . 516
8.502.2 Field Documentation . . . 516
8.502.2.1 pDetachAction . . . 516
8.503 pack_pds_PDSInjectTimeReference_t Struct Reference . . . 516
8.503.1 Detailed Description . . . 516
8.503.2 Field Documentation . . . 517

8.503.2.1 systemDiscontinuities . . . 517
8.503.2.2 systemTime . . . 517
8.504 pack_pds_ResetPDSData_t Struct Reference . . . 517
8.504.1 Detailed Description . . . 517
8.504.2 Field Documentation . . . 517
8.504.2.1 pCellDataMask . . . 518
8.504.2.2 pGPSDataMask . . . 518
8.505 pack_pds_SetEventReportCallback_t Struct Reference . . . 518
8.505.1 Detailed Description . . . 518
8.505.2 Field Documentation . . . 518
8.505.2.1 posDataNmea . . . 518
8.505.2.2 rptPosData . . . 518
8.506 pack_pds_SetPDSDefaults_t Struct Reference . . . 518
8.506.1 Detailed Description . . . 518
8.506.2 Field Documentation . . . 519
8.506.2.1 accuracy . . . 519
8.506.2.2 interval . . . 519
8.506.2.3 operation . . . 519
8.506.2.4 timeout . . . 519
8.507 pack_pds_SetPDSState_t Struct Reference . . . 519
8.507.1 Detailed Description . . . 519
8.507.2 Field Documentation . . . 519
8.507.2.1 enable . . . 519
8.508 pack_pds_SetPortAutomaticTracking_t Struct Reference . . . 519
8.508.1 Detailed Description . . . 520
8.508.2 Field Documentation . . . 520
8.508.2.1 bAuto . . . 520
8.509 pack_pds_SetServiceAutomaticTracking_t Struct Reference . . . 520
8.509.1 Detailed Description . . . 520
8.509.2 Field Documentation . . . 520
8.509.2.1 bAuto . . . 520
8.510 pack_pds_SetXTRAAutomaticDownload_t Struct Reference . . . 520
8.510.1 Detailed Description . . . 521
8.510.2 Field Documentation . . . 521
8.510.2.1 bEnabled . . . 521
8.510.2.2 interval . . . 521
8.511 pack_pds_SetXTRANetwork_t Struct Reference . . . 521
8.511.1 Detailed Description . . . 521
8.511.2 Field Documentation . . . 521
8.511.2.1 preference . . . 521

8.512 pack_pds_SLQSGetAGPSConfig_t Struct Reference . . . 521
8.512.1 Detailed Description . . . 522
8.512.2 Field Documentation . . . 522
8.512.2.1 pNetworkMode . . . 522
8.513 pack_pds_SLQSPDSInjectAbsoluteTimeReference_t Struct Reference . . . 522
8.513.1 Detailed Description . . . 522
8.513.2 Field Documentation . . . 522
8.513.2.1 forceFlag . . . 522
8.513.2.2 timeBase . . . 522
8.513.2.3 timeMsec . . . 523
8.513.2.4 timeUncMsec . . . 523
8.514 pack_pds_SLQSPDSInjectPositionData_t Struct Reference . . . 523
8.514.1 Detailed Description . . . 523
8.514.2 Field Documentation . . . 524
8.514.2.1 pAltitudeWrtEllipsoid . . . 524
8.514.2.2 pAltitudeWrtSealevel . . . 524
8.514.2.3 pHorizontalConfidence . . . 524
8.514.2.4 pHorizontalUncCircular . . . 524
8.514.2.5 pLatitude . . . 524
8.514.2.6 pLongitude . . . 524
8.514.2.7 pPositionSource . . . 524
8.514.2.8 pTimeStamp . . . 524
8.514.2.9 pTimeType . . . 524
8.514.2.10 pVerticalConfidence . . . 524
8.514.2.11 pVerticalUnc . . . 525
8.515 pack_pds_SLQSSetAGPSConfig_t Struct Reference . . . 525
8.515.1 Detailed Description . . . 525
8.515.2 Field Documentation . . . 525
8.515.2.1 pNetworkMode . . . 525
8.515.2.2 pServerAddress . . . 525
8.515.2.3 pServerPort . . . 525
8.515.2.4 pServerURL . . . 525
8.515.2.5 pServerURLLength . . . 525
8.516 pack_pds_SLQSSetPositionMethodState_t Struct Reference . . . 525
8.516.1 Detailed Description . . . 526
8.516.2 Field Documentation . . . 526
8.516.2.1 pWifiState . . . 526
8.516.2.2 pXtraDataState . . . 526
8.516.2.3 pXtraTimeState . . . 526
8.517 pack_pds_StartPDSTrackingSessionExt_t Struct Reference . . . 526

8.517.1 Detailed Description . . . 526
8.517.2 Field Documentation . . . 527
8.517.2.1 fixAccuracy . . . 527
8.517.2.2 fixCount . . . 527
8.517.2.3 fixInterval . . . 527
8.517.2.4 fixTimeout . . . 527
8.517.2.5 sessionControl . . . 527
8.517.2.6 sessionOperation . . . 527
8.517.2.7 sessionServerOption . . . 527
8.517.2.8 sessionType . . . 527
8.518 pack_qmi_t Struct Reference . . . 527
8.518.1 Detailed Description . . . 528
8.518.2 Field Documentation . . . 528
8.518.2.1 msgid . . . 528
8.518.2.2 svc . . . 528
8.518.2.3 timeout . . . 528
8.518.2.4 xid . . . 528
8.519 pack_qos_BindDataPort_t Struct Reference . . . 528
8.519.1 Detailed Description . . . 528
8.519.2 Field Documentation . . . 529
8.519.2.1 pMuxID . . . 529
8.519.2.2 pPeripheralEndPointID . . . 529
8.519.2.3 pSIODataPort . . . 529
8.520 pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 529
8.520.1 Detailed Description . . . 529
8.520.2 Field Documentation . . . 529
8.520.2.1 apnId . . . 529
8.521 pack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 529
8.521.1 Detailed Description . . . 529
8.521.2 Field Documentation . . . 529
8.521.2.1 apnId . . . 529
8.522 pack_qos_SLQSSetQosEventCallback_t Struct Reference . . . 529
8.522.1 Detailed Description . . . 530
8.522.2 Field Documentation . . . 530
8.522.2.1 enable . . . 530
8.523 pack_rms_SetSMSWake_t Struct Reference . . . 530
8.523.1 Detailed Description . . . 530
8.523.2 Field Documentation . . . 530
8.523.2.1 enabled . . . 530
8.523.2.2 wake_mask . . . 530

8.524 pack_sar_SLQSSetRfSarState_t Struct Reference . . . 530
8.524.1 Detailed Description . . . 531
8.524.2 Field Documentation . . . 531
8.524.2.1 RfSarState . . . 531
8.525 pack_sms_SaveSMS_t Struct Reference . . . 531
8.525.1 Detailed Description . . . 531
8.525.2 Field Documentation . . . 532
8.525.2.1 messageFormat . . . 532
8.525.2.2 messageSize . . . 532
8.525.2.3 pMessage . . . 532
8.525.2.4 storageType . . . 532
8.526 pack_sms_SendSMS_t Struct Reference . . . 532
8.526.1 Detailed Description . . . 532
8.526.2 Field Documentation . . . 532
8.526.2.1 messageFormat . . . 532
8.526.2.2 messageSize . . . 532
8.526.2.3 pLinktimer . . . 533
8.526.2.4 pMessage . . . 533
8.527 pack_sms_SetNewSMSCallback_t Struct Reference . . . 533
8.527.1 Detailed Description . . . 533
8.527.2 Field Documentation . . . 533
8.527.2.1 status . . . 533
8.528 pack_sms_SetSMSCAddress_t Struct Reference . . . 533
8.528.1 Detailed Description . . . 533
8.528.2 Field Documentation . . . 533
8.528.2.1 pSMSCAddress . . . 533
8.528.2.2 pSMSCType . . . 533
8.529 pack_sms_SLQSDeleteSMS_t Struct Reference . . . 533
8.529.1 Detailed Description . . . 534
8.529.2 Field Documentation . . . 534
8.529.2.1 pMessageIndex . . . 534
8.529.2.2 pMessageMode . . . 534
8.529.2.3 pMessageTag . . . 534
8.529.2.4 storageType . . . 534
8.530 pack_sms_SLQSGetSMS_t Struct Reference . . . 534
8.530.1 Detailed Description . . . 535
8.530.2 Field Documentation . . . 535
8.530.2.1 messageIndex . . . 535
8.530.2.2 pMessageMode . . . 535
8.530.2.3 storageType . . . 535

8.531 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 535
8.531.1 Detailed Description . . . 535
8.531.2 Field Documentation . . . 535
8.531.2.1 mode . . . 535
8.532 pack_sms_SLQSGetSMSList_t Struct Reference . . . 535
8.532.1 Detailed Description . . . 536
8.532.2 Field Documentation . . . 536
8.532.2.1 pMessageMode . . . 536
8.532.2.2 pRequestedTag . . . 536
8.532.2.3 storageType . . . 536
8.533 pack_sms_SLQSModifySMSStatus_t Struct Reference . . . 536
8.533.1 Detailed Description . . . 536
8.533.2 Field Documentation . . . 537
8.533.2.1 messageIndex . . . 537
8.533.2.2 messageTag . . . 537
8.533.2.3 pMessageMode . . . 537
8.533.2.4 storageType . . . 537
8.534 pack_sms_SLQSSendAsyncSMS_t Struct Reference . . . 537
8.534.1 Detailed Description . . . 537
8.534.2 Field Documentation . . . 537
8.534.2.1 pSendSmsParams . . . 537
8.535 pack_sms_SLQSSetIndicationRegister_t Struct Reference . . . 537
8.535.1 Detailed Description . . . 538
8.535.2 Field Documentation . . . 538
8.535.2.1 pSetIndicationRegReq . . . 538
8.536 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference . . . 538
8.536.1 Detailed Description . . . 538
8.536.2 Field Documentation . . . 538
8.536.2.1 broadcastActivate . . . 538
8.536.2.2 mode . . . 538
8.537 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference . . . 538
8.537.1 Detailed Description . . . 539
8.537.2 Field Documentation . . . 539
8.537.2.1 mode . . . 539
8.537.2.2 pBroadcastConfig . . . 539
8.537.2.3 pCDMABroadcastConfig . . . 539
8.538 pack_sms_SLQSSetSmsStorage_t Struct Reference . . . 539
8.538.1 Detailed Description . . . 539
8.538.2 Field Documentation . . . 539
8.538.2.1 smsStorage . . . 540

8.539 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 540
8.539.1 Detailed Description . . . 540
8.539.2 Field Documentation . . . 540
8.539.2.1 pMaxStorageSizeReq . . . 540
8.540 pack_sms_SLQSSmsSetRoutes_t Struct Reference . . . 540
8.540.1 Detailed Description . . . 540
8.540.2 Field Documentation . . . 540
8.540.2.1 pSetRoutesReq . . . 540
8.541 pack_swiaudio_SLQSGetM2MAudioProfile_t Struct Reference . . . 540
8.541.1 Detailed Description . . . 541
8.541.2 Field Documentation . . . 541
8.541.2.1 pGenerator . . . 541
8.542 pack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference . . . 541
8.542.1 Detailed Description . . . 541
8.542.2 Field Documentation . . . 541
8.542.2.1 Generator . . . 541
8.542.2.2 Profile . . . 541
8.543 pack_swiaudio_SLQSGetM2MAVMute_t Struct Reference . . . 541
8.543.1 Detailed Description . . . 541
8.543.2 Field Documentation . . . 542
8.543.2.1 Profile . . . 542
8.544 pack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference . . . 542
8.544.1 Detailed Description . . . 542
8.544.2 Field Documentation . . . 542
8.544.2.1 Profile . . . 542
8.545 pack_swiaudio_SLQSSetM2MAudioAVCFG_t Struct Reference . . . 542
8.545.1 Detailed Description . . . 542
8.545.2 Field Documentation . . . 543
8.545.2.1 Device . . . 543
8.545.2.2 PIFACEId . . . 543
8.545.2.3 pPCMParams . . . 543
8.545.2.4 Profile . . . 543
8.546 pack_swiaudio_SLQSSetM2MAudioLPBK_t Struct Reference . . . 543
8.546.1 Detailed Description . . . 543
8.546.2 Field Documentation . . . 543
8.546.2.1 Enable . . . 543
8.547 pack_swiaudio_SLQSSetM2MAudioProfile_t Struct Reference . . . 543
8.547.1 Detailed Description . . . 544
8.547.2 Field Documentation . . . 544
8.547.2.1 pCwtMute . . . 544

8.547.2.2 pEarMute . . . 544
8.547.2.3 pGenerator . . . 544
8.547.2.4 pMicMute . . . 544
8.547.2.5 Profile . . . 544
8.547.2.6 pVolume . . . 544
8.548 pack_swiaudio_SLQSSetM2MAudioVolume_t Struct Reference . . . 545
8.548.1 Detailed Description . . . 545
8.548.2 Field Documentation . . . 545
8.548.2.1 Generator . . . 545
8.548.2.2 Level . . . 545
8.548.2.3 Profile . . . 545
8.549 pack_swiaudio_SLQSSetM2MAVMute_t Struct Reference . . . 545
8.549.1 Detailed Description . . . 545
8.549.2 Field Documentation . . . 546
8.549.2.1 EarMute . . . 546
8.549.2.2 MicMute . . . 546
8.549.2.3 pCwtMute . . . 546
8.549.2.4 Profile . . . 546
8.550 pack_swiaudio_SLQSSetM2MSpkrGain_t Struct Reference . . . 546
8.550.1 Detailed Description . . . 546
8.550.2 Field Documentation . . . 546
8.550.2.1 Profile . . . 546
8.550.2.2 Value . . . 546
8.551 pack_swiavms_SLQSAVMSSendSelection_t Struct Reference . . . 547
8.551.1 Detailed Description . . . 547
8.551.2 Field Documentation . . . 547
8.551.2.1 pClientPerformOperationFlag . . . 547
8.551.2.2 pDeferTime . . . 547
8.551.2.3 pPackageID . . . 547
8.551.2.4 pRejectReason . . . 547
8.551.2.5 selection . . . 547
8.552 pack_swiavms_SLQSAVMSSetSettings_t Struct Reference . . . 547
8.552.1 Detailed Description . . . 548
8.552.2 Field Documentation . . . 549
8.552.2.1 AutoConnect . . . 549
8.552.2.2 AutoReboot . . . 549
8.552.2.3 pAPNInfo . . . 549
8.552.2.4 pConnectionRetryTimers . . . 549
8.552.2.5 pFwAutoSDM . . . 549
8.552.2.6 pNotifStore . . . 549

8.552.2.7 pPeriodInfo . . . 549
8.552.2.8 pPollingTimer . . . 549
8.552.2.9 PromptFwDownload . . . 549
8.552.2.10 PromptFwUpdate . . . 549
8.553 pack_swiavms_SLQSAVMSSetSettings_v2_t Struct Reference . . . 549
8.553.1 Detailed Description . . . 549
8.553.2 Field Documentation . . . 550
8.553.2.1 AutoConnect . . . 551
8.553.2.2 pAPNInfo . . . 551
8.553.2.3 pAutoReboot . . . 551
8.553.2.4 pConnectionRetryTimers . . . 551
8.553.2.5 pFwAutoSDM . . . 551
8.553.2.6 pNotifStore . . . 551
8.553.2.7 pPeriodInfo . . . 551
8.553.2.8 pPollingTimer . . . 551
8.553.2.9 PromptFwDownload . . . 551
8.553.2.10 PromptFwUpdate . . . 551
8.554 pack_swiavms_SLQSAVMSStartSession_t Struct Reference . . . 551
8.554.1 Detailed Description . . . 551
8.554.2 Field Documentation . . . 551
8.554.2.1 sessionType . . . 551
8.555 pack_swiavms_SLQSAVMSStopSession_t Struct Reference . . . 551
8.555.1 Detailed Description . . . 551
8.555.2 Field Documentation . . . 552
8.555.2.1 sessionType . . . 552
8.556 pack_swidms_SLQSSwiDmsSetHWWatchdog_t Struct Reference . . . 552
8.556.1 Detailed Description . . . 552
8.556.2 Field Documentation . . . 552
8.556.2.1 enable . . . 552
8.556.2.2 resetDelay . . . 552
8.556.2.3 timeout . . . 552
8.557 pack_swidms_SLQSSwiDmsSetMTU_t Struct Reference . . . 552
8.557.1 Detailed Description . . . 553
8.557.2 Field Documentation . . . 553
8.557.2.1 MTUSize . . . 553
8.558 pack_swidms_SLQSSwiDmsSetUimSlot_t Struct Reference . . . 553
8.558.1 Detailed Description . . . 553
8.558.2 Field Documentation . . . 554
8.558.2.1 pAutoSwitchMode . . . 554
8.558.2.2 pMultiplexedSimMode . . . 554

8.558.2.3 pSlotSelect . . . 554
8.558.2.4 slotSelectaction . . . 554
8.559 pack_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference . . . 554
8.559.1 Detailed Description . . . 554
8.559.2 Field Documentation . . . 555
8.559.2.1 CfgValue . . . 555
8.560 pack_swidms_SLQSSwiDmsSetUsbNetNum_t Struct Reference . . . 555
8.560.1 Detailed Description . . . 555
8.560.2 Field Documentation . . . 555
8.560.2.1 nUsbNetNum . . . 556
8.561 pack_swiloc_SwiLocSetAutoStart_t Struct Reference . . . 556
8.561.1 Detailed Description . . . 556
8.561.2 Field Documentation . . . 557
8.561.2.1 fix_rate . . . 557
8.561.2.2 fix_type . . . 557
8.561.2.3 function . . . 557
8.561.2.4 max_dist . . . 557
8.561.2.5 max_time . . . 557
8.561.2.6 set_fix_rate . . . 557
8.561.2.7 set_fix_type . . . 557
8.561.2.8 set_function . . . 557
8.561.2.9 set_max_dist . . . 557
8.561.2.10 set_max_time . . . 557
8.562 pack_swioma_SLQSOMADMCancelSession_t Struct Reference . . . 557
8.562.1 Detailed Description . . . 557
8.562.2 Field Documentation . . . 558
8.562.2.1 sessionType . . . 558
8.563 pack_swioma_SLQSOMADMCancelSessionExt_t Struct Reference . . . 558
8.563.1 Detailed Description . . . 558
8.563.2 Field Documentation . . . 558
8.563.2.1 sessionType . . . 558
8.564 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 558
8.564.1 Detailed Description . . . 558
8.564.2 Field Documentation . . . 559
8.564.2.1 SessionType . . . 559
8.565 pack_swioma_SLQSOMADMSendSelection_t Struct Reference . . . 559
8.565.1 Detailed Description . . . 559
8.565.2 Field Documentation . . . 559
8.565.2.1 pDeferTime . . . 559
8.565.2.2 pRejectReason . . . 559

8.565.2.3 selection . . . 559
8.566 pack_swioma_SLQSOMADMSendSelectionExt_t Struct Reference . . . 559
8.566.1 Detailed Description . . . 559
8.566.2 Field Documentation . . . 560
8.566.2.1 selection . . . 560
8.567 pack_swioma_SLQSOMADMSetSettings_t Struct Reference . . . 560
8.567.1 Detailed Description . . . 560
8.567.2 Field Documentation . . . 561
8.567.2.1 FOTAdownload . . . 561
8.567.2.2 FOTAUpdate . . . 561
8.567.2.3 pAutosdm . . . 561
8.567.2.4 pFwAutoCheck . . . 561
8.568 pack_swioma_SLQSOMADMSetSettingsExt_t Struct Reference . . . 561
8.568.1 Detailed Description . . . 561
8.568.2 Field Documentation . . . 562
8.568.2.1 FOTAdownload . . . 562
8.568.2.2 FOTAUpdate . . . 562
8.568.2.3 FUMOEnable . . . 562
8.568.2.4 OMADMEnable . . . 562
8.568.2.5 OMADMLogEnable . . . 562
8.568.2.6 PRLEnable . . . 562
8.569 pack_swioma_SLQSOMADMStartSession_t Struct Reference . . . 562
8.569.1 Detailed Description . . . 562
8.569.2 Field Documentation . . . 562
8.569.2.1 sessionType . . . 562
8.570 pack_swioma_SLQSOMADMStartSessionExt_t Struct Reference . . . 562
8.570.1 Detailed Description . . . 563
8.570.2 Field Documentation . . . 563
8.570.2.1 sessionType . . . 563
8.571 pack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference . . . 563
8.571.1 Detailed Description . . . 563
8.571.2 Field Documentation . . . 563
8.571.2.1 mitigationDevID . . . 563
8.571.2.2 mitigationDevIDLen . . . 563
8.572 pack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference . . . 563
8.572.1 Detailed Description . . . 564
8.572.2 Field Documentation . . . 564
8.572.2.1 mitigationDevID . . . 564
8.572.2.2 mitigationDevIDLen . . . 564
8.573 pack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference . . . 564

8.573.1 Detailed Description . . . 564
8.573.2 Field Documentation . . . 564
8.573.2.1 mitigationDevID . . . 564
8.573.2.2 mitigationDevIDLen . . . 564
8.574 pack_uim_ChangePin_t Struct Reference . . . 564
8.574.1 Detailed Description . . . 565
8.574.2 Field Documentation . . . 565
8.574.2.1 changePIN . . . 565
8.574.2.2 EncryptedPIN1 . . . 565
8.574.2.3 pIndicationToken . . . 565
8.574.2.4 pKeyReferenceID . . . 565
8.574.2.5 sessionInfo . . . 565
8.574.2.6 Tlvresult . . . 565
8.575 pack_uim_ReadTransparent_t Struct Reference . . . 565
8.575.1 Detailed Description . . . 566
8.575.2 Field Documentation . . . 566
8.575.2.1 fileIndex . . . 566
8.575.2.2 pEncryptData . . . 566
8.575.2.3 pIndicationToken . . . 566
8.575.2.4 readTransparent . . . 566
8.575.2.5 sessionInfo . . . 566
8.575.2.6 Tlvresult . . . 566
8.576 pack_uim_SetPinProtection_t Struct Reference . . . 567
8.576.1 Detailed Description . . . 567
8.576.2 Field Documentation . . . 567
8.576.2.1 EncryptedPIN1 . . . 567
8.576.2.2 pIndicationToken . . . 567
8.576.2.3 pinProtection . . . 567
8.576.2.4 pKeyReferenceID . . . 567
8.576.2.5 sessionInfo . . . 567
8.576.2.6 Tlvresult . . . 568
8.577 pack_uim_SLQSUIMAddProfile_t Struct Reference . . . 568
8.577.1 Detailed Description . . . 568
8.577.2 Field Documentation . . . 568
8.577.2.1 pProfileConfInfo . . . 568
8.577.2.2 profileActInfo . . . 568
8.577.2.3 pUserConsent . . . 568
8.577.2.4 slot . . . 568
8.578 pack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 568
8.578.1 Detailed Description . . . 569

8.578.2 Field Documentation . . . 569
8.578.2.1 authData . . . 569
8.578.2.2 pIndicationToken . . . 569
8.578.2.3 sessionInfo . . . 569
8.579 pack_uim_SLQSUIMDeleteProfile_t Struct Reference . . . 569
8.579.1 Detailed Description . . . 569
8.579.2 Field Documentation . . . 569
8.579.2.1 profileId . . . 569
8.579.2.2 slot . . . 570
8.580 pack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 570
8.580.1 Detailed Description . . . 570
8.580.2 Field Documentation . . . 570
8.580.2.1 depersonilisationInfo . . . 570
8.581 pack_uim_SLQSUIMeUICCMemReset_t Struct Reference . . . 570
8.581.1 Detailed Description . . . 570
8.581.2 Field Documentation . . . 571
8.581.2.1 pResetOption . . . 571
8.581.2.2 uimSlot . . . 571
8.582 pack_uim_SLQSUIMEventRegister_t Struct Reference . . . 571
8.582.1 Detailed Description . . . 571
8.582.2 Field Documentation . . . 571
8.582.2.1 eventMask . . . 571
8.583 pack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 571
8.583.1 Detailed Description . . . 571
8.583.2 Field Documentation . . . 572
8.583.2.1 pConfigurationMask . . . 572
8.584 pack_uim_SLQSUIMGetEID_t Struct Reference . . . 572
8.584.1 Detailed Description . . . 572
8.584.2 Field Documentation . . . 572
8.584.2.1 pUimSlotID . . . 572
8.585 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 572
8.585.1 Detailed Description . . . 572
8.585.2 Field Documentation . . . 573
8.585.2.1 fileIndex . . . 573
8.585.2.2 pIndicationToken . . . 573
8.585.2.3 sessionInfo . . . 573
8.586 pack_uim_SLQSUIMGetProfileInfo_t Struct Reference . . . 573
8.586.1 Detailed Description . . . 573
8.586.2 Field Documentation . . . 574
8.586.2.1 profileId . . . 574

8.586.2.2 slotNo . . . 574
8.587 pack_uim_SLQSUIMGetServiceStatus_t Struct Reference . . . 574
8.587.1 Detailed Description . . . 574
8.587.2 Field Documentation . . . 574
8.587.2.1 capMask . . . 575
8.587.2.2 sessionInfo . . . 575
8.588 pack_uim_SLQSUIMGetSIMProfile_t Struct Reference . . . 575
8.588.1 Detailed Description . . . 575
8.588.2 Field Documentation . . . 575
8.588.2.1 slotNo . . . 575
8.589 pack_uim_SLQSUIMPowerDown_t Struct Reference . . . 575
8.589.1 Detailed Description . . . 575
8.589.2 Field Documentation . . . 575
8.589.2.1 slot . . . 575
8.590 pack_uim_SLQSUIMPowerUp_t Struct Reference . . . 576
8.590.1 Detailed Description . . . 576
8.590.2 Field Documentation . . . 576
8.590.2.1 pIgnoreHotSwapSwitch . . . 576
8.590.2.2 slot . . . 576
8.591 pack_uim_SLQSUIMReadRecord_t Struct Reference . . . 576
8.591.1 Detailed Description . . . 576
8.591.2 Field Documentation . . . 577
8.591.2.1 fileIndex . . . 577
8.591.2.2 pIndicationToken . . . 577
8.591.2.3 pLastRecord . . . 577
8.591.2.4 readRecord . . . 577
8.591.2.5 sessionInfo . . . 577
8.592 pack_uim_SLQSUIMRefreshComplete_t Struct Reference . . . 577
8.592.1 Detailed Description . . . 577
8.592.2 Field Documentation . . . 577
8.592.2.1 refreshComplete . . . 578
8.592.2.2 sessionInfo . . . 578
8.593 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 578
8.593.1 Detailed Description . . . 578
8.593.2 Field Documentation . . . 578
8.593.2.1 sessionInfo . . . 578
8.594 pack_uim_SLQSUIMRefreshOK_t Struct Reference . . . 578
8.594.1 Detailed Description . . . 578
8.594.2 Field Documentation . . . 578
8.594.2.1 OKtoRefresh . . . 578

8.594.2.2 sessionInfo . . . 579
8.595 pack_uim_SLQSUIMRefreshRegister_t Struct Reference . . . 579
8.595.1 Detailed Description . . . 579
8.595.2 Field Documentation . . . 579
8.595.2.1 regRefresh . . . 579
8.595.2.2 sessionInfo . . . 579
8.596 pack_uim_SLQSUIMSetServiceStatus_t Struct Reference . . . 579
8.596.1 Detailed Description . . . 579
8.596.2 Field Documentation . . . 580
8.596.2.1 pFDNStatus . . . 580
8.596.2.2 sessionInfo . . . 580
8.597 pack_uim_SLQSUIMSetSIMProfile_t Struct Reference . . . 580
8.597.1 Detailed Description . . . 580
8.597.2 Field Documentation . . . 581
8.597.2.1 pProfileStatus . . . 581
8.597.2.2 profileId . . . 581
8.597.2.3 setSlot . . . 581
8.598 pack_uim_SLQSUIMSwitchSlot_t Struct Reference . . . 581
8.598.1 Detailed Description . . . 581
8.598.2 Field Documentation . . . 581
8.598.2.1 bLogicalSlot . . . 581
8.598.2.2 ulPhysicalSlot . . . 581
8.599 pack_uim_SLQSUIMUpdateProfile_t Struct Reference . . . 581
8.599.1 Detailed Description . . . 582
8.599.2 Field Documentation . . . 582
8.599.2.1 pNickName . . . 582
8.599.2.2 profileId . . . 582
8.599.2.3 slot . . . 582
8.600 pack_uim_SLQSUIMWriteRecord_t Struct Reference . . . 582
8.600.1 Detailed Description . . . 582
8.600.2 Field Documentation . . . 583
8.600.2.1 fileIndex . . . 583
8.600.2.2 pIndicationToken . . . 583
8.600.2.3 sessionInfo . . . 583
8.600.2.4 writeRecord . . . 583
8.601 pack_uim_SLQSUIMWriteTransparent_t Struct Reference . . . 583
8.601.1 Detailed Description . . . 583
8.601.2 Field Documentation . . . 584
8.601.2.1 fileIndex . . . 584
8.601.2.2 pIndicationToken . . . 584

8.601.2.3 sessionInfo . . . 584
8.601.2.4 writeTransparent . . . 584
8.602 pack_uim_UnblockPin_t Struct Reference . . . 584
8.602.1 Detailed Description . . . 584
8.602.2 Field Documentation . . . 584
8.602.2.1 EncryptedPIN1 . . . 585
8.602.2.2 pIndicationToken . . . 585
8.602.2.3 pinProtection . . . 585
8.602.2.4 pKeyReferenceID . . . 585
8.602.2.5 sessionInfo . . . 585
8.602.2.6 TlvResult . . . 585
8.603 pack_uim_VerifyPin_t Struct Reference . . . 585
8.603.1 Detailed Description . . . 585
8.603.2 Field Documentation . . . 586
8.603.2.1 pEncryptedPIN1 . . . 586
8.603.2.2 pIndicationToken . . . 586
8.603.2.3 pKeyReferenceID . . . 586
8.603.2.4 sessionInfo . . . 586
8.603.2.5 TlvResult . . . 586
8.603.2.6 verifyPIN . . . 586
8.604 pack_voice_AnswerUSSD_t Struct Reference . . . 586
8.604.1 Detailed Description . . . 586
8.604.2 Field Documentation . . . 586
8.604.2.1 pInfo . . . 586
8.605 pack_voice_OriginateUSSD_t Struct Reference . . . 586
8.605.1 Detailed Description . . . 586
8.605.2 Field Documentation . . . 587
8.605.2.1 pInfo . . . 587
8.606 pack_voice_SLQSOriginateUSSD_t Struct Reference . . . 587
8.606.1 Detailed Description . . . 587
8.606.2 Field Documentation . . . 587
8.606.2.1 ussData . . . 587
8.606.2.2 ussDCS . . . 587
8.606.2.3 ussLen . . . 587
8.607 pack_voice_SLQSVoiceALSSelectLine_t Struct Reference . . . 587
8.607.1 Detailed Description . . . 588
8.607.2 Field Documentation . . . 588
8.607.2.1 lineValue . . . 588
8.608 pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference . . . 588
8.608.1 Detailed Description . . . 588

8.608.2 Field Documentation . . . 588
8.608.2.1 switchOption . . . 588
8.609pack_voice_SLQSVoiceAnswerCall_t Struct Reference . . . 588
8.609.1 Detailed Description . . . 588
8.609.2 Field Documentation . . . 589
8.609.2.1 pCallId . . . 589
8.610pack_voice_SLQSVoiceBindSubscription_t Struct Reference . . . 589
8.610.1 Detailed Description . . . 589
8.610.2 Field Documentation . . . 589
8.610.2.1 subsType . . . 589
8.611pack_voice_SLQSVoiceBurstDTMF_t Struct Reference . . . 589
8.611.1 Detailed Description . . . 589
8.611.2 Field Documentation . . . 589
8.611.2.1 BurstDTMFInfo . . . 590
8.611.2.2 pBurstDTMFLengths . . . 590
8.612pack_voice_SLQSVoiceDialCall_t Struct Reference . . . 590
8.612.1 Detailed Description . . . 590
8.612.2 Field Documentation . . . 591
8.612.2.1 callNumber . . . 591
8.612.2.2 pCallPartySubAdd . . . 591
8.612.2.3 pCallType . . . 591
8.612.2.4 pCLIRType . . . 591
8.612.2.5 pCUGInfo . . . 591
8.612.2.6 pEmergencyCategory . . . 591
8.612.2.7 pSvcType . . . 591
8.612.2.8 pUUSInFo . . . 591
8.613pack_voice_SLQSVoiceEndCall_t Struct Reference . . . 591
8.613.1 Detailed Description . . . 592
8.613.2 Field Documentation . . . 592
8.613.2.1 pCallId . . . 592
8.614pack_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 592
8.614.1 Detailed Description . . . 592
8.614.2 Field Documentation . . . 592
8.614.2.1 pSvcClass . . . 592
8.614.2.2 reason . . . 593
8.615pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 593
8.615.1 Detailed Description . . . 593
8.615.2 Field Documentation . . . 593
8.615.2.1 pSvcClass . . . 593
8.615.2.2 Reason . . . 593

8.616pack_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 593
8.616.1 Detailed Description . . . 593
8.616.2 Field Documentation . . . 594
8.616.2.1 callID . . . 594
8.617pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 594
8.617.1 Detailed Description . . . 594
8.617.2 Field Documentation . . . 594
8.617.2.1 pSvcClass . . . 594
8.618pack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 594
8.618.1 Detailed Description . . . 594
8.618.2 Field Documentation . . . 595
8.618.2.1 pAirTimer . . . 595
8.618.2.2 pAMRStatus . . . 596
8.618.2.3 pAutoAnswer . . . 596
8.618.2.4 pNamID . . . 596
8.618.2.5 pPrefVoicePrivacy . . . 596
8.618.2.6 pPrefVoiceSO . . . 596
8.618.2.7 pRoamTimer . . . 596
8.618.2.8 pTTYMode . . . 596
8.618.2.9 pVoiceDomainPref . . . 596
8.619pack_voice_SLQSVoiceIndicationRegister_t Struct Reference . . . 596
8.619.1 Detailed Description . . . 596
8.619.2 Field Documentation . . . 597
8.619.2.1 pRegDTMFEvents . . . 597
8.619.2.2 pRegVoicePrivacyEvents . . . 597
8.619.2.3 pSuppsNotifEvents . . . 597
8.620pack_voice_SLQSVoiceManageCalls_t Struct Reference . . . 597
8.620.1 Detailed Description . . . 597
8.620.2 Field Documentation . . . 598
8.620.2.1 pCallID . . . 598
8.620.2.2 SUPSType . . . 598
8.621pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference . . . 598
8.621.1 Detailed Description . . . 598
8.621.2 Field Documentation . . . 598
8.621.2.1 USSInformation . . . 598
8.622pack_voice_SLQSVoiceSendFlash_t Struct Reference . . . 598
8.622.1 Detailed Description . . . 598
8.622.2 Field Documentation . . . 599
8.622.2.1 pCallID . . . 599
8.622.2.2 pFlashPayLd . . . 599

8.622.2.3 pFlashType . . . 599
8.623 pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 599
8.623.1 Detailed Description . . . 599
8.623.2 Field Documentation . . . 600
8.623.2.1 newPasswd . . . 600
8.623.2.2 newPasswdAgain . . . 600
8.623.2.3 oldPasswd . . . 600
8.623.2.4 Reason . . . 600
8.624 pack_voice_SLQSVoiceSetConfig_t Struct Reference . . . 600
8.624.1 Detailed Description . . . 600
8.624.2 Field Documentation . . . 601
8.624.2.1 pAirTimerConfig . . . 601
8.624.2.2 pAutoAnswer . . . 601
8.624.2.3 pPrefVoiceDomain . . . 601
8.624.2.4 pPrefVoiceSO . . . 601
8.624.2.5 pRoamTimerConfig . . . 601
8.624.2.6 pTTYMode . . . 601
8.625 pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference . . . 601
8.625.1 Detailed Description . . . 601
8.625.2 Field Documentation . . . 601
8.625.2.1 privacyPref . . . 601
8.626 pack_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 601
8.626.1 Detailed Description . . . 602
8.626.2 Field Documentation . . . 604
8.626.2.1 pCallBarringPasswd . . . 604
8.626.2.2 pCallForwardingNumber . . . 604
8.626.2.3 pCallFwdTypeAndPlan . . . 604
8.626.2.4 pServiceClass . . . 604
8.626.2.5 pTimerVal . . . 604
8.626.2.6 reason . . . 604
8.626.2.7 voiceSvc . . . 604
8.627 pack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 604
8.627.1 Detailed Description . . . 604
8.627.2 Field Documentation . . . 604
8.627.2.1 DTMFdigit . . . 604
8.627.2.2 pCallID . . . 605
8.628 pack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 605
8.628.1 Detailed Description . . . 605
8.628.2 Field Documentation . . . 605
8.628.2.1 callID . . . 605

8.629 pack_wda_GetDataFormat_t Struct Reference . . . 605
8.629.1 Detailed Description . . . 605
8.629.2 Field Documentation . . . 606
8.629.2.1 ep_type . . . 606
8.629.2.2 iface_id . . . 606
8.630 pack_wda_GetQmapSettings_t Struct Reference . . . 606
8.630.1 Detailed Description . . . 606
8.630.2 Field Documentation . . . 606
8.630.2.1 pEp_type . . . 606
8.630.2.2 pIface_id . . . 606
8.631 pack_wda_SetDataFormat_t Struct Reference . . . 606
8.631.1 Detailed Description . . . 607
8.631.2 Field Documentation . . . 609
8.631.2.1 pDl_data_aggregation_max_datagrams . . . 609
8.631.2.2 pDl_data_aggregation_max_size . . . 609
8.631.2.3 pDl_data_aggregation_protocol . . . 609
8.631.2.4 pDl_minimum_padding . . . 609
8.631.2.5 pEp_type . . . 609
8.631.2.6 pFlow_control . . . 609
8.631.2.7 pIface_id . . . 609
8.631.2.8 pNdp_signature . . . 609
8.631.2.9 pQlink_prot . . . 609
8.631.2.10 pQos_format . . . 609
8.631.2.11 pQos_header_format . . . 609
8.631.2.12 pTcp_coalescing . . . 609
8.631.2.13 pUdp_coalescing . . . 609
8.631.2.14 pUl_data_aggregation_max_datagrams . . . 609
8.631.2.15 pUl_data_aggregation_max_size . . . 609
8.631.2.16 pUl_data_aggregation_protocol . . . 609
8.632 pack_wda_SetQmapSettings_t Struct Reference . . . 609
8.632.1 Detailed Description . . . 610
8.632.2 Field Documentation . . . 610
8.632.2.1 pEp_type . . . 610
8.632.2.2 pIface_id . . . 610
8.632.2.3 pQmap_dfc . . . 610
8.633 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference . . . 610
8.633.1 Detailed Description . . . 610
8.633.2 Field Documentation . . . 611
8.633.2.1 pEnableNotification . . . 611
8.634 pack_wds_GetDefaultProfile_t Struct Reference . . . 611

8.634.1 Detailed Description . . . 611
8.634.2 Field Documentation . . . 611
8.634.2.1 profiletype . . . 611
8.635pack_wds_GetDefaultProfileNum_t Struct Reference . . . 611
8.635.1 Detailed Description . . . 611
8.635.2 Field Documentation . . . 612
8.635.2.1 family . . . 612
8.635.2.2 type . . . 612
8.636pack_wds_GetDefaultProfileV2_t Struct Reference . . . 612
8.636.1 Detailed Description . . . 612
8.636.2 Field Documentation . . . 612
8.636.2.1 profiletype . . . 612
8.637pack_wds_GetDormancyState_t Struct Reference . . . 612
8.637.1 Detailed Description . . . 612
8.638pack_wds_GetLastMobileIPError_t Struct Reference . . . 612
8.638.1 Detailed Description . . . 612
8.639pack_wds_GetMobileIP_t Struct Reference . . . 613
8.639.1 Detailed Description . . . 613
8.640pack_wds_GetMobileIPProfile_t Struct Reference . . . 613
8.640.1 Detailed Description . . . 613
8.640.2 Field Documentation . . . 613
8.640.2.1 index . . . 613
8.641pack_wds_GetPacketStatistics_t Struct Reference . . . 613
8.641.1 Detailed Description . . . 613
8.641.2 Field Documentation . . . 614
8.641.2.1 pStatMask . . . 614
8.642pack_wds_GetPacketStatus_t Struct Reference . . . 614
8.642.1 Detailed Description . . . 614
8.642.2 Field Documentation . . . 614
8.642.2.1 statmask . . . 614
8.643pack_wds_GetSessionDuration_t Struct Reference . . . 614
8.643.1 Detailed Description . . . 615
8.644pack_wds_RMSetTransferStatistics_t Struct Reference . . . 615
8.644.1 Detailed Description . . . 615
8.644.2 Field Documentation . . . 615
8.644.2.1 RmTrasnferStaticsReq . . . 615
8.645pack_wds_SetAutoconnect_t Struct Reference . . . 615
8.645.1 Detailed Description . . . 615
8.645.2 Field Documentation . . . 616
8.645.2.1 acroamsetting . . . 616

8.645.2.2 acsetting . . . 616
8.646 pack_wds_SetDefaultProfile_t Struct Reference . . . 616
8.646.1 Detailed Description . . . 616
8.646.2 Field Documentation . . . 617
8.646.2.1 authentication . . . 617
8.646.2.2 ipAddress . . . 617
8.646.2.3 pApnname . . . 617
8.646.2.4 pdpType . . . 617
8.646.2.5 pName . . . 617
8.646.2.6 pPassword . . . 617
8.646.2.7 primaryDNS . . . 617
8.646.2.8 profileType . . . 617
8.646.2.9 pUsername . . . 617
8.646.2.10 secondaryDNS . . . 617
8.647 pack_wds_SetDefaultProfileNum_t Struct Reference . . . 618
8.647.1 Detailed Description . . . 618
8.647.2 Field Documentation . . . 618
8.647.2.1 family . . . 618
8.647.2.2 index . . . 618
8.647.2.3 type . . . 618
8.648 pack_wds_SetMobileIP_t Struct Reference . . . 618
8.648.1 Detailed Description . . . 618
8.648.2 Field Documentation . . . 618
8.648.2.1 mode . . . 619
8.649 pack_wds_SetMobileIPParameters_t Struct Reference . . . 619
8.649.1 Detailed Description . . . 619
8.649.2 Field Documentation . . . 620
8.649.2.1 pHA2002bis . . . 620
8.649.2.2 pHAAuthenticator . . . 620
8.649.2.3 pMode . . . 620
8.649.2.4 pReRegPeriod . . . 620
8.649.2.5 pReRegTraffic . . . 620
8.649.2.6 pRetryInterval . . . 620
8.649.2.7 pRetryLimit . . . 620
8.649.2.8 pSPC . . . 620
8.650 pack_wds_SetMobileIPProfile_t Struct Reference . . . 620
8.650.1 Detailed Description . . . 620
8.650.2 Field Documentation . . . 621
8.650.2.1 index . . . 621
8.650.2.2 pAAASPI . . . 621

8.650.2.3 pAddress . . . 621
8.650.2.4 pEnabled . . . 621
8.650.2.5 pHASPI . . . 621
8.650.2.6 pMNAAA . . . 621
8.650.2.7 pMNHA . . . 621
8.650.2.8 pNAI . . . 621
8.650.2.9 pPrimaryHA . . . 621
8.650.2.10 pRevTunneling . . . 621
8.650.2.11 pSecondaryHA . . . 622
8.650.2.12 spc . . . 622
8.651 pack_wds_SLQSCreateProfile_t Struct Reference . . . 622
8.651.1 Detailed Description . . . 622
8.651.2 Field Documentation . . . 622
8.651.2.1 pCurProfile . . . 622
8.651.2.2 pProfileId . . . 622
8.651.2.3 pProfileType . . . 622
8.652 pack_wds_SLQSDeleteProfile_t Struct Reference . . . 622
8.652.1 Detailed Description . . . 623
8.652.2 Field Documentation . . . 623
8.652.2.1 profileIndex . . . 623
8.652.2.2 profileType . . . 623
8.653 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 623
8.653.1 Detailed Description . . . 623
8.654 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 623
8.654.1 Detailed Description . . . 623
8.655 pack_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 623
8.655.1 Detailed Description . . . 624
8.655.2 Field Documentation . . . 624
8.655.2.1 Mask . . . 625
8.655.2.2 pReportChannelRate . . . 625
8.655.2.3 pReportConnStatus . . . 625
8.655.2.4 pReportDataBearerTech . . . 625
8.655.2.5 pReportDormStatus . . . 625
8.655.2.6 pTransferStatInd . . . 625
8.656 pack_wds_SLQSGetProfileSettings_t Struct Reference . . . 625
8.656.1 Detailed Description . . . 625
8.656.2 Field Documentation . . . 625
8.656.2.1 ProfileId . . . 625
8.656.2.2 ProfileType . . . 626
8.657 pack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 626

8.657.1 Detailed Description . . . 626
8.657.2 Field Documentation . . . 626
8.657.2.1 pReqSettings . . . 626
8.658pack_wds_SLQSModifyProfile_t Struct Reference . . . 626
8.658.1 Detailed Description . . . 627
8.658.2 Field Documentation . . . 627
8.658.2.1 curProfile . . . 627
8.658.2.2 pProfileId . . . 627
8.658.2.3 pProfileType . . . 627
8.659pack_wds_SLQSSet3GPPConfigItem_t Struct Reference . . . 627
8.659.1 Detailed Description . . . 628
8.659.2 Field Documentation . . . 629
8.659.2.1 LTEAttachProfileListLen . . . 629
8.659.2.2 p3gppRelease . . . 629
8.659.2.3 pDefaultPDNEnabled . . . 629
8.659.2.4 pLTEAttachProfile . . . 629
8.659.2.5 pLTEAttachProfileList . . . 629
8.659.2.6 pProfileList . . . 629
8.660pack_wds_SLQSSetIPFamilyPreference_t Struct Reference . . . 629
8.660.1 Detailed Description . . . 629
8.660.2 Field Documentation . . . 629
8.660.2.1 IPFamilyPreference . . . 629
8.661pack_wds_SLQSSetWdsEventCallback_t Struct Reference . . . 629
8.661.1 Detailed Description . . . 629
8.661.2 Field Documentation . . . 630
8.661.2.1 currentDataBearer . . . 630
8.661.2.2 dataBearer . . . 630
8.661.2.3 dataBearerTechExt . . . 630
8.661.2.4 dataSystemStatus . . . 630
8.661.2.5 dormancyStatus . . . 630
8.661.2.6 interval . . . 630
8.661.2.7 mobileIP . . . 630
8.661.2.8 transferStats . . . 630
8.662pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 630
8.662.1 Detailed Description . . . 630
8.662.2 Field Documentation . . . 630
8.662.2.1 pProfileId . . . 630
8.663pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference . . . 630
8.663.1 Detailed Description . . . 631
8.663.2 Field Documentation . . . 631

8.663.2.1 pHwConfig . . . 631
8.663.2.2 pProfileId . . . 631
8.663.2.3 pRequestOptionList . . . 631
8.664 pack_wds_SLQSSetLoopback_t Struct Reference . . . 631
8.664.1 Detailed Description . . . 631
8.664.2 Field Documentation . . . 632
8.664.2.1 loopbackMode . . . 632
8.664.2.2 loopbackMultiplier . . . 632
8.665 pack_wds_SLQSStartDataSession_t Struct Reference . . . 632
8.665.1 Detailed Description . . . 632
8.665.2 Field Documentation . . . 633
8.665.2.1 pAuth . . . 633
8.665.2.2 pPass . . . 633
8.665.2.3 pprofileid3gpp . . . 633
8.665.2.4 pprofileid3gpp2 . . . 633
8.665.2.5 pTech . . . 633
8.665.2.6 pUser . . . 633
8.666 pack_wds_SLQSStopDataSession_t Struct Reference . . . 633
8.666.1 Detailed Description . . . 633
8.666.2 Field Documentation . . . 633
8.666.2.1 psid . . . 633
8.667 pack_wds_SLQSSwiProfileChangeCallback_t Struct Reference . . . 633
8.667.1 Detailed Description . . . 633
8.667.2 Field Documentation . . . 633
8.667.2.1 pProfileChangeInd . . . 633
8.668 pack_wds_SLQSWdsSetEventReport_t Struct Reference . . . 634
8.668.1 Detailed Description . . . 634
8.668.2 Field Documentation . . . 635
8.668.2.1 pCurrChannelRateInd . . . 635
8.668.2.2 pCurrDataBearerTechInd . . . 635
8.668.2.3 pCurrPrefDataSysInd . . . 635
8.668.2.4 pDataBearerTechInd . . . 635
8.668.2.5 pDataCallStatusChangeInd . . . 635
8.668.2.6 pDataSystemStatusChangeInd . . . 635
8.668.2.7 pDormancyStatusInd . . . 635
8.668.2.8 pEVDOPageMonPerChangeInd . . . 635
8.668.2.9 pMIPStatusInd . . . 635
8.668.2.10 pTransferStatInd . . . 635
8.669 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 635
8.669.1 Detailed Description . . . 635

8.669.2 Field Documentation . . . 636
8.669.2.1 contextId . . . 636
8.669.2.2 contextType . . . 636
8.670 PackCreateProfileOut Struct Reference . . . 636
8.670.1 Detailed Description . . . 636
8.670.2 Field Documentation . . . 636
8.670.2.1 ExtErrorCode . . . 636
8.670.2.2 ProfileIndex . . . 636
8.670.2.3 ProfileType . . . 636
8.671 packgetDyingGaspCfg Struct Reference . . . 637
8.671.1 Detailed Description . . . 637
8.671.2 Field Documentation . . . 637
8.671.2.1 pDestSMSContent . . . 637
8.671.2.2 pDestSMSNum . . . 637
8.672 packgetDyingGaspStatistics Struct Reference . . . 637
8.672.1 Detailed Description . . . 637
8.672.2 Field Documentation . . . 638
8.672.2.1 pSMSAttemptedFlag . . . 638
8.672.2.2 pTimeStamp . . . 638
8.673 PackSwiAvmsSetSettingsAPNInfo Struct Reference . . . 638
8.673.1 Detailed Description . . . 638
8.673.2 Field Documentation . . . 638
8.673.2.1 bAPNLength . . . 638
8.673.2.2 bPWDLength . . . 638
8.673.2.3 bUnameLength . . . 638
8.673.2.4 szAPN . . . 638
8.673.2.5 szPWD . . . 638
8.673.2.6 szUname . . . 638
8.674 PackSwiAvmsSetSettingsConnectionRetryTimers Struct Reference . . . 638
8.674.1 Detailed Description . . . 638
8.674.2 Field Documentation . . . 639
8.674.2.1 Timers . . . 639
8.675 PackSwiAvmsSetSettingsPeriodInfo Struct Reference . . . 639
8.675.1 Detailed Description . . . 639
8.675.2 Field Documentation . . . 639
8.675.2.1 ulMax . . . 639
8.675.2.2 ulMin . . . 639
8.676 PackSwiAVMSSettingsAPNInfo Struct Reference . . . 640
8.676.1 Detailed Description . . . 640
8.676.2 Field Documentation . . . 640

8.676.2.1 bAPNLength . . . 640
8.676.2.2 bPWDLength . . . 640
8.676.2.3 bUnameLength . . . 640
8.676.2.4 pAPN . . . 640
8.676.2.5 pPWD . . . 640
8.676.2.6 pUname . . . 640
8.677 PackSwiAVMSSettingsConnectionRetryTimers Struct Reference . . . 640
8.677.1 Detailed Description . . . 640
8.677.2 Field Documentation . . . 641
8.677.2.1 Timers . . . 641
8.678 PackSwiAVMSSettingsPeriodsInfo Struct Reference . . . 641
8.678.1 Detailed Description . . . 641
8.678.2 Field Documentation . . . 641
8.678.2.1 max . . . 641
8.678.2.2 min . . . 641
8.679 QcqmiService Struct Reference . . . 641
8.679.1 Detailed Description . . . 642
8.679.2 Field Documentation . . . 642
8.679.2.1 devicePath . . . 642
8.679.2.2 readyToRead . . . 642
8.679.2.3 useQcqmi . . . 642
8.680 QmiHdr Struct Reference . . . 642
8.680.1 Field Documentation . . . 642
8.680.1.1 length . . . 642
8.680.1.2 messageId . . . 642
8.681 QmiService Struct Reference . . . 642
8.681.1 Detailed Description . . . 643
8.681.2 Field Documentation . . . 643
8.681.2.1 clientId . . . 644
8.681.2.2 deviceFd . . . 644
8.681.2.3 pIndicationCallback . . . 644
8.681.2.4 pIndicationCallbackContext . . . 644
8.681.2.5 pNext . . . 644
8.681.2.6 pQcqmitransport . . . 644
8.681.2.7 pTransport . . . 644
8.681.2.8 serviceType . . . 644
8.681.2.9 transactionId . . . 644
8.681.2.10 transactionIdLock . . . 644
8.681.2.11 transactionPool . . . 644
8.681.2.12 transactionPoolLock . . . 644

8.682 qmiSmsMessageList Struct Reference . . . 644
8.682.1 Detailed Description . . . 644
8.682.2 Field Documentation . . . 644
8.682.2.1 messageIndex . . . 644
8.682.2.2 messageTag . . . 644
8.683 QmiSyncObject Struct Reference . . . 644
8.683.1 Detailed Description . . . 645
8.683.2 Field Documentation . . . 645
8.683.2.1 buffer . . . 645
8.683.2.2 bufferSize . . . 645
8.683.2.3 condition . . . 645
8.683.2.4 mutex . . . 645
8.684 QmiTransaction Struct Reference . . . 645
8.684.1 Detailed Description . . . 646
8.684.2 Field Documentation . . . 646
8.684.2.1 inUse . . . 646
8.684.2.2 messageId . . . 646
8.684.2.3 pResponseCallback . . . 646
8.684.2.4 pResponseCallbackContext . . . 646
8.684.2.5 transactionId . . . 646
8.685 QmiTransactionHdr Struct Reference . . . 646
8.685.1 Field Documentation . . . 646
8.685.1.1 flags . . . 646
8.685.1.2 transactionId . . . 646
8.686 qmiWDSDataBearerTechnology Struct Reference . . . 646
8.686.1 Detailed Description . . . 647
8.686.2 Field Documentation . . . 647
8.686.2.1 currentNetwork . . . 647
8.686.2.2 ratMask . . . 647
8.686.2.3 soMask . . . 647
8.687 qmTlvResult Struct Reference . . . 647
8.687.1 Field Documentation . . . 647
8.687.1.1 DeviceError . . . 647
8.687.1.2 DeviceResult . . . 647
8.687.1.3 TlvPresenceMask . . . 647
8.687.1.4 TlvResultCode . . . 647
8.688 QMUX_HDR Struct Reference . . . 647
8.688.1 Field Documentation . . . 648
8.688.1.1 ClientId . . . 648
8.688.1.2 CtlFlags . . . 648

8.688.1.3 IfType . . . 648
8.688.1.4 Length . . . 648
8.688.1.5 QMIType . . . 648
8.689 QmuxTransport Struct Reference . . . 648
8.689.1 Detailed Description . . . 648
8.689.2 Field Documentation . . . 649
8.689.2.1 deviceFd . . . 649
8.689.2.2 pDataLayer . . . 649
8.689.2.3 qcqmi . . . 649
8.689.2.4 qmiServiceList . . . 649
8.689.2.5 qmiServiceListLock . . . 649
8.689.2.6 readThread . . . 649
8.689.2.7 shutdownFd . . . 649
8.689.2.8 useMbim . . . 649
8.689.2.9 writeLock . . . 649
8.690 qos_BindDataPortMuxID_t Struct Reference . . . 649
8.690.1 Detailed Description . . . 649
8.690.2 Field Documentation . . . 649
8.690.2.1 MuxID . . . 649
8.691 qos_BindDataPortPeripheralEndPointID_t Struct Reference . . . 649
8.691.1 Detailed Description . . . 650
8.691.2 Field Documentation . . . 650
8.691.2.1 EndPointType . . . 650
8.691.2.2 IfaceID . . . 650
8.692 qos_BindDataPortSIODataPort_t Struct Reference . . . 650
8.692.1 Detailed Description . . . 650
8.692.2 Field Documentation . . . 650
8.692.2.1 SIODataPort . . . 650
8.693 ReleaseClientIdReqMsg Struct Reference . . . 650
8.693.1 Field Documentation . . . 651
8.693.1.1 clientId . . . 651
8.693.1.2 length . . . 651
8.693.1.3 messageId . . . 651
8.693.1.4 serviceType . . . 651
8.693.1.5 tlvLength . . . 651
8.693.1.6 tlvType . . . 651
8.694 ReleaseClientIdRspMsg Struct Reference . . . 651
8.694.1 Field Documentation . . . 651
8.694.1.1 clientId . . . 651
8.694.1.2 error . . . 651

8.694.1.3 length . . . 651
8.694.1.4 messageId . . . 651
8.694.1.5 result . . . 651
8.694.1.6 serviceType . . . 651
8.694.1.7 tlv2Length . . . 651
8.694.1.8 tlv2Type . . . 651
8.694.1.9 tlvLength . . . 651
8.694.1.10 tlvType . . . 651
8.695 ResultTlv Struct Reference . . . 652
8.695.1 Field Documentation . . . 652
8.695.1.1 error . . . 652
8.695.1.2 result . . . 652
8.695.1.3 tlvLength . . . 652
8.695.1.4 tlvType . . . 652
8.696 RFBandInfoElements Struct Reference . . . 652
8.696.1 Detailed Description . . . 652
8.696.2 Field Documentation . . . 652
8.696.2.1 activeBandClass . . . 652
8.696.2.2 activeChannel . . . 652
8.696.2.3 radioInterface . . . 652
8.697 rmTrasnferStaticsReq Struct Reference . . . 653
8.697.1 Detailed Description . . . 653
8.697.2 Field Documentation . . . 653
8.697.2.1 bResetStatistics . . . 653
8.697.2.2 ulMask . . . 653
8.698 sensorData_t Struct Reference . . . 653
8.698.1 Detailed Description . . . 653
8.698.2 Field Documentation . . . 654
8.698.2.1 flags . . . 654
8.698.2.2 sensorDataLen . . . 654
8.698.2.3 timeOfFirstSample . . . 654
8.698.2.4 timeOffset . . . 654
8.698.2.5 xAxis . . . 654
8.698.2.6 yAxis . . . 655
8.698.2.7 zAxis . . . 655
8.699 slot_t Struct Reference . . . 655
8.699.1 Detailed Description . . . 655
8.699.2 Field Documentation . . . 655
8.699.2.1 bICCID . . . 655
8.699.2.2 bICCIDLength . . . 655

8.699.2.3 bLogicalSlot . . . 656
8.699.2.4 uPhyCardStatus . . . 656
8.699.2.5 uPhySlotStatus . . . 656
8.700 slotInf Struct Reference . . . 656
8.700.1 Detailed Description . . . 656
8.700.2 Field Documentation . . . 657
8.700.2.1 AppStatus . . . 657
8.700.2.2 cardState . . . 657
8.700.2.3 errorState . . . 657
8.700.2.4 numApp . . . 657
8.700.2.5 upinRetries . . . 657
8.700.2.6 upinState . . . 657
8.700.2.7 upukRetries . . . 657
8.701 slots_t Struct Reference . . . 657
8.701.1 Detailed Description . . . 657
8.701.2 Field Documentation . . . 658
8.701.2.1 uimSlotStatus . . . 658
8.702 sms_BroadcastConfig Struct Reference . . . 658
8.702.1 Detailed Description . . . 658
8.702.2 Field Documentation . . . 658
8.702.2.1 fromServiceId . . . 658
8.702.2.2 selected . . . 658
8.702.2.3 toServiceId . . . 658
8.703 sms_CDMABroadcastConfig Struct Reference . . . 658
8.703.1 Detailed Description . . . 659
8.703.2 Field Documentation . . . 659
8.703.2.1 language . . . 659
8.703.2.2 selected . . . 659
8.703.2.3 serviceCategory . . . 659
8.704 sms_getIndicationReg Struct Reference . . . 659
8.704.1 Detailed Description . . . 659
8.704.2 Field Documentation . . . 660
8.704.2.1 pRegCallStatInfoEvt . . . 660
8.704.2.2 pRegTransLayerInfoEvt . . . 660
8.704.2.3 pRegTransNWRegInfoEvt . . . 660
8.705 sms_getMsgWaitingInfo Struct Reference . . . 660
8.705.1 Detailed Description . . . 660
8.705.2 Field Documentation . . . 660
8.705.2.1 msgWaitInfo . . . 660
8.705.2.2 numInstances . . . 660

8.706 sms_getTransLayerInfo Struct Reference . . . 660
8.706.1 Detailed Description . . . 661
8.706.2 Field Documentation . . . 661
8.706.2.1 pRegInd . . . 661
8.706.2.2 pTransLayerInfo . . . 661
8.707 sms_getTransNWRegInfo Struct Reference . . . 661
8.707.1 Detailed Description . . . 661
8.707.2 Field Documentation . . . 661
8.707.2.1 pRegStatus . . . 662
8.708 sms_maxStorageSizeReq Struct Reference . . . 662
8.708.1 Detailed Description . . . 662
8.708.2 Field Documentation . . . 662
8.708.2.1 pMessageMode . . . 662
8.708.2.2 storageType . . . 662
8.709 sms_maxStorageSizeResp Struct Reference . . . 662
8.709.1 Detailed Description . . . 662
8.709.2 Field Documentation . . . 663
8.709.2.1 freeSlots . . . 663
8.709.2.2 maxStorageSize . . . 663
8.710 sms_messageWaitingInfoContent Struct Reference . . . 663
8.710.1 Detailed Description . . . 663
8.710.2 Field Documentation . . . 663
8.710.2.1 activeInd . . . 663
8.710.2.2 msgCount . . . 663
8.710.2.3 msgType . . . 663
8.711 sms_msgProtocolResp Struct Reference . . . 664
8.711.1 Detailed Description . . . 664
8.711.2 Field Documentation . . . 664
8.711.2.1 msgProtocol . . . 664
8.712 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference . . . 664
8.712.1 Detailed Description . . . 664
8.712.2 Field Documentation . . . 665
8.712.2.1 activated_ind . . . 665
8.712.2.2 CDMABroadcastConfig . . . 665
8.712.2.3 num_instances . . . 665
8.713 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference . . . 665
8.713.1 Detailed Description . . . 665
8.713.2 Field Documentation . . . 665
8.713.2.1 activated_ind . . . 665
8.713.2.2 broadcastConfig . . . 665

8.713.2.3 num_instances . . . 665
8.714 sms_routeEntry Struct Reference . . . 665
8.714.1 Detailed Description . . . 666
8.714.2 Field Documentation . . . 666
8.714.2.1 messageClass . . . 666
8.714.2.2 messageType . . . 666
8.714.2.3 receiptAction . . . 666
8.714.2.4 routeStorage . . . 666
8.715 sms_sendAsyncsmsParams Struct Reference . . . 666
8.715.1 Detailed Description . . . 667
8.715.2 Field Documentation . . . 668
8.715.2.1 messageFormat . . . 668
8.715.2.2 messageSize . . . 668
8.715.2.3 pFollowOnDC . . . 668
8.715.2.4 pForceOnDC . . . 668
8.715.2.5 pLinktimer . . . 668
8.715.2.6 pMessage . . . 668
8.715.2.7 pRetryMessage . . . 668
8.715.2.8 pRetryMessageId . . . 668
8.715.2.9 pServiceOption . . . 668
8.715.2.10 pSmsOnIms . . . 668
8.715.2.11 pUserData . . . 668
8.716 sms_setIndicationReg Struct Reference . . . 669
8.716.1 Detailed Description . . . 669
8.716.2 Field Documentation . . . 669
8.716.2.1 pRegCallStatInfoEvt . . . 669
8.716.2.2 pRegTransLayerInfoEvt . . . 669
8.716.2.3 pRegTransNWRegInfoEvt . . . 669
8.717 sms_setRoutesReq Struct Reference . . . 669
8.717.1 Detailed Description . . . 670
8.717.2 Field Documentation . . . 670
8.717.2.1 numOfRoutes . . . 670
8.717.2.2 pTransferStatusReport . . . 670
8.717.2.3 routeList . . . 670
8.718 sms_transLayerInfo Struct Reference . . . 670
8.718.1 Detailed Description . . . 670
8.718.2 Field Documentation . . . 670
8.718.2.1 TransCap . . . 671
8.718.2.2 TransType . . . 671
8.719 sMSCAddressInfo Struct Reference . . . 671

8.719.1 Detailed Description . . . 671
8.719.2 Field Documentation . . . 671
8.719.2.1 data . . . 671
8.719.2.2 length . . . 671
8.720 sMSCAddressTlv Struct Reference . . . 671
8.720.1 Detailed Description . . . 671
8.720.2 Field Documentation . . . 671
8.720.2.1 SMSCInfo . . . 672
8.720.2.2 TlvPresent . . . 672
8.721 sMSEtwsMessageInfo Struct Reference . . . 672
8.721.1 Detailed Description . . . 672
8.721.2 Field Documentation . . . 672
8.721.2.1 data . . . 672
8.721.2.2 length . . . 672
8.721.2.3 notificationType . . . 672
8.722 sMSEtwsMessageTlv Struct Reference . . . 672
8.722.1 Detailed Description . . . 672
8.722.2 Field Documentation . . . 673
8.722.2.1 EtwsMessageInfo . . . 673
8.722.2.2 TlvPresent . . . 673
8.723 sMSEtwsPlmnInfo Struct Reference . . . 673
8.723.1 Detailed Description . . . 673
8.723.2 Field Documentation . . . 673
8.723.2.1 mobileCountryCode . . . 673
8.723.2.2 mobileNetworkCode . . . 673
8.724 sMSMessageModeInfo Struct Reference . . . 673
8.724.1 Detailed Description . . . 673
8.724.2 Field Documentation . . . 674
8.724.2.1 messageMode . . . 674
8.725 sMSMTMessageInfo Struct Reference . . . 674
8.725.1 Detailed Description . . . 674
8.725.2 Field Documentation . . . 674
8.725.2.1 messageIndex . . . 674
8.725.2.2 storageType . . . 674
8.726 sMSOnIMSInfo Struct Reference . . . 674
8.726.1 Detailed Description . . . 674
8.726.2 Field Documentation . . . 674
8.726.2.1 smsOnIMS . . . 674
8.727 sMSOnIMSTlv Struct Reference . . . 675
8.727.1 Detailed Description . . . 675

8.727.2 Field Documentation . . . 675
8.727.2.1 IMSInfo . . . 675
8.727.2.2 TlvPresent . . . 675
8.728 sMSTransferRouteMTMessageInfo Struct Reference . . . 675
8.728.1 Detailed Description . . . 675
8.728.2 Field Documentation . . . 676
8.728.2.1 ackIndicator . . . 676
8.728.2.2 data . . . 676
8.728.2.3 format . . . 676
8.728.2.4 length . . . 676
8.728.2.5 transactionID . . . 676
8.729 swi_uint256_t Struct Reference . . . 676
8.729.1 Detailed Description . . . 676
8.729.2 Field Documentation . . . 676
8.729.2.1 word . . . 676
8.730 swiaudio_PCMparams Struct Reference . . . 676
8.730.1 Detailed Description . . . 676
8.730.2 Field Documentation . . . 677
8.730.2.1 iFaceTab . . . 677
8.730.2.2 iFaceTabLen . . . 677
8.731 swidms_AutoUimSwitchModeTlv Struct Reference . . . 677
8.731.1 Detailed Description . . . 677
8.731.2 Field Documentation . . . 677
8.731.2.1 TlvPresent . . . 677
8.731.2.2 uimSlot . . . 677
8.732 swidms_ehrpdMTUSizeTlv Struct Reference . . . 677
8.732.1 Detailed Description . . . 677
8.732.2 Field Documentation . . . 678
8.732.2.1 ehrpdMTUSize . . . 678
8.732.2.2 TlvPresent . . . 678
8.733 swidms_hrpdMTUSizeTlv Struct Reference . . . 678
8.733.1 Detailed Description . . . 678
8.733.2 Field Documentation . . . 678
8.733.2.1 hrpdMTUSize . . . 678
8.733.2.2 TlvPresent . . . 678
8.734 swidms_intfaceCfgTlv Struct Reference . . . 678
8.734.1 Detailed Description . . . 678
8.734.2 Field Documentation . . . 679
8.734.2.1 CfgValue . . . 679
8.734.2.2 CurrentCfgType . . . 679

8.734.2.3 TlvPresent . . . 679
8.735 swidms_mtuSize3gppTlv Struct Reference . . . 679
8.735.1 Detailed Description . . . 680
8.735.2 Field Documentation . . . 680
8.735.2.1 MTUSize3gpp . . . 680
8.735.2.2 TlvPresent . . . 680
8.736 swidms_MultiplexedSimModeTlv Struct Reference . . . 680
8.736.1 Detailed Description . . . 680
8.736.2 Field Documentation . . . 681
8.736.2.1 mode . . . 681
8.736.2.2 TlvPresent . . . 681
8.736.2.3 uimSlot . . . 681
8.737 swidms_SetMultiplexedModeTlv Struct Reference . . . 681
8.737.1 Detailed Description . . . 681
8.737.2 Field Documentation . . . 681
8.737.2.1 mode . . . 681
8.737.2.2 uimSlot . . . 681
8.738 swidms_supportedIntBitmaskTlv Struct Reference . . . 681
8.738.1 Detailed Description . . . 681
8.738.2 Field Documentation . . . 682
8.738.2.1 TlvPresent . . . 682
8.738.2.2 ValidBitmasks . . . 682
8.739 swidms_SwiDmsGetHWWatchdog Struct Reference . . . 682
8.739.1 Detailed Description . . . 682
8.739.2 Field Documentation . . . 682
8.739.2.1 count . . . 682
8.739.2.2 enable . . . 682
8.739.2.3 resetDelay . . . 682
8.739.2.4 timeout . . . 682
8.740 swidms_UimSlotSelectedTlv Struct Reference . . . 683
8.740.1 Detailed Description . . . 683
8.740.2 Field Documentation . . . 683
8.740.2.1 TlvPresent . . . 683
8.740.2.2 uimSlot . . . 683
8.741 swidms_usbMTUSizeTlv Struct Reference . . . 683
8.741.1 Detailed Description . . . 683
8.741.2 Field Documentation . . . 683
8.741.2.1 TlvPresent . . . 684
8.741.2.2 UsbMTUSize . . . 684
8.742 tdscdmaSigInfoExt Struct Reference . . . 684

8.742.1 Detailed Description . . . 684
8.742.2 Field Documentation . . . 684
8.742.2.1 ecio . . . 684
8.742.2.2 rscp . . . 684
8.742.2.3 rssi . . . 684
8.742.2.4 sinr . . . 684
8.743tempData_t Struct Reference . . . 684
8.743.1 Detailed Description . . . 685
8.743.2 Field Documentation . . . 685
8.743.2.1 temperature . . . 685
8.743.2.2 temperatureDataLen . . . 685
8.743.2.3 timeOfFirstSample . . . 685
8.743.2.4 timeOffset . . . 685
8.743.2.5 timeSource . . . 685
8.744tmd_mitigationDevList Struct Reference . . . 685
8.744.1 Detailed Description . . . 686
8.744.2 Field Documentation . . . 686
8.744.2.1 maxMitigationLevel . . . 686
8.744.2.2 mitigationDevId . . . 686
8.744.2.3 mitigationDevIdLen . . . 686
8.745transferRouteMessageTlv Struct Reference . . . 686
8.745.1 Detailed Description . . . 686
8.745.2 Field Documentation . . . 686
8.745.2.1 TlvPresent . . . 686
8.745.2.2 TransferRouteMTMessageInfo . . . 686
8.746uim_activeProfileInfo Struct Reference . . . 687
8.746.1 Detailed Description . . . 687
8.746.2 Field Documentation . . . 687
8.746.2.1 profileId . . . 687
8.746.2.2 profileType . . . 687
8.746.2.3 TlvPresent . . . 687
8.747uim_additionalReadResult Struct Reference . . . 687
8.747.1 Detailed Description . . . 688
8.747.2 Field Documentation . . . 688
8.747.2.1 additionalRecord . . . 688
8.747.2.2 additionalRecordLen . . . 688
8.747.2.3 TlvPresent . . . 688
8.748uim_appStatus Struct Reference . . . 688
8.748.1 Detailed Description . . . 688
8.748.2 Field Documentation . . . 690

8.748.2.1 aidLength . . . 690
8.748.2.2 aidVal . . . 690
8.748.2.3 appState . . . 691
8.748.2.4 appType . . . 691
8.748.2.5 persoFeature . . . 691
8.748.2.6 persoRetries . . . 691
8.748.2.7 persoState . . . 691
8.748.2.8 persoUnblockRetries . . . 691
8.748.2.9 pin1Retries . . . 691
8.748.2.10 pin1State . . . 691
8.748.2.11 pin2Retries . . . 691
8.748.2.12 pin2State . . . 691
8.748.2.13 puk1Retries . . . 691
8.748.2.14 puk2Retries . . . 691
8.748.2.15 univPin . . . 691
8.749 uim_authenticateResult Struct Reference . . . 691
8.749.1 Detailed Description . . . 691
8.749.2 Field Documentation . . . 691
8.749.2.1 content . . . 691
8.749.2.2 contentLen . . . 691
8.750 uim_authenticationData Struct Reference . . . 692
8.750.1 Detailed Description . . . 692
8.750.2 Field Documentation . . . 692
8.750.2.1 context . . . 692
8.750.2.2 data . . . 692
8.750.2.3 dataLen . . . 692
8.751 uim_cardResult Struct Reference . . . 693
8.751.1 Detailed Description . . . 693
8.751.2 Field Documentation . . . 693
8.751.2.1 sw1 . . . 693
8.751.2.2 sw2 . . . 693
8.752 uim_cardResultInfo Struct Reference . . . 693
8.752.1 Detailed Description . . . 693
8.752.2 Field Documentation . . . 693
8.752.2.1 sw1 . . . 694
8.752.2.2 sw2 . . . 694
8.752.2.3 TlvPresent . . . 694
8.753 uim_cardStatus Struct Reference . . . 694
8.753.1 Detailed Description . . . 694
8.753.2 Field Documentation . . . 695

8.753.2.1 index1xPri . . . 695
8.753.2.2 index1xSec . . . 695
8.753.2.3 indexGwPri . . . 695
8.753.2.4 indexGwSec . . . 695
8.753.2.5 numSlot . . . 695
8.753.2.6 SlotInfo . . . 695
8.754 uim_changeUIMPIN Struct Reference . . . 695
8.754.1 Detailed Description . . . 695
8.754.2 Field Documentation . . . 695
8.754.2.1 oldPINLen . . . 696
8.754.2.2 oldPINVal . . . 696
8.754.2.3 pinID . . . 696
8.754.2.4 pinLen . . . 696
8.754.2.5 pinVal . . . 696
8.755 uim_depersonalizationInformation Struct Reference . . . 696
8.755.1 Detailed Description . . . 696
8.755.2 Field Documentation . . . 697
8.755.2.1 ckLen . . . 697
8.755.2.2 ckVal . . . 697
8.755.2.3 feature . . . 697
8.755.2.4 operation . . . 697
8.756 uim_eidInfo Struct Reference . . . 697
8.756.1 Detailed Description . . . 697
8.756.2 Field Documentation . . . 697
8.756.2.1 eidVal . . . 697
8.756.2.2 eidValLen . . . 697
8.756.2.3 TlvPresent . . . 697
8.757 uim_encryptedPIN1 Struct Reference . . . 697
8.757.1 Detailed Description . . . 697
8.757.2 Field Documentation . . . 698
8.757.2.1 pin1Len . . . 698
8.757.2.2 pin1Val . . . 698
8.758 uim_fileAttributes Struct Reference . . . 698
8.758.1 Detailed Description . . . 698
8.758.2 Field Documentation . . . 701
8.758.2.1 fileID . . . 701
8.758.2.2 fileSize . . . 701
8.758.2.3 fileType . . . 701
8.758.2.4 rawLen . . . 701
8.758.2.5 rawValue . . . 701

8.758.2.6 recordCount . . . 701
8.758.2.7 recordSize . . . 701
8.758.2.8 secActivate . . . 701
8.758.2.9 secActivateMask . . . 701
8.758.2.10 secDeactivate . . . 701
8.758.2.11 secDeactivateMask . . . 701
8.758.2.12 secIncrease . . . 701
8.758.2.13 secIncreaseMask . . . 701
8.758.2.14 secRead . . . 701
8.758.2.15 secReadMask . . . 701
8.758.2.16 secWrite . . . 701
8.758.2.17 secWriteMask . . . 701
8.759 uim_fileInfo Struct Reference . . . 701
8.759.1 Detailed Description . . . 701
8.759.2 Field Documentation . . . 702
8.759.2.1 fileID . . . 702
8.759.2.2 path . . . 702
8.759.2.3 pathLen . . . 702
8.760 uim_getSimProfileInfo Struct Reference . . . 702
8.760.1 Detailed Description . . . 702
8.760.2 Field Documentation . . . 703
8.760.2.1 profileId . . . 703
8.760.2.2 profileInfoLen . . . 703
8.760.2.3 profileType . . . 703
8.760.2.4 TlvPresent . . . 703
8.761 uim_GetSlotsInfoTlv Struct Reference . . . 703
8.761.1 Detailed Description . . . 703
8.761.2 Field Documentation . . . 703
8.761.2.1 NumberOfPhySlotInfo . . . 703
8.761.2.2 TlvPresent . . . 703
8.761.2.3 uimSlotInfo . . . 703
8.762 uim_GetSlotsStatusTlv Struct Reference . . . 704
8.762.1 Detailed Description . . . 704
8.762.2 Field Documentation . . . 704
8.762.2.1 NumberOfPhySlot . . . 704
8.762.2.2 TlvPresent . . . 704
8.762.2.3 uimSlotStatus . . . 704
8.763 uim_hotSwapStatus Struct Reference . . . 704
8.763.1 Detailed Description . . . 704
8.763.2 Field Documentation . . . 705

8.763.2.1 hotSwap . . . 705
8.763.2.2 hotSwapLength . . . 705
8.764 uim_indToken Struct Reference . . . 705
8.764.1 Detailed Description . . . 705
8.764.2 Field Documentation . . . 705
8.764.2.1 TlvPresent . . . 705
8.764.2.2 token . . . 705
8.765 uim_personalizationStatus Struct Reference . . . 705
8.765.1 Detailed Description . . . 706
8.765.2 Field Documentation . . . 706
8.765.2.1 feature . . . 706
8.765.2.2 numFeatures . . . 706
8.765.2.3 unblockLeft . . . 706
8.765.2.4 verifyLeft . . . 706
8.766 uim_physlotInfo Struct Reference . . . 706
8.766.1 Detailed Description . . . 707
8.766.2 Field Documentation . . . 707
8.766.2.1 atrValue . . . 707
8.766.2.2 atrValueLen . . . 707
8.766.2.3 cardProtocol . . . 707
8.766.2.4 iseUICC . . . 707
8.766.2.5 numApp . . . 707
8.767 uim_physlotsInfo Struct Reference . . . 707
8.767.1 Detailed Description . . . 707
8.767.2 Field Documentation . . . 708
8.767.2.1 uimSlotInfo . . . 708
8.768 uim_profileClassTlv Struct Reference . . . 708
8.768.1 Detailed Description . . . 708
8.768.2 Field Documentation . . . 708
8.768.2.1 profileClass . . . 708
8.768.2.2 TlvPresent . . . 708
8.769 uim_profileICCIDTlv Struct Reference . . . 708
8.769.1 Detailed Description . . . 708
8.769.2 Field Documentation . . . 709
8.769.2.1 profileICCID . . . 709
8.769.2.2 profileICCIDLen . . . 709
8.769.2.3 TlvPresent . . . 709
8.770 uim_profileIconTlv Struct Reference . . . 709
8.770.1 Detailed Description . . . 709
8.770.2 Field Documentation . . . 709

8.770.2.1 profileIconData . . . 709
8.770.2.2 profileIconDataLen . . . 709
8.770.2.3 TlvPresent . . . 709
8.771 uim_profileIconTypeTlv Struct Reference . . . 710
8.771.1 Detailed Description . . . 710
8.771.2 Field Documentation . . . 710
8.771.2.1 profileIconType . . . 710
8.771.2.2 TlvPresent . . . 710
8.772 uim_profileNameTlv Struct Reference . . . 710
8.772.1 Detailed Description . . . 710
8.772.2 Field Documentation . . . 711
8.772.2.1 profileName . . . 711
8.772.2.2 profileNameLen . . . 711
8.772.2.3 TlvPresent . . . 711
8.773 uim_profileNicknameTlv Struct Reference . . . 711
8.773.1 Detailed Description . . . 711
8.773.2 Field Documentation . . . 711
8.773.2.1 profileNickname . . . 711
8.773.2.2 profileNicknameLen . . . 711
8.773.2.3 TlvPresent . . . 711
8.774 uim_profilePolicyRulesTlv Struct Reference . . . 711
8.774.1 Detailed Description . . . 711
8.774.2 Field Documentation . . . 712
8.774.2.1 profilePolicyRules . . . 712
8.774.2.2 TlvPresent . . . 712
8.775 uim_profileSPNTlv Struct Reference . . . 712
8.775.1 Detailed Description . . . 712
8.775.2 Field Documentation . . . 712
8.775.2.1 profileSPN . . . 712
8.775.2.2 profileSPNLen . . . 712
8.775.2.3 TlvPresent . . . 712
8.776 uim_profileStateTlv Struct Reference . . . 712
8.776.1 Detailed Description . . . 713
8.776.2 Field Documentation . . . 713
8.776.2.1 profileState . . . 713
8.776.2.2 TlvPresent . . . 713
8.777 uim_readRecordInfo Struct Reference . . . 713
8.777.1 Detailed Description . . . 713
8.777.2 Field Documentation . . . 713
8.777.2.1 length . . . 713

8.777.2.2 record . . . 714
8.778 uim_readResult Struct Reference . . . 714
8.778.1 Detailed Description . . . 714
8.778.2 Field Documentation . . . 714
8.778.2.1 content . . . 714
8.778.2.2 contentLen . . . 714
8.779 uim_readResultInfo Struct Reference . . . 714
8.779.1 Detailed Description . . . 714
8.779.2 Field Documentation . . . 715
8.779.2.1 content . . . 715
8.779.2.2 contentLen . . . 715
8.779.2.3 TlvPresent . . . 715
8.780 uim_readTransparentInfo Struct Reference . . . 715
8.780.1 Detailed Description . . . 715
8.780.2 Field Documentation . . . 715
8.780.2.1 length . . . 715
8.780.2.2 offset . . . 715
8.781 uim_refreshevent Struct Reference . . . 715
8.781.1 Detailed Description . . . 715
8.781.2 Field Documentation . . . 716
8.781.2.1 aid . . . 717
8.781.2.2 aidLength . . . 717
8.781.2.3 arrfileInfo . . . 717
8.781.2.4 mode . . . 717
8.781.2.5 numOfFiles . . . 717
8.781.2.6 sessionType . . . 717
8.781.2.7 stage . . . 717
8.782 uim_registerRefresh Struct Reference . . . 717
8.782.1 Detailed Description . . . 717
8.782.2 Field Documentation . . . 717
8.782.2.1 arrfileInfo . . . 718
8.782.2.2 numFiles . . . 718
8.782.2.3 registerFlag . . . 718
8.782.2.4 voteForInit . . . 718
8.783 uim_remainingRetries Struct Reference . . . 718
8.783.1 Detailed Description . . . 718
8.783.2 Field Documentation . . . 718
8.783.2.1 unblockLeft . . . 718
8.783.2.2 verifyLeft . . . 718
8.784 uim_sessionInformation Struct Reference . . . 718

8.784.1 Detailed Description . . . 718
8.784.2 Field Documentation . . . 719
8.784.2.1 aid . . . 719
8.784.2.2 aidLength . . . 719
8.784.2.3 sessionType . . . 719
8.785 uim_setPINProtection Struct Reference . . . 719
8.785.1 Detailed Description . . . 719
8.785.2 Field Documentation . . . 720
8.785.2.1 pinID . . . 720
8.785.2.2 pinLength . . . 720
8.785.2.3 pinOperation . . . 720
8.785.2.4 pinValue . . . 720
8.786 uim_simBusyStatus Struct Reference . . . 720
8.786.1 Detailed Description . . . 720
8.786.2 Field Documentation . . . 720
8.786.2.1 simBusy . . . 720
8.786.2.2 simBusyLength . . . 720
8.787 uim_slotInfo Struct Reference . . . 721
8.787.1 Detailed Description . . . 721
8.787.2 Field Documentation . . . 722
8.787.2.1 AppStatus . . . 722
8.787.2.2 cardState . . . 722
8.787.2.3 errorState . . . 722
8.787.2.4 numApp . . . 722
8.787.2.5 upinRetries . . . 722
8.787.2.6 upinState . . . 722
8.787.2.7 upukRetries . . . 722
8.788 uim_UIMGetFDNStatus Struct Reference . . . 722
8.788.1 Detailed Description . . . 722
8.788.2 Field Documentation . . . 723
8.788.2.1 FDNStatus . . . 723
8.788.2.2 TlvPresent . . . 723
8.789 uim_UIMGetHiddenKeyStatus Struct Reference . . . 723
8.789.1 Detailed Description . . . 723
8.789.2 Field Documentation . . . 723
8.789.2.1 hiddenKey . . . 723
8.789.2.2 TlvPresent . . . 723
8.790 uim_UIMGetIndex Struct Reference . . . 723
8.790.1 Detailed Description . . . 723
8.790.2 Field Documentation . . . 724

8.790.2.1 index . . . 724
8.790.2.2 TlvPresent . . . 724
8.791 uim_UIMNickName Struct Reference . . . 724
8.791.1 Detailed Description . . . 724
8.791.2 Field Documentation . . . 724
8.791.2.1 data . . . 724
8.791.2.2 dataLen . . . 724
8.792 uim_UIMProfileCode Struct Reference . . . 724
8.792.1 Detailed Description . . . 724
8.792.2 Field Documentation . . . 725
8.792.2.1 data . . . 725
8.792.2.2 dataLen . . . 725
8.793 uim_UIMSessionInformation Struct Reference . . . 725
8.793.1 Detailed Description . . . 725
8.793.2 Field Documentation . . . 725
8.793.2.1 aid . . . 726
8.793.2.2 aidLength . . . 726
8.793.2.3 sessionType . . . 726
8.794 uim_unblockUIMPIN Struct Reference . . . 726
8.794.1 Detailed Description . . . 726
8.794.2 Field Documentation . . . 726
8.794.2.1 newPINLen . . . 726
8.794.2.2 newPINVal . . . 726
8.794.2.3 pinID . . . 726
8.794.2.4 pukLen . . . 726
8.794.2.5 pukVal . . . 727
8.795 uim_validCardStatus Struct Reference . . . 727
8.795.1 Detailed Description . . . 727
8.795.2 Field Documentation . . . 727
8.795.2.1 validCard . . . 727
8.795.2.2 validCardLength . . . 727
8.796 uim_verifyUIMPIN Struct Reference . . . 727
8.796.1 Detailed Description . . . 727
8.796.2 Field Documentation . . . 728
8.796.2.1 pinID . . . 728
8.796.2.2 pinLen . . . 728
8.796.2.3 pinVal . . . 728
8.797 uim_writeRecordInfo Struct Reference . . . 728
8.797.1 Detailed Description . . . 728
8.797.2 Field Documentation . . . 728

8.797.2.1 data . . . 728
8.797.2.2 dataLen . . . 728
8.797.2.3 record . . . 728
8.798 uim_writeTransparentInfo Struct Reference . . . 728
8.798.1 Detailed Description . . . 729
8.798.2 Field Documentation . . . 729
8.798.2.1 data . . . 729
8.798.2.2 dataLen . . . 729
8.798.2.3 offset . . . 729
8.799 unpack_audio_SLQSGetAudioPathConfig_t Struct Reference . . . 729
8.799.1 Detailed Description . . . 729
8.799.2 Field Documentation . . . 731
8.799.2.1 ParamPresenceMask . . . 731
8.799.2.2 pCodecSTGain . . . 731
8.799.2.3 pDTMFTXGain . . . 731
8.799.2.4 pECMode . . . 731
8.799.2.5 pMICGainSelect . . . 731
8.799.2.6 pNSEnable . . . 731
8.799.2.7 pRXAGCList . . . 731
8.799.2.8 pRXAVCAGCSwitch . . . 731
8.799.2.9 pRXAVCList . . . 731
8.799.2.10 pRXPCMIIRFltr . . . 731
8.799.2.11 pTXAGCList . . . 731
8.799.2.12 pTXAVCSwitch . . . 731
8.799.2.13 pTXGain . . . 731
8.799.2.14 pTXPCMIIRFltr . . . 731
8.800 unpack_audio_SLQSGetAudioProfile_t Struct Reference . . . 731
8.800.1 Detailed Description . . . 732
8.800.2 Field Documentation . . . 732
8.800.2.1 EarMute . . . 732
8.800.2.2 MicMute . . . 732
8.800.2.3 ParamPresenceMask . . . 732
8.800.2.4 Profile . . . 732
8.800.2.5 Volume . . . 732
8.801 unpack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference . . . 732
8.801.1 Detailed Description . . . 733
8.801.2 Field Documentation . . . 733
8.801.2.1 ParamPresenceMask . . . 733
8.801.2.2 ResCode . . . 733
8.802 unpack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference . . . 733

8.802.1 Detailed Description . . . 733
8.802.2 Field Documentation . . . 733
8.802.2.1 ParamPresenceMask . . . 733
8.802.2.2 ResCode . . . 733
8.803 unpack_cat_SetCatEventCallback_ind_t Struct Reference . . . 733
8.803.1 Detailed Description . . . 734
8.803.2 Field Documentation . . . 734
8.803.2.1 CCETlv . . . 734
8.803.2.2 event_Index . . . 734
8.803.2.3 ParamPresenceMask . . . 734
8.804 unpack_cat_SetCATEventCallback_t Struct Reference . . . 734
8.804.1 Detailed Description . . . 734
8.804.2 Field Documentation . . . 735
8.804.2.1 errorMask . . . 735
8.804.2.2 ParamPresenceMask . . . 735
8.804.2.3 Tlvresult . . . 735
8.805 unpack_ctl_GetClientId_t Struct Reference . . . 735
8.805.1 Detailed Description . . . 735
8.805.2 Field Documentation . . . 736
8.805.2.1 clientId . . . 736
8.805.2.2 ParamPresenceMask . . . 736
8.805.2.3 serviceType . . . 736
8.805.2.4 Tlvresult . . . 736
8.806 unpack_ctl_ReleaseClientId_t Struct Reference . . . 736
8.806.1 Detailed Description . . . 736
8.806.2 Field Documentation . . . 736
8.806.2.1 clientId . . . 736
8.806.2.2 ParamPresenceMask . . . 736
8.806.2.3 serviceType . . . 737
8.806.2.4 Tlvresult . . . 737
8.807 unpack_dms_GetActivationState_t Struct Reference . . . 737
8.807.1 Detailed Description . . . 737
8.807.2 Field Documentation . . . 737
8.807.2.1 ParamPresenceMask . . . 737
8.807.2.2 state . . . 737
8.808 unpack_dms_GetBandCapability_t Struct Reference . . . 737
8.808.1 Detailed Description . . . 737
8.808.2 Field Documentation . . . 739
8.808.2.1 BandCapability . . . 739
8.808.2.2 ParamPresenceMask . . . 739

8.808.2.3 TlvResult . . . 739
8.809 unpack_dms_GetCrashAction_t Struct Reference . . . 739
8.809.1 Detailed Description . . . 739
8.809.2 Field Documentation . . . 739
8.809.2.1 DevCrashState . . . 739
8.809.2.2 ParamPresenceMask . . . 740
8.809.2.3 TlvResult . . . 740
8.810 unpack_dms_GetCustFeature_t Struct Reference . . . 740
8.810.1 Detailed Description . . . 740
8.810.2 Field Documentation . . . 742
8.810.2.1 DHCPRelayEnabled . . . 742
8.810.2.2 DisableIMSI . . . 742
8.810.2.3 GpsEnable . . . 742
8.810.2.4 GPSLPM . . . 742
8.810.2.5 GPSSel . . . 742
8.810.2.6 IPFamSupport . . . 742
8.810.2.7 IsVoiceEnabled . . . 742
8.810.2.8 ParamPresenceMask . . . 742
8.810.2.9 RMAutoConnect . . . 742
8.810.2.10 SMSSupport . . . 742
8.810.2.11 TlvResult . . . 742
8.811 unpack_dms_GetCustFeaturesV2_t Struct Reference . . . 742
8.811.1 Detailed Description . . . 742
8.811.2 Field Documentation . . . 743
8.811.2.1 GetCustomFeatureV2 . . . 743
8.811.2.2 ParamPresenceMask . . . 743
8.811.2.3 TlvResult . . . 743
8.812 unpack_dms_GetDeviceCap_t Struct Reference . . . 743
8.812.1 Detailed Description . . . 743
8.812.2 Field Documentation . . . 744
8.812.2.1 DataServiceCapability . . . 744
8.812.2.2 MaxRXChannelRate . . . 744
8.812.2.3 MaxTXChannelRate . . . 744
8.812.2.4 ParamPresenceMask . . . 744
8.812.2.5 RadioIfaces . . . 744
8.812.2.6 RadioIfacesSize . . . 744
8.812.2.7 SimCapability . . . 744
8.812.2.8 TlvResult . . . 744
8.813 unpack_dms_GetDeviceCapabilities_t Struct Reference . . . 744
8.813.1 Detailed Description . . . 745

8.813.2 Field Documentation . . . 745
8.813.2.1 dataServiceCaCapability . . . 746
8.813.2.2 maxRxChannelRate . . . 746
8.813.2.3 maxTxChannelRate . . . 746
8.813.2.4 ParamPresenceMask . . . 746
8.813.2.5 RadioIfaces . . . 746
8.813.2.6 radioIfacesSize . . . 746
8.813.2.7 simCapability . . . 746
8.814 unpack_dms_GetDeviceCapabilitiesV2_t Struct Reference . . . 746
8.814.1 Detailed Description . . . 746
8.814.2 Field Documentation . . . 748
8.814.2.1 DevCaps . . . 748
8.814.2.2 ParamPresenceMask . . . 748
8.814.2.3 pDevCurSubsCaps . . . 748
8.814.2.4 pDevExplicitCfgIndex . . . 748
8.814.2.5 pDevMaxActDataSubsCaps . . . 748
8.814.2.6 pDevMaxCfgListCaps . . . 748
8.814.2.7 pDevMaxSubsCaps . . . 748
8.814.2.8 pDevMultiSimCaps . . . 748
8.814.2.9 pDevMultiSimVoiceDataCaps . . . 748
8.814.2.10 pDevSrvCaps . . . 748
8.814.2.11 pDevSubsFeatureModeCaps . . . 748
8.814.2.12 pDevSubsVoiceDataCaps . . . 748
8.814.2.13 pDevVoiceCaps . . . 748
8.814.2.14 pDevVoiceDataCaps . . . 748
8.814.2.15 Tlvresult . . . 748
8.815 unpack_dms_GetDeviceHardwareRev_t Struct Reference . . . 748
8.815.1 Detailed Description . . . 749
8.815.2 Field Documentation . . . 749
8.815.2.1 ParamPresenceMask . . . 749
8.815.2.2 String . . . 749
8.815.2.3 stringSize . . . 749
8.815.2.4 Tlvresult . . . 749
8.816 unpack_dms_GetDeviceMfr_t Struct Reference . . . 749
8.816.1 Detailed Description . . . 749
8.816.2 Field Documentation . . . 750
8.816.2.1 ParamPresenceMask . . . 750
8.816.2.2 String . . . 750
8.816.2.3 stringSize . . . 750
8.816.2.4 Tlvresult . . . 750

8.817 unpack_dms_GetDeviceSerialNumbers_t Struct Reference . . . 750
8.817.1 Detailed Description . . . 750
8.817.2 Field Documentation . . . 751
8.817.2.1 esnSize . . . 752
8.817.2.2 ESNString . . . 752
8.817.2.3 imeiSize . . . 752
8.817.2.4 IMEIString . . . 752
8.817.2.5 imeiSvnSize . . . 752
8.817.2.6 ImeiSvnString . . . 752
8.817.2.7 meidSize . . . 752
8.817.2.8 MEIDString . . . 752
8.817.2.9 ParamPresenceMask . . . 752
8.817.2.10 Tlvresult . . . 752
8.818 unpack_dms_GetFirmwareInfo_t Struct Reference . . . 752
8.818.1 Detailed Description . . . 752
8.818.2 Field Documentation . . . 753
8.818.2.1 appversion_str . . . 753
8.818.2.2 bootversion_str . . . 753
8.818.2.3 carrier_str . . . 753
8.818.2.4 cur_carr_name . . . 753
8.818.2.5 cur_carr_rev . . . 753
8.818.2.6 modelid_str . . . 753
8.818.2.7 packageid_str . . . 753
8.818.2.8 ParamPresenceMask . . . 753
8.818.2.9 priversion_str . . . 753
8.818.2.10 sku_str . . . 753
8.818.2.11 Tlvresult . . . 754
8.819 unpack_dms_GetFirmwareRevision_t Struct Reference . . . 754
8.819.1 Detailed Description . . . 754
8.819.2 Field Documentation . . . 754
8.819.2.1 amssSize . . . 754
8.819.2.2 AMSSString . . . 754
8.819.2.3 ParamPresenceMask . . . 754
8.819.2.4 PRIString . . . 754
8.819.2.5 Tlvresult . . . 754
8.820 unpack_dms_GetFirmwareRevisions_t Struct Reference . . . 755
8.820.1 Detailed Description . . . 755
8.820.2 Field Documentation . . . 756
8.820.2.1 amssSize . . . 756
8.820.2.2 AMSSString . . . 756

8.820.2.3 bootSize . . . 756
8.820.2.4 BootString . . . 756
8.820.2.5 ParamPresenceMask . . . 756
8.820.2.6 priSize . . . 756
8.820.2.7 PRIString . . . 756
8.820.2.8 Tlvresult . . . 756
8.821 unpack_dms_GetFSN_t Struct Reference . . . 756
8.821.1 Detailed Description . . . 756
8.821.2 Field Documentation . . . 756
8.821.2.1 ParamPresenceMask . . . 756
8.821.2.2 String . . . 756
8.821.2.3 Tlvresult . . . 756
8.822 unpack_dms_GetHardwareRevision_t Struct Reference . . . 757
8.822.1 Detailed Description . . . 757
8.822.2 Field Documentation . . . 757
8.822.2.1 hwVer . . . 757
8.822.2.2 ParamPresenceMask . . . 757
8.823 unpack_dms_GetIMSI_t Struct Reference . . . 757
8.823.1 Detailed Description . . . 757
8.823.2 Field Documentation . . . 757
8.823.2.1 imsi . . . 758
8.823.2.2 ParamPresenceMask . . . 758
8.823.2.3 Tlvresult . . . 758
8.824 unpack_dms_GetManufacturer_t Struct Reference . . . 758
8.824.1 Detailed Description . . . 758
8.824.2 Field Documentation . . . 758
8.824.2.1 manufacturer . . . 758
8.824.2.2 ParamPresenceMask . . . 758
8.824.2.3 Tlvresult . . . 758
8.825 unpack_dms_GetModelID_t Struct Reference . . . 758
8.825.1 Detailed Description . . . 758
8.825.2 Field Documentation . . . 759
8.825.2.1 modelid . . . 759
8.825.2.2 ParamPresenceMask . . . 759
8.825.2.3 Tlvresult . . . 759
8.826 unpack_dms_GetNetworkTime_t Struct Reference . . . 759
8.826.1 Detailed Description . . . 759
8.826.2 Field Documentation . . . 760
8.826.2.1 ParamPresenceMask . . . 760
8.826.2.2 source . . . 760

8.826.2.3 timestamp . . . 760
8.826.2.4 TlvResult . . . 760
8.827 unpack_dms_GetNetworkTimeV2_t Struct Reference . . . 760
8.827.1 Detailed Description . . . 760
8.827.2 Field Documentation . . . 761
8.827.2.1 ParamPresenceMask . . . 761
8.827.2.2 pSysTime . . . 761
8.827.2.3 pUsrTime . . . 761
8.827.2.4 source . . . 761
8.827.2.5 timestamp . . . 761
8.827.2.6 TlvResult . . . 761
8.828 unpack_dms_GetOfflineReason_t Struct Reference . . . 761
8.828.1 Detailed Description . . . 761
8.828.2 Field Documentation . . . 762
8.828.2.1 ParamPresenceMask . . . 762
8.828.2.2 pbPlatform . . . 762
8.828.2.3 pReasonMask . . . 762
8.828.2.4 TlvResult . . . 762
8.829 unpack_dms_GetPower_t Struct Reference . . . 762
8.829.1 Detailed Description . . . 762
8.829.2 Field Documentation . . . 763
8.829.2.1 HardwareControlledMode . . . 763
8.829.2.2 OfflineReason . . . 763
8.829.2.3 OperationMode . . . 763
8.829.2.4 ParamPresenceMask . . . 763
8.829.2.5 TlvResult . . . 763
8.830 unpack_dms_GetPRLVersion_t Struct Reference . . . 763
8.830.1 Detailed Description . . . 763
8.830.2 Field Documentation . . . 764
8.830.2.1 ParamPresenceMask . . . 764
8.830.2.2 TlvResult . . . 764
8.830.2.3 u16PRLVersion . . . 764
8.830.2.4 u8PRLPreference . . . 764
8.831 unpack_dms_GetSerialNumbers_t Struct Reference . . . 764
8.831.1 Detailed Description . . . 764
8.831.2 Field Documentation . . . 765
8.831.2.1 esn . . . 765
8.831.2.2 imei_no . . . 765
8.831.2.3 imeisv_svn . . . 765
8.831.2.4 meid . . . 765

8.831.2.5 ParamPresenceMask . . . 765
8.832 unpack_dms_GetUSBComp_t Struct Reference . . . 765
8.832.1 Detailed Description . . . 765
8.832.2 Field Documentation . . . 767
8.832.2.1 NumSupUSBComps . . . 767
8.832.2.2 ParamPresenceMask . . . 767
8.832.2.3 SupUSBComps . . . 767
8.832.2.4 Tlvresult . . . 767
8.832.2.5 USBComp . . . 767
8.833 unpack_dms_GetVoiceNumber_t Struct Reference . . . 767
8.833.1 Detailed Description . . . 768
8.833.2 Field Documentation . . . 768
8.833.2.1 MIN . . . 768
8.833.2.2 minSize . . . 768
8.833.2.3 ParamPresenceMask . . . 768
8.833.2.4 Tlvresult . . . 768
8.833.2.5 VoiceNumber . . . 768
8.833.2.6 voiceNumberSize . . . 768
8.834 unpack_dms_PSMCfgChange_ind_t Struct Reference . . . 769
8.834.1 Detailed Description . . . 769
8.834.2 Field Documentation . . . 769
8.834.2.1 ActiveTimer . . . 769
8.834.2.2 EnableState . . . 769
8.834.2.3 ParamPresenceMask . . . 769
8.834.2.4 PeriodicUpdateTimer . . . 769
8.834.2.5 Tlvresult . . . 769
8.835 unpack_dms_ResetToFactoryDefaults_t Struct Reference . . . 769
8.835.1 Detailed Description . . . 770
8.835.2 Field Documentation . . . 770
8.835.2.1 ParamPresenceMask . . . 770
8.835.2.2 Tlvresult . . . 770
8.836 unpack_dms_SetActivationStatusCallback_t Struct Reference . . . 770
8.836.1 Detailed Description . . . 770
8.836.2 Field Documentation . . . 770
8.836.2.1 ParamPresenceMask . . . 770
8.836.2.2 Tlvresult . . . 770
8.837 unpack_dms_SetCrashAction_t Struct Reference . . . 770
8.837.1 Detailed Description . . . 771
8.837.2 Field Documentation . . . 771
8.837.2.1 notused . . . 771

8.837.2.2 ParamPresenceMask . . . 771
8.838 unpack_dms_SetCustFeature_t Struct Reference . . . 771
8.838.1 Detailed Description . . . 771
8.838.2 Field Documentation . . . 771
8.838.2.1 ParamPresenceMask . . . 771
8.838.2.2 Tlvresult . . . 771
8.839 unpack_dms_SetCustFeaturesV2_t Struct Reference . . . 771
8.839.1 Detailed Description . . . 772
8.839.2 Field Documentation . . . 772
8.839.2.1 ParamPresenceMask . . . 772
8.839.2.2 Tlvresult . . . 772
8.840 unpack_dms_SetEventReport_ind_t Struct Reference . . . 772
8.840.1 Detailed Description . . . 772
8.840.2 Field Documentation . . . 772
8.840.2.1 ActivationStatusTlv . . . 772
8.840.2.2 OperatingModeTlv . . . 773
8.840.2.3 ParamPresenceMask . . . 773
8.840.2.4 Tlvresult . . . 773
8.841 unpack_dms_SetEventReport_t Struct Reference . . . 773
8.841.1 Detailed Description . . . 773
8.841.2 Field Documentation . . . 773
8.841.2.1 ParamPresenceMask . . . 773
8.841.2.2 Tlvresult . . . 773
8.842 unpack_dms_SetFirmwarePreference_t Struct Reference . . . 773
8.842.1 Detailed Description . . . 773
8.842.2 Field Documentation . . . 774
8.842.2.1 ParamPresenceMask . . . 774
8.842.2.2 Tlvresult . . . 774
8.843 unpack_dms_SetIndicationRegister_t Struct Reference . . . 774
8.843.1 Detailed Description . . . 774
8.843.2 Field Documentation . . . 774
8.843.2.1 ParamPresenceMask . . . 774
8.843.2.2 Tlvresult . . . 774
8.844 unpack_dms_SetPower_t Struct Reference . . . 774
8.844.1 Detailed Description . . . 774
8.844.2 Field Documentation . . . 775
8.844.2.1 ParamPresenceMask . . . 775
8.844.2.2 Tlvresult . . . 775
8.845 unpack_dms_SetUSBComp_t Struct Reference . . . 775
8.845.1 Detailed Description . . . 775

8.845.2 Field Documentation . . . 775
8.845.2.1 ParamPresenceMask . . . 775
8.845.2.2 Tlvresult . . . 775
8.846 unpack_dms_SLQSDmsSwiGetPCInfo_t Struct Reference . . . 775
8.846.1 Detailed Description . . . 775
8.846.2 Field Documentation . . . 777
8.846.2.1 has_LpmFlag . . . 777
8.846.2.2 has_PersistentLpm . . . 777
8.846.2.3 has_PowerOffMode . . . 777
8.846.2.4 has_Wdisable . . . 778
8.846.2.5 LpmFlag . . . 778
8.846.2.6 opMode . . . 778
8.846.2.7 ParamPresenceMask . . . 778
8.846.2.8 PersistentLpm . . . 778
8.846.2.9 PowerOffMode . . . 778
8.846.2.10 Wdisable . . . 778
8.847 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference . . . 778
8.847.1 Detailed Description . . . 778
8.847.2 Field Documentation . . . 779
8.847.2.1 ParamPresenceMask . . . 779
8.847.2.2 source . . . 779
8.847.2.3 Tlvresult . . . 779
8.847.2.4 type . . . 779
8.848 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference . . . 779
8.848.1 Detailed Description . . . 779
8.848.2 Field Documentation . . . 780
8.848.2.1 ParamPresenceMask . . . 780
8.848.2.2 source . . . 780
8.848.2.3 Tlvresult . . . 780
8.848.2.4 type . . . 780
8.849 unpack_dms_SLQSDmsSwiGetUimSelection_t Struct Reference . . . 780
8.849.1 Detailed Description . . . 780
8.849.2 Field Documentation . . . 781
8.849.2.1 ParamPresenceMask . . . 781
8.849.2.2 pUimAutoSwitchActSlot . . . 781
8.849.2.3 uimSelect . . . 781
8.850 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 781
8.850.1 Detailed Description . . . 781
8.850.2 Field Documentation . . . 782
8.850.2.1 ParamPresenceMask . . . 782

8.850.2.2 TlvResult . . . 782
8.851 unpack_dms_SLQSGetBandCapability_t Struct Reference . . . 782
8.851.1 Detailed Description . . . 782
8.851.2 Field Documentation . . . 786
8.851.2.1 bandCapability . . . 786
8.851.2.2 is_LteBandCapability_Available . . . 786
8.851.2.3 is_TdsBandCapability_Available . . . 786
8.851.2.4 LteBandCapability . . . 786
8.851.2.5 ParamPresenceMask . . . 786
8.851.2.6 TdsBandCapability . . . 786
8.852 unpack_dms_SLQSGetBandCapabilityExt_t Struct Reference . . . 786
8.852.1 Detailed Description . . . 786
8.852.2 Field Documentation . . . 790
8.852.2.1 bandCapability . . . 790
8.852.2.2 is_LteBandCapability_Available . . . 790
8.852.2.3 is_TdsBandCapability_Available . . . 790
8.852.2.4 LteBandCapability . . . 790
8.852.2.5 LteBandsSupport . . . 790
8.852.2.6 ParamPresenceMask . . . 790
8.852.2.7 TdsBandCapability . . . 790
8.853 unpack_dms_SLQSGetERIFile_t Struct Reference . . . 790
8.853.1 Detailed Description . . . 790
8.853.2 Field Documentation . . . 791
8.853.2.1 eriFile . . . 791
8.853.2.2 ParamPresenceMask . . . 791
8.853.2.3 TlvResult . . . 791
8.854 unpack_dms_SLQSGetPowerSaveModeConfig_t Struct Reference . . . 791
8.854.1 Detailed Description . . . 791
8.854.2 Field Documentation . . . 792
8.854.2.1 pActiveTimer . . . 792
8.854.2.2 ParamPresenceMask . . . 792
8.854.2.3 pDurationDueToOOS . . . 792
8.854.2.4 pDurationThreshold . . . 792
8.854.2.5 pEarlyWakeupTime . . . 792
8.854.2.6 pPeriodicUpdateTimer . . . 792
8.854.2.7 pPsmEnableState . . . 792
8.854.2.8 pRandomizationWindow . . . 792
8.855 unpack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference . . . 792
8.855.1 Detailed Description . . . 792
8.855.2 Field Documentation . . . 793

8.855.2.1 ParamPresenceMask . . . 793
8.855.2.2 Tlvresult . . . 793
8.856 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference . . . 793
8.856.1 Detailed Description . . . 793
8.856.2 Field Documentation . . . 793
8.856.2.1 ParamPresenceMask . . . 793
8.856.2.2 Tlvresult . . . 793
8.857 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 793
8.857.1 Detailed Description . . . 793
8.857.2 Field Documentation . . . 794
8.857.2.1 crashInfoParam . . . 794
8.857.2.2 ParamPresenceMask . . . 794
8.857.2.3 Tlvresult . . . 794
8.858 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference . . . 794
8.858.1 Detailed Description . . . 794
8.858.2 Field Documentation . . . 795
8.858.2.1 ParamPresenceMask . . . 795
8.858.2.2 pGetDyingGaspCfg . . . 795
8.858.2.3 Tlvresult . . . 795
8.859 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference . . . 795
8.859.1 Detailed Description . . . 795
8.859.2 Field Documentation . . . 795
8.859.2.1 ParamPresenceMask . . . 795
8.859.2.2 pGetDyingGaspStatistics . . . 795
8.859.2.3 Tlvresult . . . 795
8.860 unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference . . . 795
8.860.1 Detailed Description . . . 796
8.860.2 Field Documentation . . . 796
8.860.2.1 carrier . . . 796
8.860.2.2 fwvers . . . 796
8.860.2.3 numEntries . . . 796
8.860.2.4 ParamPresenceMask . . . 796
8.860.2.5 pCurrImgInfo . . . 796
8.860.2.6 pkgver . . . 796
8.860.2.7 priver . . . 796
8.861 unpack_dms_SLQSSwiGetFwUpdateStatus_t Struct Reference . . . 796
8.861.1 Detailed Description . . . 797
8.861.2 Field Documentation . . . 798
8.861.2.1 imgType . . . 798
8.861.2.2 logString . . . 798

8.861.2.3 ParamPresenceMask . . . 798
8.861.2.4 refData . . . 798
8.861.2.5 refString . . . 798
8.861.2.6 ResCode . . . 798
8.861.2.7 Tlvresult . . . 798
8.862 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference . . . 798
8.862.1 Detailed Description . . . 798
8.862.2 Field Documentation . . . 799
8.862.2.1 hostID . . . 799
8.862.2.2 manString . . . 799
8.862.2.3 modelString . . . 799
8.862.2.4 ParamPresenceMask . . . 799
8.862.2.5 plasmaIDString . . . 799
8.862.2.6 swVerString . . . 799
8.862.2.7 Tlvresult . . . 799
8.863 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference . . . 799
8.863.1 Detailed Description . . . 800
8.863.2 Field Documentation . . . 800
8.863.2.1 nameString . . . 800
8.863.2.2 ParamPresenceMask . . . 800
8.863.2.3 Tlvresult . . . 800
8.863.2.4 versionString . . . 800
8.864 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference . . . 800
8.864.1 Detailed Description . . . 800
8.864.2 Field Documentation . . . 801
8.864.2.1 meidString . . . 801
8.864.2.2 ParamPresenceMask . . . 801
8.864.2.3 Tlvresult . . . 801
8.865 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 801
8.865.1 Detailed Description . . . 801
8.865.2 Field Documentation . . . 801
8.865.2.1 ParamPresenceMask . . . 801
8.865.2.2 Tlvresult . . . 801
8.866 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 801
8.866.1 Detailed Description . . . 801
8.866.2 Field Documentation . . . 802
8.866.2.1 ParamPresenceMask . . . 802
8.866.2.2 Tlvresult . . . 802
8.867 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 802
8.867.1 Detailed Description . . . 802

8.867.2 Field Documentation . . . 802
8.867.2.1 ParamPresenceMask . . . 802
8.867.2.2 Tlvresult . . . 802
8.868 unpack_dms_SLQSUIMGetState_t Struct Reference . . . 802
8.868.1 Detailed Description . . . 802
8.868.2 Field Documentation . . . 803
8.868.2.1 ParamPresenceMask . . . 803
8.868.2.2 state . . . 803
8.868.2.3 Tlvresult . . . 803
8.869 unpack_dms_SwiEventReportCallBack_ind_t Struct Reference . . . 803
8.869.1 Detailed Description . . . 803
8.869.2 Field Documentation . . . 804
8.869.2.1 ParamPresenceMask . . . 804
8.869.2.2 TempTlv . . . 804
8.869.2.3 UimStatusTlv . . . 804
8.869.2.4 VoltTlv . . . 804
8.870 unpack_dms_SwiSetEventReport_t Struct Reference . . . 804
8.870.1 Detailed Description . . . 804
8.870.2 Field Documentation . . . 804
8.870.2.1 ParamPresenceMask . . . 804
8.870.2.2 Tlvresult . . . 804
8.871 unpack_dms_SwiUimSelect_t Struct Reference . . . 804
8.871.1 Detailed Description . . . 804
8.871.2 Field Documentation . . . 805
8.871.2.1 ParamPresenceMask . . . 805
8.871.2.2 Tlvresult . . . 805
8.872 unpack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 805
8.872.1 Detailed Description . . . 805
8.872.2 Field Documentation . . . 806
8.872.2.1 facilityState . . . 806
8.872.2.2 ParamPresenceMask . . . 806
8.872.2.3 Tlvresult . . . 806
8.872.2.4 unblockRetriesLeft . . . 806
8.872.2.5 verifyRetriesLeft . . . 806
8.873 unpack_dms_UIMGetICCID_t Struct Reference . . . 806
8.873.1 Detailed Description . . . 806
8.873.2 Field Documentation . . . 806
8.873.2.1 ParamPresenceMask . . . 806
8.873.2.2 String . . . 806
8.873.2.3 stringSize . . . 806

8.873.2.4 TlvResult . . . 806
8.874 unpack_dms_UIMGetPINStatus_t Struct Reference . . . 807
8.874.1 Detailed Description . . . 807
8.874.2 Field Documentation . . . 808
8.874.2.1 p1Status . . . 808
8.874.2.2 p1UnblockRetriesLeft . . . 808
8.874.2.3 p1VerifyRetriesLeft . . . 808
8.874.2.4 p2Status . . . 808
8.874.2.5 p2UnblockRetriesLeft . . . 808
8.874.2.6 p2VerifyRetriesLeft . . . 808
8.874.2.7 ParamPresenceMask . . . 808
8.874.2.8 TlvResult . . . 808
8.875 unpack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 808
8.875.1 Detailed Description . . . 808
8.875.2 Field Documentation . . . 809
8.875.2.1 ParamPresenceMask . . . 809
8.875.2.2 TlvResult . . . 809
8.875.2.3 verifyRetriesLeft . . . 809
8.876 unpack_dms_UIMSetPINProtection_t Struct Reference . . . 809
8.876.1 Detailed Description . . . 809
8.876.2 Field Documentation . . . 810
8.876.2.1 ParamPresenceMask . . . 810
8.876.2.2 TlvResult . . . 810
8.876.2.3 unblockRetriesLeft . . . 810
8.876.2.4 verifyRetriesLeft . . . 810
8.877 unpack_dms_UIMUnblockControlKey_t Struct Reference . . . 810
8.877.1 Detailed Description . . . 810
8.877.2 Field Documentation . . . 810
8.877.2.1 ParamPresenceMask . . . 810
8.877.2.2 TlvResult . . . 810
8.877.2.3 unblockRetriesLeft . . . 810
8.878 unpack_dsd_SLQSGetDsdUIInfo_t Struct Reference . . . 810
8.878.1 Detailed Description . . . 811
8.878.2 Field Documentation . . . 811
8.878.2.1 Apn_info_len . . . 811
8.878.2.2 ParamPresenceMask . . . 811
8.878.2.3 pDsdApnAvailableSystemInfo . . . 811
8.878.2.4 pDsdGlobalUIInformation . . . 811
8.878.2.5 pDsdNullBearerReason . . . 811
8.879 unpack_dsd_SLQSGetSystemStatus_t Struct Reference . . . 811

8.879.1 Detailed Description . . . 812
8.879.2 Field Documentation . . . 812
8.879.2.1 Apn_avail_sys_info_len . . . 812
8.879.2.2 Apn_pref_sys_len . . . 812
8.879.2.3 Avail_sys_len . . . 812
8.879.2.4 ParamPresenceMask . . . 812
8.879.2.5 pDsdApnAvailableSystemInfo . . . 812
8.879.2.6 pDsdApnCurRecGlobalPrefSystem . . . 812
8.879.2.7 pDsdAvailableSystem . . . 812
8.879.2.8 pDsdCurRecGlobalPrefSystem . . . 812
8.879.2.9 pDsdNullBearerReason . . . 813
8.880 unpack_dsd_SLQSSystemStatusChange_Ind_t Struct Reference . . . 813
8.880.1 Detailed Description . . . 813
8.880.2 Field Documentation . . . 814
8.880.2.1 Apn_avail_sys_info_len . . . 814
8.880.2.2 Apn_pref_sys_len . . . 814
8.880.2.3 Avail_sys_len . . . 814
8.880.2.4 CurRecGlobalPrefTlvPresent . . . 814
8.880.2.5 NullBearerReasonTlvPresent . . . 814
8.880.2.6 ParamPresenceMask . . . 814
8.880.2.7 pDsdApnAvailableSystemInfo . . . 814
8.880.2.8 pDsdApnCurRecGlobalPrefSystem . . . 814
8.880.2.9 pDsdAvailableSystem . . . 814
8.880.2.10 pDsdCurRecGlobalPrefSystem . . . 814
8.880.2.11 pDsdNullBearerReason . . . 814
8.880.2.12 Tlvresult . . . 814
8.881 unpack_dsd_SLQSUIInfo_Ind_t Struct Reference . . . 814
8.881.1 Detailed Description . . . 814
8.881.2 Field Documentation . . . 815
8.881.2.1 Apn_info_len . . . 815
8.881.2.2 GlobalUIInfoTlvPresent . . . 815
8.881.2.3 NullBearerReasonTlvPresent . . . 815
8.881.2.4 ParamPresenceMask . . . 815
8.881.2.5 pDsdApnAvailableSystemInfo . . . 815
8.881.2.6 pDsdGlobalUIInformation . . . 815
8.881.2.7 pDsdNullBearerReason . . . 815
8.881.2.8 Tlvresult . . . 815
8.882 unpack_fms_GetImagesPreference_t Struct Reference . . . 815
8.882.1 Detailed Description . . . 816
8.882.2 Field Documentation . . . 816

8.882.2.1 ImageListSize . . . 816
8.882.2.2 ParamPresenceMask . . . 816
8.882.2.3 pImageList . . . 816
8.882.2.4 Tlvresult . . . 816
8.883 unpack_fms_GetStoredImages_t Struct Reference . . . 816
8.883.1 Detailed Description . . . 816
8.883.2 Field Documentation . . . 817
8.883.2.1 imageList . . . 817
8.883.2.2 imagelistSize . . . 817
8.883.2.3 ParamPresenceMask . . . 817
8.883.2.4 Tlvresult . . . 817
8.884 unpack_fms_SetImagesPreference_t Struct Reference . . . 817
8.884.1 Detailed Description . . . 817
8.884.2 Field Documentation . . . 817
8.884.2.1 ImageTypes . . . 817
8.884.2.2 ImageTypesSize . . . 817
8.884.2.3 ParamPresenceMask . . . 817
8.884.2.4 Tlvresult . . . 817
8.885 unpack_ims_SLQSGetIMSSMSConfig_t Struct Reference . . . 817
8.885.1 Detailed Description . . . 818
8.885.2 Field Documentation . . . 818
8.885.2.1 ParamPresenceMask . . . 818
8.885.2.2 pPhoneCtxtURI . . . 818
8.885.2.3 pPhoneCtxtURILen . . . 818
8.885.2.4 pSettingResp . . . 818
8.885.2.5 pSMSFormat . . . 818
8.885.2.6 pSMSOverIPNwInd . . . 818
8.886 unpack_ims_SLQSGetIMSUserConfig_t Struct Reference . . . 819
8.886.1 Detailed Description . . . 819
8.886.2 Field Documentation . . . 819
8.886.2.1 ParamPresenceMask . . . 819
8.886.2.2 pIMSDomain . . . 819
8.886.2.3 pIMSDomainLen . . . 819
8.886.2.4 pSettingResp . . . 819
8.887 unpack_ims_SLQSGetIMSVoIPConfig_t Struct Reference . . . 819
8.887.1 Detailed Description . . . 820
8.887.2 Field Documentation . . . 822
8.887.2.1 pAmrMode . . . 822
8.887.2.2 pAmrOctetAligned . . . 822
8.887.2.3 pAmrWbEnable . . . 822

8.887.2.4 pAmrWBMode . . . 822
8.887.2.5 pAmrWBOctetAligned . . . 822
8.887.2.6 ParamPresenceMask . . . 822
8.887.2.7 pMinSessionExpiryTimer . . . 822
8.887.2.8 pRingBackTimer . . . 822
8.887.2.9 pRingingTimer . . . 822
8.887.2.10 pRTPRTCPInactTimer . . . 822
8.887.2.11 pScrAmrEnable . . . 822
8.887.2.12 pScrAmrWbEnable . . . 822
8.887.2.13 pSessionExpiryTimer . . . 822
8.887.2.14 pSettingResp . . . 822
8.888 unpack_ims_SLQSGetRegMgrConfig_t Struct Reference . . . 822
8.888.1 Detailed Description . . . 822
8.888.2 Field Documentation . . . 823
8.888.2.1 ParamPresenceMask . . . 823
8.888.2.2 pIMSTestMode . . . 823
8.888.2.3 pPCSCFPort . . . 823
8.888.2.4 pPriCSCFPortName . . . 823
8.888.2.5 pPriCSCFPortNameLen . . . 823
8.888.2.6 pSettingResp . . . 823
8.889 unpack_ims_SLQSGetSIPConfig_t Struct Reference . . . 823
8.889.1 Detailed Description . . . 824
8.889.2 Field Documentation . . . 824
8.889.2.1 ParamPresenceMask . . . 824
8.889.2.2 pSettingResp . . . 824
8.889.2.3 pSigCompEnabled . . . 824
8.889.2.4 pSIPLocalPort . . . 824
8.889.2.5 pSubscribeTimer . . . 825
8.889.2.6 pTimerSIPReg . . . 825
8.889.2.7 pTimerT1 . . . 825
8.889.2.8 pTimerT2 . . . 825
8.889.2.9 pTimerTf . . . 825
8.890 unpack_ims_SLQSRegMgrCfgCallBack_ind_t Struct Reference . . . 825
8.890.1 Detailed Description . . . 825
8.890.2 Field Documentation . . . 825
8.890.2.1 ParamPresenceMask . . . 825
8.890.2.2 PCTlv . . . 825
8.890.2.3 PNTlv . . . 825
8.890.2.4 TMTlv . . . 826
8.891 unpack_ims_SLQSSetIMSSMSConfig_t Struct Reference . . . 826

8.891.1 Detailed Description . . . 826
8.891.2 Field Documentation . . . 826
8.891.2.1 ParamPresenceMask . . . 826
8.891.2.2 pSettingResp . . . 826
8.892 unpack_ims_SLQSSetIMSUserConfig_t Struct Reference . . . 826
8.892.1 Detailed Description . . . 826
8.892.2 Field Documentation . . . 826
8.892.2.1 ParamPresenceMask . . . 826
8.892.2.2 pSettingResp . . . 826
8.893 unpack_ims_SLQSSetIMSVoIPConfig_t Struct Reference . . . 827
8.893.1 Detailed Description . . . 827
8.893.2 Field Documentation . . . 827
8.893.2.1 ParamPresenceMask . . . 827
8.893.2.2 pSettingResp . . . 827
8.894 unpack_ims_SLQSSetRegMgrConfig_t Struct Reference . . . 827
8.894.1 Detailed Description . . . 827
8.894.2 Field Documentation . . . 827
8.894.2.1 ParamPresenceMask . . . 827
8.894.2.2 pSettingResp . . . 827
8.895 unpack_ims_SLQSSetSIPConfig_t Struct Reference . . . 827
8.895.1 Detailed Description . . . 828
8.895.2 Field Documentation . . . 828
8.895.2.1 ParamPresenceMask . . . 828
8.895.2.2 pSettingResp . . . 828
8.896 unpack_ims_SLQSSIPCfgCallBack_ind_t Struct Reference . . . 828
8.896.1 Detailed Description . . . 828
8.896.2 Field Documentation . . . 829
8.896.2.1 ParamPresenceMask . . . 829
8.896.2.2 SCTlv . . . 829
8.896.2.3 SPTlv . . . 829
8.896.2.4 SRTlv . . . 829
8.896.2.5 STTlv . . . 829
8.896.2.6 TT1Tlv . . . 829
8.896.2.7 TT2Tlv . . . 829
8.896.2.8 TTfTlv . . . 829
8.897 unpack_ims_SLQSSMSCfgCallBack_ind_t Struct Reference . . . 829
8.897.1 Detailed Description . . . 830
8.897.2 Field Documentation . . . 830
8.897.2.1 ParamPresenceMask . . . 830
8.897.2.2 PCURTlv . . . 830

8.897.2.3 SFTlv . . . 830
8.897.2.4 SINTlv . . . 830
8.898 unpack_ims_SLQSUserCfgCallBack_ind_t Struct Reference . . . 830
8.898.1 Detailed Description . . . 830
8.898.2 Field Documentation . . . 831
8.898.2.1 IDTlv . . . 831
8.898.2.2 ParamPresenceMask . . . 831
8.899 unpack_ims_SLQSVoIPCfgCallBack_ind_t Struct Reference . . . 831
8.899.1 Detailed Description . . . 831
8.899.2 Field Documentation . . . 833
8.899.2.1 AMTlv . . . 833
8.899.2.2 AOATlv . . . 833
8.899.2.3 AWMTlv . . . 833
8.899.2.4 AWOATlv . . . 833
8.899.2.5 EAWTlv . . . 833
8.899.2.6 ESATlv . . . 833
8.899.2.7 ESAWTlv . . . 833
8.899.2.8 MSETlv . . . 833
8.899.2.9 ParamPresenceMask . . . 833
8.899.2.10 RBTTlv . . . 833
8.899.2.11 RTIDTlv . . . 833
8.899.2.12 RTTlv . . . 833
8.899.2.13 SDTlv . . . 833
8.900 unpack_imsa_SLQSGetIMSARegStatus_t Struct Reference . . . 833
8.900.1 Detailed Description . . . 833
8.900.2 Field Documentation . . . 834
8.900.2.1 ImsRegErrCode . . . 834
8.900.2.2 ImsRegStatus . . . 834
8.900.2.3 NewImsRegStatus . . . 834
8.900.2.4 ParamPresenceMask . . . 834
8.901 unpack_imsa_SLQSGetIMSAServiceStatus_t Struct Reference . . . 834
8.901.1 Detailed Description . . . 834
8.901.2 Field Documentation . . . 836
8.901.2.1 ParamPresenceMask . . . 836
8.901.2.2 SmsServiceRat . . . 836
8.901.2.3 SmsServiceStatus . . . 836
8.901.2.4 UtServiceRat . . . 836
8.901.2.5 UtServiceStatus . . . 836
8.901.2.6 VoipServiceRat . . . 836
8.901.2.7 VoipServiceStatus . . . 836

8.901.2.8 VsServiceRat . . . 836
8.901.2.9 VsServiceStatus . . . 836
8.901.2.10VtServiceRat . . . 837
8.901.2.11VtServiceStatus . . . 837
8.902unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t Struct Reference . . . 837
8.902.1 Detailed Description . . . 837
8.902.2 Field Documentation . . . 837
8.902.2.1 FailErrCode . . . 837
8.902.2.2 ParamPresenceMask . . . 837
8.902.2.3 PdpConnState . . . 837
8.903unpack_imsa_SLQSImsaRatStatusCallBack_ind_t Struct Reference . . . 837
8.903.1 Detailed Description . . . 837
8.903.2 Field Documentation . . . 838
8.903.2.1 ParamPresenceMask . . . 838
8.903.2.2 RatHandover . . . 838
8.904unpack_imsa_SLQSImsaRegStatusCallBack_ind_t Struct Reference . . . 838
8.904.1 Detailed Description . . . 838
8.904.2 Field Documentation . . . 839
8.904.2.1 IMSRegistration . . . 839
8.904.2.2 IMSRegistrationError . . . 839
8.904.2.3 NewIMSRegistration . . . 839
8.904.2.4 ParamPresenceMask . . . 839
8.905unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t Struct Reference . . . 839
8.905.1 Detailed Description . . . 839
8.905.2 Field Documentation . . . 840
8.905.2.1 ParamPresenceMask . . . 840
8.905.2.2 SmsRat . . . 840
8.905.2.3 SmsService . . . 840
8.905.2.4 UtRat . . . 840
8.905.2.5 UtService . . . 840
8.905.2.6 VoipRat . . . 840
8.905.2.7 VoipService . . . 840
8.905.2.8 VtRat . . . 840
8.905.2.9 VtService . . . 840
8.906unpack_loc_BestAvailPos_Ind_t Struct Reference . . . 840
8.906.1 Detailed Description . . . 841
8.906.2 Field Documentation . . . 846
8.906.2.1 pAltitudeWrtEllipsoid . . . 846
8.906.2.2 pAltitudeWrtMeanSeaLevel . . . 846
8.906.2.3 ParamPresenceMask . . . 846

8.906.2.4 pGpsTime . . . 846
8.906.2.5 pHeading . . . 846
8.906.2.6 pHeadingUnc . . . 846
8.906.2.7 pHorCirConf . . . 846
8.906.2.8 pHorEllpConf . . . 846
8.906.2.9 pHorReliability . . . 846
8.906.2.10 pHorUncCircular . . . 846
8.906.2.11 pHorUncEllipseOrientAzimuth . . . 846
8.906.2.12 pHorUncEllipseSemiMajor . . . 846
8.906.2.13 pHorUncEllipseSemiMinor . . . 846
8.906.2.14 pLatitude . . . 846
8.906.2.15 pLongitude . . . 846
8.906.2.16 pMagneticDeviation . . . 846
8.906.2.17 pPrecisionDilution . . . 846
8.906.2.18 pSensorDataUsage . . . 846
8.906.2.19 pSpeedHorizontal . . . 846
8.906.2.20 pSpeedUnc . . . 846
8.906.2.21 pSpeedVertical . . . 846
8.906.2.22 pSpeedVerticalUnc . . . 846
8.906.2.23 pSvUsedforFix . . . 846
8.906.2.24 pTechnologyMask . . . 846
8.906.2.25 pTimeSrc . . . 847
8.906.2.26 pTimestampUtc . . . 847
8.906.2.27 pTimeUnc . . . 847
8.906.2.28 pVertConfidence . . . 847
8.906.2.29 pVertReliability . . . 847
8.906.2.30 pVertUnc . . . 847
8.906.2.31 pXid . . . 847
8.906.2.32 status . . . 847
8.906.2.33 TlvResult . . . 847
8.907 unpack_loc_CradleMountCallback_Ind_t Struct Reference . . . 847
8.907.1 Detailed Description . . . 847
8.907.2 Field Documentation . . . 847
8.907.2.1 cradleMountConfigStatus . . . 848
8.907.2.2 ParamPresenceMask . . . 848
8.908 unpack_loc_Delete_Assist_Data_t Struct Reference . . . 848
8.908.1 Detailed Description . . . 848
8.908.2 Field Documentation . . . 848
8.908.2.1 ParamPresenceMask . . . 848
8.908.2.2 TlvResult . . . 848

8.909 unpack_loc_DeleteAssistData_Ind_t Struct Reference . . . 848
8.909.1 Detailed Description . . . 848
8.909.2 Field Documentation . . . 849
8.909.2.1 ParamPresenceMask . . . 849
8.909.2.2 status . . . 849
8.909.2.3 Tlvresult . . . 849
8.910 unpack_loc_EngineState_Ind_t Struct Reference . . . 849
8.910.1 Detailed Description . . . 849
8.910.2 Field Documentation . . . 850
8.910.2.1 engineState . . . 850
8.910.2.2 ParamPresenceMask . . . 850
8.910.2.3 Tlvresult . . . 850
8.911 unpack_loc_EventNMEA_Ind_t Struct Reference . . . 850
8.911.1 Detailed Description . . . 850
8.911.2 Field Documentation . . . 850
8.911.2.1 NMEAData . . . 851
8.911.2.2 ParamPresenceMask . . . 851
8.911.2.3 Tlvresult . . . 851
8.912 unpack_loc_EventRegister_t Struct Reference . . . 851
8.912.1 Detailed Description . . . 851
8.912.2 Field Documentation . . . 851
8.912.2.1 ParamPresenceMask . . . 851
8.912.2.2 Tlvresult . . . 851
8.913 unpack_loc_EventTimeSyncCallback_Ind_t Struct Reference . . . 851
8.913.1 Detailed Description . . . 851
8.913.2 Field Documentation . . . 852
8.913.2.1 ParamPresenceMask . . . 852
8.913.2.2 timeSyncRefCounter . . . 852
8.914 unpack_loc_FixCriteria_Ind_t Struct Reference . . . 852
8.914.1 Detailed Description . . . 852
8.914.2 Field Documentation . . . 852
8.914.2.1 loc_appProviderInfo . . . 853
8.914.2.2 loc_fixCriteriaStatus . . . 853
8.914.2.3 loc_horAccuracy . . . 853
8.914.2.4 loc_intermediateRptState . . . 853
8.914.2.5 loc_minInterval . . . 853
8.914.2.6 ParamPresenceMask . . . 853
8.915 unpack_loc_GetOpMode_Ind_t Struct Reference . . . 853
8.915.1 Detailed Description . . . 853
8.915.2 Field Documentation . . . 854

8.915.2.1 ParamPresenceMask . . . 854
8.915.2.2 pMode . . . 854
8.915.2.3 Status . . . 854
8.916 unpack_loc_GetServer_Ind_t Struct Reference . . . 854
8.916.1 Detailed Description . . . 854
8.916.2 Field Documentation . . . 855
8.916.2.1 ParamPresenceMask . . . 855
8.916.2.2 pIPv4AddrInfo . . . 855
8.916.2.3 pIPv6AddrInfo . . . 856
8.916.2.4 pURL . . . 856
8.916.2.5 serverStatus . . . 856
8.916.2.6 serverType . . . 856
8.916.2.7 Tlvresult . . . 856
8.917 unpack_loc_GnssSvInfo_Ind_t Struct Reference . . . 856
8.917.1 Detailed Description . . . 856
8.917.2 Field Documentation . . . 856
8.917.2.1 altitudeAssumed . . . 856
8.917.2.2 ParamPresenceMask . . . 856
8.917.2.3 pSatelliteInfo . . . 856
8.917.2.4 Tlvresult . . . 856
8.918 unpack_loc_InjectPositionCallback_Ind_t Struct Reference . . . 857
8.918.1 Detailed Description . . . 857
8.918.2 Field Documentation . . . 857
8.918.2.1 ParamPresenceMask . . . 857
8.918.2.2 status . . . 857
8.919 unpack_loc_InjectSensorDataCallback_Ind_t Struct Reference . . . 857
8.919.1 Detailed Description . . . 858
8.919.2 Field Documentation . . . 859
8.919.2.1 injectSensorDataStatus . . . 859
8.919.2.2 pAccelSamplesAccepted . . . 859
8.919.2.3 pAccelTempSamplesAccepted . . . 859
8.919.2.4 ParamPresenceMask . . . 859
8.919.2.5 pGyroSamplesAccepted . . . 859
8.919.2.6 pGyroTempSamplesAccepted . . . 859
8.919.2.7 pOpaqueIdentifier . . . 859
8.920 unpack_loc_InjectTimeSyncDataCallback_Ind_t Struct Reference . . . 859
8.920.1 Detailed Description . . . 859
8.920.2 Field Documentation . . . 860
8.920.2.1 injectTimeSyncStatus . . . 860
8.920.2.2 ParamPresenceMask . . . 860

8.921 unpack_loc_InjectUTCTimeCallback_Ind_t Struct Reference . . . 860
8.921.1 Detailed Description . . . 860
8.921.2 Field Documentation . . . 860
8.921.2.1 ParamPresenceMask . . . 860
8.921.2.2 status . . . 861
8.922 unpack_loc_PositionRpt_Ind_t Struct Reference . . . 861
8.922.1 Detailed Description . . . 861
8.922.2 Field Documentation . . . 866
8.922.2.1 pAltitudeAssumed . . . 866
8.922.2.2 pAltitudeWrtEllipsoid . . . 866
8.922.2.3 pAltitudeWrtMeanSeaLevel . . . 866
8.922.2.4 ParamPresenceMask . . . 866
8.922.2.5 pFixId . . . 866
8.922.2.6 pGpsTime . . . 866
8.922.2.7 pHeading . . . 866
8.922.2.8 pHeadingUnc . . . 866
8.922.2.9 pHorConfidence . . . 866
8.922.2.10 pHorReliability . . . 866
8.922.2.11 pHorUncCircular . . . 866
8.922.2.12 pHorUncEllipseOrientAzimuth . . . 866
8.922.2.13 pHorUncEllipseSemiMajor . . . 866
8.922.2.14 pHorUncEllipseSemiMinor . . . 867
8.922.2.15 pLatitude . . . 867
8.922.2.16 pLeapSeconds . . . 867
8.922.2.17 pLongitude . . . 867
8.922.2.18 pMagneticDeviation . . . 867
8.922.2.19 pPrecisionDilution . . . 867
8.922.2.20 pSensorDataUsage . . . 867
8.922.2.21 pSpeedHorizontal . . . 867
8.922.2.22 pSpeedUnc . . . 867
8.922.2.23 pSpeedVertical . . . 867
8.922.2.24 pSvUsedforFix . . . 867
8.922.2.25 pTechnologyMask . . . 867
8.922.2.26 pTimeSrc . . . 867
8.922.2.27 pTimestampUtc . . . 867
8.922.2.28 pTimeUnc . . . 867
8.922.2.29 pVertConfidence . . . 867
8.922.2.30 pVertReliability . . . 867
8.922.2.31 pVertUnc . . . 867
8.922.2.32 sessionId . . . 867

8.922.2.33 sessionStatus . . . 867
8.922.2.34 TlvResult . . . 867
8.923 unpack_loc_SensorStreamingCallback_Ind_t Struct Reference . . . 867
8.923.1 Detailed Description . . . 867
8.923.2 Field Documentation . . . 868
8.923.2.1 pAccelAcceptReady . . . 868
8.923.2.2 pAccelTempAcceptReady . . . 868
8.923.2.3 ParamPresenceMask . . . 868
8.923.2.4 pGyroAcceptReady . . . 868
8.923.2.5 pGyroTempAcceptReady . . . 868
8.924 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference . . . 868
8.924.1 Detailed Description . . . 868
8.924.2 Field Documentation . . . 869
8.924.2.1 ParamPresenceMask . . . 869
8.924.2.2 status . . . 869
8.924.2.3 TlvResult . . . 869
8.925 unpack_loc_SetExtPowerState_t Struct Reference . . . 869
8.925.1 Detailed Description . . . 869
8.925.2 Field Documentation . . . 870
8.925.2.1 ParamPresenceMask . . . 870
8.925.2.2 TlvResult . . . 870
8.926 unpack_loc_SetOperationMode_Ind_t Struct Reference . . . 870
8.926.1 Detailed Description . . . 870
8.926.2 Field Documentation . . . 870
8.926.2.1 ParamPresenceMask . . . 870
8.926.2.2 status . . . 870
8.926.2.3 TlvResult . . . 870
8.927 unpack_loc_SetOperationMode_t Struct Reference . . . 871
8.927.1 Detailed Description . . . 871
8.927.2 Field Documentation . . . 871
8.927.2.1 ParamPresenceMask . . . 871
8.927.2.2 TlvResult . . . 871
8.928 unpack_loc_SetServer_Ind_t Struct Reference . . . 871
8.928.1 Detailed Description . . . 871
8.928.2 Field Documentation . . . 872
8.928.2.1 ParamPresenceMask . . . 872
8.928.2.2 serverStatus . . . 872
8.928.2.3 TlvResult . . . 872
8.929 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 872
8.929.1 Detailed Description . . . 872

8.929.2 Field Documentation . . . 873
8.929.2.1 ParamPresenceMask . . . 873
8.929.2.2 TlvResult . . . 873
8.930 unpack_loc_SLQSLOCGetOpMode_t Struct Reference . . . 873
8.930.1 Detailed Description . . . 873
8.930.2 Field Documentation . . . 873
8.930.2.1 ParamPresenceMask . . . 873
8.930.2.2 TlvResult . . . 873
8.931 unpack_loc_Start_t Struct Reference . . . 873
8.931.1 Detailed Description . . . 873
8.931.2 Field Documentation . . . 874
8.931.2.1 ParamPresenceMask . . . 874
8.931.2.2 TlvResult . . . 874
8.932 unpack_loc_Stop_t Struct Reference . . . 874
8.932.1 Detailed Description . . . 874
8.932.2 Field Documentation . . . 874
8.932.2.1 ParamPresenceMask . . . 874
8.932.2.2 TlvResult . . . 874
8.933 unpack_nas_GetACCOLC_t Struct Reference . . . 874
8.933.1 Detailed Description . . . 875
8.933.2 Field Documentation . . . 875
8.933.2.1 pAccolc . . . 875
8.933.2.2 ParamPresenceMask . . . 875
8.934 unpack_nas_GetANAAAAuthenticationStatus_t Struct Reference . . . 875
8.934.1 Detailed Description . . . 875
8.934.2 Field Documentation . . . 875
8.934.2.1 ParamPresenceMask . . . 875
8.934.2.2 pAuthStatus . . . 875
8.935 unpack_nas_GetCDMANetworkParameters_t Struct Reference . . . 875
8.935.1 Detailed Description . . . 876
8.935.2 Field Documentation . . . 877
8.935.2.1 Application . . . 877
8.935.2.2 Broadcast . . . 877
8.935.2.3 CustomSCP . . . 877
8.935.2.4 ForceRev0 . . . 877
8.935.2.5 ParamPresenceMask . . . 878
8.935.2.6 Protocol . . . 878
8.935.2.7 RegForeignNID . . . 878
8.935.2.8 RegForeignSID . . . 878
8.935.2.9 RegHomeSID . . . 878

8.935.2.10 Roaming . . . 878
8.935.2.11 SCI . . . 878
8.935.2.12 SCM . . . 878
8.936 unpack_nas_GetHomeNetwork3GPP2_t Struct Reference . . . 878
8.936.1 Detailed Description . . . 878
8.936.2 Field Documentation . . . 880
8.936.2.1 nameSize . . . 880
8.936.2.2 ParamPresenceMask . . . 880
8.936.2.3 pMCC . . . 880
8.936.2.4 pMNC . . . 880
8.936.2.5 pName . . . 880
8.936.2.6 pNID . . . 880
8.936.2.7 pNw2DescDisp . . . 880
8.936.2.8 pNw2DescEnc . . . 880
8.936.2.9 pNw2DescLen . . . 880
8.936.2.10 pNw2MCC . . . 880
8.936.2.11 pNw2MNC . . . 880
8.936.2.12 pNw2Name . . . 880
8.936.2.13 pSID . . . 880
8.937 unpack_nas_GetHomeNetwork_t Struct Reference . . . 880
8.937.1 Detailed Description . . . 880
8.937.2 Field Documentation . . . 881
8.937.2.1 mcc . . . 881
8.937.2.2 mnc . . . 881
8.937.2.3 name . . . 881
8.937.2.4 nid . . . 881
8.937.2.5 ParamPresenceMask . . . 881
8.937.2.6 sid . . . 881
8.938 unpack_nas_GetNetworkPreference_t Struct Reference . . . 881
8.938.1 Detailed Description . . . 882
8.938.2 Field Documentation . . . 882
8.938.2.1 ActiveTechPref . . . 882
8.938.2.2 Duration . . . 882
8.938.2.3 ParamPresenceMask . . . 882
8.938.2.4 PersistentTechPref . . . 882
8.938.2.5 Tlvresult . . . 883
8.939 unpack_nas_GetRFInfo_t Struct Reference . . . 883
8.939.1 Detailed Description . . . 883
8.939.2 Field Documentation . . . 883
8.939.2.1 instancesSize . . . 883

8.939.2.2 ParamPresenceMask . . . 883
8.939.2.3 RFBandInfoElements . . . 883
8.940 unpack_nas_GetServingNetwork_t Struct Reference . . . 883
8.940.1 Detailed Description . . . 884
8.940.2 Field Documentation . . . 885
8.940.2.1 CSDomain . . . 885
8.940.2.2 DataCaps . . . 885
8.940.2.3 DataCapsLen . . . 885
8.940.2.4 MCC . . . 886
8.940.2.5 MNC . . . 886
8.940.2.6 Name . . . 886
8.940.2.7 nameSize . . . 886
8.940.2.8 ParamPresenceMask . . . 886
8.940.2.9 PSDomain . . . 886
8.940.2.10 RadioIfaces . . . 886
8.940.2.11 RadioIfacesSize . . . 886
8.940.2.12 RAN . . . 886
8.940.2.13 RegistrationState . . . 886
8.940.2.14 Roaming . . . 886
8.941 unpack_nas_GetServingNetworkCapabilities_t Struct Reference . . . 886
8.941.1 Detailed Description . . . 886
8.941.2 Field Documentation . . . 887
8.941.2.1 DataCaps . . . 887
8.941.2.2 DataCapsLen . . . 887
8.941.2.3 ParamPresenceMask . . . 887
8.942 unpack_nas_GetSignalStrengths_t Struct Reference . . . 887
8.942.1 Detailed Description . . . 887
8.942.2 Field Documentation . . . 888
8.942.2.1 len . . . 888
8.942.2.2 ParamPresenceMask . . . 888
8.942.2.3 radio . . . 888
8.942.2.4 rssi . . . 888
8.943 unpack_nas_PerformNetworkScan_t Struct Reference . . . 888
8.943.1 Detailed Description . . . 888
8.943.2 Field Documentation . . . 889
8.943.2.1 p3GppNetworkInfoInstances . . . 889
8.943.2.2 p3GppNetworkInstanceSize . . . 889
8.943.2.3 ParamPresenceMask . . . 890
8.943.2.4 pLteOpModeTlv . . . 890
8.943.2.5 pPCIInfo . . . 890

8.943.2.6 pPCSInstance . . . 890
8.943.2.7 pPCSInstanceSize . . . 890
8.943.2.8 pRATINstance . . . 890
8.943.2.9 pRATInstanceSize . . . 890
8.943.2.10 pScanResult . . . 890
8.944 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference . . . 890
8.944.1 Detailed Description . . . 890
8.944.2 Field Documentation . . . 890
8.944.2.1 dataCaps . . . 890
8.944.2.2 dataCapsSize . . . 890
8.944.2.3 ParamPresenceMask . . . 890
8.945 unpack_nas_SetEventReportInd_t Struct Reference . . . 890
8.945.1 Detailed Description . . . 891
8.945.2 Field Documentation . . . 891
8.945.2.1 ParamPresenceMask . . . 891
8.945.2.2 RFTlv . . . 891
8.945.2.3 RRTlv . . . 891
8.945.2.4 SLQSSSTlv . . . 891
8.945.2.5 SSTlv . . . 891
8.946 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference . . . 892
8.946.1 Detailed Description . . . 892
8.946.2 Field Documentation . . . 892
8.946.2.1 ParamPresenceMask . . . 893
8.946.2.2 sNumScellsConfig . . . 893
8.946.2.3 sPhyCaAggDlBW . . . 893
8.946.2.4 sPhyCaAggPcellInfo . . . 893
8.946.2.5 sPhyCaAggScellDlBw . . . 893
8.946.2.6 sPhyCaAggScellIndex . . . 893
8.946.2.7 sPhyCaAggScellIndType . . . 893
8.946.2.8 sPhyCaAggScellInfo . . . 893
8.947 unpack_nas_SetNetworkPreference_t Struct Reference . . . 893
8.947.1 Detailed Description . . . 893
8.947.2 Field Documentation . . . 893
8.947.2.1 ParamPresenceMask . . . 893
8.947.2.2 Tlvresult . . . 893
8.948 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference . . . 893
8.948.1 Detailed Description . . . 893
8.948.2 Field Documentation . . . 894
8.948.2.1 ParamPresenceMask . . . 894
8.948.2.2 roaming . . . 894

8.949 unpack_nas_SetServingSystemCallback_ind_t Struct Reference . . . 894
8.949.1 Detailed Description . . . 894
8.949.2 Field Documentation . . . 894
8.949.2.1 ParamPresenceMask . . . 894
8.949.2.2 SSInfo . . . 894
8.949.2.3 Tlvresult . . . 895
8.950 unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t Struct Reference . . . 895
8.950.1 Detailed Description . . . 895
8.950.2 Field Documentation . . . 895
8.950.2.1 ddtm_Settings . . . 895
8.950.2.2 ParamPresenceMask . . . 895
8.951 unpack_nas_SLQSGetErrorRate_t Struct Reference . . . 895
8.951.1 Detailed Description . . . 895
8.951.2 Field Documentation . . . 896
8.951.2.1 ParamPresenceMask . . . 896
8.951.2.2 pCDMAFrameErrRate . . . 896
8.951.2.3 pGSMBER . . . 896
8.951.2.4 pHDRPackErrRate . . . 896
8.951.2.5 pWCDMABER . . . 896
8.952 unpack_nas_SLQSGetHomeNetwork_t Struct Reference . . . 896
8.952.1 Detailed Description . . . 896
8.952.2 Field Documentation . . . 897
8.952.2.1 mcc . . . 897
8.952.2.2 mnc . . . 897
8.952.2.3 name . . . 897
8.952.2.4 nid . . . 897
8.952.2.5 ParamPresenceMask . . . 898
8.952.2.6 pHomeNwMNC3Gpp . . . 898
8.952.2.7 pNwNameSrc3Gpp . . . 898
8.952.2.8 sid . . . 898
8.953 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference . . . 898
8.953.1 Detailed Description . . . 898
8.953.2 Field Documentation . . . 898
8.953.2.1 LTECphyCaInfo . . . 898
8.953.2.2 ParamPresenceMask . . . 898
8.953.2.3 Tlvresult . . . 898
8.954 unpack_nas_SLQSGetNetworkTime_t Struct Reference . . . 898
8.954.1 Detailed Description . . . 899
8.954.2 Field Documentation . . . 899
8.954.2.1 p3GPP2TimeInfo . . . 899

8.954.2.2 p3GPPTimeInfo . . . 899
8.954.2.3 ParamPresenceMask . . . 899
8.955 unpack_nas_SLQSGetOperatorNameData_t Struct Reference . . . 899
8.955.1 Detailed Description . . . 899
8.955.2 Field Documentation . . . 900
8.955.2.1 ParamPresenceMask . . . 900
8.955.2.2 pNITZInformation . . . 900
8.955.2.3 pOperatorNameString . . . 900
8.955.2.4 pOperatorPLMNList . . . 900
8.955.2.5 pPLMNNetworkName . . . 900
8.955.2.6 pSrvcProviderName . . . 900
8.956 unpack_nas_SLQSGetPLMNName_t Struct Reference . . . 900
8.956.1 Detailed Description . . . 901
8.956.2 Field Documentation . . . 903
8.956.2.1 longName . . . 903
8.956.2.2 longNameCI . . . 903
8.956.2.3 longNameEn . . . 903
8.956.2.4 longNameLen . . . 903
8.956.2.5 longNameSB . . . 903
8.956.2.6 ParamPresenceMask . . . 903
8.956.2.7 shortName . . . 903
8.956.2.8 shortNameCI . . . 903
8.956.2.9 shortNameEn . . . 903
8.956.2.10 shortNameLen . . . 903
8.956.2.11 shortNameSB . . . 903
8.956.2.12 spn . . . 903
8.956.2.13 spnEncoding . . . 903
8.956.2.14 spnLength . . . 903
8.957 unpack_nas_SLQSGetServingSystem_t Struct Reference . . . 903
8.957.1 Detailed Description . . . 904
8.957.2 Field Documentation . . . 907
8.957.2.1 BasestationID . . . 907
8.957.2.2 BasestationLatitude . . . 907
8.957.2.3 BasestationLongitude . . . 907
8.957.2.4 CallBarStatus . . . 907
8.957.2.5 CDMA_P_Rev . . . 907
8.957.2.6 CDMASystemInfoExt . . . 907
8.957.2.7 CellID . . . 907
8.957.2.8 ConcSvcInfo . . . 907
8.957.2.9 CurrentPLMN . . . 907

8.957.2.10 DataSrvCapabilities . . . 907
8.957.2.11 DefaultRoamInd . . . 907
8.957.2.12 DetailedSvcInfo . . . 907
8.957.2.13 DTMInd . . . 907
8.957.2.14 Gpp2TimeZone . . . 907
8.957.2.15 GppNetworkDSTAdjustment . . . 907
8.957.2.16 GppTimeZone . . . 907
8.957.2.17 HdrPersonality . . . 907
8.957.2.18 Lac . . . 907
8.957.2.19 NetworkID . . . 907
8.957.2.20 ParamPresenceMask . . . 907
8.957.2.21 PRLInd . . . 907
8.957.2.22 RoamIndicatorVal . . . 907
8.957.2.23 RoamingIndicatorList . . . 907
8.957.2.24 ServingSystem . . . 907
8.957.2.25 SystemID . . . 907
8.957.2.26 TrackAreaCode . . . 907
8.958 unpack_nas_SLQSGetServingSystemV2_t Struct Reference . . . 907
8.958.1 Detailed Description . . . 908
8.958.2 Field Documentation . . . 911
8.958.2.1 BasestationID . . . 911
8.958.2.2 BasestationLatitude . . . 911
8.958.2.3 BasestationLongitude . . . 911
8.958.2.4 CallBarStatus . . . 911
8.958.2.5 CDMA_P_Rev . . . 911
8.958.2.6 CDMASystemInfoExt . . . 911
8.958.2.7 CellID . . . 911
8.958.2.8 ConcSvcInfo . . . 911
8.958.2.9 CurrentPLMN . . . 911
8.958.2.10 DataSrvCapabilities . . . 911
8.958.2.11 DefaultRoamInd . . . 912
8.958.2.12 DetailedSvcInfo . . . 912
8.958.2.13 DTMInd . . . 912
8.958.2.14 Gpp2TimeZone . . . 912
8.958.2.15 GppNetworkDSTAdjustment . . . 912
8.958.2.16 GppTimeZone . . . 912
8.958.2.17 HdrPersonality . . . 912
8.958.2.18 Lac . . . 912
8.958.2.19 MNCPCSDigitStatTlv . . . 912
8.958.2.20 NetworkID . . . 912

8.958.2.21 ParamPresenceMask . . . 912
8.958.2.22 PRLInd . . . 912
8.958.2.23 RoamIndicatorVal . . . 912
8.958.2.24 RoamingIndicatorList . . . 912
8.958.2.25 ServingSystem . . . 912
8.958.2.26 SystemID . . . 912
8.958.2.27 TrackAreaCode . . . 912
8.959 unpack_nas_SLQSGetSignalStrength_t Struct Reference . . . 912
8.959.1 Detailed Description . . . 913
8.959.2 Field Documentation . . . 914
8.959.2.1 ecioList . . . 914
8.959.2.2 ecioListLen . . . 914
8.959.2.3 errorRateList . . . 914
8.959.2.4 errorRateListLen . . . 914
8.959.2.5 Io . . . 914
8.959.2.6 Itersrp . . . 914
8.959.2.7 Itesnr . . . 915
8.959.2.8 ParamPresenceMask . . . 915
8.959.2.9 rsrqInfo . . . 915
8.959.2.10 rxSignalStrengthList . . . 915
8.959.2.11 rxSignalStrengthListLen . . . 915
8.959.2.12 signalStrengthReqMask . . . 915
8.959.2.13 sinr . . . 915
8.960 unpack_nas_SLQSGetSysInfo_t Struct Reference . . . 915
8.960.1 Detailed Description . . . 915
8.960.2 Field Documentation . . . 917
8.960.2.1 pAddCDMASysInfo . . . 918
8.960.2.2 pAddGSMSysInfo . . . 918
8.960.2.3 pAddHDRSysInfo . . . 918
8.960.2.4 pAddLTESysInfo . . . 918
8.960.2.5 pAddWCDMASysInfo . . . 918
8.960.2.6 ParamPresenceMask . . . 918
8.960.2.7 pCampedCiotLteOpMode . . . 918
8.960.2.8 pCDMASrvStatusInfo . . . 918
8.960.2.9 pCDMASysInfo . . . 918
8.960.2.10 pGSMCallBarringSysInfo . . . 918
8.960.2.11 pGSMCipherDomainSysInfo . . . 918
8.960.2.12 pGSMSrvStatusInfo . . . 918
8.960.2.13 pGSMSysInfo . . . 918
8.960.2.14 pHDRSrvStatusInfo . . . 918

8.960.2.15 pHDRSysInfo . . . 918
8.960.2.16 pLTESrvStatusInfo . . . 918
8.960.2.17 pLTESysInfo . . . 918
8.960.2.18 pLTEVoiceSupportSysInfo . . . 918
8.960.2.19 pWCDMACallBarringSysInfo . . . 918
8.960.2.20 pWCDMACipherDomainSysInfo . . . 918
8.960.2.21 pWCDMASrvStatusInfo . . . 918
8.960.2.22 pWCDMASysInfo . . . 918
8.961 unpack_nas_SLQSGetSysInfoV2_t Struct Reference . . . 918
8.961.1 Detailed Description . . . 919
8.961.2 Field Documentation . . . 922
8.961.2.1 pAddCDMASysInfo . . . 922
8.961.2.2 pAddGSMSysInfo . . . 922
8.961.2.3 pAddHDRSysInfo . . . 922
8.961.2.4 pAddLTESysInfo . . . 922
8.961.2.5 pAddWCDMASysInfo . . . 922
8.961.2.6 ParamPresenceMask . . . 922
8.961.2.7 pCampedCiotLteOpMode . . . 922
8.961.2.8 pCDMASrvStatusInfo . . . 922
8.961.2.9 pCDMASysInfo . . . 922
8.961.2.10 pGSMCallBarringSysInfo . . . 922
8.961.2.11 pGSMCipherDomainSysInfo . . . 922
8.961.2.12 pGSMSrvStatusInfo . . . 922
8.961.2.13 pGSMSysInfo . . . 922
8.961.2.14 pHDRSrvStatusInfo . . . 922
8.961.2.15 pHDRSysInfo . . . 923
8.961.2.16 pImsVoiceSupportLte . . . 923
8.961.2.17 pLteEmbmsCoverage . . . 923
8.961.2.18 pLteEmbmsTraceId . . . 923
8.961.2.19 pLteRegDomain . . . 923
8.961.2.20 pLTESrvStatusInfo . . . 923
8.961.2.21 pLTESysInfo . . . 923
8.961.2.22 pLteVoiceDomain . . . 923
8.961.2.23 pLTEVoiceSupportSysInfo . . . 923
8.961.2.24 pNR5GCellStatusInfo . . . 923
8.961.2.25 pNR5GSrvStatusinfo . . . 923
8.961.2.26 pNR5GSysInfo . . . 923
8.961.2.27 pSimRejInfo . . . 923
8.961.2.28 pSrvRegRestriction . . . 923
8.961.2.29 pWCDMACallBarringSysInfo . . . 923

8.961.2.30 pWCDMACipherDomainSysInfo . . . 923
8.961.2.31 pWCDMASrvStatusInfo . . . 923
8.961.2.32 pWCDMASysInfo . . . 923
8.962 unpack_nas_SLQSGetSysSelectionPref_t Struct Reference . . . 923
8.962.1 Detailed Description . . . 923
8.962.2 Field Documentation . . . 928
8.962.2.1 ParamPresenceMask . . . 928
8.962.2.2 pBandPref . . . 928
8.962.2.3 pEmerMode . . . 928
8.962.2.4 pGWAcqOrderPref . . . 928
8.962.2.5 pLTEBandPref . . . 928
8.962.2.6 pModePref . . . 928
8.962.2.7 pNetSelPref . . . 928
8.962.2.8 pPRLPref . . . 928
8.962.2.9 pRoamPref . . . 928
8.962.2.10 pSrvDomainPref . . . 928
8.963 unpack_nas_SLQSGetSysSelectionPrefExt_t Struct Reference . . . 928
8.963.1 Detailed Description . . . 929
8.963.2 Field Documentation . . . 930
8.963.2.1 pAcqOrderPref . . . 930
8.963.2.2 pBandPref . . . 930
8.963.2.3 pCiotAcqOrderPref . . . 930
8.963.2.4 pCiotLteOpModePref . . . 930
8.963.2.5 pEmerMode . . . 930
8.963.2.6 pGWAcqOrderPref . . . 930
8.963.2.7 pLTEBandPref . . . 931
8.963.2.8 pLTEBandPrefExt . . . 931
8.963.2.9 pLteM1BandPref . . . 931
8.963.2.10 pLteNb1BandPref . . . 931
8.963.2.11 pModePref . . . 931
8.963.2.12 pNetSelPref . . . 931
8.963.2.13 pNr5gBandPref . . . 931
8.963.2.14 pPRLPref . . . 931
8.963.2.15 pRatDisabledMask . . . 931
8.963.2.16 pRoamPref . . . 931
8.963.2.17 pSrvDomainPref . . . 931
8.964 unpack_nas_SLQSGetSysSelectionPrefExtV2_t Struct Reference . . . 931
8.964.1 Detailed Description . . . 931
8.964.2 Field Documentation . . . 934
8.964.2.1 pAcqOrderPref . . . 934

8.964.2.2 ParamPresenceMask . . . 934
8.964.2.3 pBandPref . . . 934
8.964.2.4 pCiotAcqOrderPref . . . 934
8.964.2.5 pCiotLteOpModePref . . . 934
8.964.2.6 pEmerMode . . . 934
8.964.2.7 pGWAcqOrderPref . . . 934
8.964.2.8 pLTEBandPref . . . 934
8.964.2.9 pLTEBandPrefExt . . . 934
8.964.2.10 pLteM1BandPref . . . 934
8.964.2.11 pLteNb1BandPref . . . 934
8.964.2.12 pModePref . . . 934
8.964.2.13 pNetSelPref . . . 934
8.964.2.14 pNr5gBandPref . . . 934
8.964.2.15 pPRLPref . . . 934
8.964.2.16 pRatDisabledMask . . . 934
8.964.2.17 pRoamPref . . . 934
8.964.2.18 pSrvDomainPref . . . 934
8.964.2.19 pSrvRegRestric . . . 934
8.964.2.20 pTDSCDMABandPref . . . 934
8.964.2.21 pUsageSetting . . . 934
8.964.2.22 pVoiceDomainPref . . . 934
8.965 unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t Struct Reference . . . 935
8.965.1 Detailed Description . . . 935
8.965.2 Field Documentation . . . 935
8.965.2.1 edrxCiotLteModeTlv . . . 935
8.965.2.2 edrxCycleLengthTlv . . . 935
8.965.2.3 edrxEnableTypeTlv . . . 935
8.965.2.4 edrxPagingTimeWindowTlv . . . 935
8.965.2.5 edrxRatTypeTlv . . . 935
8.965.2.6 ParamPresenceMask . . . 936
8.966 unpack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 936
8.966.1 Detailed Description . . . 936
8.966.2 Field Documentation . . . 937
8.966.2.1 ParamPresenceMask . . . 937
8.966.2.2 pCDMAChannel . . . 937
8.966.2.3 pDirNum . . . 937
8.966.2.4 pHomeSIDNID . . . 937
8.966.2.5 pMdn . . . 937
8.966.2.6 pMinBasedIMSI . . . 937
8.966.2.7 pNAMNameInfo . . . 937

8.966.2.8 pTrueIMSI . . . 937
8.967 unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference . . . 937
8.967.1 Detailed Description . . . 937
8.967.2 Field Documentation . . . 938
8.967.2.1 ParamPresenceMask . . . 938
8.967.2.2 pCDMAInfo . . . 938
8.967.2.3 pGERANInfo . . . 938
8.967.2.4 pLTEInfoInterfreq . . . 938
8.967.2.5 pLTEInfoIntrafreq . . . 938
8.967.2.6 pLTEInfoNeighboringGSM . . . 938
8.967.2.7 pLTEInfoNeighboringWCDMA . . . 938
8.967.2.8 pUMTSCellID . . . 938
8.967.2.9 pUMTSInfo . . . 938
8.967.2.10 pWCDMAInfoLTENeighborCell . . . 938
8.968 unpack_nas_SLQSNasGetCellLocationInfoV2_t Struct Reference . . . 938
8.968.1 Detailed Description . . . 939
8.968.2 Field Documentation . . . 940
8.968.2.1 ParamPresenceMask . . . 940
8.968.2.2 pCDMAInfo . . . 940
8.968.2.3 pGERANInfo . . . 940
8.968.2.4 pLteEarfcnInfo . . . 940
8.968.2.5 pLTEInfoInterfreq . . . 940
8.968.2.6 pLTEInfoIntrafreq . . . 940
8.968.2.7 pLTEInfoNeighboringGSM . . . 940
8.968.2.8 pLTEInfoNeighboringWCDMA . . . 940
8.968.2.9 pUMTSCellID . . . 940
8.968.2.10 pUMTSExtInfo . . . 940
8.968.2.11 pUMTSInfo . . . 940
8.968.2.12 pWCDMACellInfoExt . . . 940
8.968.2.13 pWCDMAInfoLTENeighborCell . . . 940
8.969 unpack_nas_SLQSNASGeteDRXParams_t Struct Reference . . . 940
8.969.1 Detailed Description . . . 941
8.969.2 Field Documentation . . . 941
8.969.2.1 ParamPresenceMask . . . 941
8.969.2.2 pCycleLen . . . 941
8.969.2.3 pEdrxEnable . . . 941
8.969.2.4 pPagingTimeWindow . . . 941
8.970 unpack_nas_SLQSNASGeteDRXParamsExt_t Struct Reference . . . 941
8.970.1 Detailed Description . . . 942
8.970.2 Field Documentation . . . 943

8.970.2.1 ParamPresenceMask . . . 943
8.970.2.2 pCycleLen . . . 943
8.970.2.3 pEdrxEnable . . . 943
8.970.2.4 pEdrxRAT . . . 943
8.970.2.5 pLteOpMode . . . 943
8.970.2.6 pPagingTimeWindow . . . 943
8.971 unpack_nas_SLQSNASGetForbiddenNetworks_t Struct Reference . . . 943
8.971.1 Detailed Description . . . 943
8.971.2 Field Documentation . . . 943
8.971.2.1 ParamPresenceMask . . . 943
8.971.2.2 pForbiddenNetworks3GPP . . . 943
8.972 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference . . . 943
8.972.1 Detailed Description . . . 944
8.972.2 Field Documentation . . . 944
8.972.2.1 ParamPresenceMask . . . 944
8.972.2.2 pColorCode . . . 944
8.973 unpack_nas_SLQSNasGetRFInfo_t Struct Reference . . . 944
8.973.1 Detailed Description . . . 944
8.973.2 Field Documentation . . . 945
8.973.2.1 pLTEOperationMode . . . 945
8.973.2.2 pRfBandInfoExtFormat . . . 945
8.973.2.3 pRfBandwidthInfo . . . 945
8.973.2.4 pRfDedicatedBandInfo . . . 945
8.973.2.5 rfbandInfoList . . . 945
8.973.2.6 Tlvresult . . . 945
8.974 unpack_nas_SLQSNasGetSigInfo_t Struct Reference . . . 945
8.974.1 Detailed Description . . . 945
8.974.2 Field Documentation . . . 946
8.974.2.1 CDMASSInfo . . . 946
8.974.2.2 GSMSSInfo . . . 946
8.974.2.3 HDRSSInfo . . . 946
8.974.2.4 LTESSInfo . . . 946
8.974.2.5 ParamPresenceMask . . . 946
8.974.2.6 WCDMASSInfo . . . 946
8.975 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 946
8.975.1 Detailed Description . . . 946
8.975.2 Field Documentation . . . 946
8.975.2.1 ParamPresenceMask . . . 946
8.975.2.2 pRXChain0Info . . . 946
8.975.2.3 pRXChain1Info . . . 946

8.975.2.4 pTXInfo . . . 947
8.976 unpack_nas_SLQSNasNetworkRejectCallback_Ind_t Struct Reference . . . 947
8.976.1 Detailed Description . . . 947
8.976.2 Field Documentation . . . 949
8.976.2.1 ParamPresenceMask . . . 949
8.976.2.2 pCsgId . . . 949
8.976.2.3 pLteOpMode . . . 949
8.976.2.4 pPlmnId . . . 949
8.976.2.5 radioIf . . . 949
8.976.2.6 rejectCause . . . 949
8.976.2.7 serviceDomain . . . 949
8.977 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference . . . 949
8.977.1 Detailed Description . . . 949
8.977.2 Field Documentation . . . 950
8.977.2.1 ParamPresenceMask . . . 950
8.977.2.2 pDayltSavAdj . . . 950
8.977.2.3 pRadioInterface . . . 950
8.977.2.4 pTimeZone . . . 950
8.977.2.5 universalTime . . . 950
8.978 unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t Struct Reference . . . 950
8.978.1 Detailed Description . . . 951
8.978.2 Field Documentation . . . 951
8.978.2.1 CSGInfo . . . 951
8.978.2.2 network3GPPScanInfo . . . 951
8.978.2.3 networkNameSrc . . . 951
8.978.2.4 networkScanStatus . . . 951
8.978.2.5 ParamPresenceMask . . . 951
8.978.2.6 signalQuality . . . 951
8.978.2.7 signalStrength . . . 951
8.979 unpack_nas_SLQSNasRFBandInfoCallback_Ind_t Struct Reference . . . 951
8.979.1 Detailed Description . . . 952
8.979.2 Field Documentation . . . 952
8.979.2.1 lteOperationalMode . . . 952
8.979.2.2 ParamPresenceMask . . . 952
8.979.2.3 rfBandInfo . . . 952
8.979.2.4 rfBandInfoExt . . . 952
8.979.2.5 rfBandwidthInfo . . . 953
8.979.2.6 rfDedicatedBandInfo . . . 953
8.980 unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference . . . 953
8.980.1 Detailed Description . . . 953

8.980.2 Field Documentation . . . 954
8.980.2.1 ParamPresenceMask . . . 954
8.980.2.2 pCDMASigInfo . . . 954
8.980.2.3 pGSMSigInfo . . . 954
8.980.2.4 pHDRSigInfo . . . 954
8.980.2.5 pLTESigInfo . . . 954
8.980.2.6 pRscp . . . 954
8.980.2.7 pTDSCDMASigInfoExt . . . 954
8.980.2.8 pWCDMASigInfo . . . 954
8.981 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference . . . 954
8.981.1 Detailed Description . . . 954
8.981.2 Field Documentation . . . 955
8.981.2.1 ParamPresenceMask . . . 955
8.981.2.2 pLteEARFCN . . . 955
8.981.2.3 pLtePCI . . . 955
8.981.2.4 pWcdmaUARFCN . . . 955
8.982 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference . . . 955
8.982.1 Detailed Description . . . 955
8.982.2 Field Documentation . . . 955
8.982.2.1 commonInfo . . . 955
8.982.2.2 ParamPresenceMask . . . 955
8.982.2.3 pLTEInfo . . . 955
8.983 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference . . . 955
8.983.1 Detailed Description . . . 956
8.983.2 Field Documentation . . . 956
8.983.2.1 Info . . . 956
8.983.2.2 ParamPresenceMask . . . 956
8.983.2.3 Tlvresult . . . 956
8.984 unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference . . . 956
8.984.1 Detailed Description . . . 956
8.984.2 Field Documentation . . . 957
8.984.2.1 ParamPresenceMask . . . 957
8.984.2.2 t3396_apn . . . 957
8.984.2.3 t3396_plmn_id . . . 957
8.984.2.4 t3396_val . . . 957
8.985 unpack_nas_SLQSPerformNetworkScanV2_t Struct Reference . . . 957
8.985.1 Detailed Description . . . 957
8.985.2 Field Documentation . . . 958
8.985.2.1 p3GppNetworkInfoInstances . . . 958
8.985.2.2 p3GppNetworkInstanceSize . . . 958

8.985.2.3 ParamPresenceMask . . . 958
8.985.2.4 pLteOpModeTlv . . . 958
8.985.2.5 pNetworkNameSrcTlv . . . 959
8.985.2.6 pPCIInfo . . . 959
8.985.2.7 pPCSInstance . . . 959
8.985.2.8 pPCSInstanceSize . . . 959
8.985.2.9 pRATINstance . . . 959
8.985.2.10 pRATInstanceSize . . . 959
8.985.2.11 pScanResult . . . 959
8.986 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference . . . 959
8.986.1 Detailed Description . . . 959
8.986.2 Field Documentation . . . 959
8.986.2.1 Info . . . 959
8.986.2.2 ParamPresenceMask . . . 959
8.986.2.3 Tlvresult . . . 959
8.987 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference . . . 959
8.987.1 Detailed Description . . . 960
8.987.2 Field Documentation . . . 960
8.987.2.1 ParamPresenceMask . . . 960
8.987.2.2 pCurrentPersonality . . . 960
8.987.2.3 pPersonalityListLength . . . 960
8.987.2.4 pProtocolSubtypeElement . . . 960
8.988 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference . . . 960
8.988.1 Detailed Description . . . 960
8.988.2 Field Documentation . . . 961
8.988.2.1 pAppSubType . . . 961
8.988.2.2 ParamPresenceMask . . . 961
8.988.2.3 pCurrentPrsnlty . . . 961
8.988.2.4 pPersonalityListLength . . . 961
8.988.2.5 pProtoSubTypElmnt . . . 961
8.989 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference . . . 961
8.989.1 Detailed Description . . . 961
8.989.2 Field Documentation . . . 962
8.989.2.1 ParamPresenceMask . . . 962
8.989.2.2 pDRCParams . . . 962
8.989.2.3 pPilotSetData . . . 962
8.989.2.4 pUATI . . . 962
8.990 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference . . . 962
8.990.1 Detailed Description . . . 962
8.990.2 Field Documentation . . . 963

8.990.2.1 CQIValueCW0 . . . 963
8.990.2.2 CQIValueCW1 . . . 963
8.990.2.3 ParamPresenceMask . . . 963
8.990.2.4 ValidityCW0 . . . 963
8.990.2.5 ValidityCW1 . . . 963
8.991 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference . . . 963
8.991.1 Detailed Description . . . 963
8.991.2 Field Documentation . . . 963
8.991.2.1 ParamPresenceMask . . . 963
8.991.2.2 pSccRxInfo . . . 964
8.992 unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t Struct Reference . . . 964
8.992.1 Detailed Description . . . 964
8.992.2 Field Documentation . . . 964
8.992.2.1 ParamPresenceMask . . . 964
8.992.2.2 pHDRPersInd . . . 964
8.993 unpack_nas_SLQSSwiNetworkDebug_t Struct Reference . . . 964
8.993.1 Detailed Description . . . 964
8.993.2 Field Documentation . . . 965
8.993.2.1 ParamPresenceMask . . . 965
8.993.2.2 pDataStatusDetail . . . 965
8.993.2.3 pDeviceConfigDetail . . . 965
8.993.2.4 pNetworkStat1x . . . 965
8.993.2.5 pNetworkStatEVDO . . . 965
8.993.2.6 pObjectVer . . . 965
8.994 unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t Struct Reference . . . 965
8.994.1 Detailed Description . . . 965
8.994.2 Field Documentation . . . 966
8.994.2.1 ParamPresenceMask . . . 966
8.994.2.2 rankIndicatorTlv . . . 966
8.995 unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference . . . 966
8.995.1 Detailed Description . . . 966
8.995.2 Field Documentation . . . 969
8.995.2.1 pAddCDMASysInfo . . . 969
8.995.2.2 pAddGSMSysInfo . . . 969
8.995.2.3 pAddHDRSysInfo . . . 969
8.995.2.4 pAddLTESysInfo . . . 969
8.995.2.5 pAddWCDMASysInfo . . . 969
8.995.2.6 ParamPresenceMask . . . 969
8.995.2.7 pCDMASrvStatusInfo . . . 969
8.995.2.8 pCDMASysInfo . . . 969

8.995.2.9 pGSMCallBarringSysInfo . . . 969
8.995.2.10 pGSMCipherDomainSysInfo . . . 969
8.995.2.11 pGSMSrvStatusInfo . . . 969
8.995.2.12 pGSMSysInfo . . . 969
8.995.2.13 pHDRSrvStatusInfo . . . 969
8.995.2.14 pHDRSysInfo . . . 969
8.995.2.15 pLteCiotOpModeTlv . . . 969
8.995.2.16 pLTESrvStatusInfo . . . 969
8.995.2.17 pLTESysInfo . . . 969
8.995.2.18 pLTEVoiceSupportSysInfo . . . 969
8.995.2.19 pNR5GCellStatus . . . 970
8.995.2.20 pNR5GSerStatTlv . . . 970
8.995.2.21 pNR5GSystemInfoTlv . . . 970
8.995.2.22 pSysInfoNoChange . . . 970
8.995.2.23 pWCDMACallBarringSysInfo . . . 970
8.995.2.24 pWCDMACipherDomainSysInfo . . . 970
8.995.2.25 pWCDMASrvStatusInfo . . . 970
8.995.2.26 pWCDMASysInfo . . . 970
8.996 unpack_omaDmConfigTlv_t Struct Reference . . . 970
8.996.1 Detailed Description . . . 970
8.996.2 Field Documentation . . . 970
8.996.2.1 alertmsg . . . 971
8.996.2.2 alertmsglength . . . 971
8.996.2.3 state . . . 971
8.996.2.4 userInputReq . . . 971
8.996.2.5 userInputTimeout . . . 971
8.997 unpack_omaDmFotaTlv_t Struct Reference . . . 971
8.997.1 Detailed Description . . . 971
8.997.2 Field Documentation . . . 972
8.997.2.1 description . . . 972
8.997.2.2 descriptionlength . . . 972
8.997.2.3 fwdloadsize . . . 972
8.997.2.4 fwloadComplete . . . 972
8.997.2.5 namelength . . . 972
8.997.2.6 package_name . . . 972
8.997.2.7 sessionType . . . 972
8.997.2.8 severity . . . 973
8.997.2.9 state . . . 973
8.997.2.10 updateCompleteStatus . . . 973
8.997.2.11 userInputReq . . . 973

8.997.2.12 userInputTimeout . . . 973
8.997.2.13 version . . . 973
8.997.2.14 versionlength . . . 973
8.998 unpack_omaDmNotificationsTlv_t Struct Reference . . . 973
8.998.1 Detailed Description . . . 973
8.998.2 Field Documentation . . . 974
8.998.2.1 notification . . . 974
8.998.2.2 sessionStatus . . . 974
8.999 unpack_pds_GetPDSDefaults_t Struct Reference . . . 974
8.999.1 Detailed Description . . . 974
8.999.2 Field Documentation . . . 974
8.999.2.1 pAccuracy . . . 974
8.999.2.2 ParamPresenceMask . . . 974
8.999.2.3 pInterval . . . 974
8.999.2.4 pOperation . . . 975
8.999.2.5 pTimeout . . . 975
8.1000 unpack_pds_GetPDSState_t Struct Reference . . . 975
8.1000.1 Detailed Description . . . 975
8.1000.2 Field Documentation . . . 975
8.1000.2.1 ParamPresenceMask . . . 975
8.1000.2.2 pEnabledStatus . . . 975
8.1000.2.3 pTrackingStatus . . . 975
8.1001 unpack_pds_GetPortAutomaticTracking_t Struct Reference . . . 975
8.1001.1 Detailed Description . . . 976
8.1001.2 Field Documentation . . . 976
8.1001.2.1 ParamPresenceMask . . . 976
8.1001.2.2 pbAuto . . . 976
8.1002 unpack_pds_GetServiceAutomaticTracking_t Struct Reference . . . 976
8.1002.1 Detailed Description . . . 976
8.1002.2 Field Documentation . . . 976
8.1002.2.1 ParamPresenceMask . . . 976
8.1002.2.2 pbAuto . . . 976
8.1003 unpack_pds_GetXTRAAutomaticDownload_t Struct Reference . . . 977
8.1003.1 Detailed Description . . . 977
8.1003.2 Field Documentation . . . 977
8.1003.2.1 ParamPresenceMask . . . 977
8.1003.2.2 pbEnabled . . . 977
8.1003.2.3 pInterval . . . 977
8.1004 unpack_pds_GetXTRANetwork_t Struct Reference . . . 977
8.1004.1 Detailed Description . . . 977

8.1004.2 Field Documentation . . . 978
8.1004.2.1 ParamPresenceMask . . . 978
8.1004.2.2 pPreference . . . 978
8.1005 npack_pds_GetXTRAValidity_t Struct Reference . . . 978
8.1005.1 Detailed Description . . . 978
8.1005.2 Field Documentation . . . 978
8.1005.2.1 ParamPresenceMask . . . 979
8.1005.2.2 pDuration . . . 979
8.1005.2.3 pGPSWeek . . . 979
8.1005.2.4 pGPSWeekOffset . . . 979
8.1006 npack_pds_SetEventReport_Ind_t Struct Reference . . . 979
8.1006.1 Detailed Description . . . 979
8.1006.2 Field Documentation . . . 980
8.1006.2.1 dLatitude . . . 980
8.1006.2.2 dLongitude . . . 980
8.1006.2.3 has_dLatitude . . . 980
8.1006.2.4 has_dLongitude . . . 980
8.1006.2.5 has_PositionDataNMEA . . . 980
8.1006.2.6 has_posSrc . . . 980
8.1006.2.7 has_SessionStatus . . . 980
8.1006.2.8 ParamPresenceMask . . . 980
8.1006.2.9 PositionDataNMEA . . . 981
8.1006.2.10 posSrc . . . 981
8.1006.2.11 SessionStatus . . . 981
8.1007 npack_pds_SetPdsState_Ind_t Struct Reference . . . 981
8.1007.1 Detailed Description . . . 981
8.1007.2 Field Documentation . . . 981
8.1007.2.1 EnabledStatus . . . 981
8.1007.2.2 ParamPresenceMask . . . 981
8.1007.2.3 TlvPresent . . . 981
8.1007.2.4 TrackingStatus . . . 981
8.1008 npack_pds_SLQSGetAGPSConfig_t Struct Reference . . . 981
8.1008.1 Detailed Description . . . 982
8.1008.2 Field Documentation . . . 982
8.1008.2.1 ParamPresenceMask . . . 982
8.1008.2.2 pServerAddress . . . 982
8.1008.2.3 pServerPort . . . 982
8.1008.2.4 pServerURL . . . 982
8.1008.2.5 pServerURLLength . . . 982
8.1009 npack_pds_SLQSGetGPSStateInfo_t Struct Reference . . . 982

8.1009.1 Detailed Description . . . 983
8.1009.2 Field Documentation . . . 986
8.1009.2.1 Altitude . . . 986
8.1009.2.2 EngineState . . . 987
8.1009.2.3 glo_almanac_sv_msk . . . 987
8.1009.2.4 glo_ephemeris_sv_msk . . . 987
8.1009.2.5 glo_health_sv_msk . . . 987
8.1009.2.6 glo_visible_sv_msk . . . 987
8.1009.2.7 gps_almanac_sv_msk . . . 987
8.1009.2.8 gps_ephemeris_sv_msk . . . 987
8.1009.2.9 gps_health_sv_msk . . . 987
8.1009.2.10 gps_visible_sv_msk . . . 987
8.1009.2.11 HorizontalUncertainty . . . 987
8.1009.2.12 lno_valid . . . 987
8.1009.2.13 Latitude . . . 987
8.1009.2.14 Longitude . . . 987
8.1009.2.15 ParamPresenceMask . . . 987
8.1009.2.16 sbas_almanac_sv_msk . . . 987
8.1009.2.17 sbas_ephemeris_sv_msk . . . 987
8.1009.2.18 sbas_health_sv_msk . . . 987
8.1009.2.19 sbas_visible_sv_msk . . . 987
8.1009.2.20 Time_uncert_ms . . . 987
8.1009.2.21 TimeStmp_gps_week . . . 987
8.1009.2.22 TimeStmp_tow_ms . . . 987
8.1009.2.23 ValidMask . . . 987
8.1009.2.24 VerticalUncertainty . . . 987
8.1009.2.25 xtra_start_gps_minutes . . . 987
8.1009.2.26 xtra_start_gps_week . . . 987
8.1009.2.27 xtra_valid_duration_hours . . . 987
8.1010 unpack_qmi_t Struct Reference . . . 987
8.1010.1 Detailed Description . . . 988
8.1010.2 Field Documentation . . . 988
8.1010.2.1 msgid . . . 988
8.1010.2.2 type . . . 988
8.1010.2.3 xid . . . 988
8.1011 unpack_qos_BindDataPort_t Struct Reference . . . 988
8.1011.1 Detailed Description . . . 988
8.1011.2 Field Documentation . . . 988
8.1011.2.1 ParamPresenceMask . . . 988
8.1012 unpack_qos_dataRate_t Struct Reference . . . 988

8.1012.1 Detailed Description . . . 989
8.1012.2 Field Documentation . . . 989
8.1012.2.1 dataRateMax . . . 989
8.1012.2.2 guaranteedRate . . . 989
8.1013 unpack_qos_IPv4Addr_t Struct Reference . . . 989
8.1013.1 Detailed Description . . . 989
8.1013.2 Field Documentation . . . 989
8.1013.2.1 addr . . . 989
8.1013.2.2 subnetMask . . . 989
8.1014 unpack_qos_IPv6Addr_t Struct Reference . . . 989
8.1014.1 Detailed Description . . . 989
8.1014.2 Field Documentation . . . 990
8.1014.2.1 addr . . . 990
8.1014.2.2 prefixLen . . . 990
8.1015 unpack_qos_IPv6TrafCls_t Struct Reference . . . 990
8.1015.1 Detailed Description . . . 990
8.1015.2 Field Documentation . . . 990
8.1015.2.1 mask . . . 990
8.1015.2.2 val . . . 990
8.1016 unpack_qos_pktErrRate_t Struct Reference . . . 990
8.1016.1 Detailed Description . . . 990
8.1016.2 Field Documentation . . . 991
8.1016.2.1 exponent . . . 991
8.1016.2.2 multiplier . . . 991
8.1017 unpack_qos_Port_t Struct Reference . . . 991
8.1017.1 Detailed Description . . . 991
8.1017.2 Field Documentation . . . 991
8.1017.2.1 port . . . 991
8.1017.2.2 range . . . 991
8.1018 unpack_qos_QosFlowInfo_t Struct Reference . . . 991
8.1018.1 Detailed Description . . . 991
8.1018.2 Field Documentation . . . 992
8.1018.2.1 BearerID . . . 992
8.1018.2.2 is_RxQFlowGranted_Available . . . 992
8.1018.2.3 is_TxQFlowGranted_Available . . . 992
8.1018.2.4 NumRxFilters . . . 992
8.1018.2.5 NumTxFilters . . . 992
8.1018.2.6 QFlowState . . . 992
8.1018.2.7 RxQFilter . . . 993
8.1018.2.8 RxQFlowGranted . . . 993

8.1018.2.9 TxQFilter . . . 993
8.1018.2.10 TxQFlowGranted . . . 993
8.1019 unpack_qos_QosFlowInfoState_t Struct Reference . . . 993
8.1019.1 Detailed Description . . . 993
8.1019.2 Field Documentation . . . 993
8.1019.2.1 id . . . 993
8.1019.2.2 isNewFlow . . . 993
8.1019.2.3 state . . . 993
8.1020 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference . . . 993
8.1020.1 Detailed Description . . . 994
8.1020.2 Field Documentation . . . 994
8.1020.2.1 NWQoSStatus . . . 994
8.1020.2.2 ParamPresenceMask . . . 994
8.1021 unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 994
8.1021.1 Detailed Description . . . 994
8.1021.2 Field Documentation . . . 995
8.1021.2.1 ambr_dl . . . 995
8.1021.2.2 ambr_dl_ext . . . 995
8.1021.2.3 ambr_dl_ext2 . . . 995
8.1021.2.4 ambr_ul . . . 995
8.1021.2.5 ambr_ul_ext . . . 995
8.1021.2.6 ambr_ul_ext2 . . . 995
8.1021.2.7 apnId . . . 995
8.1021.2.8 ParamPresenceMask . . . 995
8.1022 unpack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 995
8.1022.1 Detailed Description . . . 996
8.1022.2 Field Documentation . . . 997
8.1022.2.1 apnId . . . 997
8.1022.2.2 numQosFlow . . . 997
8.1022.2.3 ParamPresenceMask . . . 997
8.1022.2.4 qosFlow . . . 997
8.1022.2.5 total_rx_bytes . . . 997
8.1022.2.6 total_rx_pkt . . . 997
8.1022.2.7 total_tx_bytes . . . 997
8.1022.2.8 total_tx_bytes_drp . . . 997
8.1022.2.9 total_tx_pkt . . . 997
8.1022.2.10 total_tx_pkt_drp . . . 997
8.1023 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference . . . 997
8.1023.1 Detailed Description . . . 997
8.1023.2 Field Documentation . . . 997

8.1023.2.1 NumFlows . . . 997
8.1023.2.2 ParamPresenceMask . . . 998
8.1023.2.3 QosFlowInfo . . . 998
8.1024 qmi::nas::unpack_qos_SLQSSetQosEventCallback_t Struct Reference . . . 998
8.1024.1 Detailed Description . . . 998
8.1024.2 Field Documentation . . . 998
8.1024.2.1 ParamPresenceMask . . . 998
8.1025 qmi::nas::unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference . . . 998
8.1025.1 Detailed Description . . . 998
8.1025.2 Field Documentation . . . 998
8.1025.2.1 ParamPresenceMask . . . 998
8.1025.2.2 status . . . 998
8.1026 qmi::nas::unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference . . . 999
8.1026.1 Detailed Description . . . 999
8.1026.2 Field Documentation . . . 999
8.1026.2.1 event . . . 999
8.1026.2.2 ParamPresenceMask . . . 999
8.1027 qmi::nas::unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference . . . 999
8.1027.1 Detailed Description . . . 999
8.1027.2 Field Documentation . . . 1000
8.1027.2.1 event . . . 1000
8.1027.2.2 id . . . 1000
8.1027.2.3 ParamPresenceMask . . . 1000
8.1027.2.4 reason . . . 1000
8.1027.2.5 status . . . 1000
8.1028 qmi::nas::unpack_qos_swiQosFilter_t Struct Reference . . . 1001
8.1028.1 Detailed Description . . . 1001
8.1028.2 Field Documentation . . . 1003
8.1028.2.1 EspSpi . . . 1003
8.1028.2.2 id . . . 1003
8.1028.2.3 index . . . 1003
8.1028.2.4 IPv4DstAddr . . . 1003
8.1028.2.5 IPv4SrcAddr . . . 1003
8.1028.2.6 IPv4Tos . . . 1003
8.1028.2.7 IPv6DstAddr . . . 1003
8.1028.2.8 IPv6Label . . . 1003
8.1028.2.9 IPv6SrcAddr . . . 1003
8.1028.2.10 IPv6TrafCls . . . 1003
8.1028.2.11 is_EspSpi_Available . . . 1003
8.1028.2.12 is_Id_Available . . . 1003

8.1028.2.13 is_IPv4DstAddr_Available . . . 1003
8.1028.2.14 is_IPv4SrcAddr_Available . . . 1003
8.1028.2.15 is_IPv4Tos_Available . . . 1003
8.1028.2.16 is_IPv6DstAddr_Available . . . 1003
8.1028.2.17 is_IPv6Label_Available . . . 1003
8.1028.2.18 is_IPv6SrcAddr_Available . . . 1003
8.1028.2.19 is_IPv6TrafCls_Available . . . 1003
8.1028.2.20 is_NxtHdrProto_Available . . . 1003
8.1028.2.21 is_Precedence_Available . . . 1003
8.1028.2.22 is_TCPDstPort_Available . . . 1003
8.1028.2.23 is_TCPSrcPort_Available . . . 1004
8.1028.2.24 is_TranDstPort_Available . . . 1004
8.1028.2.25 is_TranSrcPort_Available . . . 1004
8.1028.2.26 is_UDPDstPort_Available . . . 1004
8.1028.2.27 is_UDPSrcPort_Available . . . 1004
8.1028.2.28 NxtHdrProto . . . 1004
8.1028.2.29 Precedence . . . 1004
8.1028.2.30 TCPDstPort . . . 1004
8.1028.2.31 TCPSrcPort . . . 1004
8.1028.2.32 TranDstPort . . . 1004
8.1028.2.33 TranSrcPort . . . 1004
8.1028.2.34 UDPDstPort . . . 1004
8.1028.2.35 UDPSrcPort . . . 1004
8.1028.2.36 Version . . . 1004
8.1029 npack_qos_swiQosFlow_t Struct Reference . . . 1004
8.1029.1 Detailed Description . . . 1005
8.1029.2 Field Documentation . . . 1007
8.1029.2.1 DataRate . . . 1007
8.1029.2.2 Index . . . 1007
8.1029.2.3 is_DataRate_Available . . . 1007
8.1029.2.4 is_Jitter_Available . . . 1007
8.1029.2.5 is_Latency_Available . . . 1007
8.1029.2.6 is_LteQci_Available . . . 1007
8.1029.2.7 is_MaxAllowedPktSz_Available . . . 1007
8.1029.2.8 is_MinPolicedPktSz_Available . . . 1007
8.1029.2.9 is_PktErrRate_Available . . . 1007
8.1029.2.10 is_ProfileId3GPP2_Available . . . 1007
8.1029.2.11 is_TokenBucket_Available . . . 1007
8.1029.2.12 is_TrafficClass_Available . . . 1007
8.1029.2.13 is_val_3GPP2Pri_Available . . . 1007

8.1029.2.14 val_3GPPImCn_Available . . . 1007
8.1029.2.15 val_3GPPResResidualBER_Available . . . 1007
8.1029.2.16 val_3GPPSigInd_Available . . . 1007
8.1029.2.17 val_3GPPTraHdlPri_Available . . . 1007
8.1029.2.18 Jitter . . . 1007
8.1029.2.19 Latency . . . 1008
8.1029.2.20 LteQci . . . 1008
8.1029.2.21 MaxAllowedPktSz . . . 1008
8.1029.2.22 MinPolicedPktSz . . . 1008
8.1029.2.23 PktErrRate . . . 1008
8.1029.2.24 ProfileId3GPP2 . . . 1008
8.1029.2.25 TokenBucket . . . 1008
8.1029.2.26 TrafficClass . . . 1008
8.1029.2.27 val_3GPP2Pri . . . 1008
8.1029.2.28 val_3GPPImCn . . . 1008
8.1029.2.29 val_3GPPResResidualBER . . . 1008
8.1029.2.30 val_3GPPSigInd . . . 1008
8.1029.2.31 val_3GPPTraHdlPri . . . 1008
8.1030 unpack_qos_tokenBucket_t Struct Reference . . . 1008
8.1030.1 Detailed Description . . . 1008
8.1030.2 Field Documentation . . . 1008
8.1030.2.1 bucketSz . . . 1008
8.1030.2.2 peakRate . . . 1008
8.1030.2.3 tokenRate . . . 1008
8.1031 unpack_qos_Tos_t Struct Reference . . . 1009
8.1031.1 Detailed Description . . . 1009
8.1031.2 Field Documentation . . . 1009
8.1031.2.1 mask . . . 1009
8.1031.2.2 val . . . 1009
8.1032 unpack_QosFlowStat_t Struct Reference . . . 1009
8.1032.1 Detailed Description . . . 1009
8.1032.2 Field Documentation . . . 1010
8.1032.2.1 bearerId . . . 1010
8.1032.2.2 tx_bytes . . . 1010
8.1032.2.3 tx_bytes_drp . . . 1010
8.1032.2.4 tx_pkt . . . 1010
8.1032.2.5 tx_pkt_drp . . . 1010
8.1033 unpack_result_t Struct Reference . . . 1010
8.1033.1 Detailed Description . . . 1010
8.1033.2 Field Documentation . . . 1010

8.1033.2.1 ParamPresenceMask . . . 1010
8.1033.2.2 Tlvresult . . . 1010
8.1034 unpack_rms_GetSMSWake_t Struct Reference . . . 1010
8.1034.1 Detailed Description . . . 1011
8.1034.2 Field Documentation . . . 1011
8.1034.2.1 enabled . . . 1011
8.1034.2.2 ParamPresenceMask . . . 1011
8.1034.2.3 wake_mask . . . 1011
8.1035 unpack_rms_SetSMSWake_t Struct Reference . . . 1011
8.1035.1 Detailed Description . . . 1011
8.1035.2 Field Documentation . . . 1011
8.1035.2.1 ParamPresenceMask . . . 1011
8.1036 unpack_RMTransferStatistics_ind_t Struct Reference . . . 1011
8.1036.1 Detailed Description . . . 1012
8.1036.2 Field Documentation . . . 1012
8.1036.2.1 ParamPresenceMask . . . 1012
8.1036.2.2 RxDropConutTlv . . . 1012
8.1036.2.3 RxOkByteCountTlv . . . 1012
8.1036.2.4 RxOkConutTlv . . . 1012
8.1036.2.5 TxDropConutTlv . . . 1012
8.1036.2.6 TxOkByteCountTlv . . . 1012
8.1036.2.7 TxOkConutTlv . . . 1012
8.1037 unpack_sar_SLQSGetRfSarState_t Struct Reference . . . 1013
8.1037.1 Detailed Description . . . 1013
8.1037.2 Field Documentation . . . 1013
8.1037.2.1 ParamPresenceMask . . . 1013
8.1037.2.2 pSarRFState . . . 1013
8.1038 unpack_sms_GetSMSCAddress_t Struct Reference . . . 1013
8.1038.1 Detailed Description . . . 1013
8.1038.2 Field Documentation . . . 1014
8.1038.2.1 addressSize . . . 1014
8.1038.2.2 ParamPresenceMask . . . 1014
8.1038.2.3 pSMSCAddress . . . 1014
8.1038.2.4 pSMSCType . . . 1014
8.1038.2.5 typeSize . . . 1014
8.1039 unpack_sms_SaveSMS_t Struct Reference . . . 1014
8.1039.1 Detailed Description . . . 1014
8.1039.2 Field Documentation . . . 1015
8.1039.2.1 ParamPresenceMask . . . 1015
8.1039.2.2 pMessageIndex . . . 1015

8.1040 unpack_sms_SendSMS_t Struct Reference . . . 1015
8.1040.1 Detailed Description . . . 1015
8.1040.2 Field Documentation . . . 1015
8.1040.2.1 messageFailureCode . . . 1015
8.1040.2.2 messageID . . . 1015
8.1040.2.3 ParamPresenceMask . . . 1015
8.1041 unpack_sms_SetNewSMSCallback_ind_t Struct Reference . . . 1015
8.1041.1 Detailed Description . . . 1016
8.1041.2 Field Documentation . . . 1016
8.1041.2.1 ETWSPLMNTlv . . . 1017
8.1041.2.2 ETWSTlv . . . 1017
8.1041.2.3 IMSTlv . . . 1017
8.1041.2.4 MMTlv . . . 1017
8.1041.2.5 NewMMTlv . . . 1017
8.1041.2.6 ParamPresenceMask . . . 1017
8.1041.2.7 SMSCTlv . . . 1017
8.1041.2.8 TRMessageTlv . . . 1017
8.1042 unpack_sms_SetNewSMSCallback_t Struct Reference . . . 1017
8.1042.1 Detailed Description . . . 1017
8.1042.2 Field Documentation . . . 1017
8.1042.2.1 ParamPresenceMask . . . 1017
8.1043 unpack_sms_SetSMSCAddress_t Struct Reference . . . 1017
8.1043.1 Detailed Description . . . 1017
8.1043.2 Field Documentation . . . 1018
8.1043.2.1 ParamPresenceMask . . . 1018
8.1044 unpack_sms_SLQSDeleteSMS_t Struct Reference . . . 1018
8.1044.1 Detailed Description . . . 1018
8.1044.2 Field Documentation . . . 1018
8.1044.2.1 ParamPresenceMask . . . 1018
8.1045 unpack_sms_SLQSGetIndicationRegister_t Struct Reference . . . 1018
8.1045.1 Detailed Description . . . 1018
8.1045.2 Field Documentation . . . 1019
8.1045.2.1 ParamPresenceMask . . . 1019
8.1045.2.2 pGetIndicationRegInfo . . . 1019
8.1046 unpack_sms_SLQSGetMessageWaiting_t Struct Reference . . . 1019
8.1046.1 Detailed Description . . . 1019
8.1046.2 Field Documentation . . . 1019
8.1046.2.1 ParamPresenceMask . . . 1019
8.1046.2.2 pGetMsgWaitingInfoResp . . . 1019
8.1047 unpack_sms_SLQSGetSMS_t Struct Reference . . . 1019

8.1047.1 Detailed Description . . . 1019
8.1047.2 Field Documentation . . . 1020
8.1047.2.1 message . . . 1020
8.1047.2.2 messageFormat . . . 1020
8.1047.2.3 messageSize . . . 1020
8.1047.2.4 messageTag . . . 1020
8.1047.2.5 ParamPresenceMask . . . 1020
8.1048 qmi_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 1020
8.1048.1 Detailed Description . . . 1021
8.1048.2 Field Documentation . . . 1021
8.1048.2.1 ParamPresenceMask . . . 1021
8.1048.2.2 pBroadcastConfig . . . 1021
8.1048.2.3 pCDMABroadcastConfig . . . 1021
8.1049 qmi_sms_SLQSGetSMSList_t Struct Reference . . . 1021
8.1049.1 Detailed Description . . . 1021
8.1049.2 Field Documentation . . . 1022
8.1049.2.1 messageList . . . 1022
8.1049.2.2 messageListSize . . . 1022
8.1049.2.3 ParamPresenceMask . . . 1022
8.1050 qmi_sms_SLQSGetTransLayerInfo_t Struct Reference . . . 1022
8.1050.1 Detailed Description . . . 1022
8.1050.2 Field Documentation . . . 1022
8.1050.2.1 ParamPresenceMask . . . 1022
8.1050.2.2 pGetTransLayerInfo . . . 1022
8.1051 qmi_sms_SLQSGetTransNWRegInfo_t Struct Reference . . . 1022
8.1051.1 Detailed Description . . . 1023
8.1051.2 Field Documentation . . . 1023
8.1051.2.1 ParamPresenceMask . . . 1023
8.1051.2.2 pGetTransNWRegInfo . . . 1023
8.1052 qmi_sms_SLQSModifySMSStatus_t Struct Reference . . . 1023
8.1052.1 Detailed Description . . . 1023
8.1052.2 Field Documentation . . . 1023
8.1052.2.1 ParamPresenceMask . . . 1023
8.1053 qmi_sms_SLQSNWRegInfoCallback_ind_t Struct Reference . . . 1023
8.1053.1 Detailed Description . . . 1024
8.1053.2 Field Documentation . . . 1024
8.1053.2.1 NWRegStat . . . 1024
8.1053.2.2 ParamPresenceMask . . . 1024
8.1054 qmi_sms_SLQSSendAsyncSMS_t Struct Reference . . . 1024
8.1054.1 Detailed Description . . . 1024

8.1054.2 Field Documentation . . . 1024
8.1054.2.1 ParamPresenceMask . . . 1024
8.1055 unpack_sms_SLQSSetIndicationRegister_t Struct Reference . . . 1024
8.1055.1 Detailed Description . . . 1025
8.1055.2 Field Documentation . . . 1025
8.1055.2.1 ParamPresenceMask . . . 1025
8.1056 unpack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference . . . 1025
8.1056.1 Detailed Description . . . 1025
8.1056.2 Field Documentation . . . 1025
8.1056.2.1 ParamPresenceMask . . . 1025
8.1057 unpack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference . . . 1025
8.1057.1 Detailed Description . . . 1025
8.1057.2 Field Documentation . . . 1026
8.1057.2.1 ParamPresenceMask . . . 1026
8.1058 unpack_sms_SLQSSetSmsStorage_t Struct Reference . . . 1026
8.1058.1 Detailed Description . . . 1026
8.1058.2 Field Documentation . . . 1026
8.1058.2.1 ParamPresenceMask . . . 1026
8.1059 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 1026
8.1059.1 Detailed Description . . . 1026
8.1059.2 Field Documentation . . . 1026
8.1059.2.1 ParamPresenceMask . . . 1027
8.1059.2.2 pMaxStorageSizeResp . . . 1027
8.1060 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference . . . 1027
8.1060.1 Detailed Description . . . 1027
8.1060.2 Field Documentation . . . 1027
8.1060.2.1 ParamPresenceMask . . . 1027
8.1060.2.2 pMessageProtocol . . . 1027
8.1061 unpack_sms_SLQSSmsSetRoutes_t Struct Reference . . . 1027
8.1061.1 Detailed Description . . . 1027
8.1061.2 Field Documentation . . . 1027
8.1061.2.1 ParamPresenceMask . . . 1027
8.1062 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference . . . 1028
8.1062.1 Detailed Description . . . 1028
8.1062.2 Field Documentation . . . 1028
8.1062.2.1 ParamPresenceMask . . . 1028
8.1062.2.2 pSmsStorage . . . 1028
8.1063 unpack_sms_SLQSTransLayerInfoCallback_ind_t Struct Reference . . . 1028
8.1063.1 Detailed Description . . . 1028
8.1063.2 Field Documentation . . . 1029

8.1063.2.1 ParamPresenceMask . . . 1029
8.1063.2.2 pTransLayerInfo . . . 1029
8.1063.2.3 regInd . . . 1029
8.1064 unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t Struct Reference . . . 1029
8.1064.1 Detailed Description . . . 1029
8.1064.2 Field Documentation . . . 1031
8.1064.2.1 alphaIDLen . . . 1031
8.1064.2.2 causeCode . . . 1031
8.1064.2.3 errorClass . . . 1031
8.1064.2.4 messageID . . . 1031
8.1064.2.5 msgDelFailureCause . . . 1031
8.1064.2.6 msgDelFailureType . . . 1031
8.1064.2.7 pAlphaID . . . 1031
8.1064.2.8 ParamPresenceMask . . . 1031
8.1064.2.9 RPCause . . . 1031
8.1064.2.10 sendStatus . . . 1031
8.1064.2.11 TPCause . . . 1031
8.1064.2.12 userData . . . 1031
8.1065 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference . . . 1031
8.1065.1 Detailed Description . . . 1031
8.1065.2 Field Documentation . . . 1032
8.1065.2.1 messageMode . . . 1032
8.1065.2.2 ParamPresenceMask . . . 1032
8.1065.2.3 storageType . . . 1032
8.1066 unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t Struct Reference . . . 1032
8.1066.1 Detailed Description . . . 1032
8.1066.2 Field Documentation . . . 1032
8.1066.2.1 msgWaitInfo . . . 1032
8.1066.2.2 numInstances . . . 1032
8.1066.2.3 ParamPresenceMask . . . 1032
8.1067 unpack_swiaudio_SLQSGetM2MAudioProfile_t Struct Reference . . . 1032
8.1067.1 Detailed Description . . . 1033
8.1067.2 Field Documentation . . . 1033
8.1067.2.1 CwtMute . . . 1033
8.1067.2.2 EarMute . . . 1033
8.1067.2.3 Generator . . . 1034
8.1067.2.4 MicMute . . . 1034
8.1067.2.5 ParamPresenceMask . . . 1034
8.1067.2.6 Profile . . . 1034
8.1067.2.7 Volume . . . 1034

8.1068 unpack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference . . . 1034
8.1068.1 Detailed Description . . . 1034
8.1068.2 Field Documentation . . . 1034
8.1068.2.1 Level . . . 1034
8.1068.2.2 ParamPresenceMask . . . 1034
8.1069 unpack_swiaudio_SLQSGetM2MAVMute_t Struct Reference . . . 1034
8.1069.1 Detailed Description . . . 1034
8.1069.2 Field Documentation . . . 1035
8.1069.2.1 CwtMute . . . 1035
8.1069.2.2 EarMute . . . 1035
8.1069.2.3 MicMute . . . 1035
8.1069.2.4 ParamPresenceMask . . . 1035
8.1070 unpack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference . . . 1035
8.1070.1 Detailed Description . . . 1035
8.1070.2 Field Documentation . . . 1036
8.1070.2.1 ParamPresenceMask . . . 1036
8.1070.2.2 Value . . . 1036
8.1071 unpack_swiavms_SLQSAVMSEventReportInd_t Struct Reference . . . 1036
8.1071.1 Detailed Description . . . 1036
8.1071.2 Field Documentation . . . 1037
8.1071.2.1 ParamPresenceMask . . . 1037
8.1071.2.2 pBinaryUpdateSessionInfo . . . 1037
8.1071.2.3 pConfig . . . 1037
8.1071.2.4 pConnectionRequest . . . 1037
8.1071.2.5 pDataSessionStatus . . . 1037
8.1071.2.6 pHTTPStatus . . . 1037
8.1071.2.7 pNotification . . . 1037
8.1071.2.8 pPackageID . . . 1037
8.1071.2.9 pRegStatus . . . 1037
8.1071.2.10 pSessionType . . . 1037
8.1071.2.11 pWAMSParaChanged . . . 1037
8.1071.2.12 resultcode . . . 1037
8.1072 unpack_swiavms_SLQSAVMSGetSettings_t Struct Reference . . . 1037
8.1072.1 Detailed Description . . . 1038
8.1072.2 Field Documentation . . . 1039
8.1072.2.1 AutoConnect . . . 1040
8.1072.2.2 AutoReboot . . . 1040
8.1072.2.3 FwAutoSDM . . . 1040
8.1072.2.4 FwPromptdownload . . . 1040
8.1072.2.5 FwPromptUpdate . . . 1040

8.1072.2.6 OMADMEnabled . . . 1040
8.1072.2.7 pAPNInfo . . . 1040
8.1072.2.8 ParamPresenceMask . . . 1040
8.1072.2.9 pConnectionRetryTimers . . . 1040
8.1072.2.10 pNotificationStore . . . 1040
8.1072.2.11 pPeroidsInfo . . . 1040
8.1072.2.12 pPollingTimer . . . 1040
8.1072.2.13 resultcode . . . 1040
8.1073 unpack_swiavms_SLQSAVMSGetSettings_v2_t Struct Reference . . . 1040
8.1073.1 Detailed Description . . . 1040
8.1073.2 Field Documentation . . . 1042
8.1073.2.1 AutoConnect . . . 1042
8.1073.2.2 FwAutoSDM . . . 1042
8.1073.2.3 FwPromptdownload . . . 1042
8.1073.2.4 FwPromptUpdate . . . 1042
8.1073.2.5 OMADMEnabled . . . 1042
8.1073.2.6 pAPNInfo . . . 1042
8.1073.2.7 ParamPresenceMask . . . 1042
8.1073.2.8 pAutoReboot . . . 1042
8.1073.2.9 pConnectionRetryTimers . . . 1042
8.1073.2.10 pNotificationStore . . . 1042
8.1073.2.11 pPeroidsInfo . . . 1042
8.1073.2.12 pPollingTimer . . . 1042
8.1073.2.13 resultcode . . . 1042
8.1074 unpack_swiavms_SLQSAVMSSendSelection_t Struct Reference . . . 1042
8.1074.1 Detailed Description . . . 1042
8.1074.2 Field Documentation . . . 1043
8.1074.2.1 ParamPresenceMask . . . 1043
8.1074.2.2 resultcode . . . 1043
8.1075 unpack_swiavms_SLQSAVMSSessionGetInfo_t Struct Reference . . . 1043
8.1075.1 Detailed Description . . . 1043
8.1075.2 Field Documentation . . . 1043
8.1075.2.1 ParamPresenceMask . . . 1043
8.1075.2.2 pBinaryUpdateSessionInfo . . . 1043
8.1075.2.3 pConfig . . . 1044
8.1075.2.4 pNotification . . . 1044
8.1075.2.5 pPackageID . . . 1044
8.1075.2.6 resultcode . . . 1044
8.1076 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference . . . 1044
8.1076.1 Detailed Description . . . 1044

8.1076.2 Field Documentation . . . 1044
8.1076.2.1 ParamPresenceMask . . . 1044
8.1076.2.2 resultcode . . . 1044
8.1077 unpack_swiavms_SLQSAVMSSetSettings_t Struct Reference . . . 1044
8.1077.1 Detailed Description . . . 1044
8.1077.2 Field Documentation . . . 1044
8.1077.2.1 ParamPresenceMask . . . 1044
8.1077.2.2 resultcode . . . 1045
8.1078 unpack_swiavms_SLQSAVMSSetSettings_v2_t Struct Reference . . . 1045
8.1078.1 Detailed Description . . . 1045
8.1078.2 Field Documentation . . . 1045
8.1078.2.1 ParamPresenceMask . . . 1045
8.1078.2.2 resultcode . . . 1045
8.1079 unpack_swiavms_SLQSAVMSStartSession_t Struct Reference . . . 1045
8.1079.1 Detailed Description . . . 1045
8.1079.2 Field Documentation . . . 1045
8.1079.2.1 ParamPresenceMask . . . 1045
8.1079.2.2 resultcode . . . 1046
8.1079.2.3 sessionResponse . . . 1046
8.1080 unpack_swiavms_SLQSAVMSStopSession_t Struct Reference . . . 1046
8.1080.1 Detailed Description . . . 1046
8.1080.2 Field Documentation . . . 1046
8.1080.2.1 ParamPresenceMask . . . 1046
8.1080.2.2 resultcode . . . 1046
8.1081 unpack_swidms_SLQSSwiDmsGetHWWatchdog_t Struct Reference . . . 1046
8.1081.1 Detailed Description . . . 1046
8.1081.2 Field Documentation . . . 1046
8.1081.2.1 ParamPresenceMask . . . 1046
8.1081.2.2 pHWWatchdog . . . 1047
8.1082 unpack_swidms_SLQSSwiDmsGetMTU_t Struct Reference . . . 1047
8.1082.1 Detailed Description . . . 1047
8.1082.2 Field Documentation . . . 1047
8.1082.2.1 ParamPresenceMask . . . 1047
8.1082.2.2 pEhrpdMTUSize . . . 1047
8.1082.2.3 pHrpdMTUSize . . . 1047
8.1082.2.4 pMTUSize3gpp . . . 1047
8.1082.2.5 pUsbMTUSize . . . 1047
8.1083 unpack_swidms_SLQSSwiDmsGetSecureInfo_t Struct Reference . . . 1047
8.1083.1 Detailed Description . . . 1048
8.1083.2 Field Documentation . . . 1048

8.1083.2.1 jtagAccessAllowed . . . 1048
8.1083.2.2 memoryDumpAllowed . . . 1048
8.1083.2.3 ParamPresenceMask . . . 1048
8.1083.2.4 secureBootEnabled . . . 1048
8.1083.2.5 TlvResult . . . 1048
8.1084 qmi_swidms_SLQSSwiDmsGetUimSlot_t Struct Reference . . . 1048
8.1084.1 Detailed Description . . . 1049
8.1084.2 Field Documentation . . . 1049
8.1084.2.1 autoUimSwitchMode . . . 1049
8.1084.2.2 multiplexedSimMode . . . 1049
8.1084.2.3 ParamPresenceMask . . . 1049
8.1084.2.4 uimSlotSelected . . . 1049
8.1085 qmi_swidms_SLQSSwiDmsGetUsbComp_t Struct Reference . . . 1049
8.1085.1 Detailed Description . . . 1049
8.1085.2 Field Documentation . . . 1050
8.1085.2.1 ParamPresenceMask . . . 1050
8.1085.2.2 pInterfaceCfg . . . 1050
8.1085.2.3 pSupportedBitmasks . . . 1050
8.1086 qmi_swidms_SLQSSwiDmsGetUsbNetNum_t Struct Reference . . . 1050
8.1086.1 Detailed Description . . . 1050
8.1086.2 Field Documentation . . . 1050
8.1086.2.1 ParamPresenceMask . . . 1050
8.1086.2.2 usbNetNum . . . 1050
8.1087 qmi_swidms_SLQSSwiDmsSetHWWatchdog_t Struct Reference . . . 1050
8.1087.1 Detailed Description . . . 1051
8.1087.2 Field Documentation . . . 1051
8.1087.2.1 ParamPresenceMask . . . 1051
8.1087.2.2 Tlvresult . . . 1051
8.1088 qmi_swidms_SLQSSwiDmsSetMTU_t Struct Reference . . . 1051
8.1088.1 Detailed Description . . . 1051
8.1088.2 Field Documentation . . . 1051
8.1088.2.1 ParamPresenceMask . . . 1051
8.1088.2.2 Tlvresult . . . 1051
8.1089 qmi_swidms_SLQSSwiDmsSetUimSlot_t Struct Reference . . . 1051
8.1089.1 Detailed Description . . . 1052
8.1089.2 Field Documentation . . . 1052
8.1089.2.1 ParamPresenceMask . . . 1052
8.1089.2.2 Tlvresult . . . 1052
8.1090 qmi_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference . . . 1052
8.1090.1 Detailed Description . . . 1052

8.1090.2 Field Documentation . . . 1052
8.1090.2.1 ParamPresenceMask . . . 1052
8.1090.2.2 Tlvresult . . . 1052
8.1091 unpack_swiloc_SwiLocGetAutoStart_t Struct Reference . . . 1052
8.1091.1 Detailed Description . . . 1053
8.1091.2 Field Documentation . . . 1054
8.1091.2.1 fix_rate . . . 1054
8.1091.2.2 fix_rate_reported . . . 1054
8.1091.2.3 fix_type . . . 1054
8.1091.2.4 fix_type_reported . . . 1054
8.1091.2.5 function . . . 1054
8.1091.2.6 function_reported . . . 1054
8.1091.2.7 max_dist . . . 1054
8.1091.2.8 max_dist_reported . . . 1054
8.1091.2.9 max_time . . . 1054
8.1091.2.10 max_time_reported . . . 1054
8.1091.2.11 ParamPresenceMask . . . 1054
8.1092 unpack_swioma_SLQSOMADMAlertCallback_ind_t Struct Reference . . . 1054
8.1092.1 Detailed Description . . . 1054
8.1092.2 Field Documentation . . . 1055
8.1092.2.1 eventType . . . 1055
8.1092.2.2 ParamPresenceMask . . . 1055
8.1092.2.3 SessionInfoConfig . . . 1055
8.1092.2.4 SessionInfoFota . . . 1055
8.1092.2.5 SessionInfoNotification . . . 1055
8.1093 unpack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 1055
8.1093.1 Detailed Description . . . 1056
8.1093.2 Field Documentation . . . 1058
8.1093.2.1 Date . . . 1058
8.1093.2.2 DateLength . . . 1058
8.1093.2.3 ParamPresenceMask . . . 1058
8.1093.2.4 PkgDescLength . . . 1058
8.1093.2.5 PkgDescription . . . 1058
8.1093.2.6 PkgName . . . 1058
8.1093.2.7 PkgNameLength . . . 1058
8.1093.2.8 RetryCount . . . 1058
8.1093.2.9 SessionState . . . 1058
8.1093.2.10 SessionType . . . 1058
8.1093.2.11 Severity . . . 1058
8.1093.2.12 Source . . . 1058

8.1093.2.13 SourceLength . . . 1058
8.1093.2.14 Status . . . 1058
8.1093.2.15 Time . . . 1058
8.1093.2.16 TimeLength . . . 1058
8.1093.2.17 UpdateCompleteStatus . . . 1058
8.1094 qmi_nas_swioma_SLQSOMADMGetSessionInfoExt_t Struct Reference . . . 1058
8.1094.1 Detailed Description . . . 1059
8.1094.2 Field Documentation . . . 1061
8.1094.2.1 fumoResultCode . . . 1061
8.1094.2.2 fumoState . . . 1061
8.1094.2.3 hfaMaxRetry . . . 1061
8.1094.2.4 hfaRetryIndex . . . 1061
8.1094.2.5 hfaRetryInterval . . . 1061
8.1094.2.6 hfaStatus . . . 1061
8.1094.2.7 ParamPresenceMask . . . 1061
8.1094.2.8 pkgDate . . . 1061
8.1094.2.9 pkgDateLength . . . 1061
8.1094.2.10 pkgDesc . . . 1061
8.1094.2.11 pkgDescLength . . . 1061
8.1094.2.12 pkgInstallTime . . . 1061
8.1094.2.13 pkgInstallTimeLength . . . 1061
8.1094.2.14 pkgName . . . 1061
8.1094.2.15 pkgNameLength . . . 1061
8.1094.2.16 pkgSize . . . 1061
8.1094.2.17 pkgVendorName . . . 1061
8.1094.2.18 pkgVendorNameLength . . . 1061
8.1094.2.19 pkgVersionName . . . 1061
8.1094.2.20 pkgVersionNameLength . . . 1061
8.1094.2.21 sessionState . . . 1061
8.1094.2.22 status . . . 1061
8.1095 qmi_nas_swioma_SLQSOMADMGetSettings_t Struct Reference . . . 1062
8.1095.1 Detailed Description . . . 1062
8.1095.2 Field Documentation . . . 1063
8.1095.2.1 Autosdm . . . 1063
8.1095.2.2 FOTAdownload . . . 1063
8.1095.2.3 FOTAUpdate . . . 1063
8.1095.2.4 FwAutoCheck . . . 1063
8.1095.2.5 OMADMEnabled . . . 1063
8.1095.2.6 ParamPresenceMask . . . 1063
8.1096 qmi_nas_swioma_SLQSOMADMStartSession_t Struct Reference . . . 1063

8.1096.1 Detailed Description . . . 1063
8.1096.2 Field Documentation . . . 1064
8.1096.2.1 FwAvailability . . . 1064
8.1096.2.2 ParamPresenceMask . . . 1064
8.1097 unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference . . . 1064
8.1097.1 Detailed Description . . . 1064
8.1097.2 Field Documentation . . . 1064
8.1097.2.1 ParamPresenceMask . . . 1064
8.1097.2.2 Tlvresult . . . 1064
8.1098 unpack_tmd_SLQSTmdGetMitigationDevList_t Struct Reference . . . 1064
8.1098.1 Detailed Description . . . 1064
8.1098.2 Field Documentation . . . 1065
8.1098.2.1 MitigationDevList . . . 1065
8.1098.2.2 MitigationDevListLen . . . 1065
8.1098.2.3 ParamPresenceMask . . . 1065
8.1098.2.4 Tlvresult . . . 1065
8.1099 unpack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference . . . 1065
8.1099.1 Detailed Description . . . 1065
8.1099.2 Field Documentation . . . 1066
8.1099.2.1 CurrentmitigationLvl . . . 1066
8.1099.2.2 ParamPresenceMask . . . 1066
8.1099.2.3 ReqMitigationLvl . . . 1066
8.1099.2.4 Tlvresult . . . 1066
8.1100 unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t Struct Reference . . . 1066
8.1100.1 Detailed Description . . . 1066
8.1100.2 Field Documentation . . . 1066
8.1100.2.1 deviceID . . . 1066
8.1100.2.2 deviceIdLen . . . 1066
8.1100.2.3 lvl . . . 1066
8.1100.2.4 ParamPresenceMask . . . 1066
8.1101 unpack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference . . . 1066
8.1101.1 Detailed Description . . . 1067
8.1101.2 Field Documentation . . . 1067
8.1101.2.1 ParamPresenceMask . . . 1067
8.1101.2.2 Tlvresult . . . 1067
8.1102 unpack_uim_ChangePin_t Struct Reference . . . 1067
8.1102.1 Detailed Description . . . 1067
8.1102.2 Field Documentation . . . 1068
8.1102.2.1 ParamPresenceMask . . . 1068
8.1102.2.2 pEncryptedPIN1 . . . 1068

8.1102.2.3 pIndicationToken . . . 1068
8.1102.2.4 pRemainingRetries . . . 1068
8.1102.2.5 Tlvresult . . . 1068
8.1103 unpack_uim_GetCardStatus_t Struct Reference . . . 1068
8.1103.1 Detailed Description . . . 1068
8.1103.2 Field Documentation . . . 1068
8.1103.2.1 ParamPresenceMask . . . 1068
8.1103.2.2 pCardStatus . . . 1068
8.1103.2.3 pHotSwapStatus . . . 1068
8.1103.2.4 Tlvresult . . . 1069
8.1104 unpack_uim_GetCardStatusV2_t Struct Reference . . . 1069
8.1104.1 Detailed Description . . . 1069
8.1104.2 Field Documentation . . . 1069
8.1104.2.1 ParamPresenceMask . . . 1069
8.1104.2.2 pCardStatus . . . 1069
8.1104.2.3 pHotSwapStatus . . . 1069
8.1104.2.4 pSimBusyStatus . . . 1069
8.1104.2.5 pValidCardStatus . . . 1069
8.1104.2.6 Tlvresult . . . 1069
8.1105 unpack_uim_ReadTransparent_t Struct Reference . . . 1070
8.1105.1 Detailed Description . . . 1070
8.1105.2 Field Documentation . . . 1070
8.1105.2.1 ParamPresenceMask . . . 1070
8.1105.2.2 pCardResult . . . 1070
8.1105.2.3 pEncryptedData . . . 1070
8.1105.2.4 pIndicationToken . . . 1070
8.1105.2.5 pReadResult . . . 1070
8.1105.2.6 Tlvresult . . . 1070
8.1106 unpack_uim_SetPinProtection_t Struct Reference . . . 1071
8.1106.1 Detailed Description . . . 1071
8.1106.2 Field Documentation . . . 1071
8.1106.2.1 ParamPresenceMask . . . 1071
8.1106.2.2 pEncryptedPIN1 . . . 1071
8.1106.2.3 pIndicationToken . . . 1071
8.1106.2.4 pRemainingRetries . . . 1071
8.1106.2.5 Tlvresult . . . 1071
8.1107 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference . . . 1071
8.1107.1 Detailed Description . . . 1072
8.1107.2 Field Documentation . . . 1072
8.1107.2.1 bNumberOfPhySlots . . . 1072

8.1107.2.2 ParamPresenceMask . . . 1072
8.1107.2.3 slotsstatusChange . . . 1072
8.1108 unpack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 1072
8.1108.1 Detailed Description . . . 1072
8.1108.2 Field Documentation . . . 1073
8.1108.2.1 ParamPresenceMask . . . 1073
8.1108.2.2 pAuthenticateResult . . . 1073
8.1108.2.3 pCardResult . . . 1073
8.1108.2.4 pIndicationToken . . . 1073
8.1109 unpack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 1073
8.1109.1 Detailed Description . . . 1073
8.1109.2 Field Documentation . . . 1073
8.1109.2.1 ParamPresenceMask . . . 1073
8.1109.2.2 pRemainingRetries . . . 1073
8.1110 unpack_uim_SLQSUIMEventRegister_t Struct Reference . . . 1073
8.1110.1 Detailed Description . . . 1073
8.1110.2 Field Documentation . . . 1074
8.1110.2.1 eventMask . . . 1074
8.1110.2.2 ParamPresenceMask . . . 1074
8.1111 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 1074
8.1111.1 Detailed Description . . . 1074
8.1111.2 Field Documentation . . . 1075
8.1111.2.1 ParamPresenceMask . . . 1075
8.1111.2.2 pAutoSelection . . . 1075
8.1111.2.3 pHaltSubscription . . . 1075
8.1111.2.4 pPersonalizationStatus . . . 1075
8.1112 unpack_uim_SLQSUIMGetEID_t Struct Reference . . . 1075
8.1112.1 Detailed Description . . . 1075
8.1112.2 Field Documentation . . . 1075
8.1112.2.1 ParamPresenceMask . . . 1075
8.1112.2.2 pEidInfo . . . 1075
8.1113 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 1075
8.1113.1 Detailed Description . . . 1076
8.1113.2 Field Documentation . . . 1076
8.1113.2.1 ParamPresenceMask . . . 1076
8.1113.2.2 pCardResult . . . 1076
8.1113.2.3 pFileAttributes . . . 1076
8.1113.2.4 pIndicationToken . . . 1076
8.1114 unpack_uim_SLQSUIMGetProfileInfo_t Struct Reference . . . 1076
8.1114.1 Detailed Description . . . 1077

8.1114.2 Field Documentation . . . 1077
8.1114.2.1 ParamPresenceMask . . . 1077
8.1114.2.2 pProfileClass . . . 1078
8.1114.2.3 pProfileICCID . . . 1078
8.1114.2.4 pProfileIcon . . . 1078
8.1114.2.5 pProfileIconType . . . 1078
8.1114.2.6 pProfileName . . . 1078
8.1114.2.7 pProfileNickname . . . 1078
8.1114.2.8 pProfilePolicyRules . . . 1078
8.1114.2.9 pProfileSPN . . . 1078
8.1114.2.10 pProfileState . . . 1078
8.1115 npack_uim_SLQSUIMGetServiceStatus_t Struct Reference . . . 1078
8.1115.1 Detailed Description . . . 1078
8.1115.2 Field Documentation . . . 1078
8.1115.2.1 ParamPresenceMask . . . 1079
8.1115.2.2 pFDNStatus . . . 1079
8.1115.2.3 pHiddenKeyStatus . . . 1079
8.1115.2.4 pIndex . . . 1079
8.1116 npack_uim_SLQSUIMGetSIMProfile_t Struct Reference . . . 1079
8.1116.1 Detailed Description . . . 1079
8.1116.2 Field Documentation . . . 1079
8.1116.2.1 pActiveProfileInfo . . . 1079
8.1116.2.2 ParamPresenceMask . . . 1079
8.1116.2.3 pProfileInfo . . . 1079
8.1117 npack_uim_SLQSUIMGetSlotsStatus_t Struct Reference . . . 1079
8.1117.1 Detailed Description . . . 1080
8.1117.2 Field Documentation . . . 1080
8.1117.2.1 ParamPresenceMask . . . 1080
8.1117.2.2 pNumberOfPhySlot . . . 1080
8.1117.2.3 pUimSlotsStatus . . . 1080
8.1118 npack_uim_SLQSUIMGetSlotsStatusV2_t Struct Reference . . . 1080
8.1118.1 Detailed Description . . . 1080
8.1118.2 Field Documentation . . . 1081
8.1118.2.1 ParamPresenceMask . . . 1081
8.1118.2.2 pGetSlotsInfoTlv . . . 1081
8.1118.2.3 pGetSlotsStatusTlv . . . 1081
8.1119 npack_uim_SLQSUIMReadRecord_t Struct Reference . . . 1081
8.1119.1 Detailed Description . . . 1081
8.1119.2 Field Documentation . . . 1082
8.1119.2.1 pAdditionalReadResult . . . 1082

8.1119.2.2 ParamPresenceMask . . . 1082
8.1119.2.3 pCardResult . . . 1082
8.1119.2.4 pIndicationToken . . . 1082
8.1119.2.5 pReadResult . . . 1082
8.1120 unpack_uim_SLQSUIMRefreshCallback_Ind_t Struct Reference . . . 1082
8.1120.1 Detailed Description . . . 1082
8.1120.2 Field Documentation . . . 1082
8.1120.2.1 ParamPresenceMask . . . 1082
8.1120.2.2 refreshEvent . . . 1082
8.1120.2.3 TlvPresent . . . 1082
8.1121 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 1082
8.1121.1 Detailed Description . . . 1083
8.1121.2 Field Documentation . . . 1083
8.1121.2.1 ParamPresenceMask . . . 1083
8.1121.2.2 pRefreshEvent . . . 1083
8.1122 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference . . . 1083
8.1122.1 Detailed Description . . . 1083
8.1122.2 Field Documentation . . . 1083
8.1122.2.1 ParamPresenceMask . . . 1083
8.1122.2.2 pCardStatus . . . 1083
8.1123 unpack_uim_SLQSUIMWriteRecord_t Struct Reference . . . 1083
8.1123.1 Detailed Description . . . 1084
8.1123.2 Field Documentation . . . 1084
8.1123.2.1 ParamPresenceMask . . . 1084
8.1123.2.2 pCardResult . . . 1084
8.1123.2.3 pIndicationToken . . . 1084
8.1124 unpack_uim_SLQSUIMWriteTransparent_t Struct Reference . . . 1084
8.1124.1 Detailed Description . . . 1084
8.1124.2 Field Documentation . . . 1085
8.1124.2.1 ParamPresenceMask . . . 1085
8.1124.2.2 pCardResult . . . 1085
8.1124.2.3 pIndicationToken . . . 1085
8.1125 unpack_uim_UnblockPin_t Struct Reference . . . 1085
8.1125.1 Detailed Description . . . 1085
8.1125.2 Field Documentation . . . 1085
8.1125.2.1 ParamPresenceMask . . . 1086
8.1125.2.2 pEncryptedPIN1 . . . 1086
8.1125.2.3 pIndicationToken . . . 1086
8.1125.2.4 pRemainingRetries . . . 1086
8.1125.2.5 Tlvresult . . . 1086

8.1126 nnpack_uim_UnblockPinV2_t Struct Reference . . . 1086
8.1126.1 Detailed Description . . . 1086
8.1126.2 Field Documentation . . . 1086
8.1126.2.1 ParamPresenceMask . . . 1086
8.1126.2.2 pCardResult . . . 1087
8.1126.2.3 pEncryptedPIN1 . . . 1087
8.1126.2.4 pIndicationToken . . . 1087
8.1126.2.5 pRemainingRetries . . . 1087
8.1126.2.6 Tlvresult . . . 1087
8.1127 nnpack_uim_VerifyPin_t Struct Reference . . . 1087
8.1127.1 Detailed Description . . . 1087
8.1127.2 Field Documentation . . . 1087
8.1127.2.1 ParamPresenceMask . . . 1087
8.1127.2.2 pEncryptedPIN1 . . . 1087
8.1127.2.3 pIndicationToken . . . 1087
8.1127.2.4 pRemainingRetries . . . 1087
8.1127.2.5 Tlvresult . . . 1088
8.1128 nnpack_voice_allCallStatusCallback_ind_t Struct Reference . . . 1088
8.1128.1 Detailed Description . . . 1088
8.1128.2 Field Documentation . . . 1089
8.1128.2.1 arrCallInfomation . . . 1089
8.1128.2.2 ParamPresenceMask . . . 1089
8.1128.2.3 pArrAlertingPattern . . . 1089
8.1128.2.4 pArrAlertingType . . . 1089
8.1128.2.5 pArrAlphaID . . . 1089
8.1128.2.6 pArrCalledPartyNum . . . 1089
8.1128.2.7 pArrCallEndReason . . . 1089
8.1128.2.8 pArrConnectPartyNum . . . 1089
8.1128.2.9 pArrDiagInfo . . . 1089
8.1128.2.10 pArrRedirPartyNum . . . 1089
8.1128.2.11 pArrRemotePartyName . . . 1089
8.1128.2.12 pArrRemotePartyNum . . . 1089
8.1128.2.13 pArrSvcOption . . . 1089
8.1129 nnpack_voice_DTMFEventCallback_ind_t Struct Reference . . . 1090
8.1129.1 Detailed Description . . . 1090
8.1129.2 Field Documentation . . . 1090
8.1129.2.1 DTMFInformation . . . 1090
8.1129.2.2 ParamPresenceMask . . . 1090
8.1129.2.3 pOffLength . . . 1090
8.1129.2.4 pOnLength . . . 1091

8.1130 unpack_voice_OTASPStatusCallback_ind_t Struct Reference . . . 1091
8.1130.1 Detailed Description . . . 1091
8.1130.2 Field Documentation . . . 1092
8.1130.2.1 callID . . . 1092
8.1130.2.2 OTASPStatus . . . 1092
8.1130.2.3 ParamPresenceMask . . . 1092
8.1131 unpack_voice_SLQSOriginateUSSD_t Struct Reference . . . 1092
8.1131.1 Detailed Description . . . 1092
8.1131.2 Field Documentation . . . 1094
8.1131.2.1 pAlphaIDInfo . . . 1094
8.1131.2.2 ParamPresenceMask . . . 1094
8.1131.2.3 pCallId . . . 1094
8.1131.2.4 pCcResultType . . . 1094
8.1131.2.5 pCCSuppsType . . . 1094
8.1131.2.6 pfailureCause . . . 1094
8.1131.2.7 pUSSDInfo . . . 1094
8.1132 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference . . . 1094
8.1132.1 Detailed Description . . . 1094
8.1132.2 Field Documentation . . . 1095
8.1132.2.1 ParamPresenceMask . . . 1095
8.1132.2.2 pCallId . . . 1095
8.1133 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference . . . 1095
8.1133.1 Detailed Description . . . 1095
8.1133.2 Field Documentation . . . 1095
8.1133.2.1 ParamPresenceMask . . . 1095
8.1133.2.2 pCallID . . . 1095
8.1134 unpack_voice_SLQSVoiceDialCall_t Struct Reference . . . 1095
8.1134.1 Detailed Description . . . 1096
8.1134.2 Field Documentation . . . 1096
8.1134.2.1 pAlphaIDInfo . . . 1096
8.1134.2.2 ParamPresenceMask . . . 1096
8.1134.2.3 pCallID . . . 1096
8.1134.2.4 pCCResultType . . . 1096
8.1134.2.5 pCCSUPSType . . . 1096
8.1135 unpack_voice_SLQSVoiceEndCall_t Struct Reference . . . 1096
8.1135.1 Detailed Description . . . 1097
8.1135.2 Field Documentation . . . 1097
8.1135.2.1 ParamPresenceMask . . . 1097
8.1135.2.2 pCallId . . . 1097
8.1136 unpack_voice_SLQSVoiceGetAllCallInfo_t Struct Reference . . . 1097

8.1136.1 Detailed Description . . . 1097
8.1136.2 Field Documentation . . . 1100
8.1136.2.1 ParamPresenceMask . . . 1100
8.1136.2.2 pArrAlertingPattern . . . 1100
8.1136.2.3 pArrAlertingType . . . 1100
8.1136.2.4 pArrAlphaID . . . 1100
8.1136.2.5 pArrCalledPartyNum . . . 1100
8.1136.2.6 pArrCallEndReason . . . 1100
8.1136.2.7 pArrCallInfo . . . 1100
8.1136.2.8 pArrConnectPartyNum . . . 1100
8.1136.2.9 pArrDiagInfo . . . 1100
8.1136.2.10 pArrRedirPartyNum . . . 1100
8.1136.2.11 pArrRemotePartyName . . . 1100
8.1136.2.12 pArrRemotePartyNum . . . 1100
8.1136.2.13 pArrSvcOption . . . 1100
8.1136.2.14 pArrUUSInfo . . . 1100
8.1136.2.15 pOTASPStatus . . . 1100
8.1136.2.16 pVoicePrivacy . . . 1100
8.1137 qmi_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 1101
8.1137.1 Detailed Description . . . 1101
8.1137.2 Field Documentation . . . 1102
8.1137.2.1 pAlphaIDInfo . . . 1102
8.1137.2.2 ParamPresenceMask . . . 1102
8.1137.2.3 pCallID . . . 1102
8.1137.2.4 pCCResType . . . 1102
8.1137.2.5 pCCSUPSType . . . 1102
8.1137.2.6 pFailCause . . . 1102
8.1137.2.7 pSvcClass . . . 1102
8.1138 qmi_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 1102
8.1138.1 Detailed Description . . . 1102
8.1138.2 Field Documentation . . . 1103
8.1138.2.1 pAlphaIDInfo . . . 1103
8.1138.2.2 ParamPresenceMask . . . 1103
8.1138.2.3 pCallID . . . 1103
8.1138.2.4 pCCResType . . . 1103
8.1138.2.5 pCCSUPSType . . . 1103
8.1138.2.6 pFailCause . . . 1104
8.1138.2.7 pGetCallFWExtInfo . . . 1104
8.1138.2.8 pGetCallFWInfo . . . 1104
8.1139 qmi_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 1104

8.1139.1 Detailed Description . . . 1104
8.1139.2 Field Documentation . . . 1106
8.1139.2.1 pAlertingPattern . . . 1106
8.1139.2.2 pAlertType . . . 1106
8.1139.2.3 pAlphaIDInfo . . . 1106
8.1139.2.4 ParamPresenceMask . . . 1106
8.1139.2.5 pCallInfo . . . 1106
8.1139.2.6 pConnectNumInfo . . . 1106
8.1139.2.7 pDiagInfo . . . 1106
8.1139.2.8 pOTASPStatus . . . 1106
8.1139.2.9 pRemotePartyName . . . 1106
8.1139.2.10 pRemotePartyNum . . . 1106
8.1139.2.11 pSrvOpt . . . 1107
8.1139.2.12 pUUSInfo . . . 1107
8.1139.2.13 pVoicePrivacy . . . 1107
8.1140 unpack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 1107
8.1140.1 Detailed Description . . . 1107
8.1140.2 Field Documentation . . . 1108
8.1140.2.1 pAlphaIDInfo . . . 1108
8.1140.2.2 ParamPresenceMask . . . 1108
8.1140.2.3 pCallID . . . 1108
8.1140.2.4 pCCResType . . . 1108
8.1140.2.5 pCCSUPSType . . . 1108
8.1140.2.6 pFailCause . . . 1108
8.1140.2.7 pSvcClass . . . 1108
8.1141 unpack_voice_SLQSVoiceGetCLIP_t Struct Reference . . . 1108
8.1141.1 Detailed Description . . . 1108
8.1141.2 Field Documentation . . . 1109
8.1141.2.1 pAlphaIDInfo . . . 1109
8.1141.2.2 ParamPresenceMask . . . 1109
8.1141.2.3 pCallID . . . 1109
8.1141.2.4 pCCResType . . . 1109
8.1141.2.5 pCCSUPSType . . . 1109
8.1141.2.6 pCLIPResp . . . 1110
8.1141.2.7 pFailCause . . . 1110
8.1142 unpack_voice_SLQSVoiceGetCLIR_t Struct Reference . . . 1110
8.1142.1 Detailed Description . . . 1110
8.1142.2 Field Documentation . . . 1111
8.1142.2.1 pAlphaIDInfo . . . 1111
8.1142.2.2 ParamPresenceMask . . . 1111

8.1142.2.3 pCallID . . . 1111
8.1142.2.4 pCCResType . . . 1111
8.1142.2.5 pCCSUPSType . . . 1111
8.1142.2.6 pCLIRResp . . . 1111
8.1142.2.7 pFailCause . . . 1111
8.1143 unpack_voice_SLQSVoiceGetCNAP_t Struct Reference . . . 1111
8.1143.1 Detailed Description . . . 1111
8.1143.2 Field Documentation . . . 1112
8.1143.2.1 pAlphaIDInfo . . . 1112
8.1143.2.2 ParamPresenceMask . . . 1112
8.1143.2.3 pCallID . . . 1112
8.1143.2.4 pCCResType . . . 1112
8.1143.2.5 pCCSUPSType . . . 1112
8.1143.2.6 pCNAPResp . . . 1112
8.1143.2.7 pFailCause . . . 1112
8.1144 unpack_voice_SLQSVoiceGetCOLP_t Struct Reference . . . 1113
8.1144.1 Detailed Description . . . 1113
8.1144.2 Field Documentation . . . 1114
8.1144.2.1 pAlphaIDInfo . . . 1114
8.1144.2.2 ParamPresenceMask . . . 1114
8.1144.2.3 pCallID . . . 1114
8.1144.2.4 pCCResType . . . 1114
8.1144.2.5 pCCSUPSType . . . 1114
8.1144.2.6 pCOLPResp . . . 1114
8.1144.2.7 pFailCause . . . 1114
8.1145 unpack_voice_SLQSVoiceGetCOLR_t Struct Reference . . . 1114
8.1145.1 Detailed Description . . . 1114
8.1145.2 Field Documentation . . . 1115
8.1145.2.1 pAlphaIDInfo . . . 1115
8.1145.2.2 ParamPresenceMask . . . 1115
8.1145.2.3 pCallID . . . 1115
8.1145.2.4 pCCResType . . . 1115
8.1145.2.5 pCCSUPSType . . . 1115
8.1145.2.6 pCOLRResp . . . 1115
8.1145.2.7 pFailCause . . . 1115
8.1146 unpack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 1115
8.1146.1 Detailed Description . . . 1116
8.1146.2 Field Documentation . . . 1117
8.1146.2.1 pAirTimerCnt . . . 1117
8.1146.2.2 ParamPresenceMask . . . 1117

8.1146.2.3 pAutoAnswerStat . . . 1117
8.1146.2.4 pCurAMRConfig . . . 1117
8.1146.2.5 pCurPrefVoiceSO . . . 1117
8.1146.2.6 pCurrTTYMode . . . 1117
8.1146.2.7 pCurVoiceDomainPref . . . 1117
8.1146.2.8 pCurVoicePrivacyPref . . . 1117
8.1146.2.9 pRoamTimerCnt . . . 1117
8.1147 qmi_voice_SLQSVoiceManageCalls_t Struct Reference . . . 1117
8.1147.1 Detailed Description . . . 1118
8.1147.2 Field Documentation . . . 1118
8.1147.2.1 ParamPresenceMask . . . 1118
8.1147.2.2 pFailCause . . . 1118
8.1148 qmi_voice_SLQSVoiceSendFlash_t Struct Reference . . . 1118
8.1148.1 Detailed Description . . . 1118
8.1148.2 Field Documentation . . . 1118
8.1148.2.1 ParamPresenceMask . . . 1118
8.1148.2.2 pCallID . . . 1119
8.1149 qmi_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 1119
8.1149.1 Detailed Description . . . 1119
8.1149.2 Field Documentation . . . 1120
8.1149.2.1 pAlphaIDInfo . . . 1120
8.1149.2.2 ParamPresenceMask . . . 1120
8.1149.2.3 pCallID . . . 1120
8.1149.2.4 pCCResType . . . 1120
8.1149.2.5 pCCSUPSType . . . 1120
8.1149.2.6 pFailCause . . . 1120
8.1150 qmi_voice_SLQSVoiceSetConfig_t Struct Reference . . . 1120
8.1150.1 Detailed Description . . . 1120
8.1150.2 Field Documentation . . . 1121
8.1150.2.1 pAirTimerStatus . . . 1121
8.1150.2.2 ParamPresenceMask . . . 1121
8.1150.2.3 pAutoAnsStatus . . . 1121
8.1150.2.4 pPrefVoiceSOStatus . . . 1121
8.1150.2.5 pRoamTimerStatus . . . 1121
8.1150.2.6 pTTYConfigStatus . . . 1121
8.1150.2.7 pVoiceDomainPrefStatus . . . 1121
8.1151 qmi_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 1122
8.1151.1 Detailed Description . . . 1122
8.1151.2 Field Documentation . . . 1123
8.1151.2.1 pAlphaIDInfo . . . 1123

8.1151.2.2 ParamPresenceMask . . . 1123
8.1151.2.3 pCallID . . . 1123
8.1151.2.4 pCCResultType . . . 1123
8.1151.2.5 pCCSUPSType . . . 1123
8.1151.2.6 pFailCause . . . 1123
8.1152 npack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 1123
8.1152.1 Detailed Description . . . 1123
8.1152.2 Field Documentation . . . 1123
8.1152.2.1 ParamPresenceMask . . . 1123
8.1152.2.2 pCallID . . . 1123
8.1153 npack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 1123
8.1153.1 Detailed Description . . . 1124
8.1153.2 Field Documentation . . . 1124
8.1153.2.1 callID . . . 1124
8.1153.2.2 ParamPresenceMask . . . 1124
8.1154 npack_voice_SLQSVoiceSUPSCallback_ind_t Struct Reference . . . 1124
8.1154.1 Detailed Description . . . 1124
8.1154.2 Field Documentation . . . 1126
8.1154.2.1 pAlphaIDInfo . . . 1126
8.1154.2.2 ParamPresenceMask . . . 1126
8.1154.2.3 pCallBarPasswd . . . 1126
8.1154.2.4 pCallFwdInfo . . . 1127
8.1154.2.5 pCallFWNum . . . 1127
8.1154.2.6 pCallFWTimerVal . . . 1127
8.1154.2.7 pCallID . . . 1127
8.1154.2.8 pCLIPstatus . . . 1127
8.1154.2.9 pCLIRstatus . . . 1127
8.1154.2.10 pCNAPstatus . . . 1127
8.1154.2.11 pCOLPstatus . . . 1127
8.1154.2.12 pCOLRstatus . . . 1127
8.1154.2.13 pDataSrc . . . 1127
8.1154.2.14 pFailCause . . . 1127
8.1154.2.15 pNewPwdData . . . 1127
8.1154.2.16 pReason . . . 1127
8.1154.2.17 pSvcClass . . . 1127
8.1154.2.18 pUSSInfo . . . 1127
8.1154.2.19 pUPSInformation . . . 1127
8.1155 npack_voice_SUPSNotificationCallback_ind_t Struct Reference . . . 1127
8.1155.1 Detailed Description . . . 1127
8.1155.2 Field Documentation . . . 1129

8.1155.2.1 callID . . . 1129
8.1155.2.2 notifType . . . 1129
8.1155.2.3 ParamPresenceMask . . . 1129
8.1155.2.4 pCUGIndex . . . 1129
8.1155.2.5 pECTNum . . . 1129
8.1156 qmi_voice_USSDNotificationCallback_ind_t Struct Reference . . . 1129
8.1156.1 Detailed Description . . . 1129
8.1156.2 Field Documentation . . . 1129
8.1156.2.1 notification_Type . . . 1129
8.1156.2.2 ParamPresenceMask . . . 1129
8.1156.2.3 USSDNotificationNetworkInfo . . . 1129
8.1157 qmi_voice_USSDNoWaitIndicationCallback_ind_t Struct Reference . . . 1129
8.1157.1 Detailed Description . . . 1130
8.1157.2 Field Documentation . . . 1130
8.1157.2.1 AIDTlv . . . 1130
8.1157.2.2 FCTlv . . . 1130
8.1157.2.3 NwETlv . . . 1130
8.1157.2.4 ParamPresenceMask . . . 1130
8.1157.2.5 UInfoTLV . . . 1130
8.1158 qmi_voice_VoiceInfoRecCallback_ind_t Struct Reference . . . 1130
8.1158.1 Detailed Description . . . 1131
8.1158.2 Field Documentation . . . 1132
8.1158.2.1 callID . . . 1132
8.1158.2.2 ParamPresenceMask . . . 1133
8.1158.2.3 pCalledPartyInfo . . . 1133
8.1158.2.4 pCallerIDInfo . . . 1133
8.1158.2.5 pCallerNameInfo . . . 1133
8.1158.2.6 pCallingPartyInfo . . . 1133
8.1158.2.7 pCallWaitInd . . . 1133
8.1158.2.8 pCLIRCause . . . 1133
8.1158.2.9 pConnectNumInfo . . . 1133
8.1158.2.10 pDispInfo . . . 1133
8.1158.2.11 pExtDispInfo . . . 1133
8.1158.2.12 pExtDispRecInfo . . . 1133
8.1158.2.13 pLineCtrlInfo . . . 1133
8.1158.2.14 pNSSAudioCtrl . . . 1133
8.1158.2.15 pNSSRelease . . . 1133
8.1158.2.16 pRedirNumInfo . . . 1133
8.1158.2.17 pSignalInfo . . . 1133
8.1159 qmi_voice_voicePrivacyChangeCallback_ind_t Struct Reference . . . 1133

8.1159.1 Detailed Description . . . 1133
8.1159.2 Field Documentation . . . 1134
8.1159.2.1 callID . . . 1134
8.1159.2.2 ParamPresenceMask . . . 1134
8.1159.2.3 voicePrivacy . . . 1134
8.1160 unpack_wda_GetDataFormat_t Struct Reference . . . 1134
8.1160.1 Detailed Description . . . 1134
8.1160.2 Field Documentation . . . 1136
8.1160.2.1 Dl_data_aggregation_max_datagrams . . . 1136
8.1160.2.2 Dl_data_aggregation_max_size . . . 1136
8.1160.2.3 Dl_data_aggregation_protocol . . . 1136
8.1160.2.4 Dl_minimum_padding . . . 1136
8.1160.2.5 Flow_control . . . 1136
8.1160.2.6 Ndp_signature . . . 1136
8.1160.2.7 ParamPresenceMask . . . 1136
8.1160.2.8 Qlink_prot . . . 1136
8.1160.2.9 Qos_format . . . 1136
8.1160.2.10 Qos_header_format . . . 1136
8.1160.2.11 Tcp_coalescing . . . 1136
8.1160.2.12 Tlvresult . . . 1136
8.1160.2.13 Udp_coalescing . . . 1136
8.1160.2.14 Ul_data_aggregation_max_datagrams . . . 1136
8.1160.2.15 Ul_data_aggregation_max_size . . . 1136
8.1160.2.16 Ul_data_aggregation_protocol . . . 1136
8.1161 unpack_wda_GetQmapSettings_t Struct Reference . . . 1136
8.1161.1 Detailed Description . . . 1137
8.1161.2 Field Documentation . . . 1137
8.1161.2.1 ParamPresenceMask . . . 1137
8.1161.2.2 pQmap_dfc . . . 1137
8.1161.2.3 Tlvresult . . . 1137
8.1162 unpack_wda_SetDataFormat_t Struct Reference . . . 1137
8.1162.1 Detailed Description . . . 1138
8.1162.2 Field Documentation . . . 1140
8.1162.2.1 ParamPresenceMask . . . 1140
8.1162.2.2 pDl_data_aggregation_max_datagrams . . . 1140
8.1162.2.3 pDl_data_aggregation_max_size . . . 1140
8.1162.2.4 pDl_data_aggregation_protocol . . . 1140
8.1162.2.5 pDl_minimum_padding . . . 1140
8.1162.2.6 pFlow_control . . . 1140
8.1162.2.7 pNdp_signature . . . 1141

8.1162.2.8 pQlink_prot . . . 1141
8.1162.2.9 pQos_format . . . 1141
8.1162.2.10 pQos_header_format . . . 1141
8.1162.2.11 pTcp_coalescing . . . 1141
8.1162.2.12 pUdp_coalescing . . . 1141
8.1162.2.13 pUl_data_aggregation_max_datagrams . . . 1141
8.1162.2.14 pUl_data_aggregation_max_size . . . 1141
8.1162.2.15 pUl_data_aggregation_protocol . . . 1141
8.1162.2.16 Tlvresult . . . 1141
8.1163 qmi_wda_SetQmapSettings_t Struct Reference . . . 1141
8.1163.1 Detailed Description . . . 1141
8.1163.2 Field Documentation . . . 1141
8.1163.2.1 ParamPresenceMask . . . 1141
8.1163.2.2 pQmap_dfc . . . 1141
8.1163.2.3 Tlvresult . . . 1141
8.1164 qmi_wds_DHCPv4ClientLease_ind_t Struct Reference . . . 1142
8.1164.1 Detailed Description . . . 1142
8.1164.2 Field Documentation . . . 1142
8.1164.2.1 DHCPv4LeaseOptTlv . . . 1142
8.1164.2.2 DHCPv4LeaseStateTlv . . . 1142
8.1164.2.3 IPv4AddrTlv . . . 1142
8.1164.2.4 ParamPresenceMask . . . 1142
8.1164.2.5 ProfileIdTlv . . . 1142
8.1165 qmi_wds_GetAutoconnect_t Struct Reference . . . 1142
8.1165.1 Detailed Description . . . 1143
8.1165.2 Field Documentation . . . 1143
8.1165.2.1 ParamPresenceMask . . . 1143
8.1165.2.2 psetting . . . 1143
8.1166 qmi_wds_GetByteTotals_t Struct Reference . . . 1143
8.1166.1 Detailed Description . . . 1143
8.1166.2 Field Documentation . . . 1143
8.1166.2.1 ParamPresenceMask . . . 1143
8.1166.2.2 pRXTotalBytes . . . 1144
8.1166.2.3 pTXTotalBytes . . . 1144
8.1167 qmi_wds_GetConnectionRate_t Struct Reference . . . 1144
8.1167.1 Detailed Description . . . 1144
8.1167.2 Field Documentation . . . 1144
8.1167.2.1 currentChannelRXRate . . . 1144
8.1167.2.2 currentChannelTXRate . . . 1144
8.1167.2.3 maxChannelRXRate . . . 1144

8.1167.2.4 maxChannelTXRate . . . 1144
8.1167.2.5 ParamPresenceMask . . . 1144
8.1168 unpack_wds_GetDataBearerTechnology_t Struct Reference . . . 1145
8.1168.1 Detailed Description . . . 1145
8.1168.2 Field Documentation . . . 1145
8.1168.2.1 ParamPresenceMask . . . 1145
8.1168.2.2 pDataBearer . . . 1145
8.1169 unpack_wds_GetDefaultProfile_t Struct Reference . . . 1145
8.1169.1 Detailed Description . . . 1146
8.1169.2 Field Documentation . . . 1147
8.1169.2.1 apnname . . . 1147
8.1169.2.2 apnsize . . . 1148
8.1169.2.3 auth . . . 1148
8.1169.2.4 ipaddr . . . 1148
8.1169.2.5 ipaddrv6 . . . 1148
8.1169.2.6 name . . . 1148
8.1169.2.7 namesize . . . 1148
8.1169.2.8 ParamPresenceMask . . . 1148
8.1169.2.9 pdptype . . . 1148
8.1169.2.10 pridns . . . 1148
8.1169.2.11 pridnsv6 . . . 1148
8.1169.2.12 secdns . . . 1148
8.1169.2.13 secdnsv6 . . . 1148
8.1169.2.14 username . . . 1148
8.1169.2.15 usersize . . . 1148
8.1170 unpack_wds_GetDefaultProfileNum_t Struct Reference . . . 1148
8.1170.1 Detailed Description . . . 1148
8.1170.2 Field Documentation . . . 1148
8.1170.2.1 index . . . 1148
8.1170.2.2 ParamPresenceMask . . . 1148
8.1171 unpack_wds_GetDefaultProfileV2_t Struct Reference . . . 1149
8.1171.1 Detailed Description . . . 1149
8.1171.2 Field Documentation . . . 1151
8.1171.2.1 apnname . . . 1151
8.1171.2.2 apnsize . . . 1151
8.1171.2.3 auth . . . 1151
8.1171.2.4 ipaddr . . . 1151
8.1171.2.5 ipaddrv6 . . . 1151
8.1171.2.6 name . . . 1151
8.1171.2.7 namesize . . . 1151

8.1171.2.8 ParamPresenceMask . . . 1151
8.1171.2.9 pdptype . . . 1151
8.1171.2.10 pridns . . . 1151
8.1171.2.11 pridnsv6 . . . 1151
8.1171.2.12 pwd . . . 1151
8.1171.2.13 pwdsize . . . 1151
8.1171.2.14 secdns . . . 1151
8.1171.2.15 secdnsv6 . . . 1151
8.1171.2.16 username . . . 1151
8.1171.2.17 usersize . . . 1151
8.1172 qmi_wds_GetDormancyState_t Struct Reference . . . 1151
8.1172.1 Detailed Description . . . 1152
8.1172.2 Field Documentation . . . 1152
8.1172.2.1 dormancyState . . . 1152
8.1172.2.2 ParamPresenceMask . . . 1152
8.1173 qmi_wds_GetLastMobileIPError_t Struct Reference . . . 1152
8.1173.1 Detailed Description . . . 1152
8.1173.2 Field Documentation . . . 1152
8.1173.2.1 error . . . 1152
8.1173.2.2 ParamPresenceMask . . . 1152
8.1174 qmi_wds_GetMobileIP_t Struct Reference . . . 1153
8.1174.1 Detailed Description . . . 1153
8.1174.2 Field Documentation . . . 1153
8.1174.2.1 mipMode . . . 1153
8.1174.2.2 ParamPresenceMask . . . 1153
8.1175 qmi_wds_GetMobileIPProfile_t Struct Reference . . . 1153
8.1175.1 Detailed Description . . . 1153
8.1175.2 Field Documentation . . . 1155
8.1175.2.1 AAASPI . . . 1155
8.1175.2.2 AAAState . . . 1155
8.1175.2.3 address . . . 1155
8.1175.2.4 enabled . . . 1155
8.1175.2.5 HASPI . . . 1155
8.1175.2.6 HAState . . . 1155
8.1175.2.7 NAI . . . 1155
8.1175.2.8 naiSize . . . 1155
8.1175.2.9 ParamPresenceMask . . . 1155
8.1175.2.10 primaryHA . . . 1155
8.1175.2.11 revTunneling . . . 1155
8.1175.2.12 secondaryHA . . . 1155

8.1176 npack_wds_GetPacketStatistics_t Struct Reference . . . 1155
8.1176.1 Detailed Description . . . 1156
8.1176.2 Field Documentation . . . 1157
8.1176.2.1 ParamPresenceMask . . . 1157
8.1176.2.2 pRXDroppedCount . . . 1157
8.1176.2.3 pRXOkBytesCount . . . 1157
8.1176.2.4 pRXOKBytesLastCall . . . 1157
8.1176.2.5 pRXPacketErrors . . . 1157
8.1176.2.6 pRXPacketOverflows . . . 1157
8.1176.2.7 pRXPacketSuccesses . . . 1157
8.1176.2.8 pTXDroppedCount . . . 1157
8.1176.2.9 pTXOkBytesCount . . . 1157
8.1176.2.10 pTXOKBytesLastCall . . . 1157
8.1176.2.11 pTXPacketErrors . . . 1157
8.1176.2.12 pTXPacketOverflows . . . 1157
8.1176.2.13 pTXPacketSuccesses . . . 1157
8.1177 npack_wds_GetPacketStatus_t Struct Reference . . . 1157
8.1177.1 Detailed Description . . . 1158
8.1177.2 Field Documentation . . . 1159
8.1177.2.1 ParamPresenceMask . . . 1159
8.1177.2.2 rXDroppedCount . . . 1159
8.1177.2.3 rXOkBytesCount . . . 1159
8.1177.2.4 rXOKBytesLastCall . . . 1159
8.1177.2.5 rXPacketErrors . . . 1159
8.1177.2.6 rXPacketOverflows . . . 1159
8.1177.2.7 rXPacketSuccesses . . . 1159
8.1177.2.8 tXDroppedCount . . . 1159
8.1177.2.9 tXOkBytesCount . . . 1159
8.1177.2.10 tXOKBytesLastCall . . . 1159
8.1177.2.11 tXPacketErrors . . . 1159
8.1177.2.12 tXPacketOverflows . . . 1159
8.1177.2.13 tXPacketSuccesses . . . 1159
8.1178 npack_wds_GetSessionDuration_t Struct Reference . . . 1159
8.1178.1 Detailed Description . . . 1159
8.1178.2 Field Documentation . . . 1160
8.1178.2.1 callDuration . . . 1160
8.1178.2.2 ParamPresenceMask . . . 1160
8.1179 npack_wds_GetSessionDurationV2_t Struct Reference . . . 1160
8.1179.1 Detailed Description . . . 1160
8.1179.2 Field Documentation . . . 1161

8.1179.2.1 callDuration . . . 1161
8.1179.2.2 ParamPresenceMask . . . 1161
8.1179.2.3 pCallActiveDuration . . . 1161
8.1179.2.4 pLastCallActiveDuration . . . 1161
8.1179.2.5 pLastCallDuration . . . 1161
8.1180 unpack_wds_GetSessionState_t Struct Reference . . . 1161
8.1180.1 Detailed Description . . . 1161
8.1180.2 Field Documentation . . . 1161
8.1180.2.1 connectionStatus . . . 1161
8.1180.2.2 ParamPresenceMask . . . 1161
8.1181 unpack_wds_RMSetTransferStatistics_t Struct Reference . . . 1161
8.1181.1 Detailed Description . . . 1161
8.1181.2 Field Documentation . . . 1162
8.1181.2.1 ParamPresenceMask . . . 1162
8.1182 unpack_wds_SetMobileIPProfile_t Struct Reference . . . 1162
8.1182.1 Detailed Description . . . 1162
8.1182.2 Field Documentation . . . 1162
8.1182.2.1 ParamPresenceMask . . . 1162
8.1183 unpack_wds_SLQSCreateProfile_t Struct Reference . . . 1162
8.1183.1 Detailed Description . . . 1162
8.1183.2 Field Documentation . . . 1163
8.1183.2.1 ParamPresenceMask . . . 1163
8.1183.2.2 pCreateProfileOut . . . 1163
8.1183.2.3 pProfileID . . . 1163
8.1183.2.4 Tlvresult . . . 1163
8.1184 unpack_wds_SLQSDeleteProfile_t Struct Reference . . . 1163
8.1184.1 Detailed Description . . . 1163
8.1184.2 Field Documentation . . . 1163
8.1184.2.1 extendedErrorCode . . . 1163
8.1184.2.2 ParamPresenceMask . . . 1163
8.1185 unpack_wds_SLQSDUNCallInfoCallBack_ind_t Struct Reference . . . 1163
8.1185.1 Detailed Description . . . 1164
8.1185.2 Field Documentation . . . 1165
8.1185.2.1 CRTlv . . . 1165
8.1185.2.2 CSTlv . . . 1165
8.1185.2.3 DBTTlv . . . 1165
8.1185.2.4 DSTlv . . . 1165
8.1185.2.5 LMCERTlv . . . 1165
8.1185.2.6 ParamPresenceMask . . . 1165
8.1185.2.7 RXBOTlv . . . 1165

8.1185.2.8 TXBOTlv . . . 1165
8.1186 qmi_wds_SLQSGet3GPPConfigItem_t Struct Reference . . . 1165
8.1186.1 Detailed Description . . . 1165
8.1186.2 Field Documentation . . . 1166
8.1186.2.1 _3gppRelease . . . 1166
8.1186.2.2 defaultPDNEnabled . . . 1167
8.1186.2.3 LTEAttachProfile . . . 1167
8.1186.2.4 LTEAttachProfileList . . . 1167
8.1186.2.5 LTEAttachProfileListLen . . . 1167
8.1186.2.6 ParamPresenceMask . . . 1167
8.1186.2.7 profileList . . . 1167
8.1187 qmi_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 1167
8.1187.1 Detailed Description . . . 1167
8.1187.2 Field Documentation . . . 1167
8.1187.2.1 currNetworkInfo . . . 1167
8.1187.2.2 networkInfoLen . . . 1168
8.1187.2.3 ParamPresenceMask . . . 1168
8.1187.2.4 prefNetwork . . . 1168
8.1188 qmi_wds_SLQSGetCurrentChannelRate_t Struct Reference . . . 1168
8.1188.1 Detailed Description . . . 1168
8.1188.2 Field Documentation . . . 1169
8.1188.2.1 current_channel_rx_rate . . . 1169
8.1188.2.2 current_channel_tx_rate . . . 1169
8.1188.2.3 max_channel_rx_rate . . . 1169
8.1188.2.4 max_channel_tx_rate . . . 1169
8.1188.2.5 ParamPresenceMask . . . 1169
8.1189 qmi_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 1169
8.1189.1 Detailed Description . . . 1169
8.1189.2 Field Documentation . . . 1169
8.1189.2.1 curDataBearerTechnology . . . 1169
8.1189.2.2 dataBearerMask . . . 1170
8.1189.2.3 lastCallDataBearerTechnology . . . 1170
8.1189.2.4 ParamPresenceMask . . . 1170
8.1190 qmi_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 1170
8.1190.1 Detailed Description . . . 1170
8.1190.2 Field Documentation . . . 1172
8.1190.2.1 callEndReason . . . 1172
8.1190.2.2 channelRate . . . 1172
8.1190.2.3 connectionStatus . . . 1172
8.1190.2.4 dataBearerTech . . . 1172

8.1190.2.5 dormancyStatus . . . 1172
8.1190.2.6 lastCallDataBearerTech . . . 1172
8.1190.2.7 lastCallRXOKBytesCnt . . . 1172
8.1190.2.8 lastCallTXOKBytesCnt . . . 1173
8.1190.2.9 mdmCallDurationActive . . . 1173
8.1190.2.10 ParamPresenceMask . . . 1173
8.1190.2.11 rxOKBytesCount . . . 1173
8.1190.2.12 txOKBytesCount . . . 1173
8.1191 unpack_wds_SLQSGetProfileSettings_t Struct Reference . . . 1173
8.1191.1 Detailed Description . . . 1173
8.1191.2 Field Documentation . . . 1173
8.1191.2.1 ParamPresenceMask . . . 1173
8.1191.2.2 pProfileSettings . . . 1173
8.1191.2.3 ProfileType . . . 1173
8.1191.2.4 Tlvresult . . . 1173
8.1192 unpack_wds_SLQSGetProfileSettingsV2_t Struct Reference . . . 1173
8.1192.1 Detailed Description . . . 1174
8.1192.2 Field Documentation . . . 1174
8.1192.2.1 ParamPresenceMask . . . 1174
8.1192.2.2 pProfileSettings . . . 1174
8.1192.2.3 ProfileType . . . 1174
8.1192.2.4 Tlvresult . . . 1174
8.1193 unpack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 1174
8.1193.1 Detailed Description . . . 1175
8.1193.2 Field Documentation . . . 1176
8.1193.2.1 APNName . . . 1176
8.1193.2.2 Authentication . . . 1176
8.1193.2.3 DomainList . . . 1176
8.1193.2.4 GPRSGrantedQoS . . . 1176
8.1193.2.5 GWAddressV4 . . . 1177
8.1193.2.6 IMCNflag . . . 1177
8.1193.2.7 IPFamilyPreference . . . 1177
8.1193.2.8 IPv4 . . . 1177
8.1193.2.9 IPV6AddrInfo . . . 1177
8.1193.2.10 IPV6GWAddrInfo . . . 1177
8.1193.2.11 Mtu . . . 1177
8.1193.2.12 ParamPresenceMask . . . 1177
8.1193.2.13 PCSCFAddrPCO . . . 1177
8.1193.2.14 PCSCFFQDNAddrList . . . 1177
8.1193.2.15 PDPType . . . 1177

8.1193.2.18 PrimaryDNSV4 . . . 1177
8.1193.2.17 PrimaryDNSV6 . . . 1177
8.1193.2.18 ProfileID . . . 1177
8.1193.2.19 ProfileName . . . 1177
8.1193.2.20 SecondaryDNSV4 . . . 1177
8.1193.2.21 SecondaryDNSV6 . . . 1177
8.1193.2.22 ServerAddrList . . . 1177
8.1193.2.23 SubnetMaskV4 . . . 1177
8.1193.2.24 Technology . . . 1177
8.1193.2.25 UMTSGrantedQoS . . . 1177
8.1193.2.26 Username . . . 1177
8.1194 qmi_wds_SLQSModifyProfile_t Struct Reference . . . 1177
8.1194.1 Detailed Description . . . 1177
8.1194.2 Field Documentation . . . 1178
8.1194.2.1 ParamPresenceMask . . . 1178
8.1194.2.2 pExtErrorCode . . . 1178
8.1195 qmi_wds_SLQSSetIPFamilyPreference_t Struct Reference . . . 1178
8.1195.1 Detailed Description . . . 1178
8.1195.2 Field Documentation . . . 1178
8.1195.2.1 ParamPresenceMask . . . 1178
8.1195.2.2 Tlvresult . . . 1178
8.1196 qmi_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference . . . 1178
8.1196.1 Detailed Description . . . 1179
8.1196.2 Field Documentation . . . 1180
8.1196.2.1 bearerID . . . 1180
8.1196.2.2 conn_status . . . 1180
8.1196.2.3 ipFamily . . . 1180
8.1196.2.4 ParamPresenceMask . . . 1180
8.1196.2.5 reconfigReqd . . . 1180
8.1196.2.6 sessionEndReason . . . 1180
8.1196.2.7 techName . . . 1180
8.1196.2.8 verboseSessnEndReason . . . 1180
8.1196.2.9 verboseSessnEndReasonType . . . 1180
8.1197 qmi_wds_SLQSSetWdsEventCallback_ind_t Struct Reference . . . 1180
8.1197.1 Detailed Description . . . 1181
8.1197.2 Field Documentation . . . 1185
8.1197.2.1 currDBTechAvail . . . 1185
8.1197.2.2 currNWInfo . . . 1185
8.1197.2.3 dataSysStatAvail . . . 1185
8.1197.2.4 dBTechAvail . . . 1185

8.1197.2.5 dBtechExtAvail . . . 1186
8.1197.2.6 dBTechExtRatValue . . . 1186
8.1197.2.7 dBTechExtSoMask . . . 1186
8.1197.2.8 dBTechnology . . . 1186
8.1197.2.9 dBtechnologyExt . . . 1186
8.1197.2.10 dormancyStatAvail . . . 1186
8.1197.2.11 dormancyStatus . . . 1186
8.1197.2.12 ipstatAvail . . . 1186
8.1197.2.13 ipStatus . . . 1186
8.1197.2.14 netInfoLen . . . 1186
8.1197.2.15 ParamPresenceMask . . . 1186
8.1197.2.16 prefNetwork . . . 1186
8.1197.2.17 ratMask . . . 1186
8.1197.2.18 rx_bytes . . . 1186
8.1197.2.19 rx_pkts . . . 1186
8.1197.2.20 soMask . . . 1186
8.1197.2.21 tx_bytes . . . 1186
8.1197.2.22 tx_pkts . . . 1186
8.1197.2.23 ueStatAvail . . . 1186
8.1198 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 1186
8.1198.1 Detailed Description . . . 1186
8.1198.2 Field Documentation . . . 1187
8.1198.2.1 ParamPresenceMask . . . 1187
8.1198.2.2 pHwConfig . . . 1187
8.1198.2.3 pRequestOptionList . . . 1187
8.1199 unpack_wds_SLQSSGetLoopback_t Struct Reference . . . 1187
8.1199.1 Detailed Description . . . 1187
8.1199.2 Field Documentation . . . 1188
8.1199.2.1 ByteLoopbackMode . . . 1188
8.1199.2.2 ByteLoopbackMultiplier . . . 1188
8.1199.2.3 ParamPresenceMask . . . 1188
8.1200 unpack_wds_SLQSStartDataSession_t Struct Reference . . . 1188
8.1200.1 Detailed Description . . . 1188
8.1200.2 Field Documentation . . . 1188
8.1200.2.1 ParamPresenceMask . . . 1188
8.1200.2.2 pFailureReason . . . 1188
8.1200.2.3 psid . . . 1189
8.1200.2.4 pVerboseFailReasonType . . . 1189
8.1200.2.5 pVerboseFailureReason . . . 1189
8.1201 unpack_wds_SLQSSwiProfileChangeCallback_Ind_t Struct Reference . . . 1189

8.1201.1 Detailed Description . . . 1189
8.1201.2 Field Documentation . . . 1189
8.1201.2.1 ParamPresenceMask . . . 1189
8.1201.2.2 ProfileTlv . . . 1189
8.1201.2.3 srcTlv . . . 1189
8.1202 unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 1189
8.1202.1 Detailed Description . . . 1190
8.1202.2 Field Documentation . . . 1191
8.1202.2.1 apnName . . . 1191
8.1202.2.2 bearerId . . . 1191
8.1202.2.3 contextId . . . 1191
8.1202.2.4 ipv4Address . . . 1191
8.1202.2.5 ipv4GWAddress . . . 1191
8.1202.2.6 ipv6Address . . . 1191
8.1202.2.7 ipv6GWAddress . . . 1191
8.1202.2.8 ParamPresenceMask . . . 1191
8.1202.2.9 prDNSIPv4Address . . . 1191
8.1202.2.10 prDNSIPv6Address . . . 1192
8.1202.2.11 prPCSCFIPv4Address . . . 1192
8.1202.2.12 prPCSCFIPv6Address . . . 1192
8.1202.2.13 seDNSIPv4Address . . . 1192
8.1202.2.14 seDNSIPv6Address . . . 1192
8.1202.2.15 sePCSCFIPv4Address . . . 1192
8.1202.2.16 sePCSCFIPv6Address . . . 1192
8.1203 UnPackGetProfileSettingOut Struct Reference . . . 1192
8.1203.1 Detailed Description . . . 1192
8.1203.2 Field Documentation . . . 1192
8.1203.2.1 curProfile . . . 1192
8.1203.2.2 pExtErrCode . . . 1192
8.1204 UnPackGetProfileSettingOutV2 Struct Reference . . . 1192
8.1204.1 Detailed Description . . . 1193
8.1204.2 Field Documentation . . . 1193
8.1204.2.1 curProfile . . . 1193
8.1204.2.2 pExtErrCode . . . 1193
8.1205 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo Struct Reference . . . 1193
8.1205.1 Detailed Description . . . 1193
8.1205.2 Field Documentation . . . 1194
8.1205.2.1 bBinaryType . . . 1194
8.1205.2.2 bSerity . . . 1194
8.1205.2.3 bState . . . 1194

8.1205.2.4 bUserInputRequest . . . 1194
8.1205.2.5 szDescription . . . 1194
8.1205.2.6 szName . . . 1195
8.1205.2.7 szVersion . . . 1195
8.1205.2.8 TlvPresent . . . 1195
8.1205.2.9 ulPkgDownloadComplete . . . 1195
8.1205.2.10 ulPkgDownloadSize . . . 1195
8.1205.2.11 wDescriptionLength . . . 1195
8.1205.2.12 wNameLength . . . 1195
8.1205.2.13 wUpdateCompeteStatus . . . 1195
8.1205.2.14 wUserInputTimeout . . . 1195
8.1205.2.15 wVersionLength . . . 1195
8.1206 UnpackSwiAvmsEventReportConfig Struct Reference . . . 1195
8.1206.1 Detailed Description . . . 1195
8.1206.2 Field Documentation . . . 1196
8.1206.2.1 bState . . . 1196
8.1206.2.2 bUserInputRequest . . . 1196
8.1206.2.3 szAlertMsg . . . 1196
8.1206.2.4 TlvPresent . . . 1196
8.1206.2.5 wAlertMsgLength . . . 1196
8.1206.2.6 wUserInputTimeout . . . 1196
8.1207 UnpackSwiAvmsEventReportConnectionRequest Struct Reference . . . 1196
8.1207.1 Detailed Description . . . 1196
8.1207.2 Field Documentation . . . 1196
8.1207.2.1 bUserInputRequest . . . 1196
8.1207.2.2 TlvPresent . . . 1196
8.1207.2.3 wUserInputTimeout . . . 1196
8.1208 UnpackSwiAvmsEventReportDataSessionStatus Struct Reference . . . 1196
8.1208.1 Detailed Description . . . 1197
8.1208.2 Field Documentation . . . 1197
8.1208.2.1 bType . . . 1197
8.1208.2.2 TlvPresent . . . 1197
8.1208.2.3 wErrorCode . . . 1197
8.1209 UnpackSwiAvmsEventReportHTTPStatus Struct Reference . . . 1197
8.1209.1 Detailed Description . . . 1197
8.1209.2 Field Documentation . . . 1198
8.1209.2.1 TlvPresent . . . 1198
8.1209.2.2 wHTTPStatus . . . 1199
8.1210 UnpackSwiAvmsEventReportNotification Struct Reference . . . 1199
8.1210.1 Detailed Description . . . 1199

8.1210.2 Field Documentation . . . 1199
8.1210.2.1 bNotification . . . 1200
8.1210.2.2 TlvPresent . . . 1200
8.1210.2.3 wSessionStatus . . . 1200
8.1211 UnpackSwiAvmsEventReportPackageID Struct Reference . . . 1200
8.1211.1 Detailed Description . . . 1200
8.1211.2 Field Documentation . . . 1200
8.1211.2.1 bPackageID . . . 1200
8.1211.2.2 TlvPresent . . . 1200
8.1212 UnpackSwiAvmsEventReportRegStatus Struct Reference . . . 1200
8.1212.1 Detailed Description . . . 1200
8.1212.2 Field Documentation . . . 1200
8.1212.2.1 bRegStatus . . . 1200
8.1212.2.2 TlvPresent . . . 1201
8.1213 UnpackSwiAvmsEventReportSessionType Struct Reference . . . 1201
8.1213.1 Detailed Description . . . 1201
8.1213.2 Field Documentation . . . 1201
8.1213.2.1 bType . . . 1201
8.1213.2.2 TlvPresent . . . 1201
8.1214 UnpackSwiAvmsEventReportWAMSParamChange Struct Reference . . . 1201
8.1214.1 Detailed Description . . . 1201
8.1214.2 Field Documentation . . . 1202
8.1214.2.1 TlvPresent . . . 1202
8.1214.2.2 wWamsChangeMask . . . 1202
8.1215 UnpackWdsProfileParam Union Reference . . . 1202
8.1215.1 Detailed Description . . . 1202
8.1215.2 Field Documentation . . . 1202
8.1215.2.1 SlqsProfile3GPP . . . 1202
8.1215.2.2 SlqsProfile3GPP2 . . . 1202
8.1216 UnpackWdsProfileParamV2 Union Reference . . . 1202
8.1216.1 Detailed Description . . . 1202
8.1216.2 Field Documentation . . . 1203
8.1216.2.1 SlqsProfile3GPP . . . 1203
8.1216.2.2 SlqsProfile3GPP2 . . . 1203
8.1217 Voice_airTimer Struct Reference . . . 1203
8.1217.1 Detailed Description . . . 1203
8.1217.2 Field Documentation . . . 1203
8.1217.2.1 airTimerValue . . . 1203
8.1217.2.2 namID . . . 1203
8.1218 Voice_allCallsAlphaIDInfo Struct Reference . . . 1203

8.1218.1 Detailed Description . . . 1203
8.1218.2 Field Documentation . . . 1204
8.1218.2.1 AlphaIDInfo . . . 1204
8.1218.2.2 callID . . . 1204
8.1219 voice_allCallsDiagInfo Struct Reference . . . 1204
8.1219.1 Detailed Description . . . 1204
8.1219.2 Field Documentation . . . 1204
8.1219.2.1 callID . . . 1204
8.1219.2.2 DiagInfo . . . 1204
8.1220 voice_allCallsUUSInfo Struct Reference . . . 1204
8.1220.1 Detailed Description . . . 1204
8.1220.2 Field Documentation . . . 1205
8.1220.2.1 callID . . . 1205
8.1220.2.2 uusInfo . . . 1205
8.1221 voice_alphaIDInfo Struct Reference . . . 1205
8.1221.1 Detailed Description . . . 1205
8.1221.2 Field Documentation . . . 1205
8.1221.2.1 alphaDcs . . . 1205
8.1221.2.2 alphaLen . . . 1205
8.1221.2.3 alphaText . . . 1205
8.1222 voice_arrAlertingPattern Struct Reference . . . 1205
8.1222.1 Detailed Description . . . 1206
8.1222.2 Field Documentation . . . 1206
8.1222.2.1 alertingPattern . . . 1206
8.1222.2.2 callID . . . 1206
8.1222.2.3 numInstances . . . 1206
8.1223 voice_arrAlertingType Struct Reference . . . 1206
8.1223.1 Detailed Description . . . 1206
8.1223.2 Field Documentation . . . 1207
8.1223.2.1 AlertingType . . . 1207
8.1223.2.2 callID . . . 1207
8.1223.2.3 numInstances . . . 1207
8.1224 voice_arrAlphaID Struct Reference . . . 1207
8.1224.1 Detailed Description . . . 1207
8.1224.2 Field Documentation . . . 1207
8.1224.2.1 allCallsAlphaIDInfoArr . . . 1207
8.1224.2.2 numInstances . . . 1207
8.1225 voice_arrCalledPartyNum Struct Reference . . . 1207
8.1225.1 Detailed Description . . . 1208
8.1225.2 Field Documentation . . . 1208

8.1225.2.1 CalledPartyNum . . . 1208
8.1225.2.2 numInstances . . . 1208
8.1226 voice_arrCallEndReason Struct Reference . . . 1208
8.1226.1 Detailed Description . . . 1208
8.1226.2 Field Documentation . . . 1209
8.1226.2.1 callEndReason . . . 1209
8.1226.2.2 callID . . . 1209
8.1226.2.3 numInstances . . . 1209
8.1227 voice_arrCallInfo Struct Reference . . . 1209
8.1227.1 Detailed Description . . . 1209
8.1227.2 Field Documentation . . . 1209
8.1227.2.1 getAllCallInfo . . . 1209
8.1227.2.2 numInstances . . . 1209
8.1228 voice_arrConnectPartyNum Struct Reference . . . 1209
8.1228.1 Detailed Description . . . 1209
8.1228.2 Field Documentation . . . 1210
8.1228.2.1 ConnectedPartyNum . . . 1210
8.1228.2.2 numInstances . . . 1210
8.1229 voice_arrDiagInfo Struct Reference . . . 1210
8.1229.1 Detailed Description . . . 1210
8.1229.2 Field Documentation . . . 1210
8.1229.2.1 DiagInfo . . . 1210
8.1229.2.2 numInstances . . . 1210
8.1230 voice_arrRedirPartyNum Struct Reference . . . 1210
8.1230.1 Detailed Description . . . 1210
8.1230.2 Field Documentation . . . 1211
8.1230.2.1 numInstances . . . 1211
8.1230.2.2 RedirPartyNum . . . 1211
8.1231 voice_arrRemotePartyName Struct Reference . . . 1211
8.1231.1 Detailed Description . . . 1211
8.1231.2 Field Documentation . . . 1211
8.1231.2.1 GetAllCallRmtPtyName . . . 1211
8.1231.2.2 numInstances . . . 1211
8.1232 voice_arrRemotePartyNum Struct Reference . . . 1211
8.1232.1 Detailed Description . . . 1212
8.1232.2 Field Documentation . . . 1212
8.1232.2.1 numInstances . . . 1212
8.1232.2.2 RmtPtyNum . . . 1212
8.1233 voice_arrSvcOption Struct Reference . . . 1212
8.1233.1 Detailed Description . . . 1212

8.1233.2 Field Documentation . . . 1212
8.1233.2.1 callID . . . 1212
8.1233.2.2 numInstances . . . 1212
8.1233.2.3 srvOption . . . 1213
8.1234 voice_arrUUSInfo Struct Reference . . . 1213
8.1234.1 Detailed Description . . . 1213
8.1234.2 Field Documentation . . . 1213
8.1234.2.1 AllCallsUUSInfo . . . 1213
8.1234.2.2 numInstances . . . 1213
8.1235 voice_burstDTMFInfo Struct Reference . . . 1213
8.1235.1 Detailed Description . . . 1213
8.1235.2 Field Documentation . . . 1214
8.1235.2.1 digitCnt . . . 1214
8.1235.2.2 pCallID . . . 1214
8.1235.2.3 pDigitBuff . . . 1214
8.1236 voice_calledPartyInfo Struct Reference . . . 1214
8.1236.1 Detailed Description . . . 1214
8.1236.2 Field Documentation . . . 1215
8.1236.2.1 number . . . 1215
8.1236.2.2 numLen . . . 1215
8.1236.2.3 numPlan . . . 1215
8.1236.2.4 numType . . . 1215
8.1236.2.5 PI . . . 1215
8.1236.2.6 SI . . . 1215
8.1237 voice_calledPartySubAdd Struct Reference . . . 1215
8.1237.1 Detailed Description . . . 1216
8.1237.2 Field Documentation . . . 1216
8.1237.2.1 extBit . . . 1216
8.1237.2.2 oddEvenInd . . . 1216
8.1237.2.3 subAddr . . . 1216
8.1237.2.4 subAddrLen . . . 1216
8.1237.2.5 subAddrType . . . 1216
8.1238 voice_callerIDInfo Struct Reference . . . 1216
8.1238.1 Detailed Description . . . 1217
8.1238.2 Field Documentation . . . 1217
8.1238.2.1 callerID . . . 1217
8.1238.2.2 callerIDLen . . . 1217
8.1238.2.3 PI . . . 1217
8.1239 voice_callFwdTypeAndPlan Struct Reference . . . 1217
8.1239.1 Detailed Description . . . 1217

8.1239.2 Field Documentation . . . 1218
8.1239.2.1 numberPlan . . . 1218
8.1239.2.2 numberType . . . 1218
8.1240 voice_callFWExtInfo Struct Reference . . . 1218
8.1240.1 Detailed Description . . . 1218
8.1240.2 Field Documentation . . . 1219
8.1240.2.1 noReplyTimer . . . 1219
8.1240.2.2 number . . . 1219
8.1240.2.3 numLen . . . 1220
8.1240.2.4 numPlan . . . 1220
8.1240.2.5 numType . . . 1220
8.1240.2.6 PI . . . 1220
8.1240.2.7 SI . . . 1220
8.1240.2.8 SvcClass . . . 1220
8.1240.2.9 SvcStatus . . . 1220
8.1241 voice_callFWInfo Struct Reference . . . 1220
8.1241.1 Detailed Description . . . 1220
8.1241.2 Field Documentation . . . 1220
8.1241.2.1 noReplyTimer . . . 1220
8.1241.2.2 number . . . 1221
8.1241.2.3 numLen . . . 1221
8.1241.2.4 SvcClass . . . 1221
8.1241.2.5 SvcStatus . . . 1221
8.1242 voice_callInfo Struct Reference . . . 1221
8.1242.1 Detailed Description . . . 1221
8.1242.2 Field Documentation . . . 1222
8.1242.2.1 callID . . . 1222
8.1242.2.2 callState . . . 1222
8.1242.2.3 callType . . . 1222
8.1242.2.4 direction . . . 1222
8.1242.2.5 mode . . . 1222
8.1243 voice_callingPartyInfo Struct Reference . . . 1222
8.1243.1 Detailed Description . . . 1223
8.1243.2 Field Documentation . . . 1224
8.1243.2.1 number . . . 1224
8.1243.2.2 numLen . . . 1224
8.1243.2.3 numPlan . . . 1224
8.1243.2.4 numType . . . 1224
8.1243.2.5 PI . . . 1224
8.1243.2.6 SI . . . 1224

8.1244 voice_ccSUPSType Struct Reference . . . 1224
8.1244.1 Detailed Description . . . 1224
8.1244.2 Field Documentation . . . 1224
8.1244.2.1 reason . . . 1225
8.1244.2.2 svcType . . . 1225
8.1245 voice_CLIPResp Struct Reference . . . 1225
8.1245.1 Detailed Description . . . 1225
8.1245.2 Field Documentation . . . 1225
8.1245.2.1 ActiveStatus . . . 1225
8.1245.2.2 ProvisionStatus . . . 1225
8.1246 voice_CLIRResp Struct Reference . . . 1225
8.1246.1 Detailed Description . . . 1225
8.1246.2 Field Documentation . . . 1226
8.1246.2.1 ActiveStatus . . . 1226
8.1246.2.2 ProvisionStatus . . . 1226
8.1247 voice_CNAPResp Struct Reference . . . 1226
8.1247.1 Detailed Description . . . 1226
8.1247.2 Field Documentation . . . 1227
8.1247.2.1 ActiveStatus . . . 1227
8.1247.2.2 ProvisionStatus . . . 1227
8.1248 voice_COLPResp Struct Reference . . . 1227
8.1248.1 Detailed Description . . . 1227
8.1248.2 Field Documentation . . . 1227
8.1248.2.1 ActiveStatus . . . 1227
8.1248.2.2 ProvisionStatus . . . 1227
8.1249 voice_COLRResp Struct Reference . . . 1227
8.1249.1 Detailed Description . . . 1227
8.1249.2 Field Documentation . . . 1228
8.1249.2.1 ActiveStatus . . . 1228
8.1249.2.2 ProvisionStatus . . . 1228
8.1250 voice_connectNumInfo Struct Reference . . . 1228
8.1250.1 Detailed Description . . . 1228
8.1250.2 Field Documentation . . . 1229
8.1250.2.1 callerID . . . 1229
8.1250.2.2 callerIDLen . . . 1229
8.1250.2.3 numPlan . . . 1229
8.1250.2.4 numPresInd . . . 1230
8.1250.2.5 numType . . . 1230
8.1250.2.6 screeningInd . . . 1230
8.1251 voice_CUGInfo Struct Reference . . . 1230

8.1251.1 Detailed Description . . . 1230
8.1251.2 Field Documentation . . . 1230
8.1251.2.1 CUGIndex . . . 1230
8.1251.2.2 SuppOA . . . 1230
8.1251.2.3 SuppPrefCUG . . . 1230
8.1252 voice_curAMRConfig Struct Reference . . . 1230
8.1252.1 Detailed Description . . . 1230
8.1252.2 Field Documentation . . . 1231
8.1252.2.1 gsmAmrStat . . . 1231
8.1252.2.2 wcdmaAmrStat . . . 1231
8.1253 voice_diagInfo Struct Reference . . . 1231
8.1253.1 Detailed Description . . . 1231
8.1253.2 Field Documentation . . . 1231
8.1253.2.1 diagInfoLen . . . 1231
8.1253.2.2 diagnosticInfo . . . 1231
8.1254 voice_DTMFInfo Struct Reference . . . 1232
8.1254.1 Detailed Description . . . 1232
8.1254.2 Field Documentation . . . 1232
8.1254.2.1 callID . . . 1232
8.1254.2.2 digitBuff . . . 1232
8.1254.2.3 digitCnt . . . 1232
8.1254.2.4 DTMFEvent . . . 1232
8.1255 voice_DTMFLengths Struct Reference . . . 1232
8.1255.1 Detailed Description . . . 1233
8.1255.2 Field Documentation . . . 1233
8.1255.2.1 DTMFInterdigitInterval . . . 1233
8.1255.2.2 DTMFPulseWidth . . . 1233
8.1256 voice_ECTNum Struct Reference . . . 1233
8.1256.1 Detailed Description . . . 1233
8.1256.2 Field Documentation . . . 1234
8.1256.2.1 ECTCallState . . . 1234
8.1256.2.2 number . . . 1234
8.1256.2.3 presentationInd . . . 1234
8.1257 voice_extDispRecInfo Struct Reference . . . 1234
8.1257.1 Detailed Description . . . 1234
8.1257.2 Field Documentation . . . 1234
8.1257.2.1 dispType . . . 1234
8.1257.2.2 extDispInfo . . . 1234
8.1257.2.3 extDispInfoLen . . . 1235
8.1258 voice_getAllCallInformation Struct Reference . . . 1235

8.1258.1 Detailed Description . . . 1235
8.1258.2 Field Documentation . . . 1235
8.1258.2.1 ALS . . . 1235
8.1258.2.2 CallInfo . . . 1235
8.1258.2.3 isEmpty . . . 1235
8.1259 voice_getAllCallRmtPtyName Struct Reference . . . 1235
8.1259.1 Detailed Description . . . 1235
8.1259.2 Field Documentation . . . 1236
8.1259.2.1 callID . . . 1236
8.1259.2.2 RemotePartyName . . . 1236
8.1260 voice_getAllCallRmtPtyNum Struct Reference . . . 1236
8.1260.1 Detailed Description . . . 1236
8.1260.2 Field Documentation . . . 1236
8.1260.2.1 callID . . . 1236
8.1260.2.2 RemotePartyNum . . . 1236
8.1261 voice_getCallFWExtInfo Struct Reference . . . 1236
8.1261.1 Detailed Description . . . 1236
8.1261.2 Field Documentation . . . 1237
8.1261.2.1 CallFWExtInfo . . . 1237
8.1261.2.2 numInstances . . . 1237
8.1262 voice_getCallFWInfo Struct Reference . . . 1237
8.1262.1 Detailed Description . . . 1237
8.1262.2 Field Documentation . . . 1237
8.1262.2.1 CallFWInfo . . . 1237
8.1262.2.2 numInstances . . . 1237
8.1263 voice_lineCtrlInfo Struct Reference . . . 1237
8.1263.1 Detailed Description . . . 1238
8.1263.2 Field Documentation . . . 1238
8.1263.2.1 polarityIncluded . . . 1238
8.1263.2.2 pwrDenialTime . . . 1238
8.1263.2.3 revPolarity . . . 1238
8.1263.2.4 toggleMode . . . 1238
8.1264 voice_newPwdData Struct Reference . . . 1238
8.1264.1 Detailed Description . . . 1238
8.1264.2 Field Documentation . . . 1239
8.1264.2.1 newPwd . . . 1239
8.1264.2.2 newPwdAgain . . . 1239
8.1265 voice_NSSAudioCtrl Struct Reference . . . 1239
8.1265.1 Detailed Description . . . 1239
8.1265.2 Field Documentation . . . 1239

8.1265.2.1 downLink . . . 1239
8.1265.2.2 upLink . . . 1239
8.1266 voice_peerNumberInfo Struct Reference . . . 1239
8.1266.1 Detailed Description . . . 1239
8.1266.2 Field Documentation . . . 1240
8.1266.2.1 callID . . . 1241
8.1266.2.2 number . . . 1241
8.1266.2.3 numLen . . . 1241
8.1266.2.4 numPI . . . 1241
8.1266.2.5 numPlan . . . 1241
8.1266.2.6 numSI . . . 1241
8.1266.2.7 numType . . . 1241
8.1267 voice_prefVoiceSO Struct Reference . . . 1241
8.1267.1 Detailed Description . . . 1241
8.1267.2 Field Documentation . . . 1243
8.1267.2.1 evrcCapability . . . 1243
8.1267.2.2 homeOrigVoiceSO . . . 1243
8.1267.2.3 homePageVoiceSO . . . 1243
8.1267.2.4 namID . . . 1243
8.1267.2.5 roamOrigVoiceSO . . . 1243
8.1268 voice_redirNumInfo Struct Reference . . . 1243
8.1268.1 Detailed Description . . . 1243
8.1268.2 Field Documentation . . . 1244
8.1268.2.1 number . . . 1244
8.1268.2.2 numLen . . . 1244
8.1268.2.3 numPlan . . . 1244
8.1268.2.4 numType . . . 1244
8.1268.2.5 PI . . . 1244
8.1268.2.6 reason . . . 1244
8.1268.2.7 SI . . . 1244
8.1269 voice_remotePartyName Struct Reference . . . 1245
8.1269.1 Detailed Description . . . 1245
8.1269.2 Field Documentation . . . 1245
8.1269.2.1 callerName . . . 1245
8.1269.2.2 codingScheme . . . 1245
8.1269.2.3 nameLen . . . 1245
8.1269.2.4 namePI . . . 1245
8.1270 voice_remotePartyNum Struct Reference . . . 1245
8.1270.1 Detailed Description . . . 1246
8.1270.2 Field Documentation . . . 1246

8.1270.2.1 numLen . . . 1246
8.1270.2.2 presentationInd . . . 1246
8.1270.2.3 remPartyNumber . . . 1246
8.1271 voice_roamTimer Struct Reference . . . 1246
8.1271.1 Detailed Description . . . 1246
8.1271.2 Field Documentation . . . 1247
8.1271.2.1 namID . . . 1247
8.1271.2.2 roamTimerValue . . . 1247
8.1272 voice_signalInfo Struct Reference . . . 1247
8.1272.1 Detailed Description . . . 1247
8.1272.2 Field Documentation . . . 1247
8.1272.2.1 alertPitch . . . 1247
8.1272.2.2 signal . . . 1247
8.1272.2.3 signalType . . . 1247
8.1273 voice_SUPSInfo Struct Reference . . . 1248
8.1273.1 Detailed Description . . . 1248
8.1273.2 Field Documentation . . . 1248
8.1273.2.1 isModByCC . . . 1248
8.1273.2.2 svcType . . . 1248
8.1274 voice_USSDNotificationNetworkInfo Struct Reference . . . 1248
8.1274.1 Detailed Description . . . 1248
8.1274.2 Field Documentation . . . 1249
8.1274.2.1 networkInfo . . . 1249
8.1274.2.2 tlvPresent . . . 1249
8.1275 voice_USSInfo Struct Reference . . . 1249
8.1275.1 Detailed Description . . . 1249
8.1275.2 Field Documentation . . . 1249
8.1275.2.1 ussData . . . 1249
8.1275.2.2 ussDCS . . . 1249
8.1275.2.3 ussLen . . . 1249
8.1276 voice_UUSInfo Struct Reference . . . 1249
8.1276.1 Detailed Description . . . 1250
8.1276.2 Field Documentation . . . 1250
8.1276.2.1 UUSData . . . 1250
8.1276.2.2 UUSDatalen . . . 1250
8.1276.2.3 UUSDcs . . . 1250
8.1276.2.4 UUSType . . . 1250
8.1277 voice_VoiceUssdNoWaitAlphaIDTlv Struct Reference . . . 1251
8.1277.1 Detailed Description . . . 1251
8.1277.2 Field Documentation . . . 1251

8.1277.2.1 Alpha_Identifier . . . 1251
8.1277.2.2 TlvPresent . . . 1251
8.1278 voice_VoiceUssdNoWaitErrorTlv Struct Reference . . . 1251
8.1278.1 Detailed Description . . . 1251
8.1278.2 Field Documentation . . . 1251
8.1278.2.1 Error . . . 1252
8.1278.2.2 TlvPresent . . . 1252
8.1279 voice_VoiceUssdNoWaitFailCauseTlv Struct Reference . . . 1252
8.1279.1 Detailed Description . . . 1252
8.1279.2 Field Documentation . . . 1252
8.1279.2.1 Failure_Cause . . . 1252
8.1279.2.2 TlvPresent . . . 1252
8.1280 voice_VoiceUssdNoWaitUSSINFOTlv Struct Reference . . . 1252
8.1280.1 Detailed Description . . . 1252
8.1280.2 Field Documentation . . . 1253
8.1280.2.1 NetworkInfo . . . 1253
8.1280.2.2 TlvPresent . . . 1253
8.1281 wds_channelRate Struct Reference . . . 1253
8.1281.1 Detailed Description . . . 1253
8.1281.2 Field Documentation . . . 1253
8.1281.2.1 CurrChanRxRate . . . 1253
8.1281.2.2 CurrChanTxRate . . . 1253
8.1282 wds_ChannelRateTlv Struct Reference . . . 1253
8.1282.1 Detailed Description . . . 1253
8.1282.2 Field Documentation . . . 1254
8.1282.2.1 ChannelRate . . . 1254
8.1282.2.2 TlvPresent . . . 1254
8.1283 wds_ConnStatusTlv Struct Reference . . . 1254
8.1283.1 Detailed Description . . . 1254
8.1283.2 Field Documentation . . . 1254
8.1283.2.1 MDMConnStatus . . . 1254
8.1283.2.2 TlvPresent . . . 1254
8.1284 wds_currNetworkInfo Struct Reference . . . 1254
8.1284.1 Detailed Description . . . 1254
8.1284.2 Field Documentation . . . 1255
8.1284.2.1 NetworkType . . . 1255
8.1284.2.2 RATMask . . . 1255
8.1284.2.3 SOMask . . . 1255
8.1285 wds_DataBearTechTlv Struct Reference . . . 1255
8.1285.1 Detailed Description . . . 1255

8.1285.2 Field Documentation . . . 1256
8.1285.2.1 DataBearerTech . . . 1256
8.1285.2.2 TlvPresent . . . 1256
8.1286 wds_DataULongLongTlv Struct Reference . . . 1256
8.1286.1 Detailed Description . . . 1256
8.1286.2 Field Documentation . . . 1256
8.1286.2.1 TlvPresent . . . 1256
8.1286.2.2 ullData . . . 1257
8.1287 wds_DataULongTlv Struct Reference . . . 1257
8.1287.1 Detailed Description . . . 1257
8.1287.2 Field Documentation . . . 1257
8.1287.2.1 TlvPresent . . . 1257
8.1287.2.2 ulData . . . 1257
8.1288 wds_DHCPLeaseOptTlv Struct Reference . . . 1257
8.1288.1 Detailed Description . . . 1257
8.1288.2 Field Documentation . . . 1257
8.1288.2.1 numOpt . . . 1257
8.1288.2.2 optList . . . 1257
8.1288.2.3 optListData . . . 1257
8.1288.2.4 TlvPresent . . . 1258
8.1289 wds_DHCPLeaseStateTlv Struct Reference . . . 1258
8.1289.1 Detailed Description . . . 1258
8.1289.2 Field Documentation . . . 1258
8.1289.2.1 leaseState . . . 1258
8.1289.2.2 TlvPresent . . . 1258
8.1290 wds_DHCPOpt Struct Reference . . . 1258
8.1290.1 Detailed Description . . . 1258
8.1290.2 Field Documentation . . . 1259
8.1290.2.1 optCode . . . 1259
8.1290.2.2 optValLen . . . 1259
8.1290.2.3 pOptVal . . . 1259
8.1291 wds_DHCPProfileIdTlv Struct Reference . . . 1259
8.1291.1 Detailed Description . . . 1259
8.1291.2 Field Documentation . . . 1259
8.1291.2.1 profileId . . . 1259
8.1291.2.2 profileType . . . 1259
8.1291.2.3 TlvPresent . . . 1259
8.1292 wds_DHCPv4HWConfig Struct Reference . . . 1259
8.1292.1 Detailed Description . . . 1260
8.1292.2 Field Documentation . . . 1260

8.1292.2.1 chaddr . . . 1260
8.1292.2.2 chaddrLen . . . 1260
8.1292.2.3 hwType . . . 1260
8.1293 wds_DHCPv4Option Struct Reference . . . 1260
8.1293.1 Detailed Description . . . 1260
8.1293.2 Field Documentation . . . 1260
8.1293.2.1 optCode . . . 1260
8.1293.2.2 optVal . . . 1261
8.1293.2.3 optValLen . . . 1261
8.1294 wds_DHCPv4OptionList Struct Reference . . . 1261
8.1294.1 Detailed Description . . . 1261
8.1294.2 Field Documentation . . . 1261
8.1294.2.1 numOpt . . . 1261
8.1294.2.2 pOptList . . . 1261
8.1295 wds_DHCPv4ProfileId Struct Reference . . . 1261
8.1295.1 Detailed Description . . . 1261
8.1295.2 Field Documentation . . . 1261
8.1295.2.1 profileId . . . 1262
8.1295.2.2 profileType . . . 1262
8.1296 wds_Domain Struct Reference . . . 1262
8.1296.1 Detailed Description . . . 1262
8.1296.2 Field Documentation . . . 1262
8.1296.2.1 domainLen . . . 1262
8.1296.2.2 domainName . . . 1262
8.1297 wds_DomainNameList Struct Reference . . . 1262
8.1297.1 Detailed Description . . . 1262
8.1297.2 Field Documentation . . . 1263
8.1297.2.1 domain . . . 1263
8.1297.2.2 numInstances . . . 1263
8.1298 wds_DormStatTlv Struct Reference . . . 1263
8.1298.1 Detailed Description . . . 1263
8.1298.2 Field Documentation . . . 1263
8.1298.2.1 DormancyStat . . . 1263
8.1298.2.2 TlvPresent . . . 1263
8.1299 wds_GPRSQoS Struct Reference . . . 1263
8.1299.1 Detailed Description . . . 1263
8.1299.2 Field Documentation . . . 1264
8.1299.2.1 delayClass . . . 1264
8.1299.2.2 meanThroughputClass . . . 1264
8.1299.2.3 peakThroughputClass . . . 1264

8.1299.2.4precedenceClass . . . 1264
8.1299.2.5reliabilityClass . . . 1264
8.1300wds_IPv4AdTlv Struct Reference . . . 1264
8.1300.1Detailed Description . . . 1264
8.1300.2Field Documentation . . . 1264
8.1300.2.1IPv4Addr . . . 1264
8.1300.2.2TlvPresent . . . 1264
8.1301wds_IPV6AddressInfo Struct Reference . . . 1265
8.1301.1Detailed Description . . . 1265
8.1301.2Field Documentation . . . 1265
8.1301.2.1IPAddressV6 . . . 1265
8.1301.2.2IPV6PrefixLen . . . 1265
8.1302wds_IPV6GWAddressInfo Struct Reference . . . 1265
8.1302.1Detailed Description . . . 1265
8.1302.2Field Documentation . . . 1266
8.1302.2.1gwAddressV6 . . . 1266
8.1302.2.2gwV6PrefixLen . . . 1266
8.1303wds_LastMdmCallEndRsnTlv Struct Reference . . . 1266
8.1303.1Detailed Description . . . 1266
8.1303.2Field Documentation . . . 1266
8.1303.2.1CallEndReason . . . 1266
8.1303.2.2TlvPresent . . . 1266
8.1304wds_PCSCFFQDNAddress Struct Reference . . . 1266
8.1304.1Detailed Description . . . 1266
8.1304.2Field Documentation . . . 1267
8.1304.2.1fqdnAddr . . . 1267
8.1304.2.2fqdnLen . . . 1267
8.1305wds_PCSCFFQDNAddressList Struct Reference . . . 1267
8.1305.1Detailed Description . . . 1267
8.1305.2Field Documentation . . . 1267
8.1305.2.1numInstances . . . 1267
8.1305.2.2pcsfFQDNAddress . . . 1267
8.1306wds_PCSCFIPv4ServerAddressList Struct Reference . . . 1267
8.1306.1Detailed Description . . . 1267
8.1306.2Field Documentation . . . 1268
8.1306.2.1numInstances . . . 1268
8.1306.2.2pscsfIPv4Addr . . . 1268
8.1307wds_profileChange Struct Reference . . . 1268
8.1307.1Detailed Description . . . 1268
8.1307.2Field Documentation . . . 1268

8.1307.2.1 profileInx . . . 1268
8.1307.2.2 profileType . . . 1268
8.1308 wds_ProfileIdentifier Struct Reference . . . 1268
8.1308.1 Detailed Description . . . 1268
8.1308.2 Field Documentation . . . 1269
8.1308.2.1 profileIndex . . . 1269
8.1308.2.2 profileType . . . 1269
8.1309 wds_profileInfo Union Reference . . . 1269
8.1309.1 Detailed Description . . . 1269
8.1309.2 Field Documentation . . . 1269
8.1309.2.1 SlqsProfile3GPP . . . 1269
8.1309.2.2 SlqsProfile3GPP2 . . . 1269
8.1310 wds_RXBytesOKTlv Struct Reference . . . 1269
8.1310.1 Detailed Description . . . 1269
8.1310.2 Field Documentation . . . 1270
8.1310.2.1 RxByteOKCnt . . . 1270
8.1310.2.2 TlvPresent . . . 1270
8.1311 wds_sourceOfChange Struct Reference . . . 1270
8.1311.1 Detailed Description . . . 1270
8.1311.2 Field Documentation . . . 1270
8.1311.2.1 source . . . 1270
8.1312 wds_transferStatInd Struct Reference . . . 1270
8.1312.1 Detailed Description . . . 1270
8.1312.2 Field Documentation . . . 1271
8.1312.2.1 StatsMask . . . 1271
8.1312.2.2 StatsPeriod . . . 1271
8.1313 wds_TrStatInd Struct Reference . . . 1271
8.1313.1 Detailed Description . . . 1271
8.1313.2 Field Documentation . . . 1271
8.1313.2.1 statsMask . . . 1271
8.1313.2.2 statsPeriod . . . 1271
8.1314 wds_TXBytesOKTlv Struct Reference . . . 1271
8.1314.1 Detailed Description . . . 1272
8.1314.2 Field Documentation . . . 1272
8.1314.2.1 TlvPresent . . . 1272
8.1314.2.2 TxByteOKCnt . . . 1272
8.1315 wds_UMTSMinQoS Struct Reference . . . 1272
8.1315.1 Detailed Description . . . 1272
8.1315.2 Field Documentation . . . 1274
8.1315.2.1 deliveryErrSDU . . . 1274

8.1315.2.2 grntDownlinkBitrate . . . 1274
8.1315.2.3 grntUplinkBitrate . . . 1274
8.1315.2.4 maxDownlinkBitrate . . . 1274
8.1315.2.5 maxSDUSize . . . 1274
8.1315.2.6 maxUplinkBitrate . . . 1274
8.1315.2.7 qosDeliveryOrder . . . 1274
8.1315.2.8 resBerRatio . . . 1274
8.1315.2.9 sduErrorRatio . . . 1274
8.1315.2.10 trafficClass . . . 1274
8.1315.2.11 trafficPriority . . . 1274
8.1315.2.12 transferDelay . . . 1274
8.1316 wdsDhcpv4HwConfig Struct Reference . . . 1274
8.1316.1 Detailed Description . . . 1274
8.1316.2 Field Documentation . . . 1275
8.1316.2.1 chaddr . . . 1275
8.1316.2.2 chaddrLen . . . 1275
8.1316.2.3 hwType . . . 1275
8.1317 wdsDhcpv4Option Struct Reference . . . 1275
8.1317.1 Detailed Description . . . 1275
8.1317.2 Field Documentation . . . 1275
8.1317.2.1 optCode . . . 1275
8.1317.2.2 optVal . . . 1275
8.1317.2.3 optValLen . . . 1275
8.1318 wdsDhcpv4OptionList Struct Reference . . . 1275
8.1318.1 Detailed Description . . . 1276
8.1318.2 Field Documentation . . . 1276
8.1318.2.1 numOpt . . . 1276
8.1318.2.2 pOptList . . . 1276
8.1319 wdsDhcpv4ProfileId Struct Reference . . . 1276
8.1319.1 Detailed Description . . . 1276
8.1319.2 Field Documentation . . . 1276
8.1319.2.1 profileId . . . 1276
8.1319.2.2 profileType . . . 1276

9 File Documentation **1277**

9.1 apdoxypages.c File Reference . . . 1277
9.1.1 Detailed Description . . . 1277
9.2 audio.h File Reference . . . 1277
9.2.1 Function Documentation . . . 1278
9.2.1.1 pack_audio_SLQSGetAudioPathConfig . . . 1278

9.2.1.2 pack_audio_SLQSGetAudioProfile . . . 1278
9.2.1.3 pack_audio_SLQSGetAudioVolTLBConfig . . . 1279
9.2.1.4 pack_audio_SLQSSetAudioPathConfig . . . 1279
9.2.1.5 pack_audio_SLQSSetAudioProfile . . . 1280
9.2.1.6 pack_audio_SLQSSetAudioVolTLBConfig . . . 1280
9.2.1.7 unpack_audio_SLQSGetAudioPathConfig . . . 1281
9.2.1.8 unpack_audio_SLQSGetAudioProfile . . . 1281
9.2.1.9 unpack_audio_SLQSGetAudioVolTLBConfig . . . 1281
9.2.1.10 unpack_audio_SLQSSetAudioPathConfig . . . 1282
9.2.1.11 unpack_audio_SLQSSetAudioProfile . . . 1282
9.2.1.12 unpack_audio_SLQSSetAudioVolTLBConfig . . . 1282
9.3 BasicConnectDeviceService.h File Reference . . . 1283
9.3.1 Typedef Documentation . . . 1283
9.3.1.1 MbimEventEntry . . . 1283
9.3.2 Function Documentation . . . 1283
9.3.2.1 BasicConnectDeviceService_DeviceServiceSubscribeParse . . . 1283
9.3.2.2 BasicConnectDeviceService_DeviceServiceSubscribeSetSend . . . 1284
9.3.2.3 BasicConnectDeviceService_Uuid . . . 1284
9.4 cat.h File Reference . . . 1284
9.4.1 Macro Definition Documentation . . . 1285
9.4.1.1 CAN_COMMON_EVENT_TLV_NUMBER . . . 1285
9.4.1.2 CAT_EVENT_DATA_MAX_LENGTH . . . 1285
9.4.2 Typedef Documentation . . . 1285
9.4.2.1 unpack_cat_CATSendEnvelopeCommand_t . . . 1285
9.4.2.2 unpack_cat_CATSendTerminalResponse_t . . . 1285
9.4.3 Function Documentation . . . 1285
9.4.3.1 pack_cat_CATSendEnvelopeCommand . . . 1285
9.4.3.2 pack_cat_CATSendTerminalResponse . . . 1286
9.4.3.3 pack_cat_SetCATEventCallback . . . 1286
9.4.3.4 unpack_cat_CATSendEnvelopeCommand . . . 1287
9.4.3.5 unpack_cat_CATSendTerminalResponse . . . 1287
9.4.3.6 unpack_cat_SetCATEventCallback . . . 1287
9.4.3.7 unpack_cat_SetCatEventCallback_ind . . . 1288
9.5 common.h File Reference . . . 1288
9.5.1 Macro Definition Documentation . . . 1290
9.5.1.1 DEAULT_LOC_TIMEOUT_IN_SEC . . . 1290
9.5.1.2 MINREQBKLEN . . . 1290
9.5.1.3 MSGID_AND_LEN . . . 1290
9.5.1.4 MSGID_DONT_CARE . . . 1290
9.5.1.5 SDK_VALIDATE_INPUT_PACK_PARAM . . . 1290

9.5.1.6 SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID . . . 1290
9.5.1.7 SDK_VALIDATE_INPUT_UNPACK_PARAM . . . 1290
9.5.1.8 SDU_HDR_LEN . . . 1291
9.5.1.9 SWI_INIT_UNPACK_RESULT_VALUE . . . 1291
9.5.1.10 UNUSEDPARAM . . . 1291
9.5.2 Enumeration Type Documentation . . . 1291
9.5.2.1 eLOG_LEVEL . . . 1291
9.5.2.2 eQMI_SVC . . . 1291
9.5.2.3 eTimeout . . . 1292
9.5.2.4 msgtype . . . 1292
9.5.3 Function Documentation . . . 1292
9.5.3.1 fill_pack_ctx . . . 1292
9.5.3.2 fill_sdu_hdr . . . 1292
9.5.3.3 get_version . . . 1292
9.5.3.4 helper_get_error_code . . . 1292
9.5.3.5 helper_get_error_reason . . . 1292
9.5.3.6 helper_get_req_str . . . 1292
9.5.3.7 helper_get_resp_ctx . . . 1293
9.5.3.8 helper_get_xid . . . 1293
9.5.3.9 helper_isBootLoader_DebugEnabled . . . 1293
9.5.3.10 helper_set_log_func . . . 1293
9.5.3.11 helper_set_log_lvl . . . 1293
9.5.3.12 liteqmi_GetVersion . . . 1293
9.5.3.13 liteqmi_helper_decode7bitAsciiEncString . . . 1294
9.5.3.14 liteqmi_log . . . 1294
9.5.3.15 unpack_result_code_only . . . 1294
9.5.4 Variable Documentation . . . 1294
9.5.4.1 glog . . . 1294
9.5.4.2 gloglvl . . . 1294
9.6 ctl.h File Reference . . . 1294
9.6.1 Function Documentation . . . 1294
9.6.1.1 helper_get_ctl_resp_ctx . . . 1294
9.6.1.2 pack_ctl_GetClientId . . . 1294
9.6.1.3 pack_ctl_ReleaseClientId . . . 1295
9.6.1.4 unpack_ctl_GetClientId . . . 1295
9.6.1.5 unpack_ctl_ReleaseClientId . . . 1296
9.7 CtlService.h File Reference . . . 1296
9.7.1 Typedef Documentation . . . 1296
9.7.1.1 CtlService . . . 1296
9.7.2 Function Documentation . . . 1296

9.7.2.1 CtlService_GetClientId . . . 1297
9.7.2.2 CtlService_Initialize . . . 1297
9.7.2.3 CtlService_InitializeRegularService . . . 1297
9.7.2.4 CtlService_ReleaseClientId . . . 1297
9.7.2.5 CtlService_ShutDown . . . 1298
9.7.2.6 CtlService_ShutDownRegularService . . . 1298
9.8 DirectLayer.h File Reference . . . 1298
9.8.1 Function Documentation . . . 1298
9.8.1.1 DirectLayer_Create . . . 1298
9.8.1.2 DirectLayer_GetClientID . . . 1298
9.8.1.3 DirectLayer_GetDeviceFD . . . 1298
9.8.1.4 DirectLayer_SendQmuxPacket . . . 1298
9.8.1.5 DirectLayer_ShutDown . . . 1298
9.8.1.6 DriectLay_GetDummyServiceClientID . . . 1298
9.9 dms.h File Reference . . . 1299
9.9.1 Macro Definition Documentation . . . 1306
9.9.1.1 ACT_CODE_MAX_SIZE . . . 1306
9.9.1.2 CK_MAX_SIZE . . . 1306
9.9.1.3 DMS_IMGDETAILS_LEN . . . 1306
9.9.1.4 DMS_MAX_CUST_ID_LEN . . . 1307
9.9.1.5 DMS_MAX_CUST_VALUE_LEN . . . 1307
9.9.1.6 DMS_MAX_FWUPDATE_LOG_STR_SZ . . . 1307
9.9.1.7 DMS_MAX_FWUPDATE_REF_STR_SZ . . . 1307
9.9.1.8 DMS_MAX_RADIO_IFCS_SIZE . . . 1307
9.9.1.9 DMS_MAX_SUBS_CFG_LIST_SIZE . . . 1307
9.9.1.10 DMS_MAX_SUBS_LIST_SIZE . . . 1307
9.9.1.11 DMS_MAX_SUPPORTED_LTE_BANDS . . . 1307
9.9.1.12 DMS_PM_FACTORY . . . 1307
9.9.1.13 DMS_PM_LOW . . . 1307
9.9.1.14 DMS_PM_OFFLINE . . . 1307
9.9.1.15 DMS_PM_ONLINE . . . 1307
9.9.1.16 DMS_PM_PERSISTENT_LOW . . . 1307
9.9.1.17 DMS_PM_RESET . . . 1307
9.9.1.18 DMS_PM_SHUT_DOWN . . . 1307
9.9.1.19 DMS_SET_REG_IND_DISABLE . . . 1307
9.9.1.20 DMS_SET_REG_IND_ENABLE . . . 1307
9.9.1.21 DMS_SET_REG_IND_NO_CHANGE . . . 1307
9.9.1.22 DMS_SET_REPORT_DISABLE . . . 1307
9.9.1.23 DMS_SET_REPORT_ENABLE . . . 1307
9.9.1.24 DMS_SLQSFWINFO_APPVERSION_SZ . . . 1307

9.9.1.25 DMS_SLQSFWINFO_BOOTVERSION_SZ . . . 1307
9.9.1.26 DMS_SLQSFWINFO_CARRIER_SZ . . . 1307
9.9.1.27 DMS_SLQSFWINFO_CUR_CARR_NAME . . . 1307
9.9.1.28 DMS_SLQSFWINFO_CUR_CARR_REV . . . 1307
9.9.1.29 DMS_SLQSFWINFO_MODELID_SZ . . . 1307
9.9.1.30 DMS_SLQSFWINFO_PACKAGEID_SZ . . . 1307
9.9.1.31 DMS_SLQSFWINFO_PRIVERSION_SZ . . . 1307
9.9.1.32 DMS_SLQSFWINFO_SKU_SZ . . . 1308
9.9.1.33 DMS_SWI_SET_IND_DISABLE . . . 1308
9.9.1.34 DMS_SWI_SET_IND_ENABLE . . . 1308
9.9.1.35 DMS_UINT8_MAX_STRING_SZ . . . 1308
9.9.1.36 DMS_VALID_FSN_LEN . . . 1308
9.9.1.37 ERI_DATA_MAX_SIZE . . . 1308
9.9.1.38 MAX_BUILD_ID_LEN . . . 1308
9.9.1.39 MEID_MAX_SIZE . . . 1308
9.9.1.40 SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH . . . 1308
9.9.1.41 SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH . . . 1308
9.9.1.42 SPC_SIZE . . . 1308
9.9.1.43 UNIQUE_ID_LEN . . . 1308
9.9.2 Function Documentation . . . 1308
9.9.2.1 pack_dms_ActivateAutomatic . . . 1308
9.9.2.2 pack_dms_GetActivationState . . . 1308
9.9.2.3 pack_dms_GetBandCapability . . . 1309
9.9.2.4 pack_dms_GetCrashAction . . . 1309
9.9.2.5 pack_dms_GetCustFeature . . . 1309
9.9.2.6 pack_dms_GetCustFeaturesV2 . . . 1310
9.9.2.7 pack_dms_GetDeviceCap . . . 1310
9.9.2.8 pack_dms_GetDeviceCapabilities . . . 1311
9.9.2.9 pack_dms_GetDeviceCapabilitiesV2 . . . 1311
9.9.2.10 pack_dms_GetDeviceHardwareRev . . . 1311
9.9.2.11 pack_dms_GetDeviceMfr . . . 1312
9.9.2.12 pack_dms_GetDeviceSerialNumbers . . . 1312
9.9.2.13 pack_dms_GetFirmwareInfo . . . 1312
9.9.2.14 pack_dms_GetFirmwareRevision . . . 1313
9.9.2.15 pack_dms_GetFirmwareRevisions . . . 1313
9.9.2.16 pack_dms_GetFSN . . . 1313
9.9.2.17 pack_dms_GetHardwareRevision . . . 1314
9.9.2.18 pack_dms_GetIMSI . . . 1314
9.9.2.19 pack_dms_GetManufacturer . . . 1314
9.9.2.20 pack_dms_GetModelID . . . 1315

9.9.2.21 pack_dms_GetNetworkTime . . . 1315
9.9.2.22 pack_dms_GetNetworkTimeV2 . . . 1315
9.9.2.23 pack_dms_GetOfflineReason . . . 1316
9.9.2.24 pack_dms_GetPower . . . 1316
9.9.2.25 pack_dms_GetPRLVersion . . . 1316
9.9.2.26 pack_dms_GetSerialNumbers . . . 1317
9.9.2.27 pack_dms_GetUSBComp . . . 1317
9.9.2.28 pack_dms_GetVoiceNumber . . . 1318
9.9.2.29 pack_dms_ResetToFactoryDefaults . . . 1318
9.9.2.30 pack_dms_SetActivationStatusCallback . . . 1318
9.9.2.31 pack_dms_SetCrashAction . . . 1319
9.9.2.32 pack_dms_SetCustFeature . . . 1319
9.9.2.33 pack_dms_SetCustFeaturesV2 . . . 1319
9.9.2.34 pack_dms_SetEventReport . . . 1320
9.9.2.35 pack_dms_SetFirmwarePreference . . . 1320
9.9.2.36 pack_dms_SetIndicationRegister . . . 1320
9.9.2.37 pack_dms_SetPower . . . 1321
9.9.2.38 pack_dms_SetUSBComp . . . 1321
9.9.2.39 pack_dms_SLQSDmsSwiGetPCInfo . . . 1322
9.9.2.40 pack_dms_SLQSDmsSwiGetResetInfo . . . 1322
9.9.2.41 pack_dms_SLQSDmsSwiGetUimSelection . . . 1322
9.9.2.42 pack_dms_SLQSDmsSwiIndicationRegister . . . 1323
9.9.2.43 pack_dms_SLQSGetBandCapability . . . 1323
9.9.2.44 pack_dms_SLQSGetERIFile . . . 1323
9.9.2.45 pack_dms_SLQSGetPowerSaveModeConfig . . . 1324
9.9.2.46 pack_dms_SLQSSetPowerSaveModeConfig . . . 1324
9.9.2.47 pack_dms_SLQSSwiClearDyingGaspStatistics . . . 1325
9.9.2.48 pack_dms_SLQSSwiGetCrashInfo . . . 1325
9.9.2.49 pack_dms_SLQSSwiGetDyingGaspCfg . . . 1325
9.9.2.50 pack_dms_SLQSSwiGetDyingGaspStatistics . . . 1326
9.9.2.51 pack_dms_SLQSSwiGetFirmwareCurr . . . 1326
9.9.2.52 pack_dms_SLQSSwiGetFwUpdateStatus . . . 1326
9.9.2.53 pack_dms_SLQSSwiGetHostDevInfo . . . 1327
9.9.2.54 pack_dms_SLQSSwiGetOSInfo . . . 1327
9.9.2.55 pack_dms_SLQSSwiGetSerialNoExt . . . 1327
9.9.2.56 pack_dms_SLQSSwiSetDyingGaspCfg . . . 1328
9.9.2.57 pack_dms_SLQSSwiSetHostDevInfo . . . 1328
9.9.2.58 pack_dms_SLQSSwiSetOSInfo . . . 1328
9.9.2.59 pack_dms_SLQSUIMGetState . . . 1329
9.9.2.60 pack_dms_SwiSetEventReport . . . 1329

9.9.2.61 pack_dms_SwiUimSelect . . . 1329
9.9.2.62 pack_dms_UIMChangePIN . . . 1330
9.9.2.63 pack_dms_UIMGetControlKeyStatus . . . 1330
9.9.2.64 pack_dms_UIMGetICCID . . . 1331
9.9.2.65 pack_dms_UIMGetPINStatus . . . 1331
9.9.2.66 pack_dms_UIMSetControlKeyProtection . . . 1331
9.9.2.67 pack_dms_UIMSetPINProtection . . . 1332
9.9.2.68 pack_dms_UIMUnblockControlKey . . . 1332
9.9.2.69 pack_dms_UIMUnblockPIN . . . 1332
9.9.2.70 pack_dms_UIMVerifyPIN . . . 1333
9.9.2.71 pack_dms_ValidateSPC . . . 1333
9.9.2.72 unpack_dms_ActivateAutomatic . . . 1334
9.9.2.73 unpack_dms_GetActivationState . . . 1334
9.9.2.74 unpack_dms_GetBandCapability . . . 1334
9.9.2.75 unpack_dms_GetCrashAction . . . 1335
9.9.2.76 unpack_dms_GetCustFeature . . . 1335
9.9.2.77 unpack_dms_GetCustFeaturesV2 . . . 1335
9.9.2.78 unpack_dms_GetDeviceCap . . . 1336
9.9.2.79 unpack_dms_GetDeviceCapabilities . . . 1336
9.9.2.80 unpack_dms_GetDeviceCapabilitiesV2 . . . 1336
9.9.2.81 unpack_dms_GetDeviceHardwareRev . . . 1337
9.9.2.82 unpack_dms_GetDeviceMfr . . . 1337
9.9.2.83 unpack_dms_GetDeviceSerialNumbers . . . 1337
9.9.2.84 unpack_dms_GetFirmwareInfo . . . 1338
9.9.2.85 unpack_dms_GetFirmwareRevision . . . 1338
9.9.2.86 unpack_dms_GetFirmwareRevisions . . . 1339
9.9.2.87 unpack_dms_GetFSN . . . 1339
9.9.2.88 unpack_dms_GetHardwareRevision . . . 1339
9.9.2.89 unpack_dms_GetIMSI . . . 1340
9.9.2.90 unpack_dms_GetManufacturer . . . 1340
9.9.2.91 unpack_dms_GetModelID . . . 1340
9.9.2.92 unpack_dms_GetNetworkTime . . . 1341
9.9.2.93 unpack_dms_GetNetworkTimeV2 . . . 1341
9.9.2.94 unpack_dms_GetOfflineReason . . . 1341
9.9.2.95 unpack_dms_GetPower . . . 1342
9.9.2.96 unpack_dms_GetPRLVersion . . . 1342
9.9.2.97 unpack_dms_GetSerialNumbers . . . 1342
9.9.2.98 unpack_dms_GetUSBComp . . . 1343
9.9.2.99 unpack_dms_GetVoiceNumber . . . 1343
9.9.2.100 unpack_dms_PSMCfgChange_ind . . . 1343

9.9.2.101 unpack_dms_ResetToFactoryDefaults . . . 1344
9.9.2.102 unpack_dms_SetActivationStatusCallback . . . 1344
9.9.2.103 unpack_dms_SetCrashAction . . . 1344
9.9.2.104 unpack_dms_SetCustFeature . . . 1345
9.9.2.105 unpack_dms_SetCustFeaturesV2 . . . 1345
9.9.2.106 unpack_dms_SetEventReport . . . 1345
9.9.2.107 unpack_dms_SetEventReport_ind . . . 1346
9.9.2.108 unpack_dms_SetFirmwarePreference . . . 1346
9.9.2.109 unpack_dms_SetIndicationRegister . . . 1347
9.9.2.110 unpack_dms_SetPower . . . 1347
9.9.2.111 unpack_dms_SetUSBComp . . . 1347
9.9.2.112 unpack_dms_SLQSDmsSwiGetPCInfo . . . 1348
9.9.2.113 unpack_dms_SLQSDmsSwiGetResetInfo . . . 1348
9.9.2.114 unpack_dms_SLQSDmsSwiGetResetInfo_Ind . . . 1348
9.9.2.115 unpack_dms_SLQSDmsSwiGetUimSelection . . . 1349
9.9.2.116 unpack_dms_SLQSDmsSwiIndicationRegister . . . 1349
9.9.2.117 unpack_dms_SLQSGetBandCapability . . . 1350
9.9.2.118 unpack_dms_SLQSGetBandCapabilityExt . . . 1350
9.9.2.119 unpack_dms_SLQSGetERIFile . . . 1350
9.9.2.120 unpack_dms_SLQSGetPowerSaveModeConfig . . . 1351
9.9.2.121 unpack_dms_SLQSSetPowerSaveModeConfig . . . 1351
9.9.2.122 unpack_dms_SLQSSwiClearDyingGaspStatistics . . . 1351
9.9.2.123 unpack_dms_SLQSSwiGetCrashInfo . . . 1352
9.9.2.124 unpack_dms_SLQSSwiGetDyingGaspCfg . . . 1352
9.9.2.125 unpack_dms_SLQSSwiGetDyingGaspStatistics . . . 1352
9.9.2.126 unpack_dms_SLQSSwiGetFirmwareCurr . . . 1353
9.9.2.127 unpack_dms_SLQSSwiGetFwUpdateStatus . . . 1353
9.9.2.128 unpack_dms_SLQSSwiGetHostDevInfo . . . 1353
9.9.2.129 unpack_dms_SLQSSwiGetOSInfo . . . 1354
9.9.2.130 unpack_dms_SLQSSwiGetSerialNoExt . . . 1354
9.9.2.131 unpack_dms_SLQSSwiSetDyingGaspCfg . . . 1354
9.9.2.132 unpack_dms_SLQSSwiSetHostDevInfo . . . 1355
9.9.2.133 unpack_dms_SLQSSwiSetOSInfo . . . 1355
9.9.2.134 unpack_dms_SLQSUIMGetState . . . 1355
9.9.2.135 unpack_dms_SwiEventReportCallBack_ind . . . 1356
9.9.2.136 unpack_dms_SwiSetEventReport . . . 1356
9.9.2.137 unpack_dms_SwiUimSelect . . . 1356
9.9.2.138 unpack_dms_UIMChangePIN . . . 1357
9.9.2.139 unpack_dms_UIMGetControlKeyStatus . . . 1357
9.9.2.140 unpack_dms_UIMGetICCID . . . 1357

9.9.2.141 unpack_dms_UIMGetPINStatus . . . 1358
9.9.2.142 unpack_dms_UIMSetControlKeyProtection . . . 1358
9.9.2.143 unpack_dms_UIMSetPINProtection . . . 1359
9.9.2.144 unpack_dms_UIMUnblockControlKey . . . 1359
9.9.2.145 unpack_dms_UIMUnblockPIN . . . 1359
9.9.2.146 unpack_dms_UIMVerifyPIN . . . 1360
9.9.2.147 unpack_dms_ValidateSPC . . . 1360
9.10 dsd.h File Reference . . . 1360
9.10.1 Detailed Description . . . 1361
9.10.2 DSDSOMask SO Mask . . . 1361
9.10.3 DSDRatMask RAT Mask . . . 1362
9.10.4 Macro Definition Documentation . . . 1363
9.10.4.1 MAX_APN_STRING . . . 1363
9.10.4.2 MAX_AVAIL_SYSTEM_INFO . . . 1363
9.10.5 Typedef Documentation . . . 1363
9.10.5.1 unpack_dsd_RegisterUIInfoChange_t . . . 1363
9.10.5.2 unpack_dsd_SystemStatusChnage_t . . . 1363
9.10.6 Function Documentation . . . 1363
9.10.6.1 pack_dsd_RegisterUIInfoChange . . . 1363
9.10.6.2 pack_dsd_SLQSGetSystemStatus . . . 1363
9.10.6.3 pack_dsd_SLQSGetUIInfo . . . 1364
9.10.6.4 pack_dsd_SystemStatusChange . . . 1364
9.10.6.5 unpack_dsd_RegisterUIInfoChange . . . 1364
9.10.6.6 unpack_dsd_SLQSGetSystemStatus . . . 1365
9.10.6.7 unpack_dsd_SLQSGetUIInfo . . . 1365
9.10.6.8 unpack_dsd_SLQSSystemStatusChangeInd . . . 1366
9.10.6.9 unpack_dsd_SLQSUIInfoChangeInd . . . 1366
9.10.6.10 unpack_dsd_SystemStatusChange . . . 1366
9.11 fms.h File Reference . . . 1367
9.11.1 Macro Definition Documentation . . . 1368
9.11.1.1 FMS_FW_PRI_BUILD_MATCH_LEN . . . 1368
9.11.1.2 FMS_GOBI_LISTENTRIES_MAX . . . 1368
9.11.1.3 FMS_GOBI_MBN_BUILD_ID_STR_LEN . . . 1368
9.11.1.4 FMS_GOBI_MBN_IMG_ID_SIZE . . . 1368
9.11.1.5 FMS_GOBI_MBN_IMG_ID_STR_LEN . . . 1368
9.11.1.6 FMS_IMAGE_ID_BUILD_ID_LEN . . . 1368
9.11.1.7 FMS_IMAGE_ID_IMG_ID_LEN . . . 1368
9.11.1.8 FMS_IMAGE_ID_MAX_ENTRIES . . . 1368
9.11.1.9 FMS_IMAGE_ID_PRI_IMGTYPE . . . 1368
9.11.1.10 FMS_MAX_IMAGE_ID_ELEMENT . . . 1368

9.11.1.11 FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE . . . 1368
9.11.1.12 LITE_TOTAL_IMAGE_ID_STRING_SIZE . . . 1368
9.11.2 Function Documentation . . . 1368
9.11.2.1 GetValidFwPriCombinations . . . 1368
9.11.2.2 pack_fms_GetImagesPreference . . . 1369
9.11.2.3 pack_fms_GetStoredImages . . . 1369
9.11.2.4 pack_fms_SetImagesPreference . . . 1369
9.11.2.5 unpack_fms_GetImagesPreference . . . 1369
9.11.2.6 unpack_fms_GetStoredImages . . . 1370
9.11.2.7 unpack_fms_SetImagesPreference . . . 1370
9.12 ims.h File Reference . . . 1370
9.12.1 Macro Definition Documentation . . . 1372
9.12.1.1 MAX_NAME_LEN . . . 1372
9.12.2 Typedef Documentation . . . 1372
9.12.2.1 unpack_ims_SLQSImsConfigIndicationRegister_t . . . 1372
9.12.3 Function Documentation . . . 1372
9.12.3.1 pack_ims_SLQSGetIMSSMSConfig . . . 1372
9.12.3.2 pack_ims_SLQSGetIMSUserConfig . . . 1373
9.12.3.3 pack_ims_SLQSGetIMSVoIPConfig . . . 1373
9.12.3.4 pack_ims_SLQSGetRegMgrConfig . . . 1374
9.12.3.5 pack_ims_SLQSGetSIPConfig . . . 1374
9.12.3.6 pack_ims_SLQSImsConfigIndicationRegister . . . 1374
9.12.3.7 pack_ims_SLQSSetIMSSMSConfig . . . 1375
9.12.3.8 pack_ims_SLQSSetIMSUserConfig . . . 1375
9.12.3.9 pack_ims_SLQSSetIMSVoIPConfig . . . 1376
9.12.3.10 pack_ims_SLQSSetRegMgrConfig . . . 1376
9.12.3.11 pack_ims_SLQSSetSIPConfig . . . 1377
9.12.3.12 unpack_ims_SLQSGetIMSSMSConfig . . . 1377
9.12.3.13 unpack_ims_SLQSGetIMSUserConfig . . . 1377
9.12.3.14 unpack_ims_SLQSGetIMSVoIPConfig . . . 1378
9.12.3.15 unpack_ims_SLQSGetRegMgrConfig . . . 1378
9.12.3.16 unpack_ims_SLQSGetSIPConfig . . . 1379
9.12.3.17 unpack_ims_SLQSImsConfigIndicationRegister . . . 1379
9.12.3.18 unpack_ims_SLQSRegMgrCfgCallBack_ind . . . 1379
9.12.3.19 unpack_ims_SLQSSetIMSSMSConfig . . . 1380
9.12.3.20 unpack_ims_SLQSSetIMSUserConfig . . . 1380
9.12.3.21 unpack_ims_SLQSSetIMSVoIPConfig . . . 1380
9.12.3.22 unpack_ims_SLQSSetRegMgrConfig . . . 1381
9.12.3.23 unpack_ims_SLQSSetSIPConfig . . . 1381
9.12.3.24 unpack_ims_SLQSSIPCfgCallBack_ind . . . 1381

9.12.3.25 unpack_ims_SLQSSMSCfgCallBack_ind . . . 1382
9.12.3.26 unpack_ims_SLQSUserCfgCallBack_ind . . . 1382
9.12.3.27 unpack_ims_SLQSVoIPCfgCallBack_ind . . . 1382
9.13 imsa.h File Reference . . . 1383
9.13.1 Macro Definition Documentation . . . 1384
9.13.1.1 MAX_ERROR_CODE_LEN . . . 1384
9.13.2 Typedef Documentation . . . 1384
9.13.2.1 unpack_imsa_SLQSRegisterIMSAIndication_t . . . 1384
9.13.3 Function Documentation . . . 1384
9.13.3.1 pack_imsa_SLQSGetIMSARegStatus . . . 1384
9.13.3.2 pack_imsa_SLQSGetIMSAServiceStatus . . . 1384
9.13.3.3 pack_imsa_SLQSRegisterIMSAIndication . . . 1385
9.13.3.4 unpack_imsa_SLQSGetIMSARegStatus . . . 1385
9.13.3.5 unpack_imsa_SLQSGetIMSAServiceStatus . . . 1386
9.13.3.6 unpack_imsa_SLQSImsaPdpStatusCallBack_ind . . . 1386
9.13.3.7 unpack_imsa_SLQSImsaRatStatusCallBack_ind . . . 1387
9.13.3.8 unpack_imsa_SLQSImsaRegStatusCallBack_ind . . . 1387
9.13.3.9 unpack_imsa_SLQSImsaSvcStatusCallBack_ind . . . 1387
9.13.3.10 unpack_imsa_SLQSRegisterIMSAIndication . . . 1388
9.14 lite-fw.h File Reference . . . 1388
9.14.1 Detailed Description . . . 1391
9.14.2 Macro Definition Documentation . . . 1391
9.14.2.1 FIRMWARE_BCHDATESIZE . . . 1391
9.14.2.2 FIRMWARE_BCHVERSTRSIZE . . . 1391
9.14.2.3 FIRMWARE_IMAGE_SIZE_MAX . . . 1391
9.14.2.4 FIRMWARE_INFO_STRING_SIZE . . . 1391
9.14.2.5 IMG_MASK_CLEAR . . . 1391
9.14.2.6 IMG_MASK_GENERIC . . . 1391
9.14.2.7 IMG_MASK_MDM . . . 1391
9.14.2.8 IMG_MASK_PRI . . . 1391
9.14.2.9 libSDP_BuildImagesPreferenceRequest . . . 1391
9.14.2.10 libSDP_CalculateImageMask . . . 1391
9.14.2.11 LIBSDP_CARRIER_PACKAGE_SKU . . . 1391
9.14.2.12 libSDP_CheckValidFirmwareInfo . . . 1391
9.14.2.13 libSDP_DownloadFW . . . 1391
9.14.2.14 libSDP_ExtractFirmwareParametersByPath . . . 1391
9.14.2.15 libSDP_getFileType . . . 1391
9.14.2.16 libSDP_GetModelFamily . . . 1391
9.14.2.17 libSDP_GetVersion . . . 1391
9.14.2.18 libsdp_set_log_func . . . 1391

9.14.2.19 libsdp_SetReadBlockSize . . . 1392
9.14.2.20 LIBSDP_SKU_STRING_LENGTH . . . 1392
9.14.2.21 LITEFW_CARRIER_PACKAGE_SKU . . . 1392
9.14.2.22 LITEFW_SKU_STRING_LENGTH . . . 1392
9.14.2.23 MAX_IMAGE_PRODUCT_LENGTH . . . 1392
9.14.2.24 NULL_TERMINATOR_CHAR_SIZE . . . 1392
9.14.3 Typedef Documentation . . . 1392
9.14.3.1 libSDP_FirmwareInfo . . . 1392
9.14.3.2 libsdplogger . . . 1392
9.14.3.3 litefw_FirmwareFileInfo . . . 1392
9.14.3.4 litefw_FirmwareInfo . . . 1393
9.14.3.5 litefw_FirmwarePartNo . . . 1393
9.14.3.6 litefwlogger . . . 1393
9.14.4 Enumeration Type Documentation . . . 1394
9.14.4.1 litefw_fileimgtype . . . 1394
9.14.4.2 litefw_Fw_Type . . . 1394
9.14.4.3 litefw_fwdwl_error_codes . . . 1394
9.14.4.4 litefw_imagetype . . . 1395
9.14.4.5 litefw_Models . . . 1395
9.14.4.6 litefw_QDL_FLOW_CONTROLs . . . 1395
9.14.4.7 litefw_QDL_MODEs . . . 1396
9.14.5 Function Documentation . . . 1396
9.14.5.1 GetFlashSectorSize . . . 1396
9.14.5.2 litefw_BuildImagesPreferenceRequest . . . 1396
9.14.5.3 litefw_CalculateImageMask . . . 1396
9.14.5.4 litefw_CheckValidFirmwareInfo . . . 1397
9.14.5.5 litefw_DownloadFW . . . 1397
9.14.5.6 litefw_ExtractFirmwareParametersByPath . . . 1398
9.14.5.7 litefw_ExtractFirmwarePartNoByPath . . . 1398
9.14.5.8 litefw_getFileType . . . 1398
9.14.5.9 litefw_GetModelFamily . . . 1399
9.14.5.10 litefw_GetQTLDownloadMode . . . 1399
9.14.5.11 litefw_GetQTLHWFlowControl . . . 1399
9.14.5.12 litefw_GetRetryFirehoseSwitch . . . 1399
9.14.5.13 litefw_GetVersion . . . 1399
9.14.5.14 litefw_logsenable . . . 1400
9.14.5.15 litefw_set_log_func . . . 1400
9.14.5.16 litefw_SetQTLDownloadMode . . . 1400
9.14.5.17 litefw_SetQTLHWFlowControl . . . 1400
9.14.5.18 litefw_SetReadBlockSize . . . 1401

9.14.5.19 litefw_SetRetryFirehoseSwitch . . . 1401
9.14.5.20 litefw_SLQSGetFirmwareFileInfo . . . 1401
9.14.5.21 litefw_switch_9x07_to_downloadmode . . . 1401
9.14.5.22 litefw_switch_to_BootHoldMode . . . 1402
9.14.5.23 SetFlashSectorSize . . . 1402
9.15 loc.h File Reference . . . 1402
9.15.1 Macro Definition Documentation . . . 1406
9.15.1.1 LOC_UINT8_MAX_STRING_SZ . . . 1406
9.15.1.2 LOCEVENTMASKBATCHFULLNOTIFICATION . . . 1406
9.15.1.3 LOCEVENTMASKENGINESTATE . . . 1406
9.15.1.4 LOCEVENTMASKFIXSESSIONSTATE . . . 1406
9.15.1.5 LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION . . . 1406
9.15.1.6 LOCEVENTMASKGEOFENCEBREACHNOTIFICATION . . . 1407
9.15.1.7 LOCEVENTMASKGEOFENCEGENALERT . . . 1407
9.15.1.8 LOCEVENTMASKGNSSMEASUREMENTREPORT . . . 1407
9.15.1.9 LOCEVENTMASKGNSSSVINFO . . . 1407
9.15.1.10 LOCEVENTMASKINJECTPOSITIONREQ . . . 1407
9.15.1.11 LOCEVENTMASKINJECTPREDICTEDORBITSREQ . . . 1407
9.15.1.12 LOCEVENTMASKINJECTTIMEREQ . . . 1407
9.15.1.13 LOCEVENTMASKINJECTWIFIAPDATAREQ . . . 1407
9.15.1.14 LOCEVENTMASKINVALIDVALUE . . . 1407
9.15.1.15 LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT . . . 1407
9.15.1.16 LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ . . . 1407
9.15.1.17 LOCEVENTMASKMOTIONDATACONTROL . . . 1408
9.15.1.18 LOCEVENTMASKNIGEOFENCENOTIFICATION . . . 1408
9.15.1.19 LOCEVENTMASKNINOTIFYVERIFYREQ . . . 1408
9.15.1.20 LOCEVENTMASKNMEA . . . 1408
9.15.1.21 LOCEVENTMASKPEDOMETERCONTROL . . . 1408
9.15.1.22 LOCEVENTMASKPOSITIONREPORT . . . 1408
9.15.1.23 LOCEVENTMASKSENSORSTREAMINGREADYSTATUS . . . 1408
9.15.1.24 LOCEVENTMASKSETSPISTREAMINGREPORT . . . 1408
9.15.1.25 LOCEVENTMASKTIMESYNCREQ . . . 1408
9.15.1.26 LOCEVENTMASKVEHICLEDATAREADYSTATUS . . . 1408
9.15.1.27 LOCEVENTMASKWIFIREQ . . . 1408
9.15.1.28 MAX_LOC_NMEA_STR_LEN . . . 1409
9.15.1.29 MAX_SENSOR_DATA_LEN . . . 1409
9.15.1.30 MAX_TEMP_DATA_LEN . . . 1409
9.15.2 Typedef Documentation . . . 1409
9.15.2.1 unpack_loc_DeleteAssistData_t . . . 1409
9.15.2.2 unpack_loc_GetFixCriteria_t . . . 1409

9.15.2.3 unpack_loc_SLQSLOCGetServer_t . . . 1409
9.15.2.4 unpack_loc_SLQSLOCInjectPosition_t . . . 1409
9.15.2.5 unpack_loc_SLQSLOCInjectSensorData_t . . . 1409
9.15.2.6 unpack_loc_SLQSLOCInjectUTCTime_t . . . 1409
9.15.2.7 unpack_loc_SLQSLOCSetCradleMountConfig_t . . . 1409
9.15.3 Enumeration Type Documentation . . . 1409
9.15.3.1 anonymous enum . . . 1409
9.15.4 Function Documentation . . . 1409
9.15.4.1 pack_loc_DeleteAssistData . . . 1409
9.15.4.2 pack_loc_EventRegister . . . 1410
9.15.4.3 pack_loc_GetFixCriteria . . . 1410
9.15.4.4 pack_loc_SetExtPowerState . . . 1410
9.15.4.5 pack_loc_SetOperationMode . . . 1411
9.15.4.6 pack_loc_SLQSLOCGetBestAvailPos . . . 1411
9.15.4.7 pack_loc_SLQSLOCGetOpMode . . . 1411
9.15.4.8 pack_loc_SLQSLOCGetServer . . . 1412
9.15.4.9 pack_loc_SLQSLOCInjectPosition . . . 1412
9.15.4.10 pack_loc_SLQSLOCInjectSensorData . . . 1413
9.15.4.11 pack_loc_SLQSLOCInjectUTCTime . . . 1413
9.15.4.12 pack_loc_SLQSLOCSetCradleMountConfig . . . 1413
9.15.4.13 pack_loc_SLQSLOCSetServer . . . 1414
9.15.4.14 pack_loc_Start . . . 1414
9.15.4.15 pack_loc_Stop . . . 1415
9.15.4.16 unpack_loc_BestAvailPos_Ind . . . 1415
9.15.4.17 unpack_loc_CradleMountCallback_Ind . . . 1415
9.15.4.18 unpack_loc_DeleteAssistData . . . 1416
9.15.4.19 unpack_loc_DeleteAssistData_Ind . . . 1416
9.15.4.20 unpack_loc_EngineState_Ind . . . 1416
9.15.4.21 unpack_loc_EventNMEA_Ind . . . 1417
9.15.4.22 unpack_loc_EventRegister . . . 1417
9.15.4.23 unpack_loc_EventTimeSyncCallback_Ind . . . 1417
9.15.4.24 unpack_loc_FixCriteria_Ind . . . 1418
9.15.4.25 unpack_loc_GetFixCriteria . . . 1418
9.15.4.26 unpack_loc_GetOpMode_Ind . . . 1418
9.15.4.27 unpack_loc_GetServer_Ind . . . 1419
9.15.4.28 unpack_loc_GnssSvInfo_Ind . . . 1419
9.15.4.29 unpack_loc_InjectPositionCallback_Ind . . . 1419
9.15.4.30 unpack_loc_InjectSensorDataCallback_Ind . . . 1420
9.15.4.31 unpack_loc_InjectTimeSyncDataCallback_Ind . . . 1420
9.15.4.32 unpack_loc_InjectUTCTimeCallback_Ind . . . 1420

9.15.4.33 unpack_loc_PositionRpt_Ind . . . 1421
9.15.4.34 unpack_loc_SensorStreamingCallback_Ind . . . 1421
9.15.4.35 unpack_loc_SetExtPowerConfig_Ind . . . 1421
9.15.4.36 unpack_loc_SetExtPowerState . . . 1422
9.15.4.37 unpack_loc_SetOperationMode . . . 1422
9.15.4.38 unpack_loc_SetOperationMode_Ind . . . 1422
9.15.4.39 unpack_loc_SetServer_Ind . . . 1423
9.15.4.40 unpack_loc_SLQSLOCGetBestAvailPos . . . 1423
9.15.4.41 unpack_loc_SLQSLOCGetOpMode . . . 1423
9.15.4.42 unpack_loc_SLQSLOCGetServer . . . 1424
9.15.4.43 unpack_loc_SLQSLOCInjectPosition . . . 1424
9.15.4.44 unpack_loc_SLQSLOCInjectSensorData . . . 1424
9.15.4.45 unpack_loc_SLQSLOCInjectUTCTime . . . 1425
9.15.4.46 unpack_loc_SLQSLOCSetCradleMountConfig . . . 1425
9.15.4.47 unpack_loc_SLQSLOCSetServer . . . 1425
9.15.4.48 unpack_loc_Start . . . 1426
9.15.4.49 unpack_loc_Stop . . . 1426
9.16 MbimIndicator.h File Reference . . . 1426
9.16.1 Typedef Documentation . . . 1427
9.16.1.1 MBIM_INDICATION_CALLBACK . . . 1427
9.16.1.2 MbimIndicator . . . 1427
9.16.2 Function Documentation . . . 1427
9.16.2.1 MbimIndicator_Initialize . . . 1427
9.17 MbimInternal.h File Reference . . . 1428
9.17.1 Macro Definition Documentation . . . 1429
9.17.1.1 MBIM_CLOSE . . . 1429
9.17.1.2 MBIM_CLOSE_DONE . . . 1429
9.17.1.3 MBIM_COMMAND . . . 1429
9.17.1.4 MBIM_COMMAND_DONE . . . 1429
9.17.1.5 MBIM_FUNCTION_ERROR . . . 1429
9.17.1.6 MBIM_HOST_ERROR . . . 1429
9.17.1.7 MBIM_INDICATE_STATUS . . . 1429
9.17.1.8 MBIM_OPEN . . . 1429
9.17.1.9 MBIM_OPEN_DONE . . . 1429
9.17.2 Typedef Documentation . . . 1429
9.17.2.1 MBIM_CLOSE_DONE_MSG . . . 1429
9.17.2.2 MBIM_CLOSE_MSG . . . 1429
9.17.2.3 MBIM_COMMAND_DONE_MSG . . . 1429
9.17.2.4 MBIM_COMMAND_MSG . . . 1429
9.17.2.5 MBIM_FRAGMENT_HEADER . . . 1429

9.17.2.6 MBIM_FRAGMENT_MSG . . . 1429
9.17.2.7 MBIM_INDICATE_STATUS_MSG . . . 1429
9.17.2.8 MBIM_MESSAGE_HEADER . . . 1429
9.17.2.9 MBIM_OPEN_DONE_MSG . . . 1429
9.17.2.10 MBIM_OPEN_MSG . . . 1429
9.18 MbimLayer.h File Reference . . . 1429
9.18.1 Function Documentation . . . 1429
9.18.1.1 MbimLayer_Create . . . 1429
9.18.1.2 MbimLayer_SendQmuxPacket . . . 1430
9.18.1.3 MbimLayer_ShutDown . . . 1430
9.19 MbimSyncObject.h File Reference . . . 1430
9.19.1 Typedef Documentation . . . 1430
9.19.1.1 MbimSyncObject . . . 1430
9.19.2 Function Documentation . . . 1431
9.19.2.1 MbimSyncObject_Destroy . . . 1431
9.19.2.2 MbimSyncObject_DoneCallback . . . 1431
9.19.2.3 MbimSyncObject_Initialize . . . 1431
9.19.2.4 MbimSyncObject_Lock . . . 1431
9.19.2.5 MbimSyncObject_Signal . . . 1432
9.19.2.6 MbimSyncObject_TimedWait . . . 1432
9.19.2.7 MbimSyncObject_Unlock . . . 1432
9.19.2.8 MbimSyncObject_Wait . . . 1432
9.20 MbimTransaction.h File Reference . . . 1433
9.20.1 Typedef Documentation . . . 1433
9.20.1.1 MBIM_DONE_CALLBACK . . . 1433
9.20.1.2 MbimTransaction . . . 1433
9.20.2 Function Documentation . . . 1434
9.20.2.1 MbimTransaction_Initialize . . . 1434
9.21 MbimTransport.h File Reference . . . 1434
9.21.1 Macro Definition Documentation . . . 1435
9.21.1.1 MBIM_COMMAND_TYPE_QUERY . . . 1435
9.21.1.2 MBIM_COMMAND_TYPE_SET . . . 1435
9.21.1.3 MBIM_MAX_CIDS . . . 1435
9.21.1.4 MBIM_MAX_CTRL_TRANSFER . . . 1435
9.21.1.5 MBIM_STATUS_FAILURE . . . 1435
9.21.1.6 MBIM_STATUS_SUCCESS . . . 1435
9.21.1.7 MBIM_UUID_SIZE . . . 1435
9.21.2 Typedef Documentation . . . 1435
9.21.2.1 MbimTransport . . . 1435
9.21.3 Function Documentation . . . 1436

9.21.3.1 MbimTransport_AttachIndicator . . . 1436
9.21.3.2 MbimTransport_CancelTransaction . . . 1436
9.21.3.3 MbimTransport_DetachIndicator . . . 1437
9.21.3.4 MbimTransport_GetNextTransactionId . . . 1437
9.21.3.5 MbimTransport_Initialize . . . 1437
9.21.3.6 MbimTransport_SendCommand . . . 1437
9.21.3.7 MbimTransport_ShutDown . . . 1438
9.22 nas.h File Reference . . . 1438
9.22.1 Macro Definition Documentation . . . 1452
9.22.1.1 LITE_MAX_ARFCN_NUM . . . 1452
9.22.1.2 LITE_MAX_NAS_3GPP2_MDN_LEN . . . 1452
9.22.1.3 LITE_MAX_NUM_OF_SET . . . 1452
9.22.1.4 LITE_MAX_PLMNLIST_NUM . . . 1452
9.22.1.5 LITE_MAX_SLQS_NAS_NW_NAME_SRC_LENGTH . . . 1452
9.22.1.6 LITE_NW_SCAN_LTE_OP_MODE_MAX_LENGTH . . . 1452
9.22.1.7 NAS_IMSI_M_S1_LENGTH . . . 1452
9.22.1.8 NAS_IMSI_M_S2_LENGTH . . . 1452
9.22.1.9 NAS_MAX_DESCRIPTION_LENGTH . . . 1452
9.22.1.10 NAS_MAX_NUM_NETWORKS . . . 1452
9.22.1.11 NAS_MAX_PHY_CA_AGG_SCELL_ARRRY_SIZE . . . 1452
9.22.1.12 NAS_MAX_SCC_RX_INFO_INSTANCES . . . 1452
9.22.1.13 NAS_MAX_SLQS_NAS_PCI_INFO_LENGTH . . . 1452
9.22.1.14 NAS_MAX_SLQS_NAS_PCI_INFO_PLMN_LENGTH . . . 1452
9.22.1.15 NAS_MCC_MNC_INST_LENGTH . . . 1452
9.22.1.16 NAS_NAM_NAME_LENGTH . . . 1452
9.22.1.17 NAS_OTA_MESSAGE_MAX_BUF_SIZE . . . 1452
9.22.1.18 NAS_PLMN_LENGTH . . . 1452
9.22.1.19 NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST . . . 1452
9.22.2 Typedef Documentation . . . 1452
9.22.2.1 nas_LTEBandPrefExtTlv . . . 1452
9.22.2.2 nas_NR5gBandPrefTlv . . . 1452
9.22.2.3 unpack_nas_InitiateDomainAttach_t . . . 1453
9.22.2.4 unpack_nas_SetACCOLC_t . . . 1453
9.22.2.5 unpack_nas_SetCDMANetworkParameters_t . . . 1453
9.22.2.6 unpack_nas_SetLURejectCallback_t . . . 1453
9.22.2.7 unpack_nas_SetRFInfoCallback_t . . . 1453
9.22.2.8 unpack_nas_SLQSConfigSigInfo_t . . . 1453
9.22.2.9 unpack_nas_SLQSInitiateNetworkRegistration_t . . . 1453
9.22.2.10 unpack_nas_SLQSNasConfigSigInfo2_t . . . 1453
9.22.2.11 unpack_nas_SLQSNasIndicationRegisterExt_t . . . 1453

9.22.2.12 unpack_nas_SLQSNasIndicationRegisterV2_t . . . 1453
9.22.2.13 unpack_nas_SLQSNASSeteDRXParams_t . . . 1453
9.22.2.14 unpack_nas_SLQSNasSwiIndicationRegister_t . . . 1453
9.22.2.15 unpack_nas_SLQSNASSwiSetChannelLock_t . . . 1453
9.22.2.16 unpack_nas_SLQSPerformIncNetworkScan_t . . . 1453
9.22.2.17 unpack_nas_SLQSSet3GPP2SubscriptionInfo_t . . . 1453
9.22.2.18 unpack_nas_SLQSSetBandPreference_t . . . 1453
9.22.2.19 unpack_nas_SLQSSetDDTMPref_t . . . 1453
9.22.2.20 unpack_nas_SLQSSetSignalStrengthsCallback_t . . . 1453
9.22.2.21 unpack_nas_SLQSSetSysSelectionPref_t . . . 1453
9.22.2.22 unpack_nas_SLQSSetSysSelectionPrefExt_t . . . 1453
9.22.2.23 unpack_nas_SLQSSwiPSDetach_t . . . 1453
9.22.3 Enumeration Type Documentation . . . 1453
9.22.3.1 LITE_TYPE_OF_SERVICE_DOMAIN . . . 1453
9.22.3.2 LITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND . . . 1454
9.22.3.3 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB . . . 1455
9.22.3.4 LITEQMI_NAS_LTE_CPHY_SCELL_STATE . . . 1455
9.22.3.5 NAS_LTE_CPHY_CA_BW_NRB_LITE . . . 1455
9.22.3.6 NAS_LTE_CPHY_SCELL_STATE_LITE . . . 1455
9.22.4 Function Documentation . . . 1456
9.22.4.1 pack_nas_GetACCOLC . . . 1456
9.22.4.2 pack_nas_GetANAAAAuthenticationStatus . . . 1456
9.22.4.3 pack_nas_GetCDMANetworkParameters . . . 1456
9.22.4.4 pack_nas_GetHomeNetwork . . . 1457
9.22.4.5 pack_nas_GetHomeNetwork3GPP2 . . . 1457
9.22.4.6 pack_nas_GetNetworkPreference . . . 1457
9.22.4.7 pack_nas_GetRFInfo . . . 1457
9.22.4.8 pack_nas_GetServingNetwork . . . 1458
9.22.4.9 pack_nas_GetServingNetworkCapabilities . . . 1458
9.22.4.10 pack_nas_GetSignalStrengths . . . 1458
9.22.4.11 pack_nas_InitiateDomainAttach . . . 1459
9.22.4.12 pack_nas_PerformNetworkScan . . . 1459
9.22.4.13 pack_nas_PerformNetworkScanPCI . . . 1459
9.22.4.14 pack_nas_SetACCOLC . . . 1460
9.22.4.15 pack_nas_SetCDMANetworkParameters . . . 1460
9.22.4.16 pack_nas_SetLURejectCallback . . . 1461
9.22.4.17 pack_nas_SetNetworkPreference . . . 1461
9.22.4.18 pack_nas_SetRFInfoCallback . . . 1461
9.22.4.19 pack_nas_SLQSConfigSigInfo . . . 1462
9.22.4.20 pack_nas_SLQSGetErrorRate . . . 1462

9.22.4.21 pack_nas_SLQSGetHomeNetwork . . . 1462
9.22.4.22 pack_nas_SlqsGetLTECphyCAInfo . . . 1463
9.22.4.23 pack_nas_SLQSGetNetworkTime . . . 1463
9.22.4.24 pack_nas_SLQSGetOperatorNameData . . . 1463
9.22.4.25 pack_nas_SLQSGetPLMNName . . . 1463
9.22.4.26 pack_nas_SLQSGetServingSystem . . . 1464
9.22.4.27 pack_nas_SLQSGetServingSystemV2 . . . 1464
9.22.4.28 pack_nas_SLQSGetSignalStrength . . . 1465
9.22.4.29 pack_nas_SLQSGetSysInfo . . . 1465
9.22.4.30 pack_nas_SLQSGetSysInfoV2 . . . 1465
9.22.4.31 pack_nas_SLQSGetSysSelectionPref . . . 1466
9.22.4.32 pack_nas_SLQSGetSysSelectionPrefExt . . . 1466
9.22.4.33 pack_nas_SLQSGetSysSelectionPrefExtV2 . . . 1466
9.22.4.34 pack_nas_SLQSInitiateNetworkRegistration . . . 1467
9.22.4.35 pack_nas_SLQSNasConfigSigInfo2 . . . 1467
9.22.4.36 pack_nas_SLQSNasGet3GPP2Subscription . . . 1467
9.22.4.37 pack_nas_SLQSNasGetCellLocationInfo . . . 1468
9.22.4.38 pack_nas_SLQSNasGetCellLocationInfoV2 . . . 1468
9.22.4.39 pack_nas_SLQSNASGeteDRXParams . . . 1468
9.22.4.40 pack_nas_SLQSNASGeteDRXParamsExt . . . 1469
9.22.4.41 pack_nas_SLQSNASGetForbiddenNetworks . . . 1469
9.22.4.42 pack_nas_SLQSNasGetHDRColorCode . . . 1469
9.22.4.43 pack_nas_SLQSNasGetRFInfo . . . 1470
9.22.4.44 pack_nas_SLQSNasGetSigInfo . . . 1470
9.22.4.45 pack_nas_SLQSNasGetTxRxInfo . . . 1470
9.22.4.46 pack_nas_SLQSNasIndicationRegisterExt . . . 1471
9.22.4.47 pack_nas_SLQSNasIndicationRegisterV2 . . . 1471
9.22.4.48 pack_nas_SLQSNASSeteDRXParams . . . 1472
9.22.4.49 pack_nas_SLQSNASSwiGetChannelLock . . . 1472
9.22.4.50 pack_nas_SLQSNasSwiIndicationRegister . . . 1472
9.22.4.51 pack_nas_SLQSNasSwiModemStatus . . . 1473
9.22.4.52 pack_nas_SLQSNASSwiSetChannelLock . . . 1473
9.22.4.53 pack_nas_SLQSPerformIncNetworkScanReq . . . 1474
9.22.4.54 pack_nas_SLQSPerformNetworkScanV2 . . . 1474
9.22.4.55 pack_nas_SLQSSet3GPP2SubscriptionInfo . . . 1474
9.22.4.56 pack_nas_SLQSSetBandPreference . . . 1475
9.22.4.57 pack_nas_SLQSSetDDTMPref . . . 1475
9.22.4.58 pack_nas_SLQSSetSignalStrengthsCallback . . . 1475
9.22.4.59 pack_nas_SLQSSetSysSelectionPref . . . 1476
9.22.4.60 pack_nas_SLQSSetSysSelectionPrefExt . . . 1476

9.22.4.61 pack_nas_SLQSSwiGetHDRPersonality . . . 1476
9.22.4.62 pack_nas_SLQSSwiGetHDRProtSubtype . . . 1477
9.22.4.63 pack_nas_SLQSSwiGetHRPDStats . . . 1477
9.22.4.64 pack_nas_SLQSSwiGetLteCQI . . . 1477
9.22.4.65 pack_nas_SLQSSwiGetLteSccRxInfo . . . 1478
9.22.4.66 pack_nas_SLQSSwiNetworkDebug . . . 1478
9.22.4.67 pack_nas_SLQSSwiPSDetach . . . 1478
9.22.4.68 unpack_nas_GetACCOLC . . . 1479
9.22.4.69 unpack_nas_GetANAAAAuthenticationStatus . . . 1479
9.22.4.70 unpack_nas_GetCDMANetworkParameters . . . 1480
9.22.4.71 unpack_nas_GetHomeNetwork . . . 1480
9.22.4.72 unpack_nas_GetHomeNetwork3GPP2 . . . 1480
9.22.4.73 unpack_nas_GetNetworkPreference . . . 1481
9.22.4.74 unpack_nas_GetRFInfo . . . 1481
9.22.4.75 unpack_nas_GetServingNetwork . . . 1481
9.22.4.76 unpack_nas_GetServingNetworkCapabilities . . . 1482
9.22.4.77 unpack_nas_GetSignalStrengths . . . 1482
9.22.4.78 unpack_nas_InitiateDomainAttach . . . 1482
9.22.4.79 unpack_nas_PerformNetworkScan . . . 1483
9.22.4.80 unpack_nas_SetACCOLC . . . 1483
9.22.4.81 unpack_nas_SetCDMANetworkParameters . . . 1483
9.22.4.82 unpack_nas_SetDataCapabilitiesCallback_ind . . . 1484
9.22.4.83 unpack_nas_SetEventReportInd . . . 1484
9.22.4.84 unpack_nas_SetLURejectCallback . . . 1485
9.22.4.85 unpack_nas_SetNasLTECphyCaIndCallback_ind . . . 1485
9.22.4.86 unpack_nas_SetNetworkPreference . . . 1485
9.22.4.87 unpack_nas_SetRFInfoCallback . . . 1486
9.22.4.88 unpack_nas_SetRoamingIndicatorCallback_ind . . . 1486
9.22.4.89 unpack_nas_SetServingSystemCallback_ind . . . 1486
9.22.4.90 unpack_nas_SLQSConfigSigInfo . . . 1487
9.22.4.91 unpack_nas_SLQSDDTMPrefIndicationCallback_Ind . . . 1487
9.22.4.92 unpack_nas_SLQSGetErrorRate . . . 1487
9.22.4.93 unpack_nas_SLQSGetHomeNetwork . . . 1488
9.22.4.94 unpack_nas_SlqsGetLTECphyCAInfo . . . 1488
9.22.4.95 unpack_nas_SLQSGetNetworkTime . . . 1488
9.22.4.96 unpack_nas_SLQSGetOperatorNameData . . . 1489
9.22.4.97 unpack_nas_SLQSGetPLMNName . . . 1489
9.22.4.98 unpack_nas_SLQSGetServingSystem . . . 1489
9.22.4.99 unpack_nas_SLQSGetServingSystemV2 . . . 1490
9.22.4.100 unpack_nas_SLQSGetSignalStrength . . . 1490

9.22.4.101unpack_nas_SLQSGetSysInfo . . . 1490
9.22.4.102unpack_nas_SLQSGetSysInfoV2 . . . 1491
9.22.4.103unpack_nas_SLQSGetSysSelectionPref . . . 1491
9.22.4.104unpack_nas_SLQSGetSysSelectionPrefExt . . . 1492
9.22.4.105unpack_nas_SLQSGetSysSelectionPrefExtV2 . . . 1492
9.22.4.106unpack_nas_SLQSInitiateNetworkRegistration . . . 1493
9.22.4.107unpack_nas_SLQSNasConfigSigInfo2 . . . 1493
9.22.4.108unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind . . . 1493
9.22.4.109unpack_nas_SLQSNasGet3GPP2Subscription . . . 1494
9.22.4.110unpack_nas_SLQSNasGetCellLocationInfo . . . 1494
9.22.4.111unpack_nas_SLQSNasGetCellLocationInfoV2 . . . 1494
9.22.4.112unpack_nas_SLQSNASGeteDRXParams . . . 1495
9.22.4.113unpack_nas_SLQSNASGeteDRXParamsExt . . . 1495
9.22.4.114unpack_nas_SLQSNASGetForbiddenNetworks . . . 1495
9.22.4.115unpack_nas_SLQSNasGetHDRColorCode . . . 1496
9.22.4.116unpack_nas_SLQSNasGetRFInfo . . . 1496
9.22.4.117unpack_nas_SLQSNasGetSigInfo . . . 1496
9.22.4.118unpack_nas_SLQSNasGetTxRxInfo . . . 1497
9.22.4.119unpack_nas_SLQSNasIndicationRegisterExt . . . 1497
9.22.4.120unpack_nas_SLQSNasIndicationRegisterV2 . . . 1498
9.22.4.121unpack_nas_SLQSNasNetworkRejectCallback_Ind . . . 1498
9.22.4.122unpack_nas_SLQSNasNetworkTimeCallBack_ind . . . 1498
9.22.4.123unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind . . . 1499
9.22.4.124unpack_nas_SLQSNasRFBandInfoCallback_Ind . . . 1499
9.22.4.125unpack_nas_SLQSNASSeteDRXParams . . . 1499
9.22.4.126unpack_nas_SLQSNasSigInfoCallback_ind . . . 1500
9.22.4.127unpack_nas_SLQSNASSwiGetChannelLock . . . 1500
9.22.4.128unpack_nas_SLQSNasSwiIndicationRegister . . . 1500
9.22.4.129unpack_nas_SLQSNasSwiModemStatus . . . 1501
9.22.4.130unpack_nas_SLQSNasSwiOTAMessageCallback_ind . . . 1501
9.22.4.131unpack_nas_SLQSNASSwiSetChannelLock . . . 1501
9.22.4.132unpack_nas_SLQSNasSysInfoCallback_ind . . . 1502
9.22.4.133unpack_nas_SLQSNasTimerCallback_ind . . . 1502
9.22.4.134unpack_nas_SLQSPerformIncNetworkScan . . . 1502
9.22.4.135unpack_nas_SLQSPerformNetworkScanV2 . . . 1503
9.22.4.136unpack_nas_SLQSSet3GPP2SubscriptionInfo . . . 1503
9.22.4.137unpack_nas_SLQSSetBandPreference . . . 1503
9.22.4.138unpack_nas_SLQSSetDDTMPref . . . 1504
9.22.4.139unpack_nas_SLQSSetSignalStrengthsCallback . . . 1504
9.22.4.140unpack_nas_SLQSSetSysSelectionPref . . . 1504

9.22.4.141 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind . . . 1505
9.22.4.142 unpack_nas_SLQSSetSysSelectionPrefExt . . . 1505
9.22.4.143 unpack_nas_SLQSSwiGetHDRPersonality . . . 1505
9.22.4.144 unpack_nas_SLQSSwiGetHDRProtSubtype . . . 1506
9.22.4.145 unpack_nas_SLQSSwiGetHRPDStats . . . 1506
9.22.4.146 unpack_nas_SLQSSwiGetLteCQI . . . 1506
9.22.4.147 unpack_nas_SLQSSwiGetLteSccRxInfo . . . 1507
9.22.4.148 unpack_nas_SLQSSwiHDRPersonalityCallback_Ind . . . 1507
9.22.4.149 unpack_nas_SLQSSwiNetworkDebug . . . 1507
9.22.4.150 unpack_nas_SLQSSwiPSDetach . . . 1508
9.22.4.151 unpack_nas_SLQSSwiRandIndicatorCallback_Ind . . . 1508
9.22.4.152 unpack_valid_nas_GetCDMANetworkParameters . . . 1508
9.22.4.153 unpack_valid_nas_SLQSGetServingSystem . . . 1509
9.22.4.154 unpack_valid_nas_SLQSGetSignalStrength . . . 1510
9.22.4.155 unpack_valid_nas_SLQSNasGetSigInfo . . . 1510
9.23 pds.h File Reference . . . 1511
9.23.1 Typedef Documentation . . . 1513
9.23.1.1 unpack_pds_ForceXTRADownload_t . . . 1513
9.23.1.2 unpack_pds_PDSInjectTimeReference_t . . . 1513
9.23.1.3 unpack_pds_ResetPDSData_t . . . 1513
9.23.1.4 unpack_pds_SetEventReportCallback_t . . . 1513
9.23.1.5 unpack_pds_SetPDSDefaults_t . . . 1513
9.23.1.6 unpack_pds_SetPDSState_t . . . 1513
9.23.1.7 unpack_pds_SetPortAutomaticTracking_t . . . 1514
9.23.1.8 unpack_pds_SetServiceAutomaticTracking_t . . . 1514
9.23.1.9 unpack_pds_SetXTRAAutomaticDownload_t . . . 1514
9.23.1.10 unpack_pds_SetXTRANetwork_t . . . 1514
9.23.1.11 unpack_pds_SLQSPDSDeterminePosition_t . . . 1514
9.23.1.12 unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t . . . 1514
9.23.1.13 unpack_pds_SLQSPDSInjectPositionData_t . . . 1514
9.23.1.14 unpack_pds_SLQSSetAGPSConfig_t . . . 1514
9.23.1.15 unpack_pds_SLQSSetPositionMethodState_t . . . 1514
9.23.1.16 unpack_pds_StartPDSTrackingSessionExt_t . . . 1514
9.23.1.17 unpack_pds_StopPDSTrackingSession_t . . . 1514
9.23.2 Function Documentation . . . 1514
9.23.2.1 pack_pds_ForceXTRADownload . . . 1514
9.23.2.2 pack_pds_GetPDSDefaults . . . 1514
9.23.2.3 pack_pds_GetPDSState . . . 1515
9.23.2.4 pack_pds_GetPortAutomaticTracking . . . 1515
9.23.2.5 pack_pds_GetServiceAutomaticTracking . . . 1515

9.23.2.6 pack_pds_GetXTRAAutomaticDownload . . . 1516
9.23.2.7 pack_pds_GetXTRANetwork . . . 1516
9.23.2.8 pack_pds_GetXTRAValidity . . . 1516
9.23.2.9 pack_pds_PDSInjectTimeReference . . . 1517
9.23.2.10 pack_pds_ResetPDSData . . . 1517
9.23.2.11 pack_pds_SetEventReportCallback . . . 1517
9.23.2.12 pack_pds_SetPDSDefaults . . . 1518
9.23.2.13 pack_pds_SetPDSState . . . 1518
9.23.2.14 pack_pds_SetPortAutomaticTracking . . . 1518
9.23.2.15 pack_pds_SetServiceAutomaticTracking . . . 1519
9.23.2.16 pack_pds_SetXTRAAutomaticDownload . . . 1519
9.23.2.17 pack_pds_SetXTRANetwork . . . 1520
9.23.2.18 pack_pds_SLQSGetAGPSConfig . . . 1520
9.23.2.19 pack_pds_SLQSGetGPSStateInfo . . . 1520
9.23.2.20 pack_pds_SLQSPDSDeterminePosition . . . 1521
9.23.2.21 pack_pds_SLQSPDSInjectAbsoluteTimeReference . . . 1521
9.23.2.22 pack_pds_SLQSPDSInjectPositionData . . . 1521
9.23.2.23 pack_pds_SLQSSetAGPSConfig . . . 1522
9.23.2.24 pack_pds_SLQSSetPositionMethodState . . . 1522
9.23.2.25 pack_pds_StartPDSTrackingSessionExt . . . 1522
9.23.2.26 pack_pds_StopPDSTrackingSession . . . 1523
9.23.2.27 unpack_pds_ForceXTRADownload . . . 1523
9.23.2.28 unpack_pds_GetPDSDefaults . . . 1524
9.23.2.29 unpack_pds_GetPDSState . . . 1524
9.23.2.30 unpack_pds_GetPortAutomaticTracking . . . 1524
9.23.2.31 unpack_pds_GetServiceAutomaticTracking . . . 1525
9.23.2.32 unpack_pds_GetXTRAAutomaticDownload . . . 1525
9.23.2.33 unpack_pds_GetXTRANetwork . . . 1525
9.23.2.34 unpack_pds_GetXTRAValidity . . . 1526
9.23.2.35 unpack_pds_PDSInjectTimeReference . . . 1526
9.23.2.36 unpack_pds_ResetPDSData . . . 1526
9.23.2.37 unpack_pds_SetEventReport_Ind . . . 1527
9.23.2.38 unpack_pds_SetEventReportCallback . . . 1527
9.23.2.39 unpack_pds_SetPDSDefaults . . . 1527
9.23.2.40 unpack_pds_SetPDSState . . . 1528
9.23.2.41 unpack_pds_SetPdsState_Ind . . . 1528
9.23.2.42 unpack_pds_SetPortAutomaticTracking . . . 1528
9.23.2.43 unpack_pds_SetServiceAutomaticTracking . . . 1529
9.23.2.44 unpack_pds_SetXTRAAutomaticDownload . . . 1529
9.23.2.45 unpack_pds_SetXTRANetwork . . . 1529

9.23.2.46 unpack_pds_SLQSGetAGPSConfig . . . 1530
9.23.2.47 unpack_pds_SLQSGetGPSStateInfo . . . 1530
9.23.2.48 unpack_pds_SLQSPDSDeterminePosition . . . 1530
9.23.2.49 unpack_pds_SLQSPDSInjectAbsoluteTimeReference . . . 1531
9.23.2.50 unpack_pds_SLQSPDSInjectPositionData . . . 1531
9.23.2.51 unpack_pds_SLQSSetAGPSConfig . . . 1531
9.23.2.52 unpack_pds_SLQSSetPositionMethodState . . . 1532
9.23.2.53 unpack_pds_StartPDSTrackingSessionExt . . . 1532
9.23.2.54 unpack_pds_StopPDSTrackingSession . . . 1532
9.24 qaGobiApiTableBandClasses.h File Reference . . . 1533
9.24.1 Detailed Description . . . 1533
9.24.2 Band Classes (Value - Description) . . . 1533
9.24.2.1 LTE Bands . . . 1534
9.25 qaGobiApiTableCallControlReturnReasons.h File Reference . . . 1536
9.25.1 Detailed Description . . . 1536
9.25.2 S1 . . . 1536
9.26 qaGobiApiTableCallEndReasons.h File Reference . . . 1537
9.26.1 Detailed Description . . . 1537
9.26.2 Call end reason codes (Code - Reason) . . . 1537
9.26.2.1 Technology-agnostic call end reasons . . . 1537
9.26.2.2 EVDO CDMA 1xEV-DO . . . 1538
9.26.2.3 WCDMA/GSM call end reasons . . . 1538
9.26.2.4 EVDO CDMA 1xEV-DO . . . 1540
9.26.2.5 call end reason type . . . 1541
9.26.2.6 Mobile IP call end reasons (Type=1) . . . 1541
9.26.2.7 Internal call end reasons (Type=2) . . . 1543
9.26.2.8 Call Manager defined call end reasons (Type=3) . . . 1545
9.26.2.9 3GPP specification defined call end reasons (Type=6) . . . 1550
9.26.2.10 PPP call end reasons (Type=7) . . . 1552
9.26.2.11 EHRPD call end reasons (Type=8) . . . 1552
9.26.2.12 IPV6 call end reasons (Type=9) . . . 1553
9.27 qaGobiApiTableCarrierCodes.h File Reference . . . 1553
9.27.1 Detailed Description . . . 1553
9.27.2 Carrier Codes (Number - Carrier) . . . 1554
9.28 qaGobiApiTableCodingScheme.h File Reference . . . 1555
9.28.1 Detailed Description . . . 1555
9.28.2 S1 . . . 1555
9.28.2.1 Use of bits 3..0\n\n . . . 1555
9.28.3 Coding Group Bits 7..4(0001) . . . 1556
9.28.3.1 use of bits 3..0 . . . 1556

9.28.4 Coding Group Bits 7..4(0010) . . . 1556
9.28.4.1 use of bits 3..0 . . . 1556
9.28.5 Coding Group Bits 7..4(0011) . . . 1557
9.28.5.1 use of bits 3..0 . . . 1557
9.28.6 Coding Group Bits 7..4(01xx) . . . 1557
9.28.6.1 use of bits 3..0 . . . 1557
9.28.7 Coding Group Bits 7..4(1001) . . . 1557
9.28.7.1 Reserved coding groups . . . 1557
9.28.8 Coding Group Bits 7..4(1010..1101) . . . 1558
9.28.8.1 Reserved coding groups . . . 1558
9.28.9 Coding Group Bits 7..4(1110) . . . 1558
9.28.9.1 Defined by the WAP Forum . . . 1558
9.28.10 Coding Group Bits 7..4 (1111) . . . 1558
9.28.10.1 Data coding / message handling . . . 1558
9.29 qaGobiApiTableFwDldErrorCodes.h File Reference . . . 1558
9.29.1 Detailed Description . . . 1558
9.29.2 FOTA update agent errors . . . 1558
9.29.2.1 defined error code (in hexadecimal) . . . 1558
9.29.2.2 defined error code (in hexadecimal) . . . 1561
9.29.3 FDT/SSDP reported errors . . . 1561
9.29.3.1 Code(in hex) - Description - Reference BUT error Message . . . 1561
9.30 qaGobiApiTableGpsCapabilityCodes.h File Reference . . . 1562
9.30.1 Detailed Description . . . 1562
9.30.2 GPS capability (Value - Capability) . . . 1562
9.31 qaGobiApiTablePowerModes.h File Reference . . . 1562
9.31.1 Detailed Description . . . 1562
9.31.2 Power Modes (Value - Description) . . . 1562
9.32 qaGobiApiTableRadioInterfaces.h File Reference . . . 1563
9.32.1 Detailed Description . . . 1563
9.32.2 Radio interface . . . 1563
9.32.2.1 Technology (Value - Radio Interface Technology) . . . 1563
9.33 qaGobiApiTableRegionCodes.h File Reference . . . 1564
9.33.1 Detailed Description . . . 1564
9.33.2 Region Codes (Code - Region) . . . 1564
9.34 qaGobiApiTableServiceOptions.h File Reference . . . 1564
9.34.1 Detailed Description . . . 1564
9.34.2 Service Option codes (Code - Reason) . . . 1564
9.34.2.1 Description . . . 1564
9.35 qaGobiApiTableSupServiceInfoClasses.h File Reference . . . 1566
9.35.1 Detailed Description . . . 1566

9.35.2 Supplementary Service Information Classes (Value - Service Class) . . . 1567
9.36 qaGobiApiTableSwiAudio.h File Reference . . . 1567
9.36.1 Detailed Description . . . 1567
9.36.2 ACDB Device (Device ID - description) . . . 1567
9.36.3 FDT/SSDP reported errors . . . 1567
9.37 qaGobiApiTableSwiOMADMSessionStatus.h File Reference . . . 1567
9.37.1 Detailed Description . . . 1568
9.37.2 DSDSOMask SO Mask . . . 1568
9.38 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference . . . 1568
9.38.1 Detailed Description . . . 1568
9.38.2 OMA DM Update Complete Status (Update Complete Status - Meaning - Usage) . . . 1569
9.39 qaGobiApiTableVoiceCallEndReasons.h File Reference . . . 1570
9.39.1 Detailed Description . . . 1570
9.39.2 Voice Call and supplementary services end reason codes (Code - Reason) . . . 1570
9.39.2.1 General . . . 1570
9.39.2.2 service Errors . . . 1572
9.39.2.3 control cause values . . . 1573
9.39.2.4 reject causes . . . 1574
9.39.2.5 reject causes . . . 1575
9.39.2.6 reject causes . . . 1575
9.39.2.7 stratum reject causes . . . 1575
9.39.2.8 reject causes . . . 1576
9.39.2.9 IP end reasons . . . 1576
9.40 qmerrno.h File Reference . . . 1576
9.40.1 Enumeration Type Documentation . . . 1578
9.40.1.1 eQCWWANError . . . 1578
9.40.1.2 qm_wds_ds_profile_extended_err_codes . . . 1583
9.41 QmiService.h File Reference . . . 1583
9.41.1 Macro Definition Documentation . . . 1584
9.41.1.1 QMI_TRANSACTION_POOL_SIZE . . . 1584
9.41.2 Typedef Documentation . . . 1584
9.41.2.1 QMI_INDICATION_CALLBACK . . . 1584
9.41.2.2 QmiService . . . 1584
9.41.3 Function Documentation . . . 1585
9.41.3.1 QmiService_CancelTransaction . . . 1585
9.41.3.2 QmiService_GetNextTransactionId . . . 1585
9.41.3.3 QmiService_HandleQmiPacket . . . 1586
9.41.3.4 QmiService_Initialize . . . 1586
9.41.3.5 QmiService_SendRequest . . . 1586
9.41.3.6 QmiService_ShutDown . . . 1587

9.42 QmiSyncObject.h File Reference . . . 1587
9.42.1 Macro Definition Documentation . . . 1587
9.42.1.1 MAX_QMI_PACKET_SIZE . . . 1587
9.42.2 Typedef Documentation . . . 1587
9.42.2.1 QmiSyncObject . . . 1587
9.42.3 Function Documentation . . . 1588
9.42.3.1 QmiSyncObject_Destroy . . . 1588
9.42.3.2 QmiSyncObject_Initialize . . . 1588
9.42.3.3 QmiSyncObject_Lock . . . 1588
9.42.3.4 QmiSyncObject_ResponseCallback . . . 1589
9.42.3.5 QmiSyncObject_Signal . . . 1589
9.42.3.6 QmiSyncObject_TimedWait . . . 1589
9.42.3.7 QmiSyncObject_Unlock . . . 1589
9.42.3.8 QmiSyncObject_Wait . . . 1590
9.43 QmiTransaction.h File Reference . . . 1590
9.43.1 Typedef Documentation . . . 1590
9.43.1.1 QMI_RESPONSE_CALLBACK . . . 1590
9.43.1.2 QmiTransaction . . . 1590
9.43.2 Function Documentation . . . 1591
9.43.2.1 QmiTransaction_Initialize . . . 1591
9.44 QmuxInternal.h File Reference . . . 1591
9.44.1 Macro Definition Documentation . . . 1592
9.44.1.1 BROADCAST_CLIENT_ID . . . 1592
9.44.1.2 eCTL_IND . . . 1592
9.44.1.3 eCTL_REQ . . . 1592
9.44.1.4 eCTL_RSP . . . 1592
9.44.1.5 eInvalidService . . . 1592
9.44.1.6 FAILURE . . . 1592
9.44.1.7 GET_CLIENT_ID . . . 1592
9.44.1.8 IF_TYPE_SIZE . . . 1592
9.44.1.9 MAX . . . 1592
9.44.1.10 MAX_QMUX_PACKET_SIZE . . . 1592
9.44.1.11 QMI_ERR_GENERAL . . . 1592
9.44.1.12 QMI_RESULT_FAILURE . . . 1592
9.44.1.13 QMI_RESULT_SUCCESS . . . 1593
9.44.1.14 QMUX_CTL_FLAG_CTL_POINT . . . 1593
9.44.1.15 QMUX_CTL_FLAG_SERVICE . . . 1593
9.44.1.16 QMUX_MSG_FLAG . . . 1593
9.44.1.17 RELEASE_CLIENT_ID . . . 1593
9.44.1.18 SUCCESS . . . 1593

9.44.1.19 TLV_TYPE_REQUIRED_PARAMETER . . . 1593
9.44.1.20 TLV_TYPE_RESULT_CODE . . . 1593
9.44.2 Typedef Documentation . . . 1593
9.44.2.1 CtlTransactionHdr . . . 1593
9.44.2.2 GetClientIdReqMsg . . . 1593
9.44.2.3 GetClientIdRspMsg . . . 1593
9.44.2.4 QmiHdr . . . 1593
9.44.2.5 QmiTransactionHdr . . . 1593
9.44.2.6 ReleaseClientIdReqMsg . . . 1593
9.44.2.7 ReleaseClientIdRspMsg . . . 1593
9.44.2.8 ResultTlv . . . 1593
9.45 QmuxTransport.h File Reference . . . 1593
9.45.1 Macro Definition Documentation . . . 1594
9.45.1.1 INVALID_CLIENT_ID . . . 1594
9.45.1.2 INVALID_TRANSACTION_ID . . . 1594
9.45.1.3 MAX_QMI_CLIENT . . . 1594
9.45.1.4 MAX_QMI_CLIENT_ID . . . 1594
9.45.1.5 QCQMI_NODE_NAME_PREFIX . . . 1594
9.45.2 Typedef Documentation . . . 1594
9.45.2.1 QmuxTransport . . . 1594
9.45.3 Function Documentation . . . 1595
9.45.3.1 QmuxTransport_AttachQmiService . . . 1595
9.45.3.2 QmuxTransport_DetachQmiService . . . 1595
9.45.3.3 QmuxTransport_Initialize . . . 1595
9.45.3.4 QmuxTransport_SendQmiPacket . . . 1595
9.45.3.5 QmuxTransport_ShutDown . . . 1596
9.46 qos.h File Reference . . . 1596
9.46.1 Macro Definition Documentation . . . 1597
9.46.1.1 LITEQMI_MAX_QOS_FILTERS . . . 1597
9.46.1.2 LITEQMI_MAX_QOS_FLOW_PER_APN_STATS . . . 1597
9.46.1.3 LITEQMI_MAX_QOS_FLOWS . . . 1597
9.46.2 Function Documentation . . . 1597
9.46.2.1 pack_qos_BindDataPort . . . 1597
9.46.2.2 pack_qos_SLQSQosGetNetworkStatus . . . 1598
9.46.2.3 pack_qos_SLQSQosSwiReadApnExtraParams . . . 1599
9.46.2.4 pack_qos_SLQSQosSwiReadDataStats . . . 1599
9.46.2.5 pack_qos_SLQSSetQosEventCallback . . . 1600
9.46.2.6 unpack_qos_BindDataPort . . . 1601
9.46.2.7 unpack_qos_SLQSQosGetNetworkStatus . . . 1601
9.46.2.8 unpack_qos_SLQSQosSwiReadApnExtraParams . . . 1602

9.46.2.9 unpack_qos_SLQSQosSwiReadDataStats . . . 1602
9.46.2.10 unpack_qos_SLQSSetQosEventCallback . . . 1602
9.46.2.11 unpack_qos_SLQSSetQosEventCallback_ind . . . 1603
9.46.2.12 unpack_qos_SLQSSetQosNWStatusCallback_ind . . . 1603
9.46.2.13 unpack_qos_SLQSSetQosPriEventCallback_ind . . . 1604
9.46.2.14 unpack_qos_SLQSSetQosStatusCallback_ind . . . 1605
9.47 rms.h File Reference . . . 1605
9.47.1 Macro Definition Documentation . . . 1606
9.47.1.1 __LITEQMI_RMS_H__ . . . 1606
9.47.2 Function Documentation . . . 1606
9.47.2.1 pack_rms_GetSMSWake . . . 1606
9.47.2.2 pack_rms_SetSMSWake . . . 1606
9.47.2.3 unpack_rms_GetSMSWake . . . 1606
9.47.2.4 unpack_rms_SetSMSWake . . . 1607
9.48 sar.h File Reference . . . 1607
9.48.1 Typedef Documentation . . . 1608
9.48.1.1 unpack_sar_SLQSSetRfSarState_t . . . 1608
9.48.2 Function Documentation . . . 1608
9.48.2.1 pack_sar_SLQSGetRfSarState . . . 1608
9.48.2.2 pack_sar_SLQSSetRfSarState . . . 1608
9.48.2.3 unpack_sar_SLQSGetRfSarState . . . 1608
9.48.2.4 unpack_sar_SLQSSetRfSarState . . . 1609
9.49 sms.h File Reference . . . 1609
9.49.1 Macro Definition Documentation . . . 1613
9.49.1.1 MAX_CDMA_ENC_MO_TXT_MSG_SIZE . . . 1613
9.49.1.2 MAX_MS_TRANSFER_ROUTE_MSG . . . 1613
9.49.1.3 MAX_MSC_ADDRESS_SIZE . . . 1613
9.49.1.4 MAX_MSE_TWS_MSG . . . 1613
9.49.1.5 MAX_SMS_LIST_SIZE . . . 1613
9.49.1.6 MAX_SMS_MESSAGE_SIZE . . . 1613
9.49.1.7 SMS_CONFIG_LEN . . . 1613
9.49.1.8 SMS_MAX_SMS_ROUTES . . . 1613
9.49.1.9 SMS_NUM_OF_SET . . . 1613
9.49.1.10 SMSC_TYPE_LEN . . . 1613
9.49.2 Enumeration Type Documentation . . . 1613
9.49.2.1 eqmiCbkSetStatus . . . 1613
9.49.3 Function Documentation . . . 1613
9.49.3.1 pack_sms_GetSMSCAddress . . . 1613
9.49.3.2 pack_sms_SaveSMS . . . 1614
9.49.3.3 pack_sms_SendSMS . . . 1614

9.49.3.4 pack_sms_SetNewSMSCallback . . . 1614
9.49.3.5 pack_sms_SetSMSCAddress . . . 1615
9.49.3.6 pack_sms_SLQSDeleteSMS . . . 1615
9.49.3.7 pack_sms_SLQSGetIndicationRegister . . . 1615
9.49.3.8 pack_sms_SLQSGetMessageWaiting . . . 1616
9.49.3.9 pack_sms_SLQSGetSMS . . . 1616
9.49.3.10 pack_sms_SLQSGetSmsBroadcastConfig . . . 1617
9.49.3.11 pack_sms_SLQSGetSMSList . . . 1617
9.49.3.12 pack_sms_SLQSGetTransLayerInfo . . . 1617
9.49.3.13 pack_sms_SLQSGetTransNWRegInfo . . . 1618
9.49.3.14 pack_sms_SLQSModifySMSStatus . . . 1618
9.49.3.15 pack_sms_SLQSSendAsyncSMS . . . 1618
9.49.3.16 pack_sms_SLQSSetIndicationRegister . . . 1619
9.49.3.17 pack_sms_SLQSSetSmsBroadcastActivation . . . 1619
9.49.3.18 pack_sms_SLQSSetSmsBroadcastConfig . . . 1619
9.49.3.19 pack_sms_SLQSSetSmsStorage . . . 1620
9.49.3.20 pack_sms_SLQSSmsGetMaxStorageSize . . . 1620
9.49.3.21 pack_sms_SLQSSmsGetMessageProtocol . . . 1621
9.49.3.22 pack_sms_SLQSSmsSetRoutes . . . 1621
9.49.3.23 pack_sms_SLQSSwiGetSMSStorage . . . 1621
9.49.3.24 unpack_sms_GetSMSCAddress . . . 1622
9.49.3.25 unpack_sms_SaveSMS . . . 1622
9.49.3.26 unpack_sms_SendSMS . . . 1622
9.49.3.27 unpack_sms_SetNewSMSCallback . . . 1623
9.49.3.28 unpack_sms_SetNewSMSCallback_ind . . . 1623
9.49.3.29 unpack_sms_SetSMSCAddress . . . 1623
9.49.3.30 unpack_sms_SLQSDeleteSMS . . . 1624
9.49.3.31 unpack_sms_SLQSGetIndicationRegister . . . 1624
9.49.3.32 unpack_sms_SLQSGetMessageWaiting . . . 1624
9.49.3.33 unpack_sms_SLQSGetSMS . . . 1625
9.49.3.34 unpack_sms_SLQSGetSmsBroadcastConfig . . . 1625
9.49.3.35 unpack_sms_SLQSGetSMSList . . . 1625
9.49.3.36 unpack_sms_SLQSGetTransLayerInfo . . . 1626
9.49.3.37 unpack_sms_SLQSGetTransNWRegInfo . . . 1626
9.49.3.38 unpack_sms_SLQSModifySMSStatus . . . 1626
9.49.3.39 unpack_sms_SLQSNWRegInfoCallback_ind . . . 1627
9.49.3.40 unpack_sms_SLQSSendAsyncSMS . . . 1627
9.49.3.41 unpack_sms_SLQSSetIndicationRegister . . . 1627
9.49.3.42 unpack_sms_SLQSSetSmsBroadcastActivation . . . 1628
9.49.3.43 unpack_sms_SLQSSetSmsBroadcastConfig . . . 1628

9.49.3.44 unpack_sms_SLQSSetSmsStorage . . . 1628
9.49.3.45 unpack_sms_SLQSSmsGetMaxStorageSize . . . 1629
9.49.3.46 unpack_sms_SLQSSmsGetMessageProtocol . . . 1629
9.49.3.47 unpack_sms_SLQSSmsSetRoutes . . . 1629
9.49.3.48 unpack_sms_SLQSSwiGetSMSStorage . . . 1630
9.49.3.49 unpack_sms_SLQSTransLayerInfoCallback_ind . . . 1630
9.49.3.50 unpack_sms_SLQSWmsAsyncRawSendCallBack_ind . . . 1631
9.49.3.51 unpack_sms_SLQSWmsMemoryFullCallBack_ind . . . 1631
9.49.3.52 unpack_sms_SLQSWmsMessageWaitingCallBack_ind . . . 1631
9.50 swiaudio.h File Reference . . . 1632
9.50.1 Macro Definition Documentation . . . 1633
9.50.1.1 SWIAUDIO_MAX_LEN_IFACE_TABLE . . . 1633
9.50.2 Typedef Documentation . . . 1633
9.50.2.1 unpack_swiaudio_SLQSSetM2MAudioAVCFG_t . . . 1633
9.50.2.2 unpack_swiaudio_SLQSSetM2MAudioLPBK_t . . . 1633
9.50.2.3 unpack_swiaudio_SLQSSetM2MAudioNVDef_t . . . 1633
9.50.2.4 unpack_swiaudio_SLQSSetM2MAudioProfile_t . . . 1633
9.50.2.5 unpack_swiaudio_SLQSSetM2MAudioVolume_t . . . 1633
9.50.2.6 unpack_swiaudio_SLQSSetM2MAVMute_t . . . 1633
9.50.2.7 unpack_swiaudio_SLQSSetM2MSpkrGain_t . . . 1633
9.50.3 Function Documentation . . . 1633
9.50.3.1 pack_swiaudio_SLQSGetM2MAudioProfile . . . 1634
9.50.3.2 pack_swiaudio_SLQSGetM2MAudioVolume . . . 1634
9.50.3.3 pack_swiaudio_SLQSGetM2MAVMute . . . 1634
9.50.3.4 pack_swiaudio_SLQSGetM2MSpkrGain . . . 1635
9.50.3.5 pack_swiaudio_SLQSSetM2MAudioAVCFG . . . 1635
9.50.3.6 pack_swiaudio_SLQSSetM2MAudioLPBK . . . 1635
9.50.3.7 pack_swiaudio_SLQSSetM2MAudioNVDef . . . 1636
9.50.3.8 pack_swiaudio_SLQSSetM2MAudioProfile . . . 1636
9.50.3.9 pack_swiaudio_SLQSSetM2MAudioVolume . . . 1637
9.50.3.10 pack_swiaudio_SLQSSetM2MAVMute . . . 1637
9.50.3.11 pack_swiaudio_SLQSSetM2MSpkrGain . . . 1637
9.50.3.12 unpack_swiaudio_SLQSGetM2MAudioProfile . . . 1638
9.50.3.13 unpack_swiaudio_SLQSGetM2MAudioVolume . . . 1638
9.50.3.14 unpack_swiaudio_SLQSGetM2MAVMute . . . 1638
9.50.3.15 unpack_swiaudio_SLQSGetM2MSpkrGain . . . 1639
9.50.3.16 unpack_swiaudio_SLQSSetM2MAudioAVCFG . . . 1639
9.50.3.17 unpack_swiaudio_SLQSSetM2MAudioLPBK . . . 1639
9.50.3.18 unpack_swiaudio_SLQSSetM2MAudioNVDef . . . 1640
9.50.3.19 unpack_swiaudio_SLQSSetM2MAudioProfile . . . 1640

9.50.3.20 unpack_swiaudio_SLQSSetM2MAudioVolume . . . 1640
9.50.3.21 unpack_swiaudio_SLQSSetM2MAVMute . . . 1641
9.50.3.22 unpack_swiaudio_SLQSSetM2MSpkrGain . . . 1641
9.51 swiavms.h File Reference . . . 1641
9.51.1 Macro Definition Documentation . . . 1643
9.51.1.1 LITEQMI_MAX_GET_SETTINGS_AVMS_APN_STRING_LENGTH . . . 1643
9.51.1.2 LITEQMI_MAX_GET_SETTINGS_AVMS_PWD_STRING_LENGTH . . . 1643
9.51.1.3 LITEQMI_MAX_GET_SETTINGS_AVMS_UNAME_STRING_LENGTH . . . 1643
9.51.1.4 LITEQMI_MAX_SWIOMA_STR_LEN . . . 1643
9.51.1.5 MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER . . . 1643
9.51.1.6 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH . . . 1643
9.51.1.7 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH . . . 1644
9.51.1.8 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH . . . 1644
9.51.1.9 MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH . . . 1644
9.51.1.10 MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH . . . 1644
9.51.1.11 MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS . 1644
9.51.1.12 MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH . . . 1644
9.51.1.13 MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH . . . 1644
9.51.2 Typedef Documentation . . . 1644
9.51.2.1 unpack_swiavms_SLQSAVMSStopSession_avc2_t . . . 1644
9.51.3 Function Documentation . . . 1644
9.51.3.1 pack_swiavms_SLQSAVMSGetSettings . . . 1644
9.51.3.2 pack_swiavms_SLQSAVMSGetSettings_v2 . . . 1644
9.51.3.3 pack_swiavms_SLQSAVMSSendSelection . . . 1645
9.51.3.4 pack_swiavms_SLQSAVMSSessionGetInfo . . . 1646
9.51.3.5 pack_swiavms_SLQSAvmsSetEventReport . . . 1646
9.51.3.6 pack_swiavms_SLQSAVMSSetSettings . . . 1647
9.51.3.7 pack_swiavms_SLQSAVMSSetSettings_v2 . . . 1647
9.51.3.8 pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField . . . 1648
9.51.3.9 pack_swiavms_SLQSAVMSStartSession . . . 1649
9.51.3.10 pack_swiavms_SLQSAVMSStopSession . . . 1649
9.51.3.11 unpack_swiavms_SLQSAVMSEventReportInd . . . 1650
9.51.3.12 unpack_swiavms_SLQSAVMSGetSettings . . . 1651
9.51.3.13 unpack_swiavms_SLQSAVMSGetSettings_v2 . . . 1651
9.51.3.14 unpack_swiavms_SLQSAVMSSendSelection . . . 1651
9.51.3.15 unpack_swiavms_SLQSAVMSSessionGetInfo . . . 1652
9.51.3.16 unpack_swiavms_SLQSAvmsSetEventReport . . . 1652

9.51.3.17 unpack_swiavms_SLQSAVMSSetSettings . . . 1653
9.51.3.18 unpack_swiavms_SLQSAVMSSetSettings_v2 . . . 1653
9.51.3.19 unpack_swiavms_SLQSAVMSStartSession . . . 1654
9.51.3.20 unpack_swiavms_SLQSAVMSStopSession . . . 1654
9.51.3.21 unpack_swiavms_SLQSAVMSStopSession_avc2 . . . 1655
9.52 SwiDataTypes.h File Reference . . . 1655
9.52.1 Detailed Description . . . 1656
9.52.2 Macro Definition Documentation . . . 1656
9.52.2.1 MEMSET_STATIC_OUTPUT_STRUCT . . . 1656
9.52.2.2 QMI_NO_LTE_FW_SUPPORT . . . 1656
9.52.2.3 QMI_TLV_PLACEHOLDER . . . 1656
9.52.2.4 SWI_API . . . 1656
9.52.2.5 UNUSEDPARAM . . . 1656
9.52.3 Typedef Documentation . . . 1656
9.52.3.1 BOOL . . . 1656
9.52.3.2 BYTE . . . 1656
9.52.3.3 CHAR . . . 1656
9.52.3.4 FLOAT . . . 1656
9.52.3.5 INT32 . . . 1656
9.52.3.6 INT8 . . . 1656
9.52.3.7 LPCSTR . . . 1657
9.52.3.8 qmuint16 . . . 1657
9.52.3.9 qmulong . . . 1657
9.52.3.10 SHORT . . . 1657
9.52.3.11 ULONG . . . 1657
9.52.3.12 ULONGLONG . . . 1657
9.52.3.13 USHORT . . . 1657
9.52.3.14 WORD . . . 1657
9.53 swidms.h File Reference . . . 1657
9.53.1 Typedef Documentation . . . 1658
9.53.1.1 unpack_swidms_SLQSSwiDmsSetUsbNetNum_t . . . 1658
9.53.2 Function Documentation . . . 1658
9.53.2.1 pack_swidms_SLQSSwiDmsGetHWWatchdog . . . 1658
9.53.2.2 pack_swidms_SLQSSwiDmsGetMTU . . . 1659
9.53.2.3 pack_swidms_SLQSSwiDmsGetSecureInfo . . . 1659
9.53.2.4 pack_swidms_SLQSSwiDmsGetUimSlot . . . 1659
9.53.2.5 pack_swidms_SLQSSwiDmsGetUsbComp . . . 1660
9.53.2.6 pack_swidms_SLQSSwiDmsGetUsbNetNum . . . 1660
9.53.2.7 pack_swidms_SLQSSwiDmsSetHWWatchdog . . . 1660
9.53.2.8 pack_swidms_SLQSSwiDmsSetMTU . . . 1661

9.53.2.9 pack_swidms_SLQSSwiDmsSetUimSlot . . . 1661
9.53.2.10 pack_swidms_SLQSSwiDmsSetUsbComp . . . 1661
9.53.2.11 pack_swidms_SLQSSwiDmsSetUsbNetNum . . . 1662
9.53.2.12 unpack_swidms_SLQSSwiDmsGetHWWatchdog . . . 1662
9.53.2.13 unpack_swidms_SLQSSwiDmsGetMTU . . . 1663
9.53.2.14 unpack_swidms_SLQSSwiDmsGetSecureInfo . . . 1663
9.53.2.15 unpack_swidms_SLQSSwiDmsGetUimSlot . . . 1663
9.53.2.16 unpack_swidms_SLQSSwiDmsGetUsbComp . . . 1664
9.53.2.17 unpack_swidms_SLQSSwiDmsGetUsbNetNum . . . 1664
9.53.2.18 unpack_swidms_SLQSSwiDmsSetHWWatchdog . . . 1664
9.53.2.19 unpack_swidms_SLQSSwiDmsSetMTU . . . 1665
9.53.2.20 unpack_swidms_SLQSSwiDmsSetUimSlot . . . 1665
9.53.2.21 unpack_swidms_SLQSSwiDmsSetUsbComp . . . 1665
9.53.2.22 unpack_swidms_SLQSSwiDmsSetUsbNetNum . . . 1666
9.54 swiloc.h File Reference . . . 1666
9.54.1 Typedef Documentation . . . 1666
9.54.1.1 unpack_swiloc_SwiLocSetAutoStart_t . . . 1666
9.54.2 Function Documentation . . . 1667
9.54.2.1 pack_swiloc_SwiLocGetAutoStart . . . 1667
9.54.2.2 pack_swiloc_SwiLocSetAutoStart . . . 1667
9.54.2.3 unpack_swiloc_SwiLocGetAutoStart . . . 1667
9.54.2.4 unpack_swiloc_SwiLocSetAutoStart . . . 1668
9.55 swioma.h File Reference . . . 1668
9.55.1 Macro Definition Documentation . . . 1669
9.55.1.1 LITEQMI_MAX_SWIOMA_STR_LEN . . . 1669
9.55.2 Typedef Documentation . . . 1669
9.55.2.1 unpack_swioma_SLQSOMADMAlertCallback_t . . . 1669
9.55.2.2 unpack_swioma_SLQSOMADMCancelSession_t . . . 1669
9.55.2.3 unpack_swioma_SLQSOMADMSendSelection_t . . . 1669
9.55.2.4 unpack_swioma_SLQSOMADMSetSettings_t . . . 1669
9.55.3 Function Documentation . . . 1669
9.55.3.1 pack_swioma_SLQSOMADMAlertCallback . . . 1669
9.55.3.2 pack_swioma_SLQSOMADMCancelSession . . . 1670
9.55.3.3 pack_swioma_SLQSOMADMGetSessionInfo . . . 1671
9.55.3.4 pack_swioma_SLQSOMADMGetSettings . . . 1671
9.55.3.5 pack_swioma_SLQSOMADMSendSelection . . . 1672
9.55.3.6 pack_swioma_SLQSOMADMSetSettings . . . 1673
9.55.3.7 pack_swioma_SLQSOMADMStartSession . . . 1673
9.55.3.8 unpack_swioma_SLQSOMADMAlertCallback . . . 1674
9.55.3.9 unpack_swioma_SLQSOMADMAlertCallback_ind . . . 1674

9.55.3.10 unpack_swioma_SLQSOMADMCancelSession . . . 1675
9.55.3.11 unpack_swioma_SLQSOMADMGetSessionInfo . . . 1675
9.55.3.12 unpack_swioma_SLQSOMADMGetSettings . . . 1676
9.55.3.13 unpack_swioma_SLQSOMADMSendSelection . . . 1676
9.55.3.14 unpack_swioma_SLQSOMADMSetSettings . . . 1677
9.55.3.15 unpack_swioma_SLQSOMADMStartSession . . . 1677
9.56 swiomaext.h File Reference . . . 1678
9.56.1 Macro Definition Documentation . . . 1678
9.56.1.1 LITE_SWIOMAEXT_MAX_UCS2_DATA_LEN . . . 1678
9.56.1.2 LITEQMI_MAX_SWIOMA_STR_LEN . . . 1678
9.56.2 Typedef Documentation . . . 1678
9.56.2.1 unpack_swioma_SLQSOMADMCancelSessionExt_t . . . 1679
9.56.2.2 unpack_swioma_SLQSOMADMSendSelectionExt_t . . . 1679
9.56.2.3 unpack_swioma_SLQSOMADMSetSettingsExt_t . . . 1679
9.56.2.4 unpack_swioma_SLQSOMADMStartSessionExt_t . . . 1679
9.56.3 Function Documentation . . . 1679
9.56.3.1 pack_swioma_SLQSOMADMCancelSessionExt . . . 1679
9.56.3.2 pack_swioma_SLQSOMADMGetSessionInfoExt . . . 1679
9.56.3.3 pack_swioma_SLQSOMADMSendSelectionExt . . . 1680
9.56.3.4 pack_swioma_SLQSOMADMSetSettingsExt . . . 1681
9.56.3.5 pack_swioma_SLQSOMADMStartSessionExt . . . 1681
9.56.3.6 unpack_swioma_SLQSOMADMCancelSessionExt . . . 1682
9.56.3.7 unpack_swioma_SLQSOMADMGetSessionInfoExt . . . 1682
9.56.3.8 unpack_swioma_SLQSOMADMSendSelectionExt . . . 1683
9.56.3.9 unpack_swioma_SLQSOMADMSetSettingsExt . . . 1683
9.56.3.10 unpack_swioma_SLQSOMADMStartSessionExt . . . 1684
9.57 switype_256bit.h File Reference . . . 1684
9.57.1 Macro Definition Documentation . . . 1685
9.57.1.1 SWI_UINT256_BIT_OFFSET . . . 1685
9.57.1.2 SWI_UINT256_BITS_PER_WORD . . . 1685
9.57.1.3 SWI_UINT256_INT_VALUE . . . 1685
9.57.1.4 SWI_UINT256_WORD_COUNT . . . 1685
9.57.1.5 SWI_UINT256_WORD_OFFSET . . . 1685
9.57.2 Typedef Documentation . . . 1685
9.57.2.1 logger . . . 1685
9.57.3 Function Documentation . . . 1685
9.57.3.1 swi_uint256_clear_bit . . . 1685
9.57.3.2 swi_uint256_get_bit . . . 1685
9.57.3.3 swi_uint256_print_mask . . . 1685
9.57.3.4 swi_uint256_set_bit . . . 1685

9.58 SWIWWANCMAPI.h File Reference . . . 1686
9.59 tmd.h File Reference . . . 1686
9.59.1 Macro Definition Documentation . . . 1686
9.59.1.1 MAX_MITIGATION_DEV_ID_LEN . . . 1686
9.59.1.2 MAX_MITIGATION_DEV_LIST_LEN . . . 1686
9.59.1.3 TMD_MAX_DEV_LIST . . . 1687
9.59.2 Function Documentation . . . 1687
9.59.2.1 pack_tmd_SLQSTmdDeRegNotMitigationLvl . . . 1687
9.59.2.2 pack_tmd_SLQSTmdGetMitigationDevList . . . 1687
9.59.2.3 pack_tmd_SLQSTmdGetMitigationLvl . . . 1687
9.59.2.4 pack_tmd_SLQSTmdRegNotMitigationLvl . . . 1688
9.59.2.5 unpack_tmd_SLQSTmdDeRegNotMitigationLvl . . . 1688
9.59.2.6 unpack_tmd_SLQSTmdGetMitigationDevList . . . 1688
9.59.2.7 unpack_tmd_SLQSTmdGetMitigationLvl . . . 1689
9.59.2.8 unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind . . . 1689
9.59.2.9 unpack_tmd_SLQSTmdRegNotMitigationLvl . . . 1690
9.60 uim.h File Reference . . . 1690
9.60.1 Macro Definition Documentation . . . 1695
9.60.1.1 MAX_ATR_LENGTH . . . 1695
9.60.1.2 MAX_DESCRIPTION_LENGTH . . . 1695
9.60.1.3 MAX_ICCID_LENGTH . . . 1695
9.60.1.4 MAX_NO_OF_APPLICATIONS . . . 1695
9.60.1.5 MAX_NO_OF_SLOTS . . . 1695
9.60.1.6 MAX_PHY_SLOTS_INFO . . . 1695
9.60.1.7 MAX_SLOTS_STATUS . . . 1695
9.60.1.8 UIM_MAX_ACTIVE_PERS_FEATURES . . . 1695
9.60.1.9 UIM_MAX_CONTENT_LENGTH . . . 1695
9.60.1.10 UIM_MAX_DESCRIPTION_LENGTH . . . 1695
9.60.1.11 UIM_MAX_NO_OF_APPLICATIONS . . . 1696
9.60.1.12 UIM_MAX_NO_OF_SLOTS . . . 1696
9.60.1.13 UIM_UINT8_MAX_STRING_SZ . . . 1696
9.60.2 Typedef Documentation . . . 1696
9.60.2.1 unpack_uim_SLQSUIMAddProfile_t . . . 1696
9.60.2.2 unpack_uim_SLQSUIMDeleteProfile_t . . . 1696
9.60.2.3 unpack_uim_SLQSUIMeUICCMemReset_t . . . 1696
9.60.2.4 unpack_uim_SLQSUIMPowerDown_t . . . 1696
9.60.2.5 unpack_uim_SLQSUIMPowerUp_t . . . 1696
9.60.2.6 unpack_uim_SLQSUIMRefreshComplete_t . . . 1696
9.60.2.7 unpack_uim_SLQSUIMRefreshOK_t . . . 1696
9.60.2.8 unpack_uim_SLQSUIMRefreshRegister_t . . . 1696

9.60.2.9 unpack_uim_SLQSUIMReset_t . . . 1696
9.60.2.10 unpack_uim_SLQSUIMSetServiceStatus_t . . . 1696
9.60.2.11 unpack_uim_SLQSUIMSetSIMProfile_t . . . 1696
9.60.2.12 unpack_uim_SLQSUIMSwitchSlot_t . . . 1696
9.60.2.13 unpack_uim_SLQSUIMUpdateProfile_t . . . 1696
9.60.3 Function Documentation . . . 1696
9.60.3.1 pack_uim_ChangePin . . . 1696
9.60.3.2 pack_uim_GetCardStatus . . . 1697
9.60.3.3 pack_uim_ReadTransparent . . . 1697
9.60.3.4 pack_uim_SetPinProtection . . . 1697
9.60.3.5 pack_uim_SLQSUIMAddProfile . . . 1698
9.60.3.6 pack_uim_SLQSUIMAuthenticate . . . 1698
9.60.3.7 pack_uim_SLQSUIMDeleteProfile . . . 1698
9.60.3.8 pack_uim_SLQSUIMDepersonalization . . . 1699
9.60.3.9 pack_uim_SLQSUIMeUICCMemReset . . . 1699
9.60.3.10 pack_uim_SLQSUIMEventRegister . . . 1699
9.60.3.11 pack_uim_SLQSUIMGetConfiguration . . . 1700
9.60.3.12 pack_uim_SLQSUIMGetEID . . . 1700
9.60.3.13 pack_uim_SLQSUIMGetFileAttributes . . . 1701
9.60.3.14 pack_uim_SLQSUIMGetProfileInfo . . . 1701
9.60.3.15 pack_uim_SLQSUIMGetServiceStatus . . . 1701
9.60.3.16 pack_uim_SLQSUIMGetSIMProfile . . . 1702
9.60.3.17 pack_uim_SLQSUIMGetSlotsStatus . . . 1702
9.60.3.18 pack_uim_SLQSUIMPowerDown . . . 1702
9.60.3.19 pack_uim_SLQSUIMPowerUp . . . 1703
9.60.3.20 pack_uim_SLQSUIMReadRecord . . . 1703
9.60.3.21 pack_uim_SLQSUIMRefreshComplete . . . 1704
9.60.3.22 pack_uim_SLQSUIMRefreshGetLastEvent . . . 1704
9.60.3.23 pack_uim_SLQSUIMRefreshOK . . . 1704
9.60.3.24 pack_uim_SLQSUIMRefreshRegister . . . 1705
9.60.3.25 pack_uim_SLQSUIMReset . . . 1705
9.60.3.26 pack_uim_SLQSUIMSetServiceStatus . . . 1705
9.60.3.27 pack_uim_SLQSUIMSetSIMProfile . . . 1706
9.60.3.28 pack_uim_SLQSUIMSwitchSlot . . . 1706
9.60.3.29 pack_uim_SLQSUIMUpdateProfile . . . 1707
9.60.3.30 pack_uim_SLQSUIMWriteRecord . . . 1707
9.60.3.31 pack_uim_SLQSUIMWriteTransparent . . . 1707
9.60.3.32 pack_uim_UnblockPin . . . 1708
9.60.3.33 pack_uim_VerifyPin . . . 1708
9.60.3.34 unpack_uim_ChangePin . . . 1708

9.60.3.35 unpack_uim_GetCardStatus . . . 1709
9.60.3.36 unpack_uim_GetCardStatusV2 . . . 1709
9.60.3.37 unpack_uim_ReadTransparent . . . 1710
9.60.3.38 unpack_uim_SetPinProtection . . . 1710
9.60.3.39 unpack_uim_SetUimSlotStatusChangeCallback_ind . . . 1710
9.60.3.40 unpack_uim_SLQSUIMAddProfile . . . 1711
9.60.3.41 unpack_uim_SLQSUIMAuthenticate . . . 1711
9.60.3.42 unpack_uim_SLQSUIMDeleteProfile . . . 1711
9.60.3.43 unpack_uim_SLQSUIMDepersonalization . . . 1712
9.60.3.44 unpack_uim_SLQSUIMeUICCMemReset . . . 1712
9.60.3.45 unpack_uim_SLQSUIMEventRegister . . . 1712
9.60.3.46 unpack_uim_SLQSUIMGetConfiguration . . . 1713
9.60.3.47 unpack_uim_SLQSUIMGetEID . . . 1713
9.60.3.48 unpack_uim_SLQSUIMGetFileAttributes . . . 1713
9.60.3.49 unpack_uim_SLQSUIMGetProfileInfo . . . 1714
9.60.3.50 unpack_uim_SLQSUIMGetServiceStatus . . . 1714
9.60.3.51 unpack_uim_SLQSUIMGetSIMProfile . . . 1715
9.60.3.52 unpack_uim_SLQSUIMGetSlotsStatus . . . 1715
9.60.3.53 unpack_uim_SLQSUIMGetSlotsStatusV2 . . . 1715
9.60.3.54 unpack_uim_SLQSUIMPowerDown . . . 1716
9.60.3.55 unpack_uim_SLQSUIMPowerUp . . . 1716
9.60.3.56 unpack_uim_SLQSUIMReadRecord . . . 1716
9.60.3.57 unpack_uim_SLQSUIMRefreshCallback_Ind . . . 1717
9.60.3.58 unpack_uim_SLQSUIMRefreshComplete . . . 1717
9.60.3.59 unpack_uim_SLQSUIMRefreshGetLastEvent . . . 1717
9.60.3.60 unpack_uim_SLQSUIMRefreshOK . . . 1718
9.60.3.61 unpack_uim_SLQSUIMRefreshRegister . . . 1718
9.60.3.62 unpack_uim_SLQSUIMReset . . . 1718
9.60.3.63 unpack_uim_SLQSUIMSetServiceStatus . . . 1719
9.60.3.64 unpack_uim_SLQSUIMSetSIMProfile . . . 1719
9.60.3.65 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind . . . 1719
9.60.3.66 unpack_uim_SLQSUIMSwitchSlot . . . 1720
9.60.3.67 unpack_uim_SLQSUIMUpdateProfile . . . 1720
9.60.3.68 unpack_uim_SLQSUIMWriteRecord . . . 1720
9.60.3.69 unpack_uim_SLQSUIMWriteTransparent . . . 1721
9.60.3.70 unpack_uim_UnblockPin . . . 1721
9.60.3.71 unpack_uim_UnblockPinV2 . . . 1721
9.60.3.72 unpack_uim_VerifyPin . . . 1722
9.61 voice.h File Reference . . . 1722
9.61.1 Detailed Description . . . 1728

9.61.2 S1 . . . 1728
9.61.3 Macro Definition Documentation . . . 1728
9.61.3.1 BARRING_PASSWORD_LENGTH . . . 1728
9.61.3.2 MAX_VOICE_CALL_NO_LEN . . . 1728
9.61.3.3 MAX_VOICE_DESCRIPTION_LENGTH . . . 1728
9.61.3.4 MAXVOICEUSSDLENGTH . . . 1728
9.61.3.5 VOICE_MAX_NO_OF_CALLS . . . 1728
9.61.4 Typedef Documentation . . . 1728
9.61.4.1 unpack_voice_AnswerUSSD_t . . . 1728
9.61.4.2 unpack_voice_CancelUSSD_t . . . 1728
9.61.4.3 unpack_voice_OriginateUSSD_t . . . 1728
9.61.4.4 unpack_voice_SLQSVoiceALSSelectLine_t . . . 1728
9.61.4.5 unpack_voice_SLQSVoiceALSSetLineSwitching_t . . . 1728
9.61.4.6 unpack_voice_SLQSVoiceBindSubscription_t . . . 1728
9.61.4.7 unpack_voice_SLQSVoiceIndicationRegister_t . . . 1728
9.61.4.8 unpack_voice_SLQSVoiceOrigUSSDNoWait_t . . . 1728
9.61.4.9 unpack_voice_SLQSVoiceSetPreferredPrivacy_t . . . 1728
9.61.5 Enumeration Type Documentation . . . 1728
9.61.5.1 liteServiceClassInformation . . . 1728
9.61.6 Function Documentation . . . 1729
9.61.6.1 pack_voice_AnswerUSSD . . . 1729
9.61.6.2 pack_voice_CancelUSSD . . . 1729
9.61.6.3 pack_voice_OriginateUSSD . . . 1729
9.61.6.4 pack_voice_SLQSOriginateUSSD . . . 1730
9.61.6.5 pack_voice_SLQSVoiceALSSelectLine . . . 1730
9.61.6.6 pack_voice_SLQSVoiceALSSetLineSwitching . . . 1730
9.61.6.7 pack_voice_SLQSVoiceAnswerCall . . . 1731
9.61.6.8 pack_voice_SLQSVoiceBindSubscription . . . 1731
9.61.6.9 pack_voice_SLQSVoiceBurstDTMF . . . 1732
9.61.6.10 pack_voice_SLQSVoiceDialCall . . . 1732
9.61.6.11 pack_voice_SLQSVoiceEndCall . . . 1732
9.61.6.12 pack_voice_SLQSVoiceGetAllCallInfo . . . 1733
9.61.6.13 pack_voice_SLQSVoiceGetCallBarring . . . 1733
9.61.6.14 pack_voice_SLQSVoiceGetCallForwardingStatus . . . 1733
9.61.6.15 pack_voice_SLQSVoiceGetCallInfo . . . 1734
9.61.6.16 pack_voice_SLQSVoiceGetCallWaiting . . . 1734
9.61.6.17 pack_voice_SLQSVoiceGetCLIP . . . 1735
9.61.6.18 pack_voice_SLQSVoiceGetCLIR . . . 1735
9.61.6.19 pack_voice_SLQSVoiceGetCNAP . . . 1735
9.61.6.20 pack_voice_SLQSVoiceGetCOLP . . . 1736

9.61.6.21 pack_voice_SLQSVoiceGetCOLR . . . 1736
9.61.6.22 pack_voice_SLQSVoiceGetConfig . . . 1736
9.61.6.23 pack_voice_SLQSVoiceIndicationRegister . . . 1737
9.61.6.24 pack_voice_SLQSVoiceManageCalls . . . 1737
9.61.6.25 pack_voice_SLQSVoiceOrigUSSDNoWait . . . 1737
9.61.6.26 pack_voice_SLQSVoiceSendFlash . . . 1738
9.61.6.27 pack_voice_SLQSVoiceSetCallBarringPassword . . . 1738
9.61.6.28 pack_voice_SLQSVoiceSetConfig . . . 1738
9.61.6.29 pack_voice_SLQSVoiceSetPreferredPrivacy . . . 1739
9.61.6.30 pack_voice_SLQSVoiceSetSUPSService . . . 1739
9.61.6.31 pack_voice_SLQSVoiceStartContDTMF . . . 1740
9.61.6.32 pack_voice_SLQSVoiceStopContDTMF . . . 1740
9.61.6.33 unpack_voice_allCallStatusCallback_ind . . . 1740
9.61.6.34 unpack_voice_AnswerUSSD . . . 1741
9.61.6.35 unpack_voice_CancelUSSD . . . 1741
9.61.6.36 unpack_voice_DTMFEventCallback_ind . . . 1741
9.61.6.37 unpack_voice_OriginateUSSD . . . 1742
9.61.6.38 unpack_voice_OTASPStatusCallback_ind . . . 1742
9.61.6.39 unpack_voice_SLQSOriginateUSSD . . . 1742
9.61.6.40 unpack_voice_SLQSVoiceALSSelectLine . . . 1743
9.61.6.41 unpack_voice_SLQSVoiceALSSetLineSwitching . . . 1743
9.61.6.42 unpack_voice_SLQSVoiceAnswerCall . . . 1743
9.61.6.43 unpack_voice_SLQSVoiceBindSubscription . . . 1744
9.61.6.44 unpack_voice_SLQSVoiceBurstDTMF . . . 1744
9.61.6.45 unpack_voice_SLQSVoiceDialCall . . . 1744
9.61.6.46 unpack_voice_SLQSVoiceEndCall . . . 1745
9.61.6.47 unpack_voice_SLQSVoiceGetAllCallInfo . . . 1745
9.61.6.48 unpack_voice_SLQSVoiceGetCallBarring . . . 1745
9.61.6.49 unpack_voice_SLQSVoiceGetCallForwardingStatus . . . 1746
9.61.6.50 unpack_voice_SLQSVoiceGetCallInfo . . . 1746
9.61.6.51 unpack_voice_SLQSVoiceGetCallWaiting . . . 1746
9.61.6.52 unpack_voice_SLQSVoiceGetCLIP . . . 1747
9.61.6.53 unpack_voice_SLQSVoiceGetCLIR . . . 1747
9.61.6.54 unpack_voice_SLQSVoiceGetCNAP . . . 1747
9.61.6.55 unpack_voice_SLQSVoiceGetCOLP . . . 1748
9.61.6.56 unpack_voice_SLQSVoiceGetCOLR . . . 1748
9.61.6.57 unpack_voice_SLQSVoiceGetConfig . . . 1748
9.61.6.58 unpack_voice_SLQSVoiceIndicationRegister . . . 1749
9.61.6.59 unpack_voice_SLQSVoiceManageCalls . . . 1749
9.61.6.60 unpack_voice_SLQSVoiceOrigUSSDNoWait . . . 1749

9.61.6.61 unpack_voice_SLQSVoiceSendFlash . . . 1750
9.61.6.62 unpack_voice_SLQSVoiceSetCallBarringPassword . . . 1750
9.61.6.63 unpack_voice_SLQSVoiceSetConfig . . . 1750
9.61.6.64 unpack_voice_SLQSVoiceSetPreferredPrivacy . . . 1751
9.61.6.65 unpack_voice_SLQSVoiceSetSUPSService . . . 1751
9.61.6.66 unpack_voice_SLQSVoiceStartContDTMF . . . 1751
9.61.6.67 unpack_voice_SLQSVoiceStopContDTMF . . . 1752
9.61.6.68 unpack_voice_SLQSVoiceSUPSCallback_ind . . . 1752
9.61.6.69 unpack_voice_SUPSNotificationCallback_ind . . . 1752
9.61.6.70 unpack_voice_USSDNotificationCallback_ind . . . 1753
9.61.6.71 unpack_voice_USSDNoWaitIndicationCallback_ind . . . 1753
9.61.6.72 unpack_voice_VoiceInfoRecCallback_ind . . . 1754
9.61.6.73 unpack_voice_voicePrivacyChangeCallback_ind . . . 1754
9.62 wda.h File Reference . . . 1754
9.62.1 Function Documentation . . . 1755
9.62.1.1 helper_get_wda_resp_ctx . . . 1755
9.62.1.2 pack_wda_GetDataFormat . . . 1755
9.62.1.3 pack_wda_GetQmapSettings . . . 1755
9.62.1.4 pack_wda_SetDataFormat . . . 1756
9.62.1.5 pack_wda_SetQmapSettings . . . 1756
9.62.1.6 unpack_wda_GetDataFormat . . . 1756
9.62.1.7 unpack_wda_GetQmapSettings . . . 1757
9.62.1.8 unpack_wda_SetDataFormat . . . 1757
9.62.1.9 unpack_wda_SetQmapSettings . . . 1758
9.63 wds.h File Reference . . . 1758
9.63.1 Detailed Description . . . 1765
9.63.2 SO Mask . . . 1765
9.63.3 RAT Mask . . . 1766
9.63.4 Macro Definition Documentation . . . 1766
9.63.4.1 BYT_STAT_STAT_MASK . . . 1766
9.63.4.2 IPV6_ADDRESS_ARRAY_SIZE . . . 1766
9.63.4.3 LITE_MAX_PCOID_LIST . . . 1766
9.63.4.4 LITE_MAX_PDN_THROTTLE_TIMER . . . 1766
9.63.4.5 MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE . . . 1767
9.63.4.6 MAX_WDS_NAME_ARRAY_SIZE . . . 1767
9.63.4.7 PACK_WDS_IPV4 . . . 1767
9.63.4.8 PACK_WDS_IPV6 . . . 1767
9.63.4.9 WDS_DHCP_MAX_NUM_OPTIONS . . . 1767
9.63.4.10 WDS_DHCP_OPTION_DATA_BUF_SIZE . . . 1767
9.63.4.11 WDS_PROFILE_3GPP . . . 1767

9.63.4.12 WDS_PROFILE_3GPP2 . . . 1767
9.63.4.13 WDS_TFTID_SOURCE_IP_SIZE . . . 1767
9.63.5 Typedef Documentation . . . 1767
9.63.5.1 unpack_wds_DHCPv4ClientLeaseChange_t . . . 1767
9.63.5.2 unpack_wds_RMTransferStatistics_ind_t . . . 1767
9.63.5.3 unpack_wds_SetAutoconnect_t . . . 1767
9.63.5.4 unpack_wds_SetDefaultProfile_t . . . 1767
9.63.5.5 unpack_wds_SetDefaultProfileNum_t . . . 1767
9.63.5.6 unpack_wds_SetMobileIP_t . . . 1767
9.63.5.7 unpack_wds_SetMobileIPParameters_t . . . 1767
9.63.5.8 unpack_wds_SetMuxID_t . . . 1767
9.63.5.9 unpack_wds_SLQSResetPacketStatics_t . . . 1767
9.63.5.10 unpack_wds_SLQSSet3GPPConfigItem_t . . . 1767
9.63.5.11 unpack_wds_SLQSSetWdsEventCallback_t . . . 1767
9.63.5.12 unpack_wds_SLQSSSetDHCPv4ClientConfig_t . . . 1767
9.63.5.13 unpack_wds_SLQSSSetLoopback_t . . . 1767
9.63.5.14 unpack_wds_SLQSStopDataSession_t . . . 1767
9.63.5.15 unpack_wds_SLQSSwiProfileChangeCallback_t . . . 1767
9.63.5.16 unpack_wds_SLQSWdsGoActive_t . . . 1767
9.63.5.17 unpack_wds_SLQSWdsGoDormant_t . . . 1767
9.63.5.18 unpack_wds_SLQSWdsSetEventReport_t . . . 1767
9.63.5.19 UnpackQmiProfileInfo . . . 1768
9.63.5.20 UnpackQmiProfileInfoV2 . . . 1768
9.63.6 Enumeration Type Documentation . . . 1768
9.63.6.1 liteQmiDataBearerMasks . . . 1768
9.63.7 Function Documentation . . . 1768
9.63.7.1 pack_wds_DHCPv4ClientLeaseChange . . . 1768
9.63.7.2 pack_wds_GetAutoconnect . . . 1769
9.63.7.3 pack_wds_GetByteTotals . . . 1769
9.63.7.4 pack_wds_GetConnectionRate . . . 1770
9.63.7.5 pack_wds_GetDataBearerTechnology . . . 1770
9.63.7.6 pack_wds_GetDefaultProfile . . . 1770
9.63.7.7 pack_wds_GetDefaultProfileNum . . . 1771
9.63.7.8 pack_wds_GetDefaultProfileV2 . . . 1771
9.63.7.9 pack_wds_GetDormancyState . . . 1772
9.63.7.10 pack_wds_GetLastMobileIPError . . . 1772
9.63.7.11 pack_wds_GetMobileIP . . . 1772
9.63.7.12 pack_wds_GetMobileIPProfile . . . 1773
9.63.7.13 pack_wds_GetPacketStatistics . . . 1773
9.63.7.14 pack_wds_GetPacketStatus . . . 1774

9.63.7.15 pack_wds_GetSessionDuration . . . 1774
9.63.7.16 pack_wds_GetSessionDurationV2 . . . 1775
9.63.7.17 pack_wds_GetSessionState . . . 1775
9.63.7.18 pack_wds_RMSetTransferStatistics . . . 1775
9.63.7.19 pack_wds_SetAutoconnect . . . 1776
9.63.7.20 pack_wds_SetDefaultProfile . . . 1776
9.63.7.21 pack_wds_SetDefaultProfileNum . . . 1777
9.63.7.22 pack_wds_SetMobileIP . . . 1777
9.63.7.23 pack_wds_SetMobileIPParameters . . . 1777
9.63.7.24 pack_wds_SetMobileIPProfile . . . 1778
9.63.7.25 pack_wds_SetMuxID . . . 1778
9.63.7.26 pack_wds_SLQSCreateProfile . . . 1779
9.63.7.27 pack_wds_SLQSDeleteProfile . . . 1779
9.63.7.28 pack_wds_SLQSGet3GPPConfigItem . . . 1779
9.63.7.29 pack_wds_SLQSGetCurrDataSystemStat . . . 1780
9.63.7.30 pack_wds_SLQSGetCurrentChannelRate . . . 1780
9.63.7.31 pack_wds_SLQSGetDataBearerTechnology . . . 1781
9.63.7.32 pack_wds_SLQSGetDUNCallInfo . . . 1781
9.63.7.33 pack_wds_SLQSGetProfileSettings . . . 1782
9.63.7.34 pack_wds_SLQSGetProfileSettingsV2 . . . 1782
9.63.7.35 pack_wds_SLQSGetRuntimeSettings . . . 1783
9.63.7.36 pack_wds_SLQSModifyProfile . . . 1783
9.63.7.37 pack_wds_SLQSResetPacketStatics . . . 1783
9.63.7.38 pack_wds_SLQSSet3GPPConfigItem . . . 1784
9.63.7.39 pack_wds_SLQSSetIPFamilyPreference . . . 1784
9.63.7.40 pack_wds_SLQSSetWdsEventCallback . . . 1785
9.63.7.41 pack_wds_SLQSSGetDHCPv4ClientConfig . . . 1785
9.63.7.42 pack_wds_SLQSSGetLoopback . . . 1785
9.63.7.43 pack_wds_SLQSSSetDHCPv4ClientConfig . . . 1786
9.63.7.44 pack_wds_SLQSSSetLoopback . . . 1786
9.63.7.45 pack_wds_SLQSStartDataSession . . . 1787
9.63.7.46 pack_wds_SLQSStopDataSession . . . 1787
9.63.7.47 pack_wds_SLQSSwiProfileChangeCallback . . . 1787
9.63.7.48 pack_wds_SLQSWdsGoActive . . . 1788
9.63.7.49 pack_wds_SLQSWdsGoDormant . . . 1788
9.63.7.50 pack_wds_SLQSWdsSetEventReport . . . 1789
9.63.7.51 pack_wds_SLQSWdsSwiPDPRuntimeSettings . . . 1789
9.63.7.52 unpack_wds_DHCPv4ClientLease_ind . . . 1790
9.63.7.53 unpack_wds_DHCPv4ClientLeaseChange . . . 1790
9.63.7.54 unpack_wds_GetAutoconnect . . . 1790

9.63.7.55 unpack_wds_GetByteTotals . . . 1791
9.63.7.56 unpack_wds_GetConnectionRate . . . 1791
9.63.7.57 unpack_wds_GetDataBearerTechnology . . . 1791
9.63.7.58 unpack_wds_GetDefaultProfile . . . 1792
9.63.7.59 unpack_wds_GetDefaultProfileNum . . . 1792
9.63.7.60 unpack_wds_GetDefaultProfileV2 . . . 1792
9.63.7.61 unpack_wds_GetDormancyState . . . 1793
9.63.7.62 unpack_wds_GetLastMobileIPError . . . 1793
9.63.7.63 unpack_wds_GetMobileIP . . . 1793
9.63.7.64 unpack_wds_GetMobileIPProfile . . . 1794
9.63.7.65 unpack_wds_GetPacketStatistics . . . 1794
9.63.7.66 unpack_wds_GetPacketStatus . . . 1794
9.63.7.67 unpack_wds_GetSessionDuration . . . 1795
9.63.7.68 unpack_wds_GetSessionDurationV2 . . . 1795
9.63.7.69 unpack_wds_GetSessionState . . . 1795
9.63.7.70 unpack_wds_RMSetTransferStatistics . . . 1796
9.63.7.71 unpack_wds_RMTransferStatistics_ind . . . 1796
9.63.7.72 unpack_wds_SetAutoconnect . . . 1796
9.63.7.73 unpack_wds_SetDefaultProfile . . . 1797
9.63.7.74 unpack_wds_SetDefaultProfileNum . . . 1797
9.63.7.75 unpack_wds_SetMobileIP . . . 1797
9.63.7.76 unpack_wds_SetMobileIPParameters . . . 1798
9.63.7.77 unpack_wds_SetMobileIPProfile . . . 1798
9.63.7.78 unpack_wds_SetMuxID . . . 1798
9.63.7.79 unpack_wds_SLQSCreateProfile . . . 1799
9.63.7.80 unpack_wds_SLQSDeleteProfile . . . 1799
9.63.7.81 unpack_wds_SLQSDUNCallInfoCallBack_ind . . . 1799
9.63.7.82 unpack_wds_SLQSGet3GPPConfigItem . . . 1800
9.63.7.83 unpack_wds_SLQSGetCurrDataSystemStat . . . 1800
9.63.7.84 unpack_wds_SLQSGetCurrentChannelRate . . . 1800
9.63.7.85 unpack_wds_SLQSGetDataBearerTechnology . . . 1801
9.63.7.86 unpack_wds_SLQSGetDUNCallInfo . . . 1801
9.63.7.87 unpack_wds_SLQSGetProfileSettings . . . 1801
9.63.7.88 unpack_wds_SLQSGetProfileSettingsV2 . . . 1802
9.63.7.89 unpack_wds_SLQSGetRuntimeSettings . . . 1802
9.63.7.90 unpack_wds_SLQSModifyProfile . . . 1802
9.63.7.91 unpack_wds_SLQSResetPacketStatics . . . 1803
9.63.7.92 unpack_wds_SLQSSet3GPPConfigItem . . . 1803
9.63.7.93 unpack_wds_SLQSSetIPFamilyPreference . . . 1803
9.63.7.94 unpack_wds_SLQSSetPacketSrvStatusCallback . . . 1804

9.63.7.95 unpack_wds_SLQSSetWdsEventCallback 1804
9.63.7.96 unpack_wds_SLQSSetWdsEventCallback_ind 1804
9.63.7.97 unpack_wds_SLQSSGetDHCPv4ClientConfig 1805
9.63.7.98 unpack_wds_SLQSSGetLoopback 1805
9.63.7.99 unpack_wds_SLQSSSetDHCPv4ClientConfig 1805
9.63.7.100 unpack_wds_SLQSSSetLoopback 1806
9.63.7.101 unpack_wds_SLQSStartDataSession 1806
9.63.7.102 unpack_wds_SLQSStopDataSession 1806
9.63.7.103 unpack_wds_SLQSSwiProfileChangeCallback 1807
9.63.7.104 unpack_wds_SLQSSwiProfileChangeCallback_Ind 1807
9.63.7.105 unpack_wds_SLQSWdsGoActive 1807
9.63.7.106 unpack_wds_SLQSWdsGoDormant 1808
9.63.7.107 unpack_wds_SLQSWdsSetEventReport 1808
9.63.7.108 unpack_wds_SLQSWdsSwiPDPRuntimeSettings 1808

Index **1810**

Chapter 1

Welcome to the Sierra Wireless Linux QMI SDK API Reference Guide

This API reference guide contains information about all the modules, in the Sierra Wireless Linux QMI SDK (SLQS). Use the tabs at the top of the page to navigate the reference guide.

- Modules tab – lists all the service modules and provides a link to the API header file in each module.
- References tab – links to reference material.

1.1 Important Notice

Due to the nature of wireless communications, transmission and reception of data can never be guaranteed. Data may be delayed, corrupted (i.e., have errors) or be totally lost. Although significant delays or losses of data are rare when wireless devices such as the Sierra Wireless modem are used in a normal manner with a well-constructed network, the Sierra Wireless modem should not be used in situations where failure to transmit or receive data could result in damage of any kind to the user or any other party, including but not limited to personal injury, death, or loss of property. Sierra Wireless accepts no responsibility for damages of any kind resulting from delays or errors in data transmitted or received using the Sierra Wireless modem, or for failure of the Sierra Wireless modem to transmit or receive such data.

1.2 Limitation of Liability

The information in this manual is subject to change without notice and does not represent a commitment on the part of Sierra Wireless. SIERRA WIRELESS AND ITS AFFILIATES SPECIFICALLY DISCLAIM LIABILITY FOR ANY AND ALL DIRECT, INDIRECT, SPECIAL, GENERAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES INCLUDING, BUT NOT LIMITED TO, LOSS OF PROFITS OR REVENUE OR ANTICIPATED PROFITS OR REVENUE ARISING OUT OF THE USE OR INABILITY TO USE ANY SIERRA WIRELESS PRODUCT, EVEN IF SIERRA WIRELESS AND/OR ITS AFFILIATES HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR THEY ARE FORESEEABLE OR FOR CLAIMS BY ANY THIRD PARTY.

Notwithstanding the foregoing, in no event shall Sierra Wireless and/or its affiliates aggregate liability arising under or in connection with the Sierra Wireless product, regardless of the number of events, occurrences, or claims giving rise to liability, be in excess of the price paid by the purchaser for the Sierra Wireless product.

1.3 Patents

Patents This product may contain technology developed by or for Sierra Wireless Inc. This product includes technology licensed from QUALCOMM® 3G. This product is manufactured or sold by Sierra Wireless Inc. or its affiliates

under one or more patents licensed from InterDigital Group.

1.4 Copyright

© 2011-2015 Sierra Wireless. All rights reserved.

1.5 Trademarks

AirCard® and Heart of the Wireless Machine® are registered trademarks of Sierra Wireless. Watcher® is a trademark of Sierra Wireless, registered in the European Community. Sierra Wireless, the Sierra Wireless logo, the red wave design, and the red-tipped antenna are trademarks of Sierra Wireless. Windows® is a registered trademark of Microsoft Corporation. QUALCOMM® is a registered trademark of QUALCOMM Incorporated. Used under license. Linux is a registered trademark of Linus Torvalds. Other trademarks are the property of the respective owners.

1.6 Contact Information

If you have any questions about the Sierra Wireless Linux SDK, contact your Sierra Wireless account manager.

Consult our website for up-to-date product descriptions, documentation, application notes, firmware upgrades, troubleshooting tips, and press releases: http://www.sierrawireless.com.

Chapter 2

Module Index

2.1 Modules

Here is a list of all modules:

QMI pack/unpack (lite-qmi) 33
Streaming Download Protocol (lite-fw) 34
QMUX layer (lite-qmux) 35
MBIM layer (lite-mbim) 36

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Tables 37

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

_litefw_FirmwareFileInfo . . . 39
_litefw_FirmwareInfo_ . . . 40
_litefw_FirmwarePartNo_ . . . 41
altSrcInfo_t . . . 42
appStats . . . 43
audio_RXAGCList . . . 46
audio_RXAVCList . . . 47
audio_RXPCMIIRFltr . . . 47
audio_TXAGCList . . . 49
audio_TXPCMIIRFltr . . . 50
CarrierImage_t . . . 51
cat_AlPhaIdentifierTlv . . . 53
cat_commonEventTlv . . . 53
cat_currentCatEvent . . . 54
cat_EndProactiveSessionTlv . . . 55
cat_EventIDDataTlv . . . 55
cat_EventListTlv . . . 55
cat_RefreshTlv . . . 56
cdmaSSInfo . . . 56
connectionStatus . . . 57
crashInfoParams . . . 58
crashInformation . . . 58
CtlService . . . 59
CtlTransactionHdr . . . 60
currNetworkInfo . . . 60
dms_ActivationStatusTlv . . . 60
dms_devCaps . . . 61
dms_devCurSubsCaps . . . 62
dms_devMaxCfgListCaps . . . 63
dms_devMaxSubsCaps . . . 64
dms_devMultiSimCaps . . . 64
dms_devMultiSimVoiceDataCaps . . . 65
dms_devSubsCfgList . . . 66
dms_devSubsFeatureModeCaps . . . 66
dms_devSubsList . . . 67
dms_devSubsVoiceDataCaps . . . 68
dms_devSubsVoiceDataList . . . 68
dms_LteBandsSupport . . . 69

dms_OperatingModeTlv . . . 69
dms_PSMActiveTimerIndTlv . . . 70
dms_PSMActiveTimerTlv . . . 70
dms_PSMDurationDueToOOSTlv . . . 71
dms_PSMDurationThresholdTlv . . . 71
dms_PSMEarlyWakeupTimeTlv . . . 72
dms_PSMEnableStateIndTlv . . . 73
dms_PSMEnableStateTlv . . . 73
dms_PSMPeriodicUpdateTimerIndTlv . . . 74
dms_PSMPeriodicUpdateTimerTlv . . . 74
dms_PSMRandomizationWindowTlv . . . 75
dms_TemperatureTlv . . . 75
dms_UimAutoSwitchActSlotTlv . . . 76
dms_UimStatusTlv . . . 77
dms_VoltageTlv . . . 77
DMScustSettingInfo . . . 78
DMScustSettingList . . . 79
DMSgetCustomFeatureV2 . . . 80
DMSgetCustomInput . . . 80
DsdApnAvailableSystemInfo_t . . . 81
DsdApnCurRecAvailableSystemInfo_t . . . 81
DsdApnCurRecPreferredSystemInfo_t . . . 82
DsdGlobalUIInformation_t . . . 83
DsdNullBearerReason_t . . . 84
DsdSystemInfo_t . . . 84
DsdUIApnAvailableSystemInfo_t . . . 87
dunchannelRate . . . 89
eriDataparams . . . 90
eTWSPLMNInfoTlv . . . 90
FMSImageElement . . . 91
FMSImageIdElement . . . 91
FMSImageIDEntries . . . 92
FMSImageList . . . 93
FMSPrefImageList . . . 94
GetClientIdReqMsg . . . 94
GetClientIdRspMsg . . . 94
hdrSSInfo . . . 95
image_info_t . . . 96
ims_AMRModeInfo . . . 97
ims_AMROctAlgnInfo . . . 98
ims_AMRWBModeInfo . . . 98
ims_AMRWBOctAlgnInfo . . . 99
ims_CSCFPortNameInfo . . . 99
ims_EnabAMRWBInfo . . . 100
ims_EnabSCRAMRInfo . . . 100
ims_EnabSCRAMRWBInfo . . . 101
ims_IMSDomainInfo . . . 101
ims_IMSTestModeInfo . . . 102
ims_MinSessExpInfo . . . 102
ims_PCSCFPortInfo . . . 103
ims_PhCtxtURIInfo . . . 103
ims_RngBkTmrInfo . . . 104
ims_RngTmrInfo . . . 104
ims_RTPRTCPInactTmrDurInfo . . . 105
ims_SessDurInfo . . . 105
ims_SigCompEnInfo . . . 106
ims_SIPPortInfo . . . 106
ims_SIPRegnTmrInfo . . . 107

ims_SMSFmtInfo . . . 107
ims_SMSoIPNwInfo . . . 108
ims_SubscrTmrInfo . . . 108
ims_TmrT1Info . . . 109
ims_TmrT2Info . . . 109
ims_TmrTfInfo . . . 109
imsa_IMSFailErrCodeTlv . . . 110
imsa_IMSRegStatusErrorCodeInfo . . . 110
imsa_IMSRegStatusInfo . . . 111
imsa_NewIMSRegStatusInfo . . . 112
imsa_RatHandoverStatusInfo . . . 112
imsa_SmsRatInfo . . . 113
imsa_SmsSvcStatusInfo . . . 114
imsa_UtRatInfo . . . 114
imsa_UtSvcStatusInfo . . . 115
imsa_VoipRatInfo . . . 115
imsa_VoipSvcStatusInfo . . . 116
imsa_VtRatInfo . . . 116
imsa_VtSvcStatusInfo . . . 117
ipv6AddressInfo . . . 117
LibPackGPRSRequestedQoS . . . 118
LibPackPCOIDList . . . 119
LibPackPDNThrottleTimer . . . 119
LibpackProfile3GPP . . . 119
LibpackProfile3GPP2 . . . 127
LibpackProfile3GPPV2 . . . 133
LibPackprofile_3GPP . . . 143
LibPackprofile_3GPP2 . . . 149
LibPackProfileMnc . . . 154
LibPackQosClassID . . . 155
LibPackTFTIDParams . . . 156
LibPackUMTSQoS . . . 158
LibPackUMTSReqQoSSigInd . . . 161
loc_accelAcceptReady . . . 161
loc_accelTempAcceptReady . . . 162
loc_AppProviderInfoTlv . . . 163
loc_BdsSV . . . 164
loc_BdsSVInfo . . . 165
loc_CellDb . . . 165
loc_ClkInfo . . . 166
loc_FixCriteriaStatusTlv . . . 167
loc_GnssData . . . 168
loc_gpsTime . . . 169
loc_gyroAcceptReady . . . 170
loc_gyroTempAcceptReady . . . 171
loc_HorAccuracyLvlTlv . . . 172
loc_IntermediateRptStateTlv . . . 172
loc_IPv4Config . . . 173
loc_IPv4Info . . . 173
loc_IPv6Config . . . 174
loc_IPv6Info . . . 174
loc_LocApplicationInfo . . . 175
loc_MinIntervalTlv . . . 176
loc_precisionDilution . . . 177
loc_satelliteInfo . . . 177
loc_sensorDataUsage . . . 180
loc_SV . . . 180
loc_SVInfo . . . 181

loc_svUsedforFix . . . 182
loc_urlAddr . . . 182
loc_URLAddrInfo . . . 183
lteSSInfo . . . 183
MBIM_CLOSE_DONE_MSG . . . 184
MBIM_CLOSE_MSG . . . 185
MBIM_COMMAND_DONE_MSG . . . 185
MBIM_COMMAND_MSG . . . 185
MBIM_FRAGMENT_HEADER . . . 186
MBIM_FRAGMENT_MSG . . . 186
MBIM_INDICATE_STATUS_MSG . . . 186
MBIM_MESSAGE_HEADER . . . 187
MBIM_OPEN_DONE_MSG . . . 187
MBIM_OPEN_MSG . . . 187
MbimEventEntry . . . 188
MbimIndicator . . . 188
MbimSyncObject . . . 189
MbimTransaction . . . 190
MbimTransport . . . 191
messageModeTlv . . . 192
nas_acqOrderPref . . . 193
nas_AcqOrderPrefTlv . . . 194
nas_ActPilotPNElement . . . 194
nas_AddCDMASysInfo . . . 195
nas_AddSysInfo . . . 195
nas_BandPrefInfoTlv . . . 196
nas_BandPrefTlv . . . 197
nas_CallBarringSysInfo . . . 199
nas_callBarStatus . . . 199
nas_CDMAChannel . . . 200
nas_CDMAECIOThresh . . . 201
nas_CDMAInfo . . . 202
nas_CDMARSSIThresh . . . 203
nas_CDMASysInfo . . . 203
nas_CDMASysInfoExt . . . 207
nas_cellParams . . . 207
nas_ciotAcqOrderPref . . . 208
nas_CiotAcqOrderPrefTlv . . . 209
nas_CiotLteOpModePrefTlv . . . 210
nas_CommInfo . . . 210
nas_CsgId . . . 212
nas_CSGID . . . 212
nas_CSGInfo . . . 213
nas_CSGInfoTlv . . . 214
nas_currentPLMN . . . 215
nas_dataSrvCapabilities . . . 216
nas_DataStatusDetail . . . 217
nas_DDTMSettings . . . 219
nas_detailSvcInfo . . . 220
nas_DeviceConfigDetail . . . 221
nas_dirNum . . . 222
nas_DRCParams . . . 223
nas_ecioListElement . . . 223
nas_ECIOThresh . . . 224
nas_EdrxCiotLteMode . . . 225
nas_EdrxCycleLength . . . 225
nas_EdrxEnableType . . . 226
nas_EdrxPagingTimeWindow . . . 226

nas_EdrxRatType . . . 227
nas_EmerModeTlv . . . 227
nas_errorRateListElement . . . 228
nas_ForbiddenNetworks3GPP . . . 229
nas_GERANInfo . . . 230
nas_geranInstArr . . . 232
nas_geranInstInfo . . . 233
nas_gsmCellInfo . . . 233
nas_GSMRSSIThresh . . . 234
nas_GSMSrvStatusInfo . . . 235
nas_GSMSysInfo . . . 236
nas_GWAcqOrderPrefTlv . . . 239
nas_HDRECIOThresh . . . 239
nas_HDRIOThresh . . . 240
nas_HDRPersonality_Ind_Data . . . 241
nas_HDRRSSIThresh . . . 241
nas_HDRSINRThresh . . . 242
nas_HDRSINRThreshold . . . 242
nas_HDRSysInfo . . . 243
nas_homeNwMNC3GppTlv . . . 245
nas_homeSIDNID . . . 246
nas_ImsVoiceSupportLteTlv . . . 246
nas_infoInterFreq . . . 247
nas_IOThresh . . . 248
nas_LTEARFCNTlv . . . 249
nas_lteBandPrefExt . . . 249
nas_LTEBandPrefTlv . . . 250
nas_LteCiotOpModeTlv . . . 252
nas_lteEARFCN . . . 252
nas_LteEarfcnInfo . . . 253
nas_LteEmbmsCoverageTlv . . . 253
nas_LteEmbmsTraceIdTlv . . . 254
nas_lteGsmCellInfo . . . 255
nas_LTEInfo . . . 256
nas_LTEInfoInterfreq . . . 258
nas_LTEInfoIntrafreq . . . 258
nas_LTEInfoNeighboringGSM . . . 260
nas_LTEInfoNeighboringWCDMA . . . 261
nas_LteM1BandPrefTlv . . . 262
nas_LteNb1BandPrefTlv . . . 264
nas_LTEOperationalModeTlv . . . 266
nas_LTEOperationMode . . . 266
nas_LteOpMode . . . 267
nas_lteOpModeTlv . . . 267
nas_ltePCI . . . 268
nas_LteRegDomainTlv . . . 269
nas_lteRsrpinformation . . . 270
nas_LTERSRPThresh . . . 270
nas_LTERSRQThresh . . . 271
nas_LTERSSIThresh . . . 271
nas_LTESigRptCfg . . . 272
nas_LTESigRptConfig . . . 273
nas_lteSnrinformation . . . 274
nas_LTESNRThresh . . . 274
nas_LTESNRThreshold . . . 275
nas_LTESysInfo . . . 276
nas_LteVoiceDomainTlv . . . 278
nas_lteWcdmaCellInfo . . . 279

nas_Mdn . . . 280
nas_minBasedIMSI . . . 280
nas_MNCPCSDigitStatus . . . 281
nas_MNRInfo . . . 282
nas_ModePrefTlv . . . 282
nas_namName . . . 283
nas_netSelectionPref . . . 284
nas_NetSelPrefTlv . . . 284
nas_NetworkNameSrcTlv . . . 285
nas_networkNameSrcTlv . . . 286
nas_NetworkScan3GPPInfo . . . 287
nas_NetworkScan3GPPInfoTlv . . . 288
nas_NetworkScanStatusTlv . . . 289
nas_NetworkStat1x . . . 290
nas_NetworkStatEVDO . . . 292
nas_nmrCellInfo . . . 294
nas_nr5gBandPref . . . 295
nas_NR5GCellStatusInfoTlv . . . 296
nas_NR5GCellStatusTlv . . . 296
nas_NR5GSerStatTlv . . . 297
nas_NR5GSrvStatusTlv . . . 298
nas_NR5GSysInfoTlv . . . 299
nas_NR5GSystemInfoTlv . . . 303
nas_NumScellsConfig . . . 306
nas_nwNameSrc3GppTlv . . . 307
nas_operatorNameString . . . 308
nas_OperatorPLMNData . . . 308
nas_operatorPLMNList . . . 309
nas_PhyCaAggDlBW . . . 309
nas_PhyCaAggPcellInfo . . . 310
nas_PhyCaAggScellDlBw . . . 311
nas_PhyCaAggScellIndex . . . 311
nas_PhyCaAggScellIndType . . . 312
nas_PhyCaAggScellInfo . . . 313
nas_PilotSetData . . . 315
nas_PilotSetParams . . . 316
nas_PlmnID . . . 316
nas_PLMNNetworkName . . . 317
nas_PLMNNetworkNameData . . . 317
nas_PrefPLMNListTlv . . . 319
nas_PRLPrefTlv . . . 320
nas_protocolSubtypeElement . . . 321
nas_qaQmi3Gpp2TimeZone . . . 322
nas_QmiNas3GppNetworkInfo . . . 323
nas_QmiNas3GppNetworkRAT . . . 324
nas_QmisNasPcsDigit . . . 325
nas_QmisNasSlqsNasPCICellInfo . . . 326
nas_QmisNasSlqsNasPCIInfo . . . 326
nas_RankIndicatorTlv . . . 328
nas_RatDisabledMaskTlv . . . 328
nas_RejectReasonTlv . . . 329
nas_RFBandInfoElements . . . 329
nas_RfBandInfoExtFormat . . . 330
nas_RfBandInfoExtFormatElements . . . 331
nas_RFBandInfoExtTlv . . . 332
nas_RfBandInfoList . . . 333
nas_RFBandInfoTlv . . . 333
nas_RfBandwidthInfo . . . 334

nas_RfBandwidthInfoElements . . . 335
nas_RFBandwidthInfoTlv . . . 336
nas_RfDedicatedBandInfo . . . 337
nas_RfDedicatedBandInfoElements . . . 338
nas_RFDedicatedBandInfoTlv . . . 339
nas_RFInfoTlv . . . 340
nas_roamIndList . . . 341
nas_RoamPrefTlv . . . 342
nas_RSRPThresh . . . 342
nas_rsrqInformation . . . 343
nas_RSRQThresh . . . 343
nas_RSSIThresh . . . 344
nas_rxInfo . . . 345
nas_RxSigInfo . . . 346
nas_rxSignalStrengthListElement . . . 347
nas_SccRxInfo . . . 348
nas_serviceProviderName . . . 349
nas_servSystem . . . 350
nas_sidNid . . . 351
nas_SignalQualityTlv . . . 352
nas_SignalStrengthTlv . . . 352
nas_SignalStrngthTlv . . . 353
nas_SimRejInfoTlv . . . 353
nas_SLQSSignalStrengthsIndReq . . . 354
nas_SLQSSignalStrengthsInformation . . . 356
nas_SLQSSignalStrengthsTlv . . . 357
nas_SrvDomainPrefTlv . . . 358
nas_SrvRegRestrictionTlv . . . 358
nas_SrvRegRestricTlv . . . 359
nas_SrvStatusInfo . . . 360
nas_sysInfoCommon . . . 360
nas_TDSCDMABandPrefTlv . . . 363
nas_TDSCDMAECIOThresh . . . 363
nas_TDSCDMARSCPThresh . . . 364
nas_TDSCDMARSSIThresh . . . 364
nas_TDSCDMASINRCONFThresh . . . 365
nas_TDSCDMASINRThresh . . . 365
nas_timeInfo . . . 366
nas_trueIMSI . . . 368
nas_txInfo . . . 369
nas_UMTSExtInfo . . . 369
nas_UMTSInfo . . . 371
nas_umtsInstArr . . . 373
nas_UMTSinstInfo . . . 374
nas_umtsLTENbrCell . . . 375
nas_UniversalTime . . . 376
nas_UsageSettingTlv . . . 377
nas_VoiceDomainPrefTlv . . . 377
nas_WCDMAARFCNTlv . . . 378
nas_wcdmaCellInfo . . . 379
nas_WCDMACellInfoExt . . . 379
nas_WCDMAECIOThresh . . . 380
nas_WCDMAInfoLTENeighborCell . . . 381
nas_WCDMARSSIThresh . . . 381
nas_WCDMASysInfo . . . 382
nas_wcdmaUARFCN . . . 385
NASAcqOrderPrefTlv . . . 386
NASBandPreferenceTlv . . . 386

NASCiotAcqOrderPrefTlv . . . 387
NASCiotLteOpModePrefTlv . . . 387
NASEmergencyModeTlv . . . 388
NasGetLTECphyCaInfo . . . 388
NASGWAcqOrderPrefTlv . . . 389
NASLTEBandPreferenceTlv . . . 390
NASLteM1BandPrefTlv . . . 390
NASLteNasReleaseInfoTlv . . . 391
NASLteNB1BandPrefTlv . . . 391
NASModePreferenceTlv . . . 392
NASNetSelPreferenceTlv . . . 392
NASNr5gBandPrefTlv . . . 393
NASOTAMessageTlv . . . 393
NASPhyCaAggPcellInfo . . . 394
NASPhyCaAggScellArray . . . 395
NASPhyCaAggScellDlBw . . . 396
NASPhyCaAggScellIndex . . . 397
NASPhyCaAggScellIndType . . . 397
NASPhyCaAggScellInfo . . . 398
NASPRLPreferenceTlv . . . 399
NASQmiCbkNasSwiOTAMessageInd . . . 400
NASQmiCbkNasSystemSelPrefInd . . . 400
NASRatDisabledMaskTlv . . . 403
NASRoamPreferenceTlv . . . 403
NASServDomainPrefTlv . . . 404
NASServingSystemInfo . . . 404
NASTimeInfoTlv . . . 405
newMTMessageTlv . . . 406
pack_audio_SLQSGetAudioPathConfig_t . . . 406
pack_audio_SLQSGetAudioProfile_t . . . 407
pack_audio_SLQSGetAudioVolTLBConfig_t . . . 408
pack_audio_SLQSSetAudioPathConfig_t . . . 408
pack_audio_SLQSSetAudioProfile_t . . . 410
pack_audio_SLQSSetAudioVolTLBConfig_t . . . 412
pack_cat_CATSendEnvelopeCommand_t . . . 412
pack_cat_CATSendTerminalResponse_t . . . 413
pack_cat_SetCATEventCallback_t . . . 414
pack_ctl_GetClientId_t . . . 415
pack_ctl_ReleaseClientId_t . . . 415
pack_dms_ActivateAutomatic_t . . . 415
pack_dms_GetCustFeaturesV2_t . . . 416
pack_dms_ResetToFactoryDefaults_t . . . 416
pack_dms_SetActivationStatusCallback_t . . . 417
pack_dms_SetCrashAction_t . . . 417
pack_dms_SetCustFeature_t . . . 418
pack_dms_SetCustFeaturesV2_t . . . 420
pack_dms_SetEventReport_t . . . 421
pack_dms_SetIndicationRegister_t . . . 421
pack_dms_SetPower_t . . . 422
pack_dms_SetUSBComp_t . . . 423
pack_dms_SLQSDmsSwiIndicationRegister_t . . . 424
pack_dms_SLQSSetPowerSaveModeConfig_t . . . 424
pack_dms_SLQSSwiGetCrashInfo_t . . . 425
pack_dms_SLQSSwiSetDyingGaspCfg_t . . . 425
pack_dms_SLQSSwiSetHostDevInfo_t . . . 426
pack_dms_SLQSSwiSetOSInfo_t . . . 427
pack_dms_SwiSetEventReport_t . . . 427
pack_dms_SwiUimSelect_t . . . 428

pack_dms_UIMChangePIN_t . . . 429
pack_dms_UIMGetControlKeyStatus_t . . . 429
pack_dms_UIMGetICCID_t . . . 430
pack_dms_UIMSetControlKeyProtection_t . . . 430
pack_dms_UIMSetPINProtection_t . . . 431
pack_dms_UIMUnblockControlKey_t . . . 432
pack_dms_UIMUnblockPIN_t . . . 432
pack_dms_UIMVerifyPIN_t . . . 433
pack_dsd_register_ui_info_changes_t . . . 433
pack_dsd_system_status_changes_t . . . 434
pack_fms_GetImagesPreference_t . . . 435
pack_fms_GetStoredImages_t . . . 436
pack_fms_SetImagesPreference_t . . . 436
pack_ims_SLQSImsConfigIndicationRegister_t . . . 437
pack_ims_SLQSSetIMSSMSConfig_t . . . 438
pack_ims_SLQSSetIMSUserConfig_t . . . 439
pack_ims_SLQSSetIMSVoIPConfig_t . . . 439
pack_ims_SLQSSetRegMgrConfig_t . . . 442
pack_ims_SLQSSetSIPConfig_t . . . 443
pack_imsa_SLQSRegisterIMSAIndication_t . . . 444
pack_loc_Delete_Assist_Data_t . . . 445
pack_loc_EventRegister_t . . . 446
pack_loc_SetExtPowerState_t . . . 448
pack_loc_SetOperationMode_t . . . 448
pack_loc_SLQSLOCGetBestAvailPos_t . . . 449
pack_loc_SLQSLOCGetServer_t . . . 450
pack_loc_SLQSLOCInjectPosition_t . . . 450
pack_loc_SLQSLOCInjectSensorData_t . . . 455
pack_loc_SLQSLOCInjectUTCTime_t . . . 457
pack_loc_SLQSLOCSetCradleMountConfig_t . . . 457
pack_loc_SLQSLOCSetServer_t . . . 458
pack_loc_Start_t . . . 459
pack_loc_Stop_t . . . 461
pack_nas_InitiateDomainAttach_t . . . 461
pack_nas_PerformNetworkScanPCI_t . . . 462
pack_nas_SetACCOLC_t . . . 465
pack_nas_SetCDMANetworkParameters_t . . . 465
pack_nas_SetNetworkPreference_t . . . 467
pack_nas_SLQSConfigSigInfo_t . . . 468
pack_nas_SLQSGetPLMNName_t . . . 469
pack_nas_SLQSInitiateNetworkRegistration_t . . . 470
pack_nas_SLQSNasConfigSigInfo2_t . . . 471
pack_nas_SLQSNasGet3GPP2Subscription_t . . . 476
pack_nas_SLQSNASGeteDRXParamsExt_t . . . 476
pack_nas_SLQSNasGetTxRxInfo_t . . . 477
pack_nas_SLQSNasIndicationRegisterExt_t . . . 477
pack_nas_SLQSNasIndicationRegisterV2_t . . . 480
pack_nas_SLQSNASSeteDRXParams_t . . . 488
pack_nas_SLQSNasSwiIndicationRegister_t . . . 490
pack_nas_SLQSNASSwiSetChannelLock_t . . . 491
pack_nas_SLQSPerformIncNetworkScanReq_t . . . 492
pack_nas_SLQSSet3GPP2SubscriptionInfo_t . . . 497
pack_nas_SLQSSetBandPreference_t . . . 498
pack_nas_SLQSSetDDTMPref_t . . . 500
pack_nas_SLQSSetSignalStrengthsCallback_t . . . 500
pack_nas_SLQSSetSysSelectionPref_t . . . 500
pack_nas_SLQSSetSysSelectionPrefExt_t . . . 506
pack_nas_SLQSSwiPSDetach_t . . . 516

pack_pds_PDSInjectTimeReference_t . . . 516
pack_pds_ResetPDSData_t . . . 517
pack_pds_SetEventReportCallback_t . . . 518
pack_pds_SetPDSDefaults_t . . . 518
pack_pds_SetPDSState_t . . . 519
pack_pds_SetPortAutomaticTracking_t . . . 519
pack_pds_SetServiceAutomaticTracking_t . . . 520
pack_pds_SetXTRAAutomaticDownload_t . . . 520
pack_pds_SetXTRANetwork_t . . . 521
pack_pds_SLQSGetAGPSConfig_t . . . 521
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t . . . 522
pack_pds_SLQSPDSInjectPositionData_t . . . 523
pack_pds_SLQSSetAGPSConfig_t . . . 525
pack_pds_SLQSSetPositionMethodState_t . . . 525
pack_pds_StartPDSTrackingSessionExt_t . . . 526
pack_qmi_t . . . 527
pack_qos_BindDataPort_t . . . 528
pack_qos_SLQSQosSwiReadApnExtraParams_t . . . 529
pack_qos_SLQSQosSwiReadDataStats_t . . . 529
pack_qos_SLQSSetQosEventCallback_t . . . 529
pack_rms_SetSMSWake_t . . . 530
pack_sar_SLQSSetRfSarState_t . . . 530
pack_sms_SaveSMS_t . . . 531
pack_sms_SendSMS_t . . . 532
pack_sms_SetNewSMSCallback_t . . . 533
pack_sms_SetSMSCAddress_t . . . 533
pack_sms_SLQSDeleteSMS_t . . . 533
pack_sms_SLQSGetSMS_t . . . 534
pack_sms_SLQSGetSmsBroadcastConfig_t . . . 535
pack_sms_SLQSGetSMSList_t . . . 535
pack_sms_SLQSModifySMSStatus_t . . . 536
pack_sms_SLQSSendAsyncSMS_t . . . 537
pack_sms_SLQSSetIndicationRegister_t . . . 537
pack_sms_SLQSSetSmsBroadcastActivation_t . . . 538
pack_sms_SLQSSetSmsBroadcastConfig_t . . . 538
pack_sms_SLQSSetSmsStorage_t . . . 539
pack_sms_SLQSSmsGetMaxStorageSize_t . . . 540
pack_sms_SLQSSmsSetRoutes_t . . . 540
pack_swiaudio_SLQSGetM2MAudioProfile_t . . . 540
pack_swiaudio_SLQSGetM2MAudioVolume_t . . . 541
pack_swiaudio_SLQSGetM2MAVMute_t . . . 541
pack_swiaudio_SLQSGetM2MSpkrGain_t . . . 542
pack_swiaudio_SLQSSetM2MAudioAVCFG_t . . . 542
pack_swiaudio_SLQSSetM2MAudioLPBK_t . . . 543
pack_swiaudio_SLQSSetM2MAudioProfile_t . . . 543
pack_swiaudio_SLQSSetM2MAudioVolume_t . . . 545
pack_swiaudio_SLQSSetM2MAVMute_t . . . 545
pack_swiaudio_SLQSSetM2MSpkrGain_t . . . 546
pack_swiavms_SLQSAVMSSendSelection_t . . . 547
pack_swiavms_SLQSAVMSSetSettings_t . . . 547
pack_swiavms_SLQSAVMSSetSettings_v2_t . . . 549
pack_swiavms_SLQSAVMSStartSession_t . . . 551
pack_swiavms_SLQSAVMSStopSession_t . . . 551
pack_swidms_SLQSSwiDmsSetHWWatchdog_t . . . 552
pack_swidms_SLQSSwiDmsSetMTU_t . . . 552
pack_swidms_SLQSSwiDmsSetUimSlot_t . . . 553
pack_swidms_SLQSSwiDmsSetUsbComp_t . . . 554
pack_swidms_SLQSSwiDmsSetUsbNetNum_t . . . 555

pack_swiloc_SwiLocSetAutoStart_t . . . 556
pack_swioma_SLQSOMADMCancelSession_t . . . 557
pack_swioma_SLQSOMADMCancelSessionExt_t . . . 558
pack_swioma_SLQSOMADMGetSessionInfo_t . . . 558
pack_swioma_SLQSOMADMSendSelection_t . . . 559
pack_swioma_SLQSOMADMSendSelectionExt_t . . . 559
pack_swioma_SLQSOMADMSetSettings_t . . . 560
pack_swioma_SLQSOMADMSetSettingsExt_t . . . 561
pack_swioma_SLQSOMADMStartSession_t . . . 562
pack_swioma_SLQSOMADMStartSessionExt_t . . . 562
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t . . . 563
pack_tmd_SLQSTmdGetMitigationLvl_t . . . 563
pack_tmd_SLQSTmdRegNotMitigationLvl_t . . . 564
pack_uim_ChangePin_t . . . 564
pack_uim_ReadTransparent_t . . . 565
pack_uim_SetPinProtection_t . . . 567
pack_uim_SLQSUIMAddProfile_t . . . 568
pack_uim_SLQSUIMAuthenticate_t . . . 568
pack_uim_SLQSUIMDeleteProfile_t . . . 569
pack_uim_SLQSUIMDepersonalization_t . . . 570
pack_uim_SLQSUIMeUICCMemReset_t . . . 570
pack_uim_SLQSUIMEventRegister_t . . . 571
pack_uim_SLQSUIMGetConfiguration_t . . . 571
pack_uim_SLQSUIMGetEID_t . . . 572
pack_uim_SLQSUIMGetFileAttributes_t . . . 572
pack_uim_SLQSUIMGetProfileInfo_t . . . 573
pack_uim_SLQSUIMGetServiceStatus_t . . . 574
pack_uim_SLQSUIMGetSIMProfile_t . . . 575
pack_uim_SLQSUIMPowerDown_t . . . 575
pack_uim_SLQSUIMPowerUp_t . . . 576
pack_uim_SLQSUIMReadRecord_t . . . 576
pack_uim_SLQSUIMRefreshComplete_t . . . 577
pack_uim_SLQSUIMRefreshGetLastEvent_t . . . 578
pack_uim_SLQSUIMRefreshOK_t . . . 578
pack_uim_SLQSUIMRefreshRegister_t . . . 579
pack_uim_SLQSUIMSetServiceStatus_t . . . 579
pack_uim_SLQSUIMSetSIMProfile_t . . . 580
pack_uim_SLQSUIMSwitchSlot_t . . . 581
pack_uim_SLQSUIMUpdateProfile_t . . . 581
pack_uim_SLQSUIMWriteRecord_t . . . 582
pack_uim_SLQSUIMWriteTransparent_t . . . 583
pack_uim_UnblockPin_t . . . 584
pack_uim_VerifyPin_t . . . 585
pack_voice_AnswerUSSD_t . . . 586
pack_voice_OriginateUSSD_t . . . 586
pack_voice_SLQSOriginateUSSD_t . . . 587
pack_voice_SLQSVoiceALSSelectLine_t . . . 587
pack_voice_SLQSVoiceALSSetLineSwitching_t . . . 588
pack_voice_SLQSVoiceAnswerCall_t . . . 588
pack_voice_SLQSVoiceBindSubscription_t . . . 589
pack_voice_SLQSVoiceBurstDTMF_t . . . 589
pack_voice_SLQSVoiceDialCall_t . . . 590
pack_voice_SLQSVoiceEndCall_t . . . 591
pack_voice_SLQSVoiceGetCallBarring_t . . . 592
pack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 593
pack_voice_SLQSVoiceGetCallInfo_t . . . 593
pack_voice_SLQSVoiceGetCallWaiting_t . . . 594
pack_voice_SLQSVoiceGetConfig_t . . . 594

pack_voice_SLQSVoiceIndicationRegister_t . . . 596
pack_voice_SLQSVoiceManageCalls_t . . . 597
pack_voice_SLQSVoiceOrigUSSDNoWait_t . . . 598
pack_voice_SLQSVoiceSendFlash_t . . . 598
pack_voice_SLQSVoiceSetCallBarringPassword_t . . . 599
pack_voice_SLQSVoiceSetConfig_t . . . 600
pack_voice_SLQSVoiceSetPreferredPrivacy_t . . . 601
pack_voice_SLQSVoiceSetSUPSService_t . . . 601
pack_voice_SLQSVoiceStartContDTMF_t . . . 604
pack_voice_SLQSVoiceStopContDTMF_t . . . 605
pack_wda_GetDataFormat_t . . . 605
pack_wda_GetQmapSettings_t . . . 606
pack_wda_SetDataFormat_t . . . 606
pack_wda_SetQmapSettings_t . . . 609
pack_wds_DHCPv4ClientLeaseChange_t . . . 610
pack_wds_GetDefaultProfile_t . . . 611
pack_wds_GetDefaultProfileNum_t . . . 611
pack_wds_GetDefaultProfileV2_t . . . 612
pack_wds_GetDormancyState_t . . . 612
pack_wds_GetLastMobileIPError_t . . . 612
pack_wds_GetMobileIP_t . . . 613
pack_wds_GetMobileIPProfile_t . . . 613
pack_wds_GetPacketStatistics_t . . . 613
pack_wds_GetPacketStatus_t . . . 614
pack_wds_GetSessionDuration_t . . . 614
pack_wds_RMSetTransferStatistics_t . . . 615
pack_wds_SetAutoconnect_t . . . 615
pack_wds_SetDefaultProfile_t . . . 616
pack_wds_SetDefaultProfileNum_t . . . 618
pack_wds_SetMobileIP_t . . . 618
pack_wds_SetMobileIPParameters_t . . . 619
pack_wds_SetMobileIPProfile_t . . . 620
pack_wds_SLQSCreateProfile_t . . . 622
pack_wds_SLQSDeleteProfile_t . . . 622
pack_wds_SLQSGetCurrDataSystemStat_t . . . 623
pack_wds_SLQSGetDataBearerTechnology_t . . . 623
pack_wds_SLQSGetDUNCallInfo_t . . . 623
pack_wds_SLQSGetProfileSettings_t . . . 625
pack_wds_SLQSGetRuntimeSettings_t . . . 626
pack_wds_SLQSModifyProfile_t . . . 626
pack_wds_SLQSSet3GPPConfigItem_t . . . 627
pack_wds_SLQSSetIPFamilyPreference_t . . . 629
pack_wds_SLQSSetWdsEventCallback_t . . . 629
pack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 630
pack_wds_SLQSSSetDHCPv4ClientConfig_t . . . 630
pack_wds_SLQSSSetLoopback_t . . . 631
pack_wds_SLQSStartDataSession_t . . . 632
pack_wds_SLQSStopDataSession_t . . . 633
pack_wds_SLQSSwiProfileChangeCallback_t . . . 633
pack_wds_SLQSWdsSetEventReport_t . . . 634
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 635
PackCreateProfileOut . . . 636
packgetDyingGaspCfg . . . 637
packgetDyingGaspStatistics . . . 637
PackSwiAvmsSetSettingsAPNInfo . . . 638
PackSwiAvmsSetSettingsConnectionRetryTimers . . . 638
PackSwiAvmsSetSettingsPeriodInfo . . . 639
PackSwiAVMSSettingsAPNInfo . . . 640

PackSwiAVMSSettingsConnectionRetryTimers . . . 640
PackSwiAVMSSettingsPeriodsInfo . . . 641
QcqmiService . . . 641
QmiHdr . . . 642
QmiService . . . 642
qmiSmsMessageList . . . 644
QmiSyncObject . . . 644
QmiTransaction . . . 645
QmiTransactionHdr . . . 646
qmiWDSDataBearerTechnology . . . 646
qmTlvResult . . . 647
QMUX_HDR . . . 647
QmuxTransport . . . 648
qos_BindDataPortMuxID_t . . . 649
qos_BindDataPortPeripheralEndPointID_t . . . 649
qos_BindDataPortSIODataPort_t . . . 650
ReleaseClientIdReqMsg . . . 650
ReleaseClientIdRspMsg . . . 651
ResultTlv . . . 652
RFBandInfoElements . . . 652
rmTrasnferStaticsReq . . . 653
sensorData_t . . . 653
slot_t . . . 655
slotInf . . . 656
slots_t . . . 657
sms_BroadcastConfig . . . 658
sms_CDMABroadcastConfig . . . 658
sms_getIndicationReg . . . 659
sms_getMsgWaitingInfo . . . 660
sms_getTransLayerInfo . . . 660
sms_getTransNWRegInfo . . . 661
sms_maxStorageSizeReq . . . 662
sms_maxStorageSizeResp . . . 662
sms_messageWaitingInfoContent . . . 663
sms_msgProtocolResp . . . 664
sms_qaQmi3GPP2BroadcastCfgInfo . . . 664
sms_qaQmi3GPPBroadcastCfgInfo . . . 665
sms_routeEntry . . . 665
sms_sendAsyncsmsParams . . . 666
sms_setIndicationReg . . . 669
sms_setRoutesReq . . . 669
sms_transLayerInfo . . . 670
sMSCAddressInfo . . . 671
sMSCAddressTlv . . . 671
sMSEtwsMessageInfo . . . 672
sMSEtwsMessageTlv . . . 672
sMSEtwsPlmnInfo . . . 673
sMSMessageModeInfo . . . 673
sMSMTMessageInfo . . . 674
sMSOnIMSInfo . . . 674
sMSOnIMSTlv . . . 675
sMSTransferRouteMTMessageInfo . . . 675
swi_uint256_t . . . 676
swiaudio_PCMparams . . . 676
swidms_AutoUimSwitchModeTlv . . . 677
swidms_ehrpdMTUSizeTlv . . . 677
swidms_hrpdMTUSizeTlv . . . 678
swidms_intfaceCfgTlv . . . 678

swidms_mtuSize3gppTlv . . . 679
swidms_MultiplexedSimModeTlv . . . 680
swidms_SetMultiplexedModeTlv . . . 681
swidms_supportedIntBitmaskTlv . . . 681
swidms_SwiDmsGetHWWatchdog . . . 682
swidms_UimSlotSelectedTlv . . . 683
swidms_usbMTUSizeTlv . . . 683
tdscdmaSigInfoExt . . . 684
tempData_t . . . 684
tmd_mitigationDevList . . . 685
transferRouteMessageTlv . . . 686
uim_activeProfileInfo . . . 687
uim_additionalReadResult . . . 687
uim_appStatus . . . 688
uim_authenticateResult . . . 691
uim_authenticationData . . . 692
uim_cardResult . . . 693
uim_cardResultInfo . . . 693
uim_cardStatus . . . 694
uim_changeUIMPIN . . . 695
uim_depersonalizationInformation . . . 696
uim_eidInfo . . . 697
uim_encryptedPIN1 . . . 697
uim_fileAttributes . . . 698
uim_fileInfo . . . 701
uim_getSimProfileInfo . . . 702
uim_GetSlotsInfoTlv . . . 703
uim_GetSlotsStatusTlv . . . 704
uim_hotSwapStatus . . . 704
uim_indToken . . . 705
uim_personalizationStatus . . . 705
uim_physlotInfo . . . 706
uim_physlotsInfo . . . 707
uim_profileClassTlv . . . 708
uim_profileICCIDTlv . . . 708
uim_profileIconTlv . . . 709
uim_profileIconTypeTlv . . . 710
uim_profileNameTlv . . . 710
uim_profileNicknameTlv . . . 711
uim_profilePolicyRulesTlv . . . 711
uim_profileSPNTlv . . . 712
uim_profileStateTlv . . . 712
uim_readRecordInfo . . . 713
uim_readResult . . . 714
uim_readResultInfo . . . 714
uim_readTransparentInfo . . . 715
uim_refreshevent . . . 715
uim_registerRefresh . . . 717
uim_remainingRetries . . . 718
uim_sessionInformation . . . 718
uim_setPINProtection . . . 719
uim_simBusyStatus . . . 720
uim_slotInfo . . . 721
uim_UIMGetFDNStatus . . . 722
uim_UIMGetHiddenKeyStatus . . . 723
uim_UIMGetIndex . . . 723
uim_UIMNickName . . . 724
uim_UIMProfileCode . . . 724

uim_UIMSessionInformation . . . 725
uim_unblockUIMPIN . . . 726
uim_validCardStatus . . . 727
uim_verifyUIMPIN . . . 727
uim_writeRecordInfo . . . 728
uim_writeTransparentInfo . . . 728
unpack_audio_SLQSGetAudioPathConfig_t . . . 729
unpack_audio_SLQSGetAudioProfile_t . . . 731
unpack_audio_SLQSGetAudioVolTLBConfig_t . . . 732
unpack_audio_SLQSSetAudioVolTLBConfig_t . . . 733
unpack_cat_SetCatEventCallback_ind_t . . . 733
unpack_cat_SetCATEventCallback_t . . . 734
unpack_ctl_GetClientId_t . . . 735
unpack_ctl_ReleaseClientId_t . . . 736
unpack_dms_GetActivationState_t . . . 737
unpack_dms_GetBandCapability_t . . . 737
unpack_dms_GetCrashAction_t . . . 739
unpack_dms_GetCustFeature_t . . . 740
unpack_dms_GetCustFeaturesV2_t . . . 742
unpack_dms_GetDeviceCap_t . . . 743
unpack_dms_GetDeviceCapabilities_t . . . 744
unpack_dms_GetDeviceCapabilitiesV2_t . . . 746
unpack_dms_GetDeviceHardwareRev_t . . . 748
unpack_dms_GetDeviceMfr_t . . . 749
unpack_dms_GetDeviceSerialNumbers_t . . . 750
unpack_dms_GetFirmwareInfo_t . . . 752
unpack_dms_GetFirmwareRevision_t . . . 754
unpack_dms_GetFirmwareRevisions_t . . . 755
unpack_dms_GetFSN_t . . . 756
unpack_dms_GetHardwareRevision_t . . . 757
unpack_dms_GetIMSI_t . . . 757
unpack_dms_GetManufacturer_t . . . 758
unpack_dms_GetModelID_t . . . 758
unpack_dms_GetNetworkTime_t . . . 759
unpack_dms_GetNetworkTimeV2_t . . . 760
unpack_dms_GetOfflineReason_t . . . 761
unpack_dms_GetPower_t . . . 762
unpack_dms_GetPRLVersion_t . . . 763
unpack_dms_GetSerialNumbers_t . . . 764
unpack_dms_GetUSBComp_t . . . 765
unpack_dms_GetVoiceNumber_t . . . 767
unpack_dms_PSMCfgChange_ind_t . . . 769
unpack_dms_ResetToFactoryDefaults_t . . . 769
unpack_dms_SetActivationStatusCallback_t . . . 770
unpack_dms_SetCrashAction_t . . . 770
unpack_dms_SetCustFeature_t . . . 771
unpack_dms_SetCustFeaturesV2_t . . . 771
unpack_dms_SetEventReport_ind_t . . . 772
unpack_dms_SetEventReport_t . . . 773
unpack_dms_SetFirmwarePreference_t . . . 773
unpack_dms_SetIndicationRegister_t . . . 774
unpack_dms_SetPower_t . . . 774
unpack_dms_SetUSBComp_t . . . 775
unpack_dms_SLQSDmsSwiGetPCInfo_t . . . 775
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t . . . 778
unpack_dms_SLQSDmsSwiGetResetInfo_t . . . 779
unpack_dms_SLQSDmsSwiGetUimSelection_t . . . 780
unpack_dms_SLQSDmsSwiIndicationRegister_t . . . 781

unpack_dms_SLQSGetBandCapability_t . . . 782
unpack_dms_SLQSGetBandCapabilityExt_t . . . 786
unpack_dms_SLQSGetERIFile_t . . . 790
unpack_dms_SLQSGetPowerSaveModeConfig_t . . . 791
unpack_dms_SLQSSetPowerSaveModeConfig_t . . . 792
unpack_dms_SLQSSwiClearDyingGaspStatistics_t . . . 793
unpack_dms_SLQSSwiGetCrashInfo_t . . . 793
unpack_dms_SLQSSwiGetDyingGaspCfg_t . . . 794
unpack_dms_SLQSSwiGetDyingGaspStatistics_t . . . 795
unpack_dms_SLQSSwiGetFirmwareCurr_t . . . 795
unpack_dms_SLQSSwiGetFwUpdateStatus_t . . . 796
unpack_dms_SLQSSwiGetHostDevInfo_t . . . 798
unpack_dms_SLQSSwiGetOSInfo_t . . . 799
unpack_dms_SLQSSwiGetSerialNoExt_t . . . 800
unpack_dms_SLQSSwiSetDyingGaspCfg_t . . . 801
unpack_dms_SLQSSwiSetHostDevInfo_t . . . 801
unpack_dms_SLQSSwiSetOSInfo_t . . . 802
unpack_dms_SLQSUIMGetState_t . . . 802
unpack_dms_SwiEventReportCallBack_ind_t . . . 803
unpack_dms_SwiSetEventReport_t . . . 804
unpack_dms_SwiUimSelect_t . . . 804
unpack_dms_UIMGetControlKeyStatus_t . . . 805
unpack_dms_UIMGetICCID_t . . . 806
unpack_dms_UIMGetPINStatus_t . . . 807
unpack_dms_UIMSetControlKeyProtection_t . . . 808
unpack_dms_UIMSetPINProtection_t . . . 809
unpack_dms_UIMUnblockControlKey_t . . . 810
unpack_dsd_SLQSGetDsdUIInfo_t . . . 810
unpack_dsd_SLQSGetSystemStatus_t . . . 811
unpack_dsd_SLQSSystemStatusChange_Ind_t . . . 813
unpack_dsd_SLQSUIInfo_Ind_t . . . 814
unpack_fms_GetImagesPreference_t . . . 815
unpack_fms_GetStoredImages_t . . . 816
unpack_fms_SetImagesPreference_t . . . 817
unpack_ims_SLQSGetIMSSMSConfig_t . . . 817
unpack_ims_SLQSGetIMSUserConfig_t . . . 819
unpack_ims_SLQSGetIMSVoIPConfig_t . . . 819
unpack_ims_SLQSGetRegMgrConfig_t . . . 822
unpack_ims_SLQSGetSIPConfig_t . . . 823
unpack_ims_SLQSRegMgrCfgCallBack_ind_t . . . 825
unpack_ims_SLQSSetIMSSMSConfig_t . . . 826
unpack_ims_SLQSSetIMSUserConfig_t . . . 826
unpack_ims_SLQSSetIMSVoIPConfig_t . . . 827
unpack_ims_SLQSSetRegMgrConfig_t . . . 827
unpack_ims_SLQSSetSIPConfig_t . . . 827
unpack_ims_SLQSSIPCfgCallBack_ind_t . . . 828
unpack_ims_SLQSSMSCfgCallBack_ind_t . . . 829
unpack_ims_SLQSUserCfgCallBack_ind_t . . . 830
unpack_ims_SLQSVoIPCfgCallBack_ind_t . . . 831
unpack_imsa_SLQSGetIMSARegStatus_t . . . 833
unpack_imsa_SLQSGetIMSAServiceStatus_t . . . 834
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t . . . 837
unpack_imsa_SLQSImsaRatStatusCallBack_ind_t . . . 837
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t . . . 838
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t . . . 839
unpack_loc_BestAvailPos_Ind_t . . . 840
unpack_loc_CradleMountCallback_Ind_t . . . 847
unpack_loc_Delete_Assist_Data_t . . . 848

unpack_loc_DeleteAssistData_Ind_t . . . 848
unpack_loc_EngineState_Ind_t . . . 849
unpack_loc_EventNMEA_Ind_t . . . 850
unpack_loc_EventRegister_t . . . 851
unpack_loc_EventTimeSyncCallback_Ind_t . . . 851
unpack_loc_FixCriteria_Ind_t . . . 852
unpack_loc_GetOpMode_Ind_t . . . 853
unpack_loc_GetServer_Ind_t . . . 854
unpack_loc_GnssSvInfo_Ind_t . . . 856
unpack_loc_InjectPositionCallback_Ind_t . . . 857
unpack_loc_InjectSensorDataCallback_Ind_t . . . 857
unpack_loc_InjectTimeSyncDataCallback_Ind_t . . . 859
unpack_loc_InjectUTCTimeCallback_Ind_t . . . 860
unpack_loc_PositionRpt_Ind_t . . . 861
unpack_loc_SensorStreamingCallback_Ind_t . . . 867
unpack_loc_SetExtPowerConfig_Ind_t . . . 868
unpack_loc_SetExtPowerState_t . . . 869
unpack_loc_SetOperationMode_Ind_t . . . 870
unpack_loc_SetOperationMode_t . . . 871
unpack_loc_SetServer_Ind_t . . . 871
unpack_loc_SLQSLOCGetBestAvailPos_t . . . 872
unpack_loc_SLQSLOCGetOpMode_t . . . 873
unpack_loc_Start_t . . . 873
unpack_loc_Stop_t . . . 874
unpack_nas_GetACCOLC_t . . . 874
unpack_nas_GetANAAAAuthenticationStatus_t . . . 875
unpack_nas_GetCDMANetworkParameters_t . . . 875
unpack_nas_GetHomeNetwork3GPP2_t . . . 878
unpack_nas_GetHomeNetwork_t . . . 880
unpack_nas_GetNetworkPreference_t . . . 881
unpack_nas_GetRFInfo_t . . . 883
unpack_nas_GetServingNetwork_t . . . 883
unpack_nas_GetServingNetworkCapabilities_t . . . 886
unpack_nas_GetSignalStrengths_t . . . 887
unpack_nas_PerformNetworkScan_t . . . 888
unpack_nas_SetDataCapabilitiesCallback_ind_t . . . 890
unpack_nas_SetEventReportInd_t . . . 890
unpack_nas_SetNasLTECphyCaIndCallback_ind_t . . . 892
unpack_nas_SetNetworkPreference_t . . . 893
unpack_nas_SetRoamingIndicatorCallback_ind_t . . . 893
unpack_nas_SetServingSystemCallback_ind_t . . . 894
unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t . . . 895
unpack_nas_SLQSGetErrorRate_t . . . 895
unpack_nas_SLQSGetHomeNetwork_t . . . 896
unpack_nas_SlqsGetLTECphyCAInfo_t . . . 898
unpack_nas_SLQSGetNetworkTime_t . . . 898
unpack_nas_SLQSGetOperatorNameData_t . . . 899
unpack_nas_SLQSGetPLMNName_t . . . 900
unpack_nas_SLQSGetServingSystem_t . . . 903
unpack_nas_SLQSGetServingSystemV2_t . . . 907
unpack_nas_SLQSGetSignalStrength_t . . . 912
unpack_nas_SLQSGetSysInfo_t . . . 915
unpack_nas_SLQSGetSysInfoV2_t . . . 918
unpack_nas_SLQSGetSysSelectionPref_t . . . 923
unpack_nas_SLQSGetSysSelectionPrefExt_t . . . 928
unpack_nas_SLQSGetSysSelectionPrefExtV2_t . . . 931
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t . . . 935
unpack_nas_SLQSNasGet3GPP2Subscription_t . . . 936

unpack_nas_SLQSNasGetCellLocationInfo_t . . . 937
unpack_nas_SLQSNasGetCellLocationInfoV2_t . . . 938
unpack_nas_SLQSNASGeteDRXParams_t . . . 940
unpack_nas_SLQSNASGeteDRXParamsExt_t . . . 941
unpack_nas_SLQSNASGetForbiddenNetworks_t . . . 943
unpack_nas_SLQSNasGetHDRColorCode_t . . . 943
unpack_nas_SLQSNasGetRFInfo_t . . . 944
unpack_nas_SLQSNasGetSigInfo_t . . . 945
unpack_nas_SLQSNasGetTxRxInfo_t . . . 946
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t . . . 947
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t . . . 949
unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t . . . 950
unpack_nas_SLQSNasRFBandInfoCallback_Ind_t . . . 951
unpack_nas_SLQSNasSigInfoCallback_ind_t . . . 953
unpack_nas_SLQSNASSwiGetChannelLock_t . . . 954
unpack_nas_SLQSNasSwiModemStatus_t . . . 955
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t . . . 955
unpack_nas_SLQSNasTimerCallback_ind_t . . . 956
unpack_nas_SLQSPerformNetworkScanV2_t . . . 957
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t . . . 959
unpack_nas_SLQSSwiGetHDRPersonality_t . . . 959
unpack_nas_SLQSSwiGetHDRProtSubtype_t . . . 960
unpack_nas_SLQSSwiGetHRPDStats_t . . . 961
unpack_nas_SLQSSwiGetLteCQI_t . . . 962
unpack_nas_SLQSSwiGetLteSccRxInfo_t . . . 963
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t . . . 964
unpack_nas_SLQSSwiNetworkDebug_t . . . 964
unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t . . . 965
unpack_nas_SLQSSysInfoCallback_ind_t . . . 966
unpack_omaDmConfigTlv_t . . . 970
unpack_omaDmFotaTlv_t . . . 971
unpack_omaDmNotificationsTlv_t . . . 973
unpack_pds_GetPDSDefaults_t . . . 974
unpack_pds_GetPDSState_t . . . 975
unpack_pds_GetPortAutomaticTracking_t . . . 975
unpack_pds_GetServiceAutomaticTracking_t . . . 976
unpack_pds_GetXTRAAutomaticDownload_t . . . 977
unpack_pds_GetXTRANetwork_t . . . 977
unpack_pds_GetXTRAValidity_t . . . 978
unpack_pds_SetEventReport_Ind_t . . . 979
unpack_pds_SetPdsState_Ind_t . . . 981
unpack_pds_SLQSGetAGPSConfig_t . . . 981
unpack_pds_SLQSGetGPSStateInfo_t . . . 982
unpack_qmi_t . . . 987
unpack_qos_BindDataPort_t . . . 988
unpack_qos_dataRate_t . . . 988
unpack_qos_IPv4Addr_t . . . 989
unpack_qos_IPv6Addr_t . . . 989
unpack_qos_IPv6TrafCls_t . . . 990
unpack_qos_pktErrRate_t . . . 990
unpack_qos_Port_t . . . 991
unpack_qos_QosFlowInfo_t . . . 991
unpack_qos_QosFlowInfoState_t . . . 993
unpack_qos_SLQSQosGetNetworkStatus_t . . . 993
unpack_qos_SLQSQosSwiReadApnExtraParams_t . . . 994
unpack_qos_SLQSQosSwiReadDataStats_t . . . 995
unpack_qos_SLQSSetQosEventCallback_ind_t . . . 997
unpack_qos_SLQSSetQosEventCallback_t . . . 998

unpack_qos_SLQSSetQosNWStatusCallback_ind_t . . . 998
unpack_qos_SLQSSetQosPriEventCallback_ind_t . . . 999
unpack_qos_SLQSSetQosStatusCallback_ind_t . . . 999
unpack_qos_swiQosFilter_t . . . 1001
unpack_qos_swiQosFlow_t . . . 1004
unpack_qos_tokenBucket_t . . . 1008
unpack_qos_Tos_t . . . 1009
unpack_QosFlowStat_t . . . 1009
unpack_result_t . . . 1010
unpack_rms_GetSMSWake_t . . . 1010
unpack_rms_SetSMSWake_t . . . 1011
unpack_RMTransferStatistics_ind_t . . . 1011
unpack_sar_SLQSGetRfSarState_t . . . 1013
unpack_sms_GetSMSCAddress_t . . . 1013
unpack_sms_SaveSMS_t . . . 1014
unpack_sms_SendSMS_t . . . 1015
unpack_sms_SetNewSMSCallback_ind_t . . . 1015
unpack_sms_SetNewSMSCallback_t . . . 1017
unpack_sms_SetSMSCAddress_t . . . 1017
unpack_sms_SLQSDeleteSMS_t . . . 1018
unpack_sms_SLQSGetIndicationRegister_t . . . 1018
unpack_sms_SLQSGetMessageWaiting_t . . . 1019
unpack_sms_SLQSGetSMS_t . . . 1019
unpack_sms_SLQSGetSmsBroadcastConfig_t . . . 1020
unpack_sms_SLQSGetSMSList_t . . . 1021
unpack_sms_SLQSGetTransLayerInfo_t . . . 1022
unpack_sms_SLQSGetTransNWRegInfo_t . . . 1022
unpack_sms_SLQSModifySMSStatus_t . . . 1023
unpack_sms_SLQSNWRegInfoCallback_ind_t . . . 1023
unpack_sms_SLQSSendAsyncSMS_t . . . 1024
unpack_sms_SLQSSetIndicationRegister_t . . . 1024
unpack_sms_SLQSSetSmsBroadcastActivation_t . . . 1025
unpack_sms_SLQSSetSmsBroadcastConfig_t . . . 1025
unpack_sms_SLQSSetSmsStorage_t . . . 1026
unpack_sms_SLQSSmsGetMaxStorageSize_t . . . 1026
unpack_sms_SLQSSmsGetMessageProtocol_t . . . 1027
unpack_sms_SLQSSmsSetRoutes_t . . . 1027
unpack_sms_SLQSSwiGetSMSStorage_t . . . 1028
unpack_sms_SLQSTransLayerInfoCallback_ind_t . . . 1028
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t . . . 1029
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t . . . 1031
unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t . . . 1032
unpack_swiaudio_SLQSGetM2MAudioProfile_t . . . 1032
unpack_swiaudio_SLQSGetM2MAudioVolume_t . . . 1034
unpack_swiaudio_SLQSGetM2MAVMute_t . . . 1034
unpack_swiaudio_SLQSGetM2MSpkrGain_t . . . 1035
unpack_swiavms_SLQSAVMSEventReportInd_t . . . 1036
unpack_swiavms_SLQSAVMSGetSettings_t . . . 1037
unpack_swiavms_SLQSAVMSGetSettings_v2_t . . . 1040
unpack_swiavms_SLQSAVMSSendSelection_t . . . 1042
unpack_swiavms_SLQSAVMSSessionGetInfo_t . . . 1043
unpack_swiavms_SLQSAvmsSetEventReport_t . . . 1044
unpack_swiavms_SLQSAVMSSetSettings_t . . . 1044
unpack_swiavms_SLQSAVMSSetSettings_v2_t . . . 1045
unpack_swiavms_SLQSAVMSStartSession_t . . . 1045
unpack_swiavms_SLQSAVMSStopSession_t . . . 1046
unpack_swidms_SLQSSwiDmsGetHWWatchdog_t . . . 1046
unpack_swidms_SLQSSwiDmsGetMTU_t . . . 1047

unpack_swidms_SLQSSwiDmsGetSecureInfo_t . . . 1047
unpack_swidms_SLQSSwiDmsGetUimSlot_t . . . 1048
unpack_swidms_SLQSSwiDmsGetUsbComp_t . . . 1049
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t . . . 1050
unpack_swidms_SLQSSwiDmsSetHWWatchdog_t . . . 1050
unpack_swidms_SLQSSwiDmsSetMTU_t . . . 1051
unpack_swidms_SLQSSwiDmsSetUimSlot_t . . . 1051
unpack_swidms_SLQSSwiDmsSetUsbComp_t . . . 1052
unpack_swiloc_SwiLocGetAutoStart_t . . . 1052
unpack_swioma_SLQSOMADMAlertCallback_ind_t . . . 1054
unpack_swioma_SLQSOMADMGetSessionInfo_t . . . 1055
unpack_swioma_SLQSOMADMGetSessionInfoExt_t . . . 1058
unpack_swioma_SLQSOMADMGetSettings_t . . . 1062
unpack_swioma_SLQSOMADMStartSession_t . . . 1063
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t . . . 1064
unpack_tmd_SLQSTmdGetMitigationDevList_t . . . 1064
unpack_tmd_SLQSTmdGetMitigationLvl_t . . . 1065
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t . . . 1066
unpack_tmd_SLQSTmdRegNotMitigationLvl_t . . . 1066
unpack_uim_ChangePin_t . . . 1067
unpack_uim_GetCardStatus_t . . . 1068
unpack_uim_GetCardStatusV2_t . . . 1069
unpack_uim_ReadTransparent_t . . . 1070
unpack_uim_SetPinProtection_t . . . 1071
unpack_uim_SetUimSlotStatusChangeCallback_ind_t . . . 1071
unpack_uim_SLQSUIMAuthenticate_t . . . 1072
unpack_uim_SLQSUIMDepersonalization_t . . . 1073
unpack_uim_SLQSUIMEventRegister_t . . . 1073
unpack_uim_SLQSUIMGetConfiguration_t . . . 1074
unpack_uim_SLQSUIMGetEID_t . . . 1075
unpack_uim_SLQSUIMGetFileAttributes_t . . . 1075
unpack_uim_SLQSUIMGetProfileInfo_t . . . 1076
unpack_uim_SLQSUIMGetServiceStatus_t . . . 1078
unpack_uim_SLQSUIMGetSIMProfile_t . . . 1079
unpack_uim_SLQSUIMGetSlotsStatus_t . . . 1079
unpack_uim_SLQSUIMGetSlotsStatusV2_t . . . 1080
unpack_uim_SLQSUIMReadRecord_t . . . 1081
unpack_uim_SLQSUIMRefreshCallback_Ind_t . . . 1082
unpack_uim_SLQSUIMRefreshGetLastEvent_t . . . 1082
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t . . . 1083
unpack_uim_SLQSUIMWriteRecord_t . . . 1083
unpack_uim_SLQSUIMWriteTransparent_t . . . 1084
unpack_uim_UnblockPin_t . . . 1085
unpack_uim_UnblockPinV2_t . . . 1086
unpack_uim_VerifyPin_t . . . 1087
unpack_voice_allCallStatusCallback_ind_t . . . 1088
unpack_voice_DTMFEventCallback_ind_t . . . 1090
unpack_voice_OTASPStatusCallback_ind_t . . . 1091
unpack_voice_SLQSOriginateUSSD_t . . . 1092
unpack_voice_SLQSVoiceAnswerCall_t . . . 1094
unpack_voice_SLQSVoiceBurstDTMF_t . . . 1095
unpack_voice_SLQSVoiceDialCall_t . . . 1095
unpack_voice_SLQSVoiceEndCall_t . . . 1096
unpack_voice_SLQSVoiceGetAllCallInfo_t . . . 1097
unpack_voice_SLQSVoiceGetCallBarring_t . . . 1101
unpack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 1102
unpack_voice_SLQSVoiceGetCallInfo_t . . . 1104
unpack_voice_SLQSVoiceGetCallWaiting_t . . . 1107

unpack_voice_SLQSVoiceGetCLIP_t . . . 1108
unpack_voice_SLQSVoiceGetCLIR_t . . . 1110
unpack_voice_SLQSVoiceGetCNAP_t . . . 1111
unpack_voice_SLQSVoiceGetCOLP_t . . . 1113
unpack_voice_SLQSVoiceGetCOLR_t . . . 1114
unpack_voice_SLQSVoiceGetConfig_t . . . 1115
unpack_voice_SLQSVoiceManageCalls_t . . . 1117
unpack_voice_SLQSVoiceSendFlash_t . . . 1118
unpack_voice_SLQSVoiceSetCallBarringPassword_t . . . 1119
unpack_voice_SLQSVoiceSetConfig_t . . . 1120
unpack_voice_SLQSVoiceSetSUPSService_t . . . 1122
unpack_voice_SLQSVoiceStartContDTMF_t . . . 1123
unpack_voice_SLQSVoiceStopContDTMF_t . . . 1123
unpack_voice_SLQSVoiceSUPSCallback_ind_t . . . 1124
unpack_voice_SUPSNotificationCallback_ind_t . . . 1127
unpack_voice_USSDNotificationCallback_ind_t . . . 1129
unpack_voice_USSDNoWaitIndicationCallback_ind_t . . . 1129
unpack_voice_VoiceInfoRecCallback_ind_t . . . 1130
unpack_voice_voicePrivacyChangeCallback_ind_t . . . 1133
unpack_wda_GetDataFormat_t . . . 1134
unpack_wda_GetQmapSettings_t . . . 1136
unpack_wda_SetDataFormat_t . . . 1137
unpack_wda_SetQmapSettings_t . . . 1141
unpack_wds_DHCPv4ClientLease_ind_t . . . 1142
unpack_wds_GetAutoconnect_t . . . 1142
unpack_wds_GetByteTotals_t . . . 1143
unpack_wds_GetConnectionRate_t . . . 1144
unpack_wds_GetDataBearerTechnology_t . . . 1145
unpack_wds_GetDefaultProfile_t . . . 1145
unpack_wds_GetDefaultProfileNum_t . . . 1148
unpack_wds_GetDefaultProfileV2_t . . . 1149
unpack_wds_GetDormancyState_t . . . 1151
unpack_wds_GetLastMobileIPError_t . . . 1152
unpack_wds_GetMobileIP_t . . . 1153
unpack_wds_GetMobileIPProfile_t . . . 1153
unpack_wds_GetPacketStatistics_t . . . 1155
unpack_wds_GetPacketStatus_t . . . 1157
unpack_wds_GetSessionDuration_t . . . 1159
unpack_wds_GetSessionDurationV2_t . . . 1160
unpack_wds_GetSessionState_t . . . 1161
unpack_wds_RMSetTransferStatistics_t . . . 1161
unpack_wds_SetMobileIPProfile_t . . . 1162
unpack_wds_SLQSCreateProfile_t . . . 1162
unpack_wds_SLQSDeleteProfile_t . . . 1163
unpack_wds_SLQSDUNCallInfoCallBack_ind_t . . . 1163
unpack_wds_SLQSGet3GPPConfigItem_t . . . 1165
unpack_wds_SLQSGetCurrDataSystemStat_t . . . 1167
unpack_wds_SLQSGetCurrentChannelRate_t . . . 1168
unpack_wds_SLQSGetDataBearerTechnology_t . . . 1169
unpack_wds_SLQSGetDUNCallInfo_t . . . 1170
unpack_wds_SLQSGetProfileSettings_t . . . 1173
unpack_wds_SLQSGetProfileSettingsV2_t . . . 1173
unpack_wds_SLQSGetRuntimeSettings_t . . . 1174
unpack_wds_SLQSModifyProfile_t . . . 1177
unpack_wds_SLQSSetIPFamilyPreference_t . . . 1178
unpack_wds_SLQSSetPacketSrvStatusCallback_t . . . 1178
unpack_wds_SLQSSetWdsEventCallback_ind_t . . . 1180
unpack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 1186

unpack_wds_SLQSSGetLoopback_t . . . 1187
unpack_wds_SLQSStartDataSession_t . . . 1188
unpack_wds_SLQSSwiProfileChangeCallback_Ind_t . . . 1189
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 1189
UnPackGetProfileSettingOut . . . 1192
UnPackGetProfileSettingOutV2 . . . 1192
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo . . . 1193
UnpackSwiAvmsEventReportConfig . . . 1195
UnpackSwiAvmsEventReportConnectionRequest . . . 1196
UnpackSwiAvmsEventReportDataSessionStatus . . . 1196
UnpackSwiAvmsEventReportHTTPStatus . . . 1197
UnpackSwiAvmsEventReportNotification . . . 1199
UnpackSwiAvmsEventReportPackageID . . . 1200
UnpackSwiAvmsEventReportRegStatus . . . 1200
UnpackSwiAvmsEventReportSessionType . . . 1201
UnpackSwiAvmsEventReportWAMSParamChange . . . 1201
unpackWdsProfileParam . . . 1202
unpackWdsProfileParamV2 . . . 1202
voice_airTimer . . . 1203
voice_allCallsAlphaIDInfo . . . 1203
voice_allCallsDiagInfo . . . 1204
voice_allCallsUUSInfo . . . 1204
voice_alphaIDInfo . . . 1205
voice_arrAlertingPattern . . . 1205
voice_arrAlertingType . . . 1206
voice_arrAlphaID . . . 1207
voice_arrCalledPartyNum . . . 1207
voice_arrCallEndReason . . . 1208
voice_arrCallInfo . . . 1209
voice_arrConnectPartyNum . . . 1209
voice_arrDiagInfo . . . 1210
voice_arrRedirPartyNum . . . 1210
voice_arrRemotePartyName . . . 1211
voice_arrRemotePartyNum . . . 1211
voice_arrSvcOption . . . 1212
voice_arrUUSInfo . . . 1213
voice_burstDTMFInfo . . . 1213
voice_calledPartyInfo . . . 1214
voice_calledPartySubAdd . . . 1215
voice_callerIDInfo . . . 1216
voice_callFwdTypeAndPlan . . . 1217
voice_callFWExtInfo . . . 1218
voice_callFWInfo . . . 1220
voice_callInfo . . . 1221
voice_callingPartyInfo . . . 1222
voice_ccSUPSType . . . 1224
voice_CLIPResp . . . 1225
voice_CLIRResp . . . 1225
voice_CNAPResp . . . 1226
voice_COLPResp . . . 1227
voice_COLRResp . . . 1227
voice_connectNumInfo . . . 1228
voice_CUGInfo . . . 1230
voice_curAMRConfig . . . 1230
voice_diagInfo . . . 1231
voice_DTMFInfo . . . 1232
voice_DTMFLengths . . . 1232
voice_ECTNum . . . 1233

voice_extDispRecInfo . . . 1234
voice_getAllCallInformation . . . 1235
voice_getAllCallRmtPtyName . . . 1235
voice_getAllCallRmtPtyNum . . . 1236
voice_getCallFWExtInfo . . . 1236
voice_getCallFWInfo . . . 1237
voice_lineCtrlInfo . . . 1237
voice_newPwdData . . . 1238
voice_NSSAudioCtrl . . . 1239
voice_peerNumberInfo . . . 1239
voice_prefVoiceSO . . . 1241
voice_redirNumInfo . . . 1243
voice_remotePartyName . . . 1245
voice_remotePartyNum . . . 1245
voice_roamTimer . . . 1246
voice_signalInfo . . . 1247
voice_SUPSInfo . . . 1248
voice_USSDNotificationNetworkInfo . . . 1248
voice_USSInfo . . . 1249
voice_UUSInfo . . . 1249
voice_VoiceUssdNoWaitAlphaIDTlv . . . 1251
voice_VoiceUssdNoWaitErrorTlv . . . 1251
voice_VoiceUssdNoWaitFailCauseTlv . . . 1252
voice_VoiceUssdNoWaitUSSINFOTlv . . . 1252
wds_channelRate . . . 1253
wds_ChannelRateTlv . . . 1253
wds_ConnStatusTlv . . . 1254
wds_currNetworkInfo . . . 1254
wds_DataBearTechTlv . . . 1255
wds_DataULongLongTlv . . . 1256
wds_DataULongTlv . . . 1257
wds_DHCPLeaseOptTlv . . . 1257
wds_DHCPLeaseStateTlv . . . 1258
wds_DHCPOpt . . . 1258
wds_DHCPProfileIdTlv . . . 1259
wds_DHCPv4HWConfig . . . 1259
wds_DHCPv4Option . . . 1260
wds_DHCPv4OptionList . . . 1261
wds_DHCPv4ProfileId . . . 1261
wds_Domain . . . 1262
wds_DomainNameList . . . 1262
wds_DormStatTlv . . . 1263
wds_GPRSQoS . . . 1263
wds_IPv4AdTlv . . . 1264
wds_IPV6AddressInfo . . . 1265
wds_IPV6GWAddressInfo . . . 1265
wds_LastMdmCallEndRsnTlv . . . 1266
wds_PCSCFFQDNAddress . . . 1266
wds_PCSCFFQDNAddressList . . . 1267
wds_PCSCFIPv4ServerAddressList . . . 1267
wds_profileChange . . . 1268
wds_ProfileIdentifier . . . 1268
wds_profileInfo . . . 1269
wds_RXBytesOKTlv . . . 1269
wds_sourceOfChange . . . 1270
wds_transferStatInd . . . 1270
wds_TrStatInd . . . 1271
wds_TXBytesOKTlv . . . 1271

wds_UMTSMinQoS 1272
wdsDhcpv4HwConfig 1274
wdsDhcpv4Option 1275
wdsDhcpv4OptionList 1275
wdsDhcpv4ProfileId 1276

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

apdoxypages.c
Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages . . . 1277
audio.h . . . 1277
BasicConnectDeviceService.h . . . 1283
cat.h . . . 1284
common.h . . . 1288
ctl.h . . . 1294
CtlService.h . . . 1296
DirectLayer.h . . . 1298
dms.h . . . 1299
dsd.h . . . 1360
fms.h . . . 1367
ims.h . . . 1370
imsa.h . . . 1383
lite-fw.h . . . 1388
loc.h . . . 1402
MbimIndicator.h . . . 1426
MbimInternal.h . . . 1428
MbimLayer.h . . . 1429
MbimSyncObject.h . . . 1430
MbimTransaction.h . . . 1433
MbimTransport.h . . . 1434
nas.h . . . 1438
pds.h . . . 1511
qaGobiApiTableBandClasses.h
Network Access Service API Band Classes table . . . 1533
qaGobiApiTableCallControlReturnReasons.h
Call Control Return Reasons table . . . 1536
qaGobiApiTableCallEndReasons.h
Wireless Data Service Call End Reasons . . . 1537
qaGobiApiTableCarrierCodes.h
Carrier Codes table . . . 1553
qaGobiApiTableCodingScheme.h
Data Coding Scheme . . . 1555
qaGobiApiTableFwDldErrorCodes.h
FOTA update agent errors and FDT/SSDP reported errors . . . 1558

qaGobiApiTableGpsCapabilityCodes.h
Position Determination Service API GPS Capability Codes . . . 1562
qaGobiApiTablePowerModes.h
Device Management Service API Power Modes table . . . 1562
qaGobiApiTableRadioInterfaces.h
Network Access Service API Radio Interfaces table . . . 1563
qaGobiApiTableRegionCodes.h
Region Codes table . . . 1564
qaGobiApiTableServiceOptions.h
Voice Service Options . . . 1564
qaGobiApiTableSupServiceInfoClasses.h
Voice Supplementary Service Information Classes . . . 1566
qaGobiApiTableSwiAudio.h
Swi Audio related tables . . . 1567
qaGobiApiTableSwiOMADMSessionStatus.h
Session Status table . . . 1567
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h
Update Complete Status table . . . 1568
qaGobiApiTableVoiceCallEndReasons.h
Voice Service Call and supplementary services end reasons . . . 1570
qmerrno.h . . . 1576
QmiService.h . . . 1583
QmiSyncObject.h . . . 1587
QmiTransaction.h . . . 1590
QmuxInternal.h . . . 1591
QmuxTransport.h . . . 1593
qos.h . . . 1596
rms.h . . . 1605
sar.h . . . 1607
sms.h . . . 1609
swiaudio.h . . . 1632
swiavms.h . . . 1641
SwiDataTypes.h
SWI data types . . . 1655
swidms.h . . . 1657
swiloc.h . . . 1666
swioma.h . . . 1668
swiomaext.h . . . 1678
switype_256bit.h . . . 1684
SWIWWANCMAPI.h . . . 1686
tmd.h . . . 1686
uim.h . . . 1690
voice.h . . . 1722
wda.h . . . 1754
wds.h . . . 1758

Chapter 6

Module Documentation

6.1 QMI pack/unpack (lite-qmi)

Files

- file common.h
- file dms.h
- file loc.h
- file nas.h
- file qos.h
- file sms.h
- file swioma.h
- file uim.h
- file wds.h
- file fms.h
- file swiloc.h
- file pds.h
- file rms.h
- file sar.h
- file swiavms.h
- file voice.h
- file cat.h
- file imsa.h
- file tmd.h
- file swiomaext.h
- file audio.h
- file swiaudio.h
- file ims.h
- file swidms.h
- file ctl.h
- file wda.h
- file dsd.h

6.1.1 Detailed Description

6.2 Streaming Download Protocol (lite-fw)

Files

- file lite-fw.h

6.2.1 Detailed Description

6.3 QMUX layer (lite-qmux)

Files

- file CtlService.h
- file DirectLayer.h
- file MbimLayer.h
- file QmiService.h
- file QmiSyncObject.h
- file QmiTransaction.h
- file QmuxTransport.h

6.3.1 Detailed Description

6.4 MBIM layer (lite-mbim)

Files

- file BasicConnectDeviceService.h
- file MbimIndicator.h
- file MbimSyncObject.h
- file MbimTransaction.h
- file MbimTransport.h

6.4.1 Detailed Description

Chapter 7

Namespace Documentation

7.1 Tables Namespace Reference

7.1.1 Detailed Description

Tables referenced in the API function headers:

- Table 1 - Call End Reason Codes The reason a call (either in process or connected) was ended. qaGobiApiTableCallEndReasons.h
- Table 2 - Carrier codes List of carrier identification codes. qaGobiApiTableCarrierCodes.h
- Table 3 - Region codes List of region identification codes. qaGobiApiTableRegionCodes.h
- Table 4 - GPS capability codes List of GPS capability codes. qaGobiApiTableGpsCapabilityCodes.h
- Table 5 - Radio Interfaces List of radio interface technologies. qaGobiApiTableRadioInterfaces.h
- Table 6 - Band classes List of band classes. qaGobiApiTableBandClasses.h
- Table 7 - Power modes List of operating modes. qaGobiApiTablePowerModes.h
- Table 8 - Service Options List of Service Options. qaGobiApiTableServiceOptions.h
- Table 9 - Voice Call End Reason List of Voice Call End Reason. qaGobiApiTableVoiceCallEndReasons.h
- Table 10 - Data Coding Scheme List of Data Coding Scheme. qaGobiApiTableCodingScheme.h
- Table 11 - Call Control Return Reasons List of Voice Call Control Return Reasons. qaGobiApiTableCallControlReturnReasons.h
- Table 12 - Supplementary Service Information Classes List of Voice Supplementary Service Information Classes. qaGobiApiTableSupServiceInfoClasses.h
- Table 13 - Audio Calibration Data Base(ACDB) Device List of ACDB devices. qaGobiApiTableSwiAudio.h
- Table 14 - PIFACE List of physical interfaces. qaGobiApiTableSwiAudio.h

Chapter 8

Data Structure Documentation

8.1 _litefw_FirmwareFileInfo Struct Reference

Data Fields

- char fullPath [512]
- uint8_t imageMask
- enum litefw_imagetype headerType
- enum litefw_fileimgtype imageType
- char modelIdStr [20+1]
- char partNoStr [20+1]
- char skuStr [20+1]
- char packageIdStr [20+1]
- char carrierStr [20+1]
- char priVersionStr [20+1]
- char versionStr [84]
- char releaseDate [8]

8.1.1 Detailed Description

This structure provided more detailed information of the provided firmware file

Parameters

fullPath	• full path of the file
imagemask	• Bitmask provides type of file – bit0 - cwe – bit1 - nvu – bit2 - spk
headerType	• see litefw_imagetype
imageType	• see litefw_fileimgtype

modelIdStr	• device model id
partNoStr	• part number id
skuStr	• sku id
packageIdStr	• cwe sierra package id
carrierStr	• carrier id
priVersionStr	• pri version
versionStr	• firmware version
releaseDate	• release date of the file

8.1.2 Field Documentation

8.1.2.1 char _litefw_FirmwareFileInfo::carrierStr[20+1]

8.1.2.2 char _litefw_FirmwareFileInfo::fullPath[512]

8.1.2.3 enum litefw_imagetype _litefw_FirmwareFileInfo::headerType

8.1.2.4 uint8_t _litefw_FirmwareFileInfo::imageMask

8.1.2.5 enum litefw_fileimgtype _litefw_FirmwareFileInfo::imageType

8.1.2.6 char _litefw_FirmwareFileInfo::modelIdStr[20+1]

8.1.2.7 char _litefw_FirmwareFileInfo::packageIdStr[20+1]

8.1.2.8 char _litefw_FirmwareFileInfo::partNoStr[20+1]

8.1.2.9 char _litefw_FirmwareFileInfo::priVersionStr[20+1]

8.1.2.10 char _litefw_FirmwareFileInfo::releaseDate[8]

8.1.2.11 char _litefw_FirmwareFileInfo::skuStr[20+1]

8.1.2.12 char _litefw_FirmwareFileInfo::versionStr[84]

8.2 _litefw_FirmwareInfo_ Struct Reference

Data Fields

- char szModelid_str [20+1]
- char szFwversion_str [20+1]
- char szSku_str [20+1]
- char szPackageid_str [20+1]
- char szCarrier_str [20+1]
- char szCarrierPriversion_str [20+1]

8.2.1 Detailed Description

This structure contains information of the provided firmware file

Parameters

szModelid_str	• Model Name String
szFwversion_str	• Firmware Version String.
szSku_str	• SKU String.
szPackageid_str	• Package ID String.
szCarrier_str	• Carrier String.
szCarrier-Priversion_str	• Carrier PRI Version String.

8.2.2 Field Documentation

8.2.2.1 char _litefw_FirmwareInfo_::szCarrier_str[20+1]

8.2.2.2 char _litefw_FirmwareInfo_::szCarrierPriversion_str[20+1]

8.2.2.3 char _litefw_FirmwareInfo_::szFwversion_str[20+1]

8.2.2.4 char _litefw_FirmwareInfo_::szModelid_str[20+1]

8.2.2.5 char _litefw_FirmwareInfo_::szPackageid_str[20+1]

8.2.2.6 char _litefw_FirmwareInfo_::szSku_str[20+1]

8.3 _litefw_FirmwarePartNo_ Struct Reference

Data Fields

- char szPartno_str [20+1]

8.3.1 Detailed Description

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

szPartno_str	• Part Number of the provided firmware image

8.3.2 Field Documentation

8.3.2.1 char _litefw_FirmwarePartNo_::szPartno_str[20+1]

8.4 altSrcInfo_t Struct Reference

Data Fields

- uint32_t source
- uint32_t linkage
- uint32_t coverage

8.4.1 Detailed Description

This structure specifies information regarding the altitude source

Parameters

source	• Specifies the source of the altitude • Valid values – 0 - Source is unknown – 1 - GPS is the source – 2 - Cell ID provided the source – 3 - Source is enhanced cell ID – 4 - Wi-Fi is the source – 5 - Terrestrial source – 6 - Hybrid terrestrial source – 7 - Altitude database is the source – 8 - Barometric altimeter is the source – 9 - Other sources
linkage	• Specifies the dependency between the horizontal and altitude position components • Valid values – 0 - Not specified – 1 - Fully interdependent – 2 - Depends on latitude and longitude – 3 - Fully independent

coverage	• Specifies the region of uncertainty. • Valid values – 0 - Not specified – 1 - Altitude uncertainty is valid at the injected horizontal position coordinates only – 2 - Altitude uncertainty applies to the position of the device regardless of horizontal position

8.4.2 Field Documentation

8.4.2.1 uint32_t altSrcInfo_t::coverage

8.4.2.2 uint32_t altSrcInfo_t::linkage

8.4.2.3 uint32_t altSrcInfo_t::source

8.5 appStats Struct Reference

Data Fields

- uint8_t appType
- uint8_t appState
- uint8_t persoState
- uint8_t persoFeature
- uint8_t persoRetries
- uint8_t persoUnblockRetries
- uint8_t aidLength
- uint8_t aidVal [255]
- uint8_t univPin
- uint8_t pin1State
- uint8_t pin1Retries
- uint8_t puk1Retries
- uint8_t pin2State
- uint8_t pin2Retries
- uint8_t puk2Retries

8.5.1 Detailed Description

This structure contains Application Status Information loaded on the card.

Parameters

appType	• Indicates the type of the application. – 0 - Unknown – 1 - SIM card – 2 - USIM application – 3 - RUIM card – 4 - CSIM application – 5 - ISIM application • Other values are reserved for the future and are to be handled as “Unknown”.

appState	• Indicates the state of the application. – 0 - Unknown – 1 - Detected – 2 - PIN1 or UPIN is required – 3 - PUK1 or PUK for UPIN is required – 4 - Personalization state must be checked – 5 - PIN1 is blocked – 6 - Illegal – 7 - Ready
persoState	• Indicates the state of the personalization for the application. – 0 - Unknown – 1 - Personalization operation is in progress – 2 - Ready – 3 - Personalization code is required – 4 - PUK for personalization code is required – 5 - Permanently blocked
persoFeature	• Indicates the personalization feature. • This applies only when a personalization code is required to deactivate or unblock personalization. – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization – 11 - Unknown
persoRetries	• Indicates the number of retries remaining to disable the personalization.
persoUnblock-Retries	• Indicates the number of retries remaining to unblock the personalization.
aidLength	• Number of sets of the following elements. i.e. aidVal • If zero(0) then no aidVal information exists.
aidVal[MAX_DESCRIPTION_LENGTH]	• Application identifier value.

univPin	• Indicates whether UPIN replaces PIN1. – 0 - PIN1 is used – 1 - UPIN replaces PIN1
pin1State	• Indicates the state of PIN1. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin1Retries	• Indicates the number of retries remaining to verify PIN1.
puk1Retries	• Indicates the number of retries remaining to unblock PIN1.
pin2State	• Indicates the state of PIN2. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin2Retries	• Indicates the number of retries remaining to verify PIN2.
puk2Retries	• Indicates the number of retries remaining to unblock PIN2.

8.5.2 Field Documentation

8.5.2.1 uint8_t appStats::aidLength

8.5.2.2 uint8_t appStats::aidVal[255]

8.5.2.3 uint8_t appStats::appState

8.5.2.4 uint8_t appStats::appType

8.5.2.5 uint8_t appStats::persoFeature

8.5.2.6 uint8_t appStats::persoRetries

8.5.2.7 uint8_t appStats::persoState

8.5.2.8 uint8_t appStats::persoUnblockRetries

8.5.2.9 uint8_t appStats::pin1Retries

8.5.2.10 uint8_t appStats::pin1State

8.5.2.11 uint8_t appStats::pin2Retries

8.5.2.12 uint8_t appStats::pin2State

8.5.2.13 uint8_t appStats::puk1Retries

8.5.2.14 uint8_t appStats::puk2Retries

8.5.2.15 uint8_t appStats::univPin

8.6 audio_RXAGCList Struct Reference

Data Fields

- uint16_t * pRXStaticGain
- uint16_t * pRXAIG
- uint16_t * pRXExpThres
- uint16_t * pRXExpSlope
- uint16_t * pRXComprThres
- uint16_t * pRXComprSlope

8.6.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_RXAGCLIST.

Parameters

pRXStaticGain	• RX pre-compressor static gain
pRXAIG	• RX pre-compressor gain selection flag
pRXExpThres	• RX expansion threshold
pRXExpSlope	• RX expansion slope
pRXComprThres	• RX compression threshold
pRXComprSlope	• RX compression slope

8.6.2 Field Documentation

8.6.2.1 uint16_t* audio_RXAGCList::pRXAIG

8.6.2.2 uint16_t* audio_RXAGCList::pRXComprSlope

8.6.2.3 uint16_t* audio_RXAGCList::pRXComprThres

8.6.2.4 uint16_t* audio_RXAGCList::pRXExpSlope

8.6.2.5 uint16_t* audio_RXAGCList::pRXExpThres

8.6.2.6 uint16_t* audio_RXAGCList::pRXStaticGain

8.7 audio_RXAVCList Struct Reference

Data Fields

- uint16_t * pAVRXAVCSens
- uint16_t * pAVRXAVCHeadroom

8.7.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_RXAVCLIST.

Parameters

pAVRXAVC-Sens	• AVC variation from nominal sensitivity
pAVRXAVC-Headroom	• AVC headroom

8.7.2 Field Documentation

8.7.2.1 uint16_t* audio_RXAVCList::pAVRXAVCHeadroom

8.7.2.2 uint16_t* audio_RXAVCList::pAVRXAVCSens

8.8 audio_RXPCMIIRFltr Struct Reference

Data Fields

- uint16_t * pFlag
- uint16_t * pStageCnt
- uint8_t * pStage0Val
- uint8_t * pStage1Val
- uint8_t * pStage2Val
- uint8_t * pStage3Val
- uint8_t * pStage4Val

8.8.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_RXPCMIIRFLTR.

Parameters

pFlag	• Flag – 0x0000 - IIR filter disable – 0xffff - IIR filter enable
pStageCnt	• Stage Count – 0-4
pStage0Val	• A 20 BYTE sized parameter indicating Stage 0 value – A1 – A2 – B0 – B1 – B2
pStage1Val	• A 20 BYTE sized parameter indicating Stage 1 value – A1 – A2 – B0 – B1 – B2
pStage2Val	• A 20 BYTE sized parameter indicating Stage 2 value – A1 – A2 – B0 – B1 – B2
pStage3Val	• A 20 BYTE sized parameter indicating Stage 3 value – A1 – A2 – B0 – B1 – B2
pStage4Val	• A 20 BYTE sized parameter indicating Stage 4 value – A1 – A2 – B0 – B1 – B2

8.8.2 Field Documentation

8.8.2.1 uint16_t* audio_RXPCMIIRFltr::pFlag

8.8.2.2 uint8_t* audio_RXPCMIIRFltr::pStage0Val

8.8.2.3 uint8_t* audio_RXPCMIIRFltr::pStage1Val

8.8.2.4 uint8_t* audio_RXPCMIIRFltr::pStage2Val

8.8.2.5 uint8_t* audio_RXPCMIIRFltr::pStage3Val

8.8.2.6 uint8_t* audio_RXPCMIIRFltr::pStage4Val

8.8.2.7 uint16_t* audio_RXPCMIIRFltr::pStageCnt

8.9 audio_TXAGCList Struct Reference

Data Fields

- uint16_t * pTXStaticGain
- uint16_t * pTXAIG
- uint16_t * pTXExpThres
- uint16_t * pTXExpSlope
- uint16_t * pTXComprThres
- uint16_t * pTXComprSlope

8.9.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_TXAGCLIST.

Parameters

pTXStaticGain	• TX pre-compressor static gain
pTXAIG	• TX pre-compressor gain selection flag
pTXExpThres	• TX expansion threshold
pTXExpSlope	• TX expansion slope
pTXComprThres	• TX compression threshold
pTXComprSlope	• TX compression slope

8.9.2 Field Documentation

8.9.2.1 uint16_t* audio_TXAGCList::pTXAIG

8.9.2.2 uint16_t* audio_TXAGCList::pTXComprSlope

8.9.2.3 uint16_t* audio_TXAGCList::pTXComprThres

8.9.2.4 uint16_t* audio_TXAGCList::pTXExpSlope

8.9.2.5 uint16_t* audio_TXAGCList::pTXExpThres

8.9.2.6 uint16_t* audio_TXAGCList::pTXStaticGain

8.10 audio_TXPCMIIRFltr Struct Reference

Data Fields

- uint16_t * pFlag
- uint16_t * pStageCnt
- uint8_t * pStage0Val
- uint8_t * pStage1Val
- uint8_t * pStage2Val
- uint8_t * pStage3Val
- uint8_t * pStage4Val

8.10.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_TXPCMIIRFLTR.

Parameters

pFlag	• Flag – 0x0000 - IIR filter disable – 0xffff - IIR filter enable
pStageCnt	• Stage Count – 0-4
pStage0Val	• A 20 BYTE sized parameter indicating Stage 0 value – A1 – A2 – B0 – B1 – B2

pStage1Val	• A 20 BYTE sized parameter indicating Stage 1 value – A1 – A2 – B0 – B1 – B2
pStage2Val	• A 20 BYTE sized parameter indicating Stage 2 value – A1 – A2 – B0 – B1 – B2
pStage3Val	• A 20 BYTE sized parameter indicating Stage 3 value – A1 – A2 – B0 – B1 – B2
pStage4Val	• A 20 BYTE sized parameter indicating Stage 4 value – A1 – A2 – B0 – B1 – B2

8.10.2 Field Documentation

8.10.2.1 uint16_t* audio_TXPCMIIRFltr::pFlag

8.10.2.2 uint8_t* audio_TXPCMIIRFltr::pStage0Val

8.10.2.3 uint8_t* audio_TXPCMIIRFltr::pStage1Val

8.10.2.4 uint8_t* audio_TXPCMIIRFltr::pStage2Val

8.10.2.5 uint8_t* audio_TXPCMIIRFltr::pStage3Val

8.10.2.6 uint8_t* audio_TXPCMIIRFltr::pStage4Val

8.10.2.7 uint16_t* audio_TXPCMIIRFltr::pStageCnt

8.11 CarrierImage_t Struct Reference

Data Fields

- uint32_t m_nCarrierId
- uint32_t m_nFolderId
- uint32_t m_nStorage
- uint8_t m_FwImageId [100]
- uint8_t m_FwBuildId [100]
- uint8_t m_PriImageId [100]
- uint8_t m_PriBuildId [100]

8.11.1 Detailed Description

This structure contains the Carrier Image parameters.

Parameters

m_nCarrierId	• Unique numeric carrier ID indicating the carrier that the following images belong to
m_nFolderId	• Unique numeric folder ID indicating the folder where the images should reside on the host storage.
m_nStorage	• Information of storage type • Values – 0 - Device – 1 - Host
m_FwImageId	• Firmware image ID
m_FwBuildId	• Firmware build ID
m_PriImageId	• PRI image ID
m_PriBuildId	• PRI build ID

8.11.2 Field Documentation

8.11.2.1 uint8_t CarrierImage_t::m_FwBuildId[100]

8.11.2.2 uint8_t CarrierImage_t::m_FwImageId[100]

8.11.2.3 uint32_t CarrierImage_t::m_nCarrierId

8.11.2.4 uint32_t CarrierImage_t::m_nFolderId

8.11.2.5 uint32_t CarrierImage_t::m_nStorage

8.11.2.6 uint8_t CarrierImage_t::m_PriBuildId[100]

8.11.2.7 uint8_t CarrierImage_t::m_PriImageId[100]

8.12 cat_AlPhaIdentifierTlv Struct Reference

Data Fields

- uint8_t ReferenceID
- uint16_t AlphaIDLength
- uint8_t AlphaID [255]

8.12.1 Detailed Description

structure used to store all Alpha Identifier parameters.

Parameters

ReferenceID	- proactive command type that included the alpha identifier - 0x01; sends SMS proactive command
AlphaIDLength	- length of AlphaID (in bytes)
AlphaID	- alpha identifier, encoded as in ETSI TS 102 223 [Section 8.2]

8.12.2 Field Documentation

8.12.2.1 uint8_t cat_AlPhaIdentifierTlv::AlphaID[255]

8.12.2.2 uint16_t cat_AlPhaIdentifierTlv::AlphaIDLength

8.12.2.3 uint8_t cat_AlPhaIdentifierTlv::ReferenceID

8.13 cat_commonEventTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EventID
- uint16_t EventLength
- union cat_currentCatEvent CatEvent

8.13.1 Detailed Description

Structure used to store all Common CAT Event TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
EventID	- Event ID. Can be any of the following 16 - Display Text 17 - Get In-Key 18 - Get Input 19 - Setup Menu 20 - Select Item 21 - Send SMS - Alpha Identifier 22 - Setup Event List 23 - Setup Idle Mode Text 24 - Language Notification 25 - Refresh 26 - End Proactive Session
EventLength	- Length of pData (in Bytes)
CatEvent	- Structure to the Data specific to the CAT event ID

8.13.2 Field Documentation

8.13.2.1 union cat_currentCatEvent cat_commonEventTlv::CatEvent

8.13.2.2 uint8_t cat_commonEventTlv::EventID

8.13.2.3 uint16_t cat_commonEventTlv::EventLength

8.13.2.4 uint8_t cat_commonEventTlv::TlvPresent

8.14 cat_currentCatEvent Union Reference

Data Fields

- struct cat_EventIDDataTlv CatEvIDData
- struct cat_AlPhaIdentifierTlv CatAlphaIdtfr
- struct cat_EventListTlv CatEventLst
- struct cat_RefreshTlv CatRefresh
- struct cat_EndProactiveSessionTlv CatEndPS

8.14.1 Detailed Description

Union used to represent the current CAT Event Data. Choose the structure based on the EventID received.

- Use cat_EventIDDataTlv if the Event ID is any of the below.
  - 16
  - 17
  - 18
  - 19
  - 20
  - 23
  - 24
- Use cat_AlPhaIdentifierTlv if the Event ID is 21
- Use cat_EventListTlv if the Event ID is 22
- Use cat_RefreshTlv if the Event ID is 25
- Use cat_EndProactiveSessionTlv if the Event ID is 26

8.14.2 Field Documentation

8.14.2.1 struct cat_AlPhaIdentifierTlv cat_currentCatEvent::CatAlphaIdtfr

8.14.2.2 struct cat_EndProactiveSessionTlv cat_currentCatEvent::CatEndPS

8.14.2.3 struct cat_EventListTlv cat_currentCatEvent::CatEventLst

8.14.2.4 struct cat_EventIDDataTlv cat_currentCatEvent::CatEvIDData

8.14.2.5 struct cat_RefreshTlv cat_currentCatEvent::CatRefresh

8.15 cat_EndProactiveSessionTlv Struct Reference

Data Fields

- uint8_t EndProactiveSession

8.15.1 Detailed Description

structure used to store End Proactive Session event parameters.

Parameters

EndProactive-Session	- The proactive session end type values are: • 0x01 - End proactive session command type received from the card • 0x02 - End proactive session internal to ME

8.15.2 Field Documentation

8.15.2.1 uint8_t cat_EndProactiveSessionTlv::EndProactiveSession

8.16 cat_EventIDDataTlv Struct Reference

Data Fields

- uint32_t ReferenceID
- uint16_t DataLength
- uint8_t Data [255]

8.16.1 Detailed Description

structure used to store all Common CAT Event parameters.

Parameters

ReferenceID	- proactive command reference ID.
DataLength	- length of pData (in Bytes)
Data	- command specific to the CAT event ID, encoded as in ETSI TS 102 223 [Section 6.6.X]

8.16.2 Field Documentation

8.16.2.1 uint8_t cat_EventIDDataTlv::Data[255]

8.16.2.2 uint16_t cat_EventIDDataTlv::DataLength

8.16.2.3 uint32_t cat_EventIDDataTlv::ReferenceID

8.17 cat_EventListTlv Struct Reference

Data Fields

- uint32_t SetupEventList

8.17.1 Detailed Description

structure used to store all Event List parameters.

Parameters

SetupEventList	- Setup event list bit mask • 0x00000001 - User Activity Notify • 0x00000002 - Idle Screen Available • 0x00000004 - Lang Selection Notify Each set bit indicates the availability of the corresponding event in Setup Event list proactive command; all unlisted bits are reserved for future use and will be ignored

8.17.2 Field Documentation

8.17.2.1 uint32_t cat_EventListTlv::SetupEventList

8.18 cat_RefreshTlv Struct Reference

Data Fields

- uint16_t RefreshMode
- uint8_t RefreshStage

8.18.1 Detailed Description

structure used to store all Refresh Event parameters.

Parameters

RefreshMode	- The Refresh Event as in ETSI TS 102 223 [Section 8.6]
RefreshStage	- Stage of a refresh procedure • 0x01 - Refresh start • 0x02 - Refresh success • 0x03 - Refresh failed

8.18.2 Field Documentation

8.18.2.1 uint16_t cat_RefreshTlv::RefreshMode

8.18.2.2 uint8_t cat_RefreshTlv::RefreshStage

8.19 cdmaSSInfo Struct Reference

Data Fields

- int8_t rssi
- int16_t ecio

8.19.1 Detailed Description

This structure contains the parameters for CDMA/WCDMA Signal Strength Information

Parameters

rssi	• RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal.
ecio	• ECIO value representing negative 0.5 dBm increments, i.e., 2 means -1 dBm (14 means -7 dBm, 63 means -31.5 dBm).

8.19.2 Field Documentation

8.19.2.1 int16_t cdmaSSInfo::ecio

8.19.2.2 int8_t cdmaSSInfo::rssi

8.20 connectionStatus Struct Reference

Data Fields

- uint8_t MDMConnStatus
- uint64_t MDMCallDuration

8.20.1 Detailed Description

This structure contains modem connection status

Parameters

MDMConn-Status	• Current link status – 0x01 - DISCONNECTED – 0x02 - CONNECTED – 0xff - Invalid data.
MDMCall-Duration	• Call duration in milliseconds. • If the modem connection status is connected, this represent the duration of the current DUN call. • If the modem connection status is disconnected, this represents the duration of the last DUN call since the device was powered up (zero, if no call has been made or if the last call was not DUN). – 0xffffffffffffffff - Invalid data.

8.20.2 Field Documentation

8.20.2.1 uint64_t connectionStatus::MDMCallDuration

8.20.2.2 uint8_t connectionStatus::MDMConnStatus

8.21 crashInfoParams Struct Reference

Data Fields

- uint8_t crashStatus
- crashInformation crashInfo

8.21.1 Detailed Description

This structure contains crash information parameters

Parameters

crashStatus[OUT]	• Device Crash Status • 0 - no crash • 1 - crash has occurred
crashInfo[OUT]	• See crashInformation

8.21.2 Field Documentation

8.21.2.1 crashInformation crashInfoParams::crashInfo

8.21.2.2 uint8_t crashInfoParams::crashStatus

8.22 crashInformation Struct Reference

Data Fields

- uint16_t numCrashes
- uint32_t crashId
- uint32_t crashData
- uint16_t crashStrlen
- char crashString [255]
- uint16_t gcdumpStrlen
- char gcdumpString [1024]

8.22.1 Detailed Description

This structure contains crash information

Parameters

numCrashes[OUT]	• number of instances of the remaining fields
crashId[OUT]	• ramdom crash id assigned at crash

crashData	• crash data[OUT]
crashStrlen[IN/-OUT]	• length of the crashString field as an input, length of the crashString field returned by API as an OUTPUT
crashString	• crash string
gcdumpStrlen[O-UT]	• length of the gcdumpString field as an input, length of the gcdumpString field returned by API as an OUTPUT
gcdumpString[O-UT]	• gcdump string for the crash

8.22.2 Field Documentation

8.22.2.1 uint32_t crashInformation::crashData

8.22.2.2 uint32_t crashInformation::crashId

8.22.2.3 char crashInformation::crashString[255]

8.22.2.4 uint16_t crashInformation::crashStrlen

8.22.2.5 char crashInformation::gcdumpString[1024]

8.22.2.6 uint16_t crashInformation::gcdumpStrlen

8.22.2.7 uint16_t crashInformation::numCrashes

8.23 CtlService Struct Reference

Data Fields

- QmiService service

8.23.1 Detailed Description

This structure abstracts a CTL QMI service. It wraps a QMI service which is initialized to type CTL. Synchronous methods are provided for getting and releasing client IDs on behalf of other (regular) QMI services.

Parameters

service	• QMI service object, initialized to be of type CTL.

8.23.2 Field Documentation

8.23.2.1 QmiService CtlService::service

8.24 CtlTransactionHdr Struct Reference

Data Fields

- uint8_t flags
- uint8_t transactionId

8.24.1 Field Documentation

8.24.1.1 uint8_t CtlTransactionHdr::flags

8.24.1.2 uint8_t CtlTransactionHdr::transactionId

8.25 currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.25.1 Detailed Description

Network information structure

Parameters

NetworkType	• Values: – 0 - 3GPP – 1 - 3GPP2
RATMask	RAT Mask
SOMask	SO Mask

8.25.2 Field Documentation

8.25.2.1 uint8_t currNetworkInfo::NetworkType

8.25.2.2 uint32_t currNetworkInfo::RATMask

8.25.2.3 uint32_t currNetworkInfo::SOMask

8.26 dms_ActivationStatusTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t activationStatus

8.26.1 Detailed Description

Activation Status Tlv

Parameters

TlvPresent	• Tlv Present
activationStatus	• Service Activation Code – 0 - Service not activated – 1 - Service activated – 2 - Activation connecting – 3 - Activation connected – 4 - OTASP security authenticated – 5 - OTASP NAM downloaded – 6 - OTASP MDN downloaded – 7 - OTASP IMSI downloaded – 8 - OTASP PRL downloaded – 9 - OTASP SPC downloaded – 10 - OTASP settings committed

8.26.2 Field Documentation

8.26.2.1 uint32_t dms_ActivationStatusTlv::activationStatus

8.26.2.2 uint16_t dms_ActivationStatusTlv::TlvPresent

8.27 dms_devCaps Struct Reference

Data Fields

- uint32_t MaxTXChannelRate
- uint32_t MaxRXChannelRate
- uint8_t DataServiceCapability
- uint8_t SimCapability
- uint8_t RadioIfacesSize
- uint8_t RadioIfaces [255]

8.27.1 Detailed Description

This structure contains the dms_devCaps parameters.

Parameters

MaxTXChannel-Rate	• Maximum Tx channel rate in bits per second
MaxRXChannel-Rate	• Maximum Rx channel rate in bits per second

DataService-Capability	• DMS_DATA_CAP_NONE - 0x00 • DMS_DATA_CAP_CS_ONLY-circuit switched only - 0x01 • DMS_DATA_CAP_PS_ONLY-packet switched only - 0x02 • DMS_DATA_CAP_SIMUL_CS_AND_PS - 0x03 • DMS_DATA_CAP_NONSIMUL_CS_AND_PS - 0x04
SimCapability	• SIM_NOT_SUPPORTED - 0x01 • SIM_SUPPORTED - 0x02
RadioIfacesSize	-Number of radio interfaces
RadioIfaces	-List of radio interfaces, each byte will be one of following • DMS_RADIO_IF_1X- CDMA2000 1X - 0x01 • DMS_RADIO_IF_1X_EVDO- CDMA2000 HRPD - 0x02 • DMS_RADIO_IF_GSM- GSM - 0x04 • DMS_RADIO_IF_UMTS - 0x05 • DMS_RADIO_IF_LTE - 0x08 • DMS_RADIO_IF_TD - 0x09

8.27.2 Field Documentation

8.27.2.1 uint8_t dms_devCaps::DataServiceCapability

8.27.2.2 uint32_t dms_devCaps::MaxRXChannelRate

8.27.2.3 uint32_t dms_devCaps::MaxTXChannelRate

8.27.2.4 uint8_t dms_devCaps::RadioIfaces[255]

8.27.2.5 uint8_t dms_devCaps::RadioIfacesSize

8.27.2.6 uint8_t dms_devCaps::SimCapability

8.28 dms_devCurSubsCaps Struct Reference

Data Fields

- uint8_t CurSubsCapsLen
- uint64_t SubsCapList [32]

8.28.1 Detailed Description

This structure contains the dms_devCurSubsCaps response parameters.

Parameters

CurSubsCapLen	• Length of subscription list

SubsCapList	• Subscription cap list, each array element will be one of following – DMS_SUBS_CAPABILITY_AMPS -0x00000001 – DMS_SUBS_CAPABILITY_CDMA -0x00000002 – DMS_SUBS_CAPABILITY_HDR -0x00000004 – DMS_SUBS_CAPABILITY_GSM -0x00000008 – DMS_SUBS_CAPABILITY_WCDMA -0x00000010 – DMS_SUBS_CAPABILITY_LTE -0x00000020 – DMS_SUBS_CAPABILITY_TDS -0x00000040

8.28.2 Field Documentation

8.28.2.1 uint8_t dms_devCurSubsCaps::CurSubsCapsLen

8.28.2.2 uint64_t dms_devCurSubsCaps::SubsCapList[32]

8.29 dms_devMaxCfgListCaps Struct Reference

Data Fields

- uint8_t MaxSubs
- uint8_t MaxActive
- uint8_t DevCfgListLen
- dms_devSubsList SubsDevList [32]
- uint8_t CurIndex

8.29.1 Detailed Description

This structure contains the devMaxCfgListCaps parameters.

Parameters

MaxSubs	• Maximum number of subscriptions supported
MaxActive	• Maximum number of subscriptions active
DevCfgListLen	• Length of subscription config list
SubsDevList	• Subscription config list – See dms_devSubsList for more information
CurIndex	• Current Config Index

8.29.2 Field Documentation

8.29.2.1 uint8_t dms_devMaxCfgListCaps::CurIndex

8.29.2.2 uint8_t dms_devMaxCfgListCaps::DevCfgListLen

8.29.2.3 uint8_t dms_devMaxCfgListCaps::MaxActive

8.29.2.4 uint8_t dms_devMaxCfgListCaps::MaxSubs

8.29.2.5 dms_devSubsList dms_devMaxCfgListCaps::SubsDevList[32]

8.30 dms_devMaxSubsCaps Struct Reference

Data Fields

- uint8_t MaxSubsCapLen
- uint64_t MaxSubsList [32]

8.30.1 Detailed Description

This structure contains the dms_devMaxSubsCaps response parameters.

Parameters

TlvPresent	• Tlv Present or not
MaxSubsCap-Len	• Length of subscription list
MaxSubsList	• Subscription cap list, each array element will be one of following – DMS_SUBS_CAPABILITY_AMPS -0x00000001 – DMS_SUBS_CAPABILITY_CDMA -0x00000002 – DMS_SUBS_CAPABILITY_HDR -0x00000004 – DMS_SUBS_CAPABILITY_GSM -0x00000008 – DMS_SUBS_CAPABILITY_WCDMA -0x00000010 – DMS_SUBS_CAPABILITY_LTE -0x00000020 – DMS_SUBS_CAPABILITY_TDS -0x00000040

8.30.2 Field Documentation

8.30.2.1 uint8_t dms_devMaxSubsCaps::MaxSubsCapLen

8.30.2.2 uint64_t dms_devMaxSubsCaps::MaxSubsList[32]

8.31 dms_devMultiSimCaps Struct Reference

Data Fields

- uint8_t MaxSubs

- uint8_t SubsCfgListLen
- dms_devSubsCfgList SubsCfgList [32]

8.31.1 Detailed Description

This structure contains the dms_devMultiSimCaps parameters.

Parameters

MaxSubs	• Maximum number of subscriptions supported
SubsCfgListLen	• Length of subscription config list
SubsCfgList	• Subscription config list – See dms_devSubsCfgList for more information

8.31.2 Field Documentation

8.31.2.1 uint8_t dms_devMultiSimCaps::MaxSubs

8.31.2.2 dms_devSubsCfgList dms_devMultiSimCaps::SubsCfgList[32]

8.31.2.3 uint8_t dms_devMultiSimCaps::SubsCfgListLen

8.32 dms_devMultiSimVoiceDataCaps Struct Reference

Data Fields

- uint8_t MaxSubs
- uint8_t MaxActive

8.32.1 Detailed Description

This structure contains the dms_devMultiSimVoiceDataCaps parameters.

Parameters

MaxSubs	• Maximum number of subscriptions supported
MaxActive	• Maximum number of subscriptions active

8.32.2 Field Documentation

8.32.2.1 uint8_t dms_devMultiSimVoiceDataCaps::MaxActive

8.32.2.2 uint8_t dms_devMultiSimVoiceDataCaps::MaxSubs

8.33 dms_devSubsCfgList Struct Reference

Data Fields

- uint8_t MaxActive
- uint8_t SubsListLen
- uint64_t SubsList [32]

8.33.1 Detailed Description

This structure contains the dms_devSubsCfgList parameters.

Parameters

MaxActive	• Maximum number of subscriptions active
SubsListLen	• Length of subscription list
SubsList	• Subscription list, each array element will be one of following – DMS_SUBS_CAPABILITY_AMPS - 0x00000001 – DMS_SUBS_CAPABILITY_CDMA - 0x00000002 – DMS_SUBS_CAPABILITY_HDR - 0x00000004 – DMS_SUBS_CAPABILITY_GSM - 0x00000008 – DMS_SUBS_CAPABILITY_WCDMA - 0x00000010 – DMS_SUBS_CAPABILITY_LTE - 0x00000020 – DMS_SUBS_CAPABILITY_TDS - 0x00000040

8.33.2 Field Documentation

8.33.2.1 uint8_t dms_devSubsCfgList::MaxActive

8.33.2.2 uint64_t dms_devSubsCfgList::SubsList[32]

8.33.2.3 uint8_t dms_devSubsCfgList::SubsListLen

8.34 dms_devSubsFeatureModeCaps Struct Reference

Data Fields

- uint8_t SubsFeatureLen
- uint32_t SubsFeatureList [32]

8.34.1 Detailed Description

This structure contains the dms_devSubsFeatureModeCaps response parameters.

Parameters

SubsFeatureLen	• Length of subscription feature list
SubsFeatureList	• Subscription feature mode list, each array element will be one of following – DMS_DEVICE_SUBS_FEATURE_MODE_NORMAL -0 – DMS_DEVICE_SUBS_FEATURE_MODE_SGLTE -1 – DMS_DEVICE_SUBS_FEATURE_MODE_SVLTE -2 – DMS_DEVICE_SUBS_FEATURE_MODE_SRLTE -3 – DMS_DEVICE_SUBS_FEATURE_MODE_DUAL_MULTIMODE -4

8.34.2 Field Documentation

8.34.2.1 uint8_t dms_devSubsFeatureModeCaps::SubsFeatureLen

8.34.2.2 uint32_t dms_devSubsFeatureModeCaps::SubsFeatureList[32]

8.35 dms_devSubsList Struct Reference

Data Fields

- uint8_t SubsListLen
- uint64_t SubsList [32]

8.35.1 Detailed Description

This structure contains the dms_devSubsList parameters.

Parameters

SubsListLen	• Length of subscription list
SubsList	• Subscription list, each array element will be one of following – DMS_SUBS_CAPABILITY_AMPS -0x00000001 – DMS_SUBS_CAPABILITY_CDMA -0x00000002 – DMS_SUBS_CAPABILITY_HDR -0x00000004 – DMS_SUBS_CAPABILITY_GSM -0x00000008 – DMS_SUBS_CAPABILITY_WCDMA -0x00000010 – DMS_SUBS_CAPABILITY_LTE -0x00000020 – DMS_SUBS_CAPABILITY_TDS -0x00000040

8.35.2 Field Documentation

8.35.2.1 uint64_t dms_devSubsList::SubsList[32]

8.35.2.2 uint8_t dms_devSubsList::SubsListLen

8.36 dms_devSubsVoiceDataCaps Struct Reference

Data Fields

- uint8_t SubsVoiceDataCapLen
- dms_devSubsVoiceDataList SubsVoiceDataList [32]

8.36.1 Detailed Description

This structure contains the dms_devSubsVoiceDataCaps response parameters.

Parameters

TlvPresent	• Tlv Present or not
SubsVoiceData-CapLen	• Length of subscription list
SubsVoiceData-List	• See dms_devSubsVoiceDataList for more information

8.36.2 Field Documentation

8.36.2.1 uint8_t dms_devSubsVoiceDataCaps::SubsVoiceDataCapLen

8.36.2.2 dms_devSubsVoiceDataList dms_devSubsVoiceDataCaps::SubsVoiceDataList[32]

8.37 dms_devSubsVoiceDataList Struct Reference

Data Fields

- uint32_t SubsVoiceDataCap
- uint8_t SimVoiceDataCap

8.37.1 Detailed Description

This structure contains the dms_devSubsVoiceDataList parameters.

Parameters

SubsVoiceData-Cap	• Voice data capabilities of a subscription – DMS_SUBS_VOICE_DATA_CAPABILITY_NORMAL -0x01 – DMS_SUBS_VOICE_DATA_CAPABILITY_SGLTE - 0x02 – DMS_SUBS_VOICE_DATA_CAPABILITY_CSFB -0x03 – DMS_SUBS_VOICE_DATA_CAPABILITY_SVLTE -0x04 – DMS_SUBS_VOICE_DATA_CAPABILITY_ SRLTE -0x05
SimVoiceData-Cap	• Simultaneous Voice data capabilities of subscription

8.37.2 Field Documentation

8.37.2.1 uint8_t dms_devSubsVoiceDataList::SimVoiceDataCap

8.37.2.2 uint32_t dms_devSubsVoiceDataList::SubsVoiceDataCap

8.38 dms_LteBandsSupport Struct Reference

Data Fields

- uint16_t supportedLteBandLen
- uint16_t lteBands [255]
- uint8_t TLVPresent

8.38.1 Detailed Description

This structure contains the TLV to get Info of Supported LTE Bands.

Parameters

OUT]	supportedLteBandLen[OUT] • Supported LTE Bands length • Number of set of following elements – lteBands
OUT]	lteBands[OUT] • Array of supported LTE bands where each entry is decimal representative of the LTE band supported.
OUT]	TLVPresent[OUT] • TLV Present

8.38.2 Field Documentation

8.38.2.1 uint16_t dms_LteBandsSupport::lteBands[255]

8.38.2.2 uint16_t dms_LteBandsSupport::supportedLteBandLen

8.38.2.3 uint8_t dms_LteBandsSupport::TLVPresent

8.39 dms_OperatingModeTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t operatingMode

8.39.1 Detailed Description

Operating Mode Tlv

Parameters

TlvPresent	• Tlv Present
operatingMode	• 0 - Online • 1 - Low power • 2 - Factory test mode • 3 - Offline • 4 - Resetting • 5 - Shutting down • 6 - Persistent low power • 7 - Mode-only low power

8.39.2 Field Documentation

8.39.2.1 uint32_t dms_OperatingModeTlv::operatingMode

8.39.2.2 uint16_t dms_OperatingModeTlv::TlvPresent

8.40 dms_PSMActiveTimerIndTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ActiveTimerInd

8.40.1 Detailed Description

This Structure used to store DMS PSM Active Timer TLV Value.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
ActiveTimerInd	• Active Timer • PSM active timer value (in seconds).

8.40.2 Field Documentation

8.40.2.1 uint32_t dms_PSMActiveTimerIndTlv::ActiveTimerInd

8.40.2.2 uint8_t dms_PSMActiveTimerIndTlv::TlvPresent

8.41 dms_PSMActiveTimerTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t activeTimer

8.41.1 Detailed Description

This structure contains the parameters for Power Save Mode active timer.

Parameters

TlvPresent	• Tlv Present.
activeTimer	Active Timer • Power Save mode active timer value (in seconds).

8.41.2 Field Documentation

8.41.2.1 uint32_t dms_PSMActiveTimerTlv::activeTimer

8.41.2.2 uint8_t dms_PSMActiveTimerTlv::TlvPresent

8.42 dms_PSMDurationDueToOOSTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t durationDueToOOS

8.42.1 Detailed Description

This structure contains the parameters for Power Save Mode Duration due to an outage.

Parameters

TlvPresent	• Tlv Present.
durationDueTo-OOS	Duration Due to OOS • Power Save mode duration due to an outage (in seconds).

8.42.2 Field Documentation

8.42.2.1 uint32_t dms_PSMDurationDueToOOSTlv::durationDueToOOS

8.42.2.2 uint8_t dms_PSMDurationDueToOOSTlv::TlvPresent

8.43 dms_PSMDurationThresholdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t durationThreshold

8.43.1 Detailed Description

This structure contains the parameters for Power Save Mode Duration Threshold.

Parameters

TlvPresent	• Tlv Present.
duration-Threshold	Duration Threshold • Minimum duration for the device to benefit by entering PSM (in seconds).

8.43.2 Field Documentation

8.43.2.1 uint32_t dms_PSMDurationThresholdTlv::durationThreshold

8.43.2.2 uint8_t dms_PSMDurationThresholdTlv::TlvPresent

8.44 dms_PSMEarlyWakeupTimeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t earlyWakeupTime

8.44.1 Detailed Description

This structure contains the parameters for Power Save Mode early wakeup time.

Parameters

TlvPresent	• Tlv Present.
earlyWakeup-Time	• Early Wakeup Time • Power Save mode early wakeup time (in seconds) indicating how early the device should exit PSM to offset for bootup and acquisition delay.

8.44.2 Field Documentation

8.44.2.1 uint32_t dms_PSMEarlyWakeupTimeTlv::earlyWakeupTime

8.44.2.2 uint8_t dms_PSMEarlyWakeupTimeTlv::TlvPresent

8.45 dms_PSMEnableStateIndTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EnableStateInd

8.45.1 Detailed Description

This Structure used to store DMS PSM Enable state TLV Value.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
EnableStateInd	• PSM Enable State • Values – 0 - PSM is not enabled – 1 - PSM is enabled

8.45.2 Field Documentation

8.45.2.1 uint8_t dms_PSMEnableStateIndTlv::EnableStateInd

8.45.2.2 uint8_t dms_PSMEnableStateIndTlv::TlvPresent

8.46 dms_PSMEnableStateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t enableState

8.46.1 Detailed Description

This structure contains the parameters for Power Save Mode Enable State.

Parameters

TlvPresent	• Tlv Present.
enableState	Power Save Mode Enable State Values • 0 - PSM is not enabled • 1 - PSM is enabled

8.46.2 Field Documentation

8.46.2.1 uint8_t dms_PSMEnableStateTlv::enableState

8.46.2.2 uint8_t dms_PSMEnableStateTlv::TlvPresent

8.47 dms_PSMPeriodicUpdateTimerIndTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t PeriodicUpdateTimerInd

8.47.1 Detailed Description

This Structure used to store DMS PSM Periodic Update Timer TLV Value.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
PeriodicUpdate-TimerInd	• Periodic Update Timer • PSM periodic update timer value (in seconds).

8.47.2 Field Documentation

8.47.2.1 uint32_t dms_PSMPeriodicUpdateTimerIndTlv::PeriodicUpdateTimerInd

8.47.2.2 uint8_t dms_PSMPeriodicUpdateTimerIndTlv::TlvPresent

8.48 dms_PSMPeriodicUpdateTimerTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t periodicUpdateTimer

8.48.1 Detailed Description

This structure contains the parameters for Power Save Mode periodic update timer.

Parameters

TlvPresent	• Tlv Present.
periodicUpdate-Timer	Periodic Update Timer • Power Save mode periodic update timer value (in seconds).

8.48.2 Field Documentation

8.48.2.1 uint32_t dms_PSMPeriodicUpdateTimerTlv::periodicUpdateTimer

8.48.2.2 uint8_t dms_PSMPeriodicUpdateTimerTlv::TlvPresent

8.49 dms_PSMRandomizationWindowTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t randomizationWindow

8.49.1 Detailed Description

This structure contains the parameters for Power Save Mode wakeup randomization window.

Parameters

TlvPresent	• Tlv Present.
randomization-Window	Randomization Window • Power Save mode wakeup randomization window (in seconds)

8.49.2 Field Documentation

8.49.2.1 uint32_t dms_PSMRandomizationWindowTlv::randomizationWindow

8.49.2.2 uint8_t dms_PSMRandomizationWindowTlv::TlvPresent

8.50 dms_TemperatureTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t TempStat
- uint16_t Temperature

8.50.1 Detailed Description

This structure stores information about temperature TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
TempStat	• Temperature state – 0 - unknown – 1 - normal – 2 - high (warning) – 3 - high (critical) – 4 - low (critical)
Temperature	• temperature in degree celsius

8.50.2 Field Documentation

8.50.2.1 uint16_t dms_TemperatureTlv::Temperature

8.50.2.2 uint8_t dms_TemperatureTlv::TempStat

8.50.2.3 uint8_t dms_TemperatureTlv::TlvPresent

8.51 dms_UimAutoSwitchActSlotTlv Struct Reference

Data Fields

- uint8_t uimAutoSwitchActSlot
- uint8_t TlvPresent

8.51.1 Detailed Description

This structure contains the TLV required to Get UIM auto active slot Tlv.

Parameters

uimAutoSwitch-ActSlot[Optional]	• When UIMAUTOSWITCH feature enabled and SIM auto-switch is activated, indicates currently active UIM slot. • Values – 0 - slot 1 (e.g. external SIM) – 1 - slot 2 (e.g. embedded SIM)
TlvPresent	• Tlv Present.

8.51.2 Field Documentation

8.51.2.1 uint8_t dms_UimAutoSwitchActSlotTlv::TlvPresent

8.51.2.2 uint8_t dms_UimAutoSwitchActSlotTlv::uimAutoSwitchActSlot

8.52 dms_UimStatusTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t intf
- uint8_t event

8.52.1 Detailed Description

This structure stores information about UIM status TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
intf	• interface type – 0 - External UIM. – 1 - Embedded UIM. – 2 - Remote UIM.
event	• event type – 0 - UIM card removed. – 1 - UIM card inserted.

8.52.2 Field Documentation

8.52.2.1 uint8_t dms_UimStatusTlv::event

8.52.2.2 uint8_t dms_UimStatusTlv::intf

8.52.2.3 uint8_t dms_UimStatusTlv::TlvPresent

8.53 dms_VoltageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t VoltStat
- uint16_t Voltage

8.53.1 Detailed Description

This structure stores information about voltage TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
VoltStat	• Voltage state – 0 - unknown – 1 - normal – 2 - low (warning) – 3 - low (critical) – 4 - high (critical)
Voltage	• Voltage in mV

8.53.2 Field Documentation

8.53.2.1 uint8_t dms_VoltageTlv::TlvPresent

8.53.2.2 uint16_t dms_VoltageTlv::Voltage

8.53.2.3 uint8_t dms_VoltageTlv::VoltStat

8.54 DMScustSettingInfo Struct Reference

Data Fields

- uint16_t id_length
- uint8_t cust_id [64+1]
- uint16_t value_length
- uint8_t cust_value [8+1]
- uint16_t cust_attr

8.54.1 Detailed Description

This structure contains information about Customization Setting. This TLV is only applicable for 9x30 modules so far

Parameters

id_length	• length of cust_id field
cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.
value_length	• length of cust_value field
cust_value	• Customization Setting Value (Maximum 8 bytes)

cust_attr	• Customization Setting attribute through QMI – bit 0: Values: * 0 - read only * 1 - read/write

8.54.2 Field Documentation

8.54.2.1 uint16_t DMScustSettingInfo::cust_attr

8.54.2.2 uint8_t DMScustSettingInfo::cust_id[64+1]

8.54.2.3 uint8_t DMScustSettingInfo::cust_value[8+1]

8.54.2.4 uint16_t DMScustSettingInfo::id_length

8.54.2.5 uint16_t DMScustSettingInfo::value_length

8.55 DMScustSettingList Struct Reference

Data Fields

- uint8_t list_type
- uint16_t num_instances
- DMScustSettingInfo custSetting [255+1]

8.55.1 Detailed Description

This structure contains the fields of TLV Customization Setting List. This TLV is only applicable for 9x30 modules so far

Parameters

list_type	• list type requested
num_instances	• number of instances of customization setting
custSetting	• See DMScustSettingInfo for more information

8.55.2 Field Documentation

8.55.2.1 DMScustSettingInfo DMScustSettingList::custSetting[255+1]

8.55.2.2 uint8_t DMScustSettingList::list_type

8.55.2.3 uint16_t DMScustSettingList::num_instances

8.56 DMSgetCustomFeatureV2 Struct Reference

Data Fields

- DMSgetCustomInput * pGetCustomInput
- DMScustSettingInfo * pCustSettingInfo
- DMScustSettingList * pCustSettingList

8.56.1 Detailed Description

This struture contains the TLV required to get the Customization Info and customization list.

Parameters

pGetCustom-Input[IN]	• See DMSgetCustomInput for more information
pCustSetting-Info[OUT]	• See DMScustSettingInfo for more information
pCustSetting-List[OUT]	• See DMScustSettingList for more information

8.56.2 Field Documentation

8.56.2.1 DMScustSettingInfo* DMSgetCustomFeatureV2::pCustSettingInfo

8.56.2.2 DMScustSettingList* DMSgetCustomFeatureV2::pCustSettingList

8.56.2.3 DMSgetCustomInput* DMSgetCustomFeatureV2::pGetCustomInput

8.57 DMSgetCustomInput Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint8_t list_type

8.57.1 Detailed Description

This structure contains which customization id or the list type want to retrieve from modem. This TLV is only applicable for 9x30 modules so far

Parameters

cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.
list_type	• list type requested

8.57.2 Field Documentation

8.57.2.1 uint8_t DMSgetCustomInput::cust_id[64+1]

8.57.2.2 uint8_t DMSgetCustomInput::list_type

8.58 DsdApnAvailableSystemInfo_t Struct Reference

Data Fields

- uint8_t apn_name_len
- char apn_name [100]
- uint8_t apn_avail_sys_len
- DsdSystemInfo_t dsdGlobalAvailableSystem [255]

8.58.1 Detailed Description

This structure contains the APN Current and Recommended Preferred System Info

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

apn_name_len	• apn name length
apn_name	• apn name
apn_avail_sys_-len	• length APN available system info
dsdGlobal-AvailableSystem	• Array of DSD Global available system info value

Note

For_5G

8.58.2 Field Documentation

8.58.2.1 uint8_t DsdApnAvailableSystemInfo_t::apn_avail_sys_len

8.58.2.2 char DsdApnAvailableSystemInfo_t::apn_name[100]

8.58.2.3 uint8_t DsdApnAvailableSystemInfo_t::apn_name_len

8.58.2.4 DsdSystemInfo_t DsdApnAvailableSystemInfo_t::dsdGlobalAvailableSystem[255]

8.59 DsdApnCurRecAvailableSystemInfo_t Struct Reference

Data Fields

- DsdSystemInfo_t dsdCurrentPreferdSystem
- DsdSystemInfo_t dsdRecommendedPreferdSystem

8.59.1 Detailed Description

This structure contains the APN Current and Recommended Preferred System Info

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

dsdCurrentPreferdSystem	• Current DSD current prefered system info
dsdRecommendedPreferdSystem	• Recommended DSD current prefered system info

Note

For_5G

8.59.2 Field Documentation

8.59.2.1 DsdSystemInfo_t DsdApnCurRecAvailableSystemInfo_t::dsdCurrentPreferdSystem

8.59.2.2 DsdSystemInfo_t DsdApnCurRecAvailableSystemInfo_t::dsdRecommendedPreferdSystem

8.60 DsdApnCurRecPreferredSystemInfo_t Struct Reference

Data Fields

- uint8_t apn_name_len
- char apn_name [100]
- DsdApnCurRecAvailableSystemInfo_t dsdApnCurRecPreferredSystemInfo

8.60.1 Detailed Description

This structure contains the APN Current and Recommended Preferred System Info

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

apn_name_len	• apn name length
apn_name	• apn name
dsdApnCurRecPreferredSystemInfo	• Array of DSD APN Current and Recommended Preferred System Info

Note

For_5G

8.60.2 Field Documentation

8.60.2.1 char DsdApnCurRecPreferredSystemInfo_t::apn_name[100]

8.60.2.2 uint8_t DsdApnCurRecPreferredSystemInfo_t::apn_name_len

8.60.2.3 DsdApnCurRecAvailableSystemInfo_t DsdApnCurRecPreferredSystemInfo_t::dsdApnCurRecPreferredSystemInfo

8.61 DsdGlobalUIInformation_t Struct Reference

Data Fields

- DsdSystemInfo_t dsdAvailableSystem
- uint64_t ui_mask

8.61.1 Detailed Description

This structure contains the GetApnOpReservedPcoList response parameters.

See Also

qaGobiApiWds.h for parameter descriptions

Parameters

dsdAvailable-System	• ref DsdSystemInfo_t.
ui_mask	• UI mask to indicate the UI icon Values: – QMI_DSD_UI_MASK_UNSPECIFIED (0x0000000000000000) - Unspecified – QMI_DSD_UI_MASK_3GPP_5G_BASIC (0x0000000000000001) - 5G basic – QMI_DSD_UI_MASK_3GPP_5G_UWB (0x0000000000000002) - 5G ultrawide band – QMI_DSD_UI_MASK_3GPP_SIB2_PLMN_R15 (0x0000000000000004) - SIB2 PLMN R15 – QMI_DSD_UI_MASK_3GPP_SIB2_ULI (0x0000000000000008) - SIB2 upper layer indication – QMI_DSD_UI_MASK_3GPP_RESTRICT_DCNR (0x0000000000000010) - Restrict decoded channel number register

Note

FOR_5G

8.61.2 Field Documentation

8.61.2.1 DsdSystemInfo_t DsdGlobalUIInformation_t::dsdAvailableSystem

8.61.2.2 uint64_t DsdGlobalUIInformation_t::ui_mask

8.62 DsdNullBearerReason_t Struct Reference

Data Fields

- uint64_t null_bearer_reason

8.62.1 Detailed Description

This structure contains the Null Bearer reason

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

null_bearer_reason	• Null Bearer reason for global current preferred system. Values : – DSD_NULL_BEARER_REASON_UNSPECIFIED (0x0000000000000000) – DSD_NULL_BEARER_REASON_CSFB (0x0000000000000001) – DSD_NULL_BEARER_REASON_OOS (0x0000000000000002) – DSD_NULL_BEARER_REASON_LIMITED_SRVC (0x0000000000000004) – DSD_NULL_BEARER_REASON_VOICE_ON_SAME_SUB (0x0000000000000008) – DSD_NULL_BEARER_REASON_VOICE_ON_OTHER_SUB (0x0000000000000010) – DSD_NULL_BEARER_REASON_SRVCC (0x0000000000000020) – DSD_NULL_BEARER_REASON_CS_ONLY (0x0000000000000040) – DSD_NULL_BEARER_REASON_ATTACH_PENDING (0x0000000000000080)

Note

For_5G

8.62.2 Field Documentation

8.62.2.1 uint64_t DsdNullBearerReason_t::null_bearer_reason

8.63 DsdSystemInfo_t Struct Reference

Data Fields

- uint32_t technology
- uint32_t rat_value
- uint64_t so_mask

8.63.1 Detailed Description

This structure contains the Dsd System Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

technology	• Technology type. Values: – DSD_SYS_NETWORK_3GPP (0) - 3GPP – DSD_SYS_NETWORK_3GPP2 (1) - 3GPP2 – DSD_SYS_NETWORK_WLAN (2) - WLAN
rat_value	• RAT value. Values: – DSD_SYS_RAT_EX_NULL_BEARER(0x0) - NULL bearer – DSD_SYS_RAT_EX_3GPP_WCDMA(0x1) - 3GPP WCDMA – DSD_SYS_RAT_EX_3GPP_GERAN(0x2) - 3GPP GERAN – DSD_SYS_RAT_EX_3GPP_LTE(0x3) - 3GPP LTE – DSD_SYS_RAT_EX_3GPP_TDSCDMA(0x4) - 3GPP TDSCDMA – DSD_SYS_RAT_EX_3GPP_WLAN(0x5) - 3GPP WLAN – DSD_SYS_RAT_EX_3GPP_5G (0x6)- 3GPP 5G – DSD_SYS_RAT_EX_3GPP_MAX(0x64) - 3GPP maximum – DSD_SYS_RAT_EX_3GPP2_1X(0x65) - 3GPP2 1X – DSD_SYS_RAT_EX_3GPP2_HRPD(0x66) - 3GPP2 HRPD – DSD_SYS_RAT_EX_3GPP2_EHRPD(0x67) - 3GPP2 EHRPD – DSD_SYS_RAT_EX_3GPP2_WLAN(0x68) - 3GPP2 WLAN – DSD_SYS_RAT_EX_3GPP2_MAX(0xC8) - 3GPP2 maximum – DSD_SYS_RAT_EX_WLAN (0xC9) - WLAN – DSD_SYS_RAT_EX_WLAN_MAX(0x12C) - WLAN maximum

so_mask

- Service Option (SO) mask to indicate the service option or type of application.
- An SO mask value of zero indicates that this field is ignored.
  - Values:
    - 0x00 - SO Mask Unspecified
- 3GPP SO Mask:
  - 0x01 - WCDMA
  - 0x02 - HSDPA
  - 0x04 - HSUPA
  - 0x08 - HSDPAPLUS
  - 0x10 - DC HSDPAPLUS
  - 0x20 - 64 QAM
  - 0x40 - HSPA
  - 0x80 - GPRS
  - 0x100 - EDGE
  - 0x200 - GSM
  - 0x400 - S2B
  - 0x800 - LTE Limited Service, , 3GPP Limited Service
  - 0x1000 - LTE FDD, 3GPP FDD
  - 0x2000 - LTE TDD
  - 0x4000 - TDSCDMA
  - 0x8000 - DC HSUPA
  - 0x10000 - LTE CA DL
  - 0x20000 - LTE CA UL
  - 0x40000 - S2B Limited Service
  - 0x80000 - 4.5G
  - 0x100000 - 4.5G+
- 3GPP2 SO Mask:
  - 0x0001000000 - 1X IS95
  - 0x0002000000 - 1X IS2000
  - 0x0004000000 - 1X IS2000 REL A
  - 0x0008000000 - HDR REV0 DPA
  - 0x0010000000 - HDR REVA DPA
  - 0x0020000000 - HDR REVB DPA
  - 0x0040000000 - HDR REVA MPA
  - 0x0080000000 - HDR REVB MPA
  - 0x0100000000 - HDR REVA EMPA
  - 0x0200000000 - HDR REVB EMPA
  - 0x0400000000 - HDR REVB MMPA
  - 0x0800000000 - HDR EVDO FMC
  - 0x1000000000 - 1X Circuit Switched
- 5G SO Mask:
  - 0x10000000000 - 5G TDD
  - 0x20000000000 - 5G SUB6
  - 0x40000000000 - 5G MMWAVE
  - 0x80000000000 - 5G NSA
  - 0x100000000000 - 5G SA

Note

For_5G

8.63.2 Field Documentation

8.63.2.1 uint32_t DsdSystemInfo_t::rat_value

8.63.2.2 uint64_t DsdSystemInfo_t::so_mask

8.63.2.3 uint32_t DsdSystemInfo_t::technology

8.64 DsdUIApnAvailableSystemInfo_t Struct Reference

Data Fields

- uint8_t apn_name_len
- char apn_name [100]
- uint64_t apn_type_mask
- uint32_t technology
- uint32_t rat_value
- uint64_t so_mask
- uint64_t ui_mask
- uint8_t is_internet

8.64.1 Detailed Description

This structure contains the APN Current and Recommended Preferred System Info

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

apn_name_len	• apn name length
apn_name	• apn name
apn_type_mask	• APN Type Mask Values – QMI_DSD_APN_TYPE_MASK_UNSPECIFIED (0x0000000000000000) – QMI_DSD_APN_TYPE_MASK_DEFAULT (0x0000000000000001) -APN type for Default/-Internet traffic – QMI_DSD_APN_TYPE_MASK_ IMS (0x0000000000000002) - APN type for IP Multimedia Subsystem – QMI_DSD_APN_TYPE_MASK_MMS (0x0000000000000004) - APN type for Multimedia Messaging Service – QMI_DSD_APN_TYPE_MASK_DUN (0x0000000000000008) - APN type for Dial Up Network – QMI_DSD_APN_TYPE_MASK_SUPL (0x0000000000000010) - APN type for Secure User Plane Location – QMI_DSD_APN_TYPE_MASK_HIPRI (0x0000000000000020) - APN type for High Priority Mobile Data – QMI_DSD_APN_TYPE_MASK_FOTA (0x0000000000000040) - APN type for over the air administration – QMI_DSD_APN_TYPE_MASK_CBS (0x0000000000000080) - APN type for Carrier Branded Services – QMI_DSD_APN_TYPE_MASK_ IA (0x0000000000000100) - APN type for Initial Attach – QMI_DSD_APN_TYPE_MASK_EMERGENCY (0x0000000000000200) - APN type for emergency
technology	• Technology type
rat_value	• RAT value
so_mask	• Service Option (SO) mask to indicate the service option or type of application.
ui_mask	
is_internet	Values: FALSE - APN is not for internet service. TRUE - APN is for internet service.

Note

For_5G

8.64.2 Field Documentation

8.64.2.1 char DsdUIApnAvailableSystemInfo_t::apn_name[100]

8.64.2.2 uint8_t DsdUIApnAvailableSystemInfo_t::apn_name_len

8.64.2.3 uint64_t DsdUIApnAvailableSystemInfo_t::apn_type_mask

8.64.2.4 uint8_t DsdUIApnAvailableSystemInfo_t::is_internet

8.64.2.5 uint32_t DsdUIApnAvailableSystemInfo_t::rat_value

8.64.2.6 uint64_t DsdUIApnAvailableSystemInfo_t::so_mask

8.64.2.7 uint32_t DsdUIApnAvailableSystemInfo_t::technology

8.64.2.8 uint64_t DsdUIApnAvailableSystemInfo_t::ui_mask

8.65 dunchannelRate Struct Reference

Data Fields

- uint32_t CurrChanTxRate
- uint32_t CurrChanRxRate
- uint32_t MaxChanTxRate
- uint32_t MaxChanRxRate

8.65.1 Detailed Description

This structure contains Channel Rate

Parameters

CurrChanTxRate	• Instantaneous channel Tx rate in bits per second • 0xffffffff - Invalid data.
CurrChanRx-Rate	• Instantaneous channel Rx rate in bits per second • 0xffffffff - Invalid data.
MaxChanTxRate	• maximum Tx rate that can be assigned to the device by the serving system in bits per second • 0xffffffff - Invalid data.
MaxChanRx-Rate	• maximum Rx rate that can be assigned to the device by the serving system in bits per second • 0xffffffff - Invalid data.

8.65.2 Field Documentation

8.65.2.1 uint32_t dunchannelRate::CurrChanRxRate

8.65.2.2 uint32_t dunchannelRate::CurrChanTxRate

8.65.2.3 uint32_t dunchannelRate::MaxChanRxRate

8.65.2.4 uint32_t dunchannelRate::MaxChanTxRate

8.66 eriDataparams Struct Reference

Data Fields

- uint16_t eriDataLen
- uint8_t eriData [1024]

8.66.1 Detailed Description

This structure contains Extended Roaming Indicator(ERI) file parameters

Parameters

eriDataLen	• Upon input, the maximum number of bytes that file contents array can contain. • Upon successful output, actual number of bytes written to file contents array
eriData	• ERI data read from persistent storage(Max size is 1024)

8.66.2 Field Documentation

8.66.2.1 uint8_t eriDataparams::eriData[1024]

8.66.2.2 uint16_t eriDataparams::eriDataLen

8.67 eTWSPLMNInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsPlmnInfo ETWSPLMNInfo

8.67.1 Detailed Description

This structure contains ETWS PLMN TLV information

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
ETWSPLMNInfo	• ETWS PLMN Information • See sMSEtwsPlmnInfo for more information

8.67.2 Field Documentation

8.67.2.1 sMSEtwsPlmnInfo eTWSPLMNInfoTlv::ETWSPLMNInfo

8.67.2.2 uint8_t eTWSPLMNInfoTlv::TlvPresent

8.68 FMSImageElement Struct Reference

Data Fields

- uint8_t imageType
- uint8_t imageId [100]
- uint8_t buildIdLength
- uint8_t buildId [100]

8.68.1 Detailed Description

This structure contains the Image Element information

Parameters

imageType	• Type of image 0 - Modem 1 - PRI
imageId	• Unique image identifier
buildIdLength	• Length of the build ID string (may be zero)
pBuildId	• Build ID ANSI string with length provided by the previous field

8.68.2 Field Documentation

8.68.2.1 uint8_t FMSImageElement::buildId[100]

8.68.2.2 uint8_t FMSImageElement::buildIdLength

8.68.2.3 uint8_t FMSImageElement::imageId[100]

8.68.2.4 uint8_t FMSImageElement::imageType

8.69 FMSImageIdElement Struct Reference

Data Fields

- uint8_t storageIndex
- uint8_t failureCount
- uint8_t imageID [100]
- uint8_t buildIDLength
- uint8_t buildID [100]

8.69.1 Detailed Description

This structure contains the Image ID list element Information

Parameters

storageIndex	• Index in storage where the image is located(a value of 0xFF indicates that the storage for this type of image is not relevant)
failureCount	• Number of consecutive write attempts to this storage index that have failed(a value of 0xFF indicates unspecified)
imageID	• Image unique identifier(max 16 chars.)
buildIDLength	• Length of the build ID string. If there is no build ID, this field will be 0 and no data will follow.
buildID	• String containing image build information(Max 100 characters)

8.69.2 Field Documentation

8.69.2.1 uint8_t FMSImageIdElement::buildID[100]

8.69.2.2 uint8_t FMSImageIdElement::buildIDLength

8.69.2.3 uint8_t FMSImageIdElement::failureCount

8.69.2.4 uint8_t FMSImageIdElement::imageID[100]

8.69.2.5 uint8_t FMSImageIdElement::storageIndex

8.70 FMSImageIDEntries Struct Reference

Data Fields

- uint8_t imageType
- uint8_t maxImages
- uint8_t executingImage
- uint8_t imageIDSize
- FMSImageIdElement imageIDElement [50]

8.70.1 Detailed Description

This structure contains the list entry Information

Parameters

imageType	• Type of image – 0 - Modem – 1 - PRI

maxImages	• Maximum number of images of this type that may be stored concurrently on the device
executingImage	• Index (into the next array) of image that is currently executing
imageIDSize	• The number of elements in the image ID list
imageIDElement	• Array of ImageIDElement Structure (Max 50 elements) • See FMSImageIdElement

8.70.2 Field Documentation

8.70.2.1 uint8_t FMSImageIDEntries::executingImage

8.70.2.2 FMSImageIdElement FMSImageIDEntries::imageIDElement[50]

8.70.2.3 uint8_t FMSImageIDEntries::imageIDSize

8.70.2.4 uint8_t FMSImageIDEntries::imageType

8.70.2.5 uint8_t FMSImageIDEntries::maxImages

8.71 FMSImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageIDEntries imageIDEntries [2]

8.71.1 Detailed Description

This structure contains the Get Stored Images List

Parameters

listSize	• The number of elements in the image list
imageIDEntries	• Array of ImageIDEntries Structure (Max 2 entries)

8.71.2 Field Documentation

8.71.2.1 FMSImageIDEntries FMSImageList::imageIDEntries[2]

8.71.2.2 uint8_t FMSImageList::listSize

8.72 FMSPrefImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageElement listEntries [2]

8.72.1 Detailed Description

This structure contains the Preference Image List information

Parameters

listSize	• The number of elements in the image list
pListEntries	• Array of Image entries with size provided by previous field • See FMSImageElement

8.72.2 Field Documentation

8.72.2.1 FMSImageElement FMSPrefImageList::listEntries[2]

8.72.2.2 uint8_t FMSPrefImageList::listSize

8.73 GetClientIdReqMsg Struct Reference

Data Fields

- uint16_t messageId
- uint16_t length
- uint8_t tlvType
- uint16_t tlvLength
- uint8_t serviceType

8.73.1 Field Documentation

8.73.1.1 uint16_t GetClientIdReqMsg::length

8.73.1.2 uint16_t GetClientIdReqMsg::messageId

8.73.1.3 uint8_t GetClientIdReqMsg::serviceType

8.73.1.4 uint16_t GetClientIdReqMsg::tlvLength

8.73.1.5 uint8_t GetClientIdReqMsg::tlvType

8.74 GetClientIdRspMsg Struct Reference

Data Fields

- uint16_t messageId
- uint16_t length
- uint8_t tlvType
- uint16_t tlvLength
- uint16_t result
- uint16_t error
- uint8_t tlv2Type
- uint16_t tlv2Length
- uint8_t serviceType
- uint8_t clientId

8.74.1 Field Documentation

8.74.1.1 uint8_t GetClientIdRspMsg::clientId

8.74.1.2 uint16_t GetClientIdRspMsg::error

8.74.1.3 uint16_t GetClientIdRspMsg::length

8.74.1.4 uint16_t GetClientIdRspMsg::messageId

8.74.1.5 uint16_t GetClientIdRspMsg::result

8.74.1.6 uint8_t GetClientIdRspMsg::serviceType

8.74.1.7 uint16_t GetClientIdRspMsg::tlv2Length

8.74.1.8 uint8_t GetClientIdRspMsg::tlv2Type

8.74.1.9 uint16_t GetClientIdRspMsg::tlvLength

8.74.1.10 uint8_t GetClientIdRspMsg::tlvType

8.75 hdrSSInfo Struct Reference

Data Fields

- int8_t rssi
- int16_t ecio
- uint8_t sinr
- int32_t io

8.75.1 Detailed Description

This structure contains the parameters for HDR Signal Strength Information

Parameters

rssi	• RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal.

ecio	• ECIO value representing negative 0.5 dBm increments, i.e., 2 means -1 dBm (14 means -7 dBm, 63 means -31.5 dBm).
sinr	• SINR level. • SINR is only applicable for 1xEV-DO. • Valid levels are 0 to 8, where the maximum value for: – 0 - SINR_LEVEL_0 is -9 dB – 1 - SINR_LEVEL_1 is -6 dB – 2 - SINR_LEVEL_2 is -4.5 dB – 3 - SINR_LEVEL_3 is -3 dB – 4 - SINR_LEVEL_4 is -2 dB – 5 - SINR_LEVEL_5 is +1 dB – 6 - SINR_LEVEL_6 is +3 dB – 7 - SINR_LEVEL_7 is +6 dB – 8 - SINR_LEVEL_8 is +9 dB – 0xFF - Not Available
io	• Received IO in dBm. • IO is only applicable for 1xEV-DO.

8.75.2 Field Documentation

8.75.2.1 int16_t hdrSSInfo::ecio

8.75.2.2 int32_t hdrSSInfo::io

8.75.2.3 int8_t hdrSSInfo::rssi

8.75.2.4 uint8_t hdrSSInfo::sinr

8.76 image_info_t Struct Reference

Data Fields

- uint8_t imageType
- uint8_t uniqueID [16]
- uint8_t buildIDLen
- uint8_t buildID [255]

8.76.1 Detailed Description

This structure is used to store image information

Parameters

imageType	• Image Type • Values: – 0 - FW – 1 - configuration
uniqueID	• Image Unique Identifier (ASCII characters)
buildIDLen	• Length of build ID string to follow • If set to zero, build ID string will be blank
buildID	• String containing image information(ASCII characters) • Maximum length of this string is 255 chars

8.76.2 Field Documentation

8.76.2.1 uint8_t image_info_t::buildID[255]

8.76.2.2 uint8_t image_info_t::buildIDLen

8.76.2.3 uint8_t image_info_t::imageType

8.76.2.4 uint8_t image_info_t::uniqueID[16]

8.77 ims_AMRModeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrMode

8.77.1 Detailed Description

This structure store information about AMR NB mode.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
amrMode	Bitmask indicating AMR modes. Values: • 0x1 - 4.75 kbps • 0x2 - 5.15 kbps • 0x4 - 5.9 kbps • 0x8 - 6.17 kbps • 0x10 - 7.4 kbps • 0x20 - 7.95 kbps • 0x40 - 10.2 kbps • 0x80 - 12.2 kbps

8.77.2 Field Documentation

8.77.2.1 uint8_t ims_AMRModeInfo::amrMode

8.77.2.2 uint8_t ims_AMRModeInfo::TlvPresent

8.78 ims_AMROctAlgnInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrOctAlgn

8.78.1 Detailed Description

This structure store information about AMR NB octet aligned.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
amrOctAlgn	-1- if octet aligned -0- if octed not aligned

8.78.2 Field Documentation

8.78.2.1 uint8_t ims_AMROctAlgnInfo::amrOctAlgn

8.78.2.2 uint8_t ims_AMROctAlgnInfo::TlvPresent

8.79 ims_AMRWBModeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t amrWBMode

8.79.1 Detailed Description

This structure store information about AMR WB mode.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
amrWBMode	Bitmask indicating AMR WB modes. Values: • 0x1 - 6.60 kbps • 0x2 - 8.85 kbps • 0x4 - 12.65 kbps • 0x8 - 14.25 kbps • 0x10 - 15.85 kbps • 0x20 - 18.25 kbps • 0x40 - 19.85 kbps • 0x80 - 23.05 kbps • 0x100 - 23.85 kbps

8.79.2 Field Documentation

8.79.2.1 uint16_t ims_AMRWBModeInfo::amrWBMode

8.79.2.2 uint8_t ims_AMRWBModeInfo::TlvPresent

8.80 ims_AMRWBOctAlgnInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrWBOctAlgn

8.80.1 Detailed Description

This structure store information about AMR WB octet aligned.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
amrWBOctAlgn	-1- if octet aligned -0- if octed not aligned

8.80.2 Field Documentation

8.80.2.1 uint8_t ims_AMRWBOctAlgnInfo::amrWBOctAlgn

8.80.2.2 uint8_t ims_AMRWBOctAlgnInfo::TlvPresent

8.81 ims_CSCFPortNameInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t cscfPortName [255]

8.81.1 Detailed Description

This structure hold parameters about CSCF port info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
cscfPortName	-CSCF port name string

8.81.2 Field Documentation

8.81.2.1 uint8_t ims_CSCFPortNameInfo::cscfPortName[255]

8.81.2.2 uint8_t ims_CSCFPortNameInfo::TlvPresent

8.82 ims_EnabAMRWBInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrWBEnable

8.82.1 Detailed Description

This structure store information about Enable AMR WB.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
amrWBEnable	-1- Enable -0- Disable

8.82.2 Field Documentation

8.82.2.1 uint8_t ims_EnabAMRWBInfo::amrWBEnable

8.82.2.2 uint8_t ims_EnabAMRWBInfo::TlvPresent

8.83 ims_EnabSCRAMRInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t scrAmrEnable

8.83.1 Detailed Description

This structure store information about Enable SCR AMR .

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
scrAmrEnable	-1- Enable -0- Disable

8.83.2 Field Documentation

8.83.2.1 uint8_t ims_EnabSCRAMRInfo::scrAmrEnable

8.83.2.2 uint8_t ims_EnabSCRAMRInfo::TlvPresent

8.84 ims_EnabSCRAMRWBInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t scrAmrWBEnable

8.84.1 Detailed Description

This structure store information about Enable SCR AMR WB.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
scrAmrWB-Enable	-1- Enable -0- Disable

8.84.2 Field Documentation

8.84.2.1 uint8_t ims_EnabSCRAMRWBInfo::scrAmrWBEnable

8.84.2.2 uint8_t ims_EnabSCRAMRWBInfo::TlvPresent

8.85 ims_IMSDomainInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t imsDomainName [255]

8.85.1 Detailed Description

This structure hold parameters about IMS domain info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
imsDomain-Name	-IMS domain name string

8.85.2 Field Documentation

8.85.2.1 uint8_t ims_IMSDomainInfo::imsDomainName[255]

8.85.2.2 uint8_t ims_IMSDomainInfo::TlvPresent

8.86 ims_IMSTestModeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t imsTestMode

8.86.1 Detailed Description

This structure hold parameters about ims test mode info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
imsTestMode	-1- if IMS test mode is enabled -0- if IMS test mode is disabled

8.86.2 Field Documentation

8.86.2.1 uint8_t ims_IMSTestModeInfo::imsTestMode

8.86.2.2 uint8_t ims_IMSTestModeInfo::TlvPresent

8.87 ims_MinSessExpInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t minSessExp

8.87.1 Detailed Description

This structure store information about minimum session expiry info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0

minSessExp	• minimum session expiry in seconds

8.87.2 Field Documentation

8.87.2.1 uint16_t ims_MinSessExpInfo::minSessExp

8.87.2.2 uint8_t ims_MinSessExpInfo::TlvPresent

8.88 ims_PCSCFPortInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t priCSCFPort

8.88.1 Detailed Description

This structure hold parameters about primary CSCF port info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
priCSCFPort	• primary CSCF port number

8.88.2 Field Documentation

8.88.2.1 uint16_t ims_PCSCFPortInfo::priCSCFPort

8.88.2.2 uint8_t ims_PCSCFPortInfo::TlvPresent

8.89 ims_PhCtxtURIInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t PhCtxtURI [255]

8.89.1 Detailed Description

This structure hold parameters about Phone context URI info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0

PhCtxtURI	-Phone context URI string

8.89.2 Field Documentation

8.89.2.1 uint8_t ims_PhCtxtURIInfo::PhCtxtURI[255]

8.89.2.2 uint8_t ims_PhCtxtURIInfo::TlvPresent

8.90 ims_RngBkTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RingBkTmr

8.90.1 Detailed Description

This structure hold parameters about Ring back timer.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
RingBkTmr	• Duration, in seconds, of the Ringback timer

8.90.2 Field Documentation

8.90.2.1 uint16_t ims_RngBkTmrInfo::RingBkTmr

8.90.2.2 uint8_t ims_RngBkTmrInfo::TlvPresent

8.91 ims_RngTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RingTmr

8.91.1 Detailed Description

This structure hold parameters about Ring timer.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0

RingTmr	• Duration, in seconds, of the Ring timer

8.91.2 Field Documentation

8.91.2.1 uint16_t ims_RngTmrInfo::RingTmr

8.91.2.2 uint8_t ims_RngTmrInfo::TlvPresent

8.92 ims_RTPRTCPInactTmrDurInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t InactTmr

8.92.1 Detailed Description

This structure hold parameters about RTP/RTCP timer.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
InactTmr	• Duration, in seconds, of the RTP/RTCP inactivity timer

8.92.2 Field Documentation

8.92.2.1 uint16_t ims_RTPRTCPInactTmrDurInfo::InactTmr

8.92.2.2 uint8_t ims_RTPRTCPInactTmrDurInfo::TlvPresent

8.93 ims_SessDurInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t sessExp

8.93.1 Detailed Description

This structure store information about session duration info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0

sessExp	• session duration in seconds

8.93.2 Field Documentation

8.93.2.1 uint16_t ims_SessDurInfo::sessExp

8.93.2.2 uint8_t ims_SessDurInfo::TlvPresent

8.94 ims_SigCompEnInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t SigCompEn

8.94.1 Detailed Description

This structure hold parameters about SigComp status.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
SigCompEn	-1- if SigComp is enabled -0- if IMS SigComp is disabled

8.94.2 Field Documentation

8.94.2.1 uint8_t ims_SigCompEnInfo::SigCompEn

8.94.2.2 uint8_t ims_SigCompEnInfo::TlvPresent

8.95 ims_SIPPortInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t SIPLocalPort

8.95.1 Detailed Description

This structure stores information about SIP port info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
SIPLocalPort	-SIP local port

8.95.2 Field Documentation

8.95.2.1 uint16_t ims_SIPPortInfo::SIPLocalPort

8.95.2.2 uint8_t ims_SIPPortInfo::TlvPresent

8.96 ims_SIPRegnTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrSIPRegn

8.96.1 Detailed Description

This structure stores information about SIP registration timer.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
tmrSIPRegn	-Initial SIP registration duration, in seconds

8.96.2 Field Documentation

8.96.2.1 uint8_t ims_SIPRegnTmrInfo::TlvPresent

8.96.2.2 uint32_t ims_SIPRegnTmrInfo::tmrSIPRegn

8.97 ims_SMSFmtInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t smsFormat

8.97.1 Detailed Description

This structure hold parameters about SMS format info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
smsFormat	• 0- IMS_SETTINGS_SMS_FORMAT_3GPP2 • 1- IMS_SETTINGS_SMS_FORMAT_3GPP

8.97.2 Field Documentation

8.97.2.1 uint8_t ims_SMSFmtInfo::smsFormat

8.97.2.2 uint8_t ims_SMSFmtInfo::TlvPresent

8.98 ims_SMSoIPNwInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t smsoIPNW

8.98.1 Detailed Description

This structure hold parameters about SMS over IP network.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
smsoIPNW	-1- if MO SMS turned on -0- if MO SMS turned off

8.98.2 Field Documentation

8.98.2.1 uint8_t ims_SMSoIPNwInfo::smsoIPNW

8.98.2.2 uint8_t ims_SMSoIPNwInfo::TlvPresent

8.99 ims_SubscrTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t subscrTmr

8.99.1 Detailed Description

This structure stores information about subscriber timer.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
subscrTmr	-Duration, in seconds, of the subscription by the UE for IMS registration notifications.

8.99.2 Field Documentation

8.99.2.1 uint32_t ims_SubscrTmrInfo::subscrTmr

8.99.2.2 uint8_t ims_SubscrTmrInfo::TlvPresent

8.100 ims_TmrT1Info Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrT1

8.100.1 Detailed Description

This structure stores information about timer T1.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
tmrT1	-RTT estimate, in milliseconds.

8.100.2 Field Documentation

8.100.2.1 uint8_t ims_TmrT1Info::TlvPresent

8.100.2.2 uint32_t ims_TmrT1Info::tmrT1

8.101 ims_TmrT2Info Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrT2

8.101.1 Detailed Description

This structure stores information about timer T2.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
tmrT2	-Maximum retransmit interval, in msec, for non-invite requests and invite responses.

8.101.2 Field Documentation

8.101.2.1 uint8_t ims_TmrT2Info::TlvPresent

8.101.2.2 uint32_t ims_TmrT2Info::tmrT2

8.102 ims_TmrTfInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrTf

8.102.1 Detailed Description

This structure store information about timer TF.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
tmrTf	-Non-invite transaction timeout timer, in msec

8.102.2 Field Documentation

8.102.2.1 uint8_t ims_TmrTfInfo::TlvPresent

8.102.2.2 uint32_t ims_TmrTfInfo::tmrTf

8.103 imsa_IMSFailErrCodeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ImsFailErrCode

8.103.1 Detailed Description

This structure hold parameters about IMS failure code for PDP connection.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
ImsFailErrCode	• 0 -IMSA_PDP_STATUS_OTHER_FAILURE - Generic failure reason • 1 -IMSA_PDP_STATUS_OPTION_UNSUBSCRIBED - Option is unsubscribed • 2 -IMSA_PDP_STATUS_UNKNOWN_PDP - PDP was unknown

8.103.2 Field Documentation

8.103.2.1 uint32_t imsa_IMSFailErrCodeTlv::ImsFailErrCode

8.103.2.2 uint8_t imsa_IMSFailErrCodeTlv::TlvPresent

8.104 imsa_IMSRegStatusErrorCodeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ErrorCode

8.104.1 Detailed Description

This structure hold parameters about IMS registration error info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
ErrorCode	Error codes when registration status is IMSA_STATUS_NOT_REGISTERED • Values: • 3xx - Redirection responses • 4xx - Client failure responses • 5xx - Server failure responses • 6xx - Global failure responses

8.104.2 Field Documentation

8.104.2.1 uint16_t imsa_IMSRegStatusErrorCodeInfo::ErrorCode

8.104.2.2 uint8_t imsa_IMSRegStatusErrorCodeInfo::TlvPresent

8.105 imsa_IMSRegStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t ImsRegistered

8.105.1 Detailed Description

This structure hold parameters about IMS registration info (decprecated tlv).

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
ImsRegistered	• Values: • 0 -Not registered • 1- Registered

8.105.2 Field Documentation

8.105.2.1 uint8_t imsa_IMSRegStatusInfo::ImsRegistered

8.105.2.2 uint8_t imsa_IMSRegStatusInfo::TlvPresent

8.106 imsa_NewIMSRegStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ImsRegStatus

8.106.1 Detailed Description

This structure hold parameters about IMS registration info (new tlv).

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
ImsRegStatus	• Values: • 0 -IMSA_STATUS_NOT_REGISTERED- Not registered for IMS • 1- IMSA_STATUS_REGISTERING - Registering for IMS • 2- IMSA_STATUS_REGISTERED - Registered for IMS

8.106.2 Field Documentation

8.106.2.1 uint32_t imsa_NewIMSRegStatusInfo::ImsRegStatus

8.106.2.2 uint8_t imsa_NewIMSRegStatusInfo::TlvPresent

8.107 imsa_RatHandoverStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t RatHandoverStatus
- uint32_t SourceRAT
- uint32_t TargetRAT
- uint8_t ErrorCodeLen
- uint8_t ErrorCodeData [256]

8.107.1 Detailed Description

This structure hold parameters about RAT handover status.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0

RatHandover-Status	• RAT handover status
SourceRAT	• source RAT info: IWLAN/WWAN
TargetRAT	• target TAT info: IWLAN/WWAN
ErrorCodeLen	• error code length
ErrorCodeData	• handover failure code string when status is IMSA_STATUS_RAT_HO_FAILURE

8.107.2 Field Documentation

8.107.2.1 uint8_t imsa_RatHandoverStatusInfo::ErrorCodeData[256]

8.107.2.2 uint8_t imsa_RatHandoverStatusInfo::ErrorCodeLen

8.107.2.3 uint32_t imsa_RatHandoverStatusInfo::RatHandoverStatus

8.107.2.4 uint32_t imsa_RatHandoverStatusInfo::SourceRAT

8.107.2.5 uint32_t imsa_RatHandoverStatusInfo::TargetRAT

8.107.2.6 uint8_t imsa_RatHandoverStatusInfo::TlvPresent

8.108 imsa_SmsRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SmsRatVal

8.108.1 Detailed Description

This structure hold parameters about SMS RAT info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
SmsRatVal	• Values: • 0-IMSA_WLAN - IMS service is registered on WLAN • 1-IMSA_WWAN - IMS service is registered on WWAN • 2-IMSA_IWLAN - IMS service is registered on interworking WLAN

8.108.2 Field Documentation

8.108.2.1 uint32_t imsa_SmsRatInfo::SmsRatVal

8.108.2.2 uint8_t imsa_SmsRatInfo::TlvPresent

8.109 imsa_SmsSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SmsSvcStatus

8.109.1 Detailed Description

This structure hold parameters about SMS service info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
SmsSvcStatus	• Values: -0 - IMS SMS service is not available -1 - IMS SMS is in limited service -2 - IMS SMS is in full service

8.109.2 Field Documentation

8.109.2.1 uint32_t imsa_SmsSvcStatusInfo::SmsSvcStatus

8.109.2.2 uint8_t imsa_SmsSvcStatusInfo::TlvPresent

8.110 imsa_UtRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t UtRatVal

8.110.1 Detailed Description

This structure hold parameters about UT RAT info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
UtRatVal	• Values: • 0-IMSA_WLAN - IMS service is registered on WLAN • 1-IMSA_WWAN - IMS service is registered on WWAN • 2-IMSA_IWLAN - IMS service is registered on interworking WLAN

8.110.2 Field Documentation

8.110.2.1 uint8_t imsa_UtRatInfo::TlvPresent

8.110.2.2 uint32_t imsa_UtRatInfo::UtRatVal

8.111 imsa_UtSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t UtSvcStatus

8.111.1 Detailed Description

This structure hold parameters about UT service info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
UtSvcStatus	• Values: -0 - IMS UT service is not available -2 - IMS UT is in full service

8.111.2 Field Documentation

8.111.2.1 uint8_t imsa_UtSvcStatusInfo::TlvPresent

8.111.2.2 uint32_t imsa_UtSvcStatusInfo::UtSvcStatus

8.112 imsa_VoipRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t VoipRatVal

8.112.1 Detailed Description

This structure hold parameters about VOIP RAT info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
VoipRatVal	• Values: • 0-IMSA_WLAN - IMS service is registered on WLAN • 1-IMSA_WWAN - IMS service is registered on WWAN • 2-IMSA_IWLAN - IMS service is registered on interworking WLAN

8.112.2 Field Documentation

8.112.2.1 uint8_t imsa_VoipRatInfo::TlvPresent

8.112.2.2 uint32_t imsa_VoipRatInfo::VoipRatVal

8.113 imsa_VoipSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t VoipSvcStatus

8.113.1 Detailed Description

This structure hold parameters about VOIP service info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
VoipSvcStatus	• Values: -0 - IMS VOIP service is not available -2 - IMS VOIP is in full service

8.113.2 Field Documentation

8.113.2.1 uint8_t imsa_VoipSvcStatusInfo::TlvPresent

8.113.2.2 uint32_t imsa_VoipSvcStatusInfo::VoipSvcStatus

8.114 imsa_VtRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent

- uint32_t VtRatVal

8.114.1 Detailed Description

This structure hold parameters about VT RAT info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
VtRatVal	• Values: • 0-IMSA_WLAN - IMS service is registered on WLAN • 1-IMSA_WWAN - IMS service is registered on WWAN • 2-IMSA_IWLAN - IMS service is registered on interworking WLAN

8.114.2 Field Documentation

8.114.2.1 uint8_t imsa_VtRatInfo::TlvPresent

8.114.2.2 uint32_t imsa_VtRatInfo::VtRatVal

8.115 imsa_VtSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t VtSvcStatus

8.115.1 Detailed Description

This structure hold parameters about VT service info.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
VtSvcStatus	• Values: -0 - IMS VT service is not available -2 - IMS VT is in full service

8.115.2 Field Documentation

8.115.2.1 uint8_t imsa_VtSvcStatusInfo::TlvPresent

8.115.2.2 uint32_t imsa_VtSvcStatusInfo::VtSvcStatus

8.116 ipv6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.116.1 Detailed Description

This structure contains the IPv6 address information

Parameters

IPV6PrefixLen	Length of the received IPv6 address • 0xff - Invalid IPv6 address information.
IPAddressV6	IPv6 address(in network byte order)

8.116.2 Field Documentation

8.116.2.1 uint16_t ipv6AddressInfo::IPAddressV6[8]

8.116.2.2 uint8_t ipv6AddressInfo::IPV6PrefixLen

8.117 LibPackGPRSRequestedQoS Struct Reference

Data Fields

- uint32_t precedenceClass
- uint32_t delayClass
- uint32_t reliabilityClass
- uint32_t peakThroughputClass
- uint32_t meanThroughputClass

8.117.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

Parameters

precedence-Class	• Precedence class
delayClass	• Delay class
reliabilityClass	• Reliability class
peak-Throughput-Class	• Peak throughput class
mean-Throughput-Class	• Mean throughput class

8.117.2 Field Documentation

8.117.2.1 uint32_t LibPackGPRSRequestedQoS::delayClass

8.117.2.2 uint32_t LibPackGPRSRequestedQoS::meanThroughputClass

8.117.2.3 uint32_t LibPackGPRSRequestedQoS::peakThroughputClass

8.117.2.4 uint32_t LibPackGPRSRequestedQoS::precedenceClass

8.117.2.5 uint32_t LibPackGPRSRequestedQoS::reliabilityClass

8.118 LibPackPCOIDList Struct Reference

Data Fields

- uint16_t PcoList [10]

8.118.1 Detailed Description

This structure contains information about the PCOID List

Parameters

PcoList	PCOID for Max 10

8.118.2 Field Documentation

8.118.2.1 uint16_t LibPackPCOIDList::PcoList[10]

8.119 LibPackPDNThrottleTimer Struct Reference

Data Fields

- uint32_t ThrottleTimer [10]

8.119.1 Detailed Description

This structure contains information about the PDN throttle timer

Parameters

ThrottleTimer	Throttle Timer for Max 10 PDN connections

8.119.2 Field Documentation

8.119.2.1 uint32_t LibPackPDNThrottleTimer::ThrottleTimer[10]

8.120 LibpackProfile3GPP Struct Reference

Data Fields

- uint8_t * pProfilename
- uint16_t * pProfilenameSize
- uint8_t * pPDPtype
- uint8_t * pPdpHdrCompType
- uint8_t * pPdpDataCompType
- uint8_t * pAPNName
- uint16_t * pAPNnameSize
- uint32_t * pPriDNSIPv4AddPref
- uint32_t * pSecDNSIPv4AddPref
- LibPackUMTSQoS * pUMTSReqQoS
- LibPackUMTSQoS * pUMTSMinQoS
- LibPackGPRSRequestedQoS * pGPRSRequestedQos
- LibPackGPRSRequestedQoS * pGPRSMinimumQoS
- uint8_t * pUsername
- uint16_t * pUsernameSize
- uint8_t * pPassword
- uint16_t * pPasswordSize
- uint8_t * pAuthenticationPref
- uint32_t * pIPv4AddrPref
- uint8_t * pPcscfAddrUsingPCO
- uint8_t * pPdpAccessConFlag
- uint8_t * pPcscfAddrUsingDhcp
- uint8_t * pImCnFlag
- LibPackTFTIDParams * pTFTID1Params
- LibPackTFTIDParams * pTFTID2Params
- uint8_t * pPdpContext
- uint8_t * pSecondaryFlag
- uint8_t * pPrimaryID
- uint16_t * pIPv6AddPref
- LibPackUMTSReqQoSSigInd * pUMTSReqQoSSigInd
- LibPackUMTSReqQoSSigInd * pUMTSMinQosSigInd
- uint16_t * pPriDNSIPv6addpref
- uint16_t * pSecDNSIPv6addpref
- uint8_t * pAddrAllocPref
- LibPackQosClassID * pQosClassID
- uint8_t * pAPNDisabledFlag
- uint32_t * pPDNInactivTimeout
- uint8_t * pAPNClass
- uint8_t * pSupportEmergencyCalls

8.120.1 Detailed Description

This structure contains Input parameters of unpack_wds_SLQSGetProfileSettings_t Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pProfileName	• One or more uint8_ts describing the profile • NULL pointer - Invalid data. • Bit to check in ParamPresenceMask - **16**
pProfilename-Size;	• This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **16**
pPDPType	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0x00 - PDP-IP (IPv4) – 0x01 - PDP-PPP – 0x02 - PDP-IPV6 – 0x03 - PDP-IPV4V6 • Bit to check in ParamPresenceMask - **17**
pPdpHdrComp-Type	• PDP header compression type – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095 • Bit to check in ParamPresenceMask - **18**
pPdpDataComp-Type	• PDP data compression type – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression • Bit to check in ParamPresenceMask - **19**
pAPNName	• Access point name • Bit to check in ParamPresenceMask - **20**
pAPNnameSize;	• This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **20**
pPriDNSIPv4-AddPref	• Primary DNS IPv4 Address Preference • Bit to check in ParamPresenceMask - **21**

pSecDNSIPv4-AddPref	• Secondary DNS IPv4 Address Preference • Bit to check in ParamPresenceMask - **22**
pUMTSReqQoS	• UMTS Requested QoS • Bit to check in ParamPresenceMask - **23**
pUMTSMinQoS	• UMTS Minimum QoS • Bit to check in ParamPresenceMask - **24**
pGPRS-RequestedQoS	• GPRS Requested QoS • Bit to check in ParamPresenceMask - **25**
pGPRS-MinimumQoS	• GPRS Minimum QoS • Bit to check in ParamPresenceMask - **26**
pUsername	• User name • Bit to check in ParamPresenceMask - **27**
pUsernameSize;	• This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **27**
pPassword	• Password • Bit to check in ParamPresenceMask - **28**
pPasswordSize;	• This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **28**
pAuthentication-Pref	• Authentication Preference – Bit map that indicates the authentication algorithm preference * Bit 0 - PAP preference · 0 - PAP is never performed · 1 - PAP may be performed * Bit 1 - CHAP preference · 0 - CHAP is never performed · 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism. • Bit to check in ParamPresenceMask - **29**

pIPv4AddrPref	• IPv4 Address Preference • Bit to check in ParamPresenceMask - **30**
pPcscfAddr-UsingPCO	• P-CSCF Address using PCO Flag – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0 • Bit to check in ParamPresenceMask - **31**
pPdpAccess-ConFlag	• PDP access control flag – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission • Bit to check in ParamPresenceMask - **32**
pPcscfAddr-UsingDhcp	• P-CSCF address using DHCP – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0 • Bit to check in ParamPresenceMask - **33**
pImCnFlag	• IM CN flag – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile • Bit to check in ParamPresenceMask - **34**
pTFTID1Params	• Traffic Flow Template • Bit to check in ParamPresenceMask - **35**
pTFTID2Params	• Traffic Flow Template • Bit to check in ParamPresenceMask - **36**
pPdpContext	• PDP context number • Bit to check in ParamPresenceMask - **37**
pSecondaryFlag	• PDP context secondary flag – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile • Bit to check in ParamPresenceMask - **38**

pPrimaryID	• PDP context primary ID • function SLQSGetProfileSettings() returns a default value 0xFF if this parameter is not returned by the device • Bit to check in ParamPresenceMask - **39**
pIPv6AddPref	• IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network • Bit to check in ParamPresenceMask - **40**
pUMTSReqQoS-SigInd	• UMTS requested QoS with Signalling Indication flag • Bit to check in ParamPresenceMask - **41**
pUMTSMinQoS-SigInd	• UMTS minimum QoS with Signalling Indication flag • Bit to check in ParamPresenceMask - **42**
pPrimaryDNSI-Pv6addpref	• Primary DNS IPv6 address preference – The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP • Bit to check in ParamPresenceMask - **43**
pSecondaryDN-SIPv6addpref	• Secondary DNS IPv6 address preference • Bit to check in ParamPresenceMask - **44**
paddrAllocation-Pref	• DHCP/NAS preference – This enumerated value may be used to indicate the address allocation preference * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation • Bit to check in ParamPresenceMask - **45**
pQosClassID	• 3GPP LTE QoS parameters • Bit to check in ParamPresenceMask - **46**

pAPNDisabled-Flag	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If set, the profile can not be used for making data calls • Any data call is failed locally • Values: – 0 - FALSE(default) – 1 - True • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings(). • Bit to check in ParamPresenceMask - **47**
pPDNInactiv-Timeout	• Optional 4 uint8_ts indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings(). • Bit to check in ParamPresenceMask - **48**
pAPNClass	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings(). • Bit to check in ParamPresenceMask - **49**
pSupport-EmergencyCalls	• Optional 1 Byte Flag indicating if the emergency call support is disabled/enabled • Can be queried to get current status • Values: – 0 - FALSE(default) – 1 - TRUE • Bit to check in ParamPresenceMask - **54**

8.120.2 Field Documentation

8.120.2.1 uint8_t* LibpackProfile3GPP::pAddrAllocPref

8.120.2.2 uint8_t* LibpackProfile3GPP::pAPNClass

8.120.2.3 uint8_t* LibpackProfile3GPP::pAPNDisabledFlag

8.120.2.4 uint8_t* LibpackProfile3GPP::pAPNName

8.120.2.5 uint16_t* LibpackProfile3GPP::pAPNnameSize

8.120.2.6 uint8_t* LibpackProfile3GPP::pAuthenticationPref

8.120.2.7 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSMinimumQoS

8.120.2.8 **LibPackGPRSRequestedQoS*** LibpackProfile3GPP::pGPRSRequestedQos

8.120.2.9 uint8_t* LibpackProfile3GPP::pImCnFlag

8.120.2.10 uint32_t* LibpackProfile3GPP::pIPv4AddrPref

8.120.2.11 uint16_t* LibpackProfile3GPP::pIPv6AddPref

8.120.2.12 uint8_t* LibpackProfile3GPP::pPassword

8.120.2.13 uint16_t* LibpackProfile3GPP::pPasswordSize

8.120.2.14 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingDhcp

8.120.2.15 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingPCO

8.120.2.16 uint32_t* LibpackProfile3GPP::pPDNInactivTimeout

8.120.2.17 uint8_t* LibpackProfile3GPP::pPdpAccessConFlag

8.120.2.18 uint8_t* LibpackProfile3GPP::pPdpContext

8.120.2.19 uint8_t* LibpackProfile3GPP::pPdpDataCompType

8.120.2.20 uint8_t* LibpackProfile3GPP::pPdpHdrCompType

8.120.2.21 uint8_t* LibpackProfile3GPP::pPDPtype

8.120.2.22 uint32_t* LibpackProfile3GPP::pPriDNSIPv4AddPref

8.120.2.23 uint16_t* LibpackProfile3GPP::pPriDNSIPv6addpref

8.120.2.24 uint8_t* LibpackProfile3GPP::pPrimaryID

8.120.2.25 uint8_t* LibpackProfile3GPP::pProfilename

8.120.2.26 uint16_t* LibpackProfile3GPP::pProfilenameSize

8.120.2.27 **LibPackQosClassID*** LibpackProfile3GPP::pQosClassID

8.120.2.28 uint32_t* LibpackProfile3GPP::pSecDNSIPv4AddPref

8.120.2.29 uint16_t* LibpackProfile3GPP::pSecDNSIPv6addpref

8.120.2.30 uint8_t* LibpackProfile3GPP::pSecondaryFlag

8.120.2.31 uint8_t* LibpackProfile3GPP::pSupportEmergencyCalls

8.120.2.32 **LibPackTFTIDParams*** LibpackProfile3GPP::pTFTID1Params

8.120.2.33 **LibPackTFTIDParams*** LibpackProfile3GPP::pTFTID2Params

8.120.2.34 **LibPackUMTSQoS*** LibpackProfile3GPP::pUMTSMinQoS

8.120.2.35 **LibPackUMTSReqQoSSigInd*** LibpackProfile3GPP::pUMTSMinQosSigInd

8.120.2.36 LibPackUMTSQoS∗ LibpackProfile3GPP::pUMTSReqQoS

8.120.2.37 LibPackUMTSReqQoSSigInd∗ LibpackProfile3GPP::pUMTSReqQoSSigInd

8.120.2.38 uint8_t∗ LibpackProfile3GPP::pUsername

8.120.2.39 uint16_t∗ LibpackProfile3GPP::pUsernameSize

8.121 LibpackProfile3GPP2 Struct Reference

Data Fields

- uint8_t ∗ pNegoDnsSrvrPref
- uint32_t ∗ pPppSessCloseTimerDO
- uint32_t ∗ pPppSessCloseTimer1x
- uint8_t ∗ pAllowLinger
- uint16_t ∗ pLcpAckTimeout
- uint16_t ∗ pIpcpAckTimeout
- uint16_t ∗ pAuthTimeout
- uint8_t ∗ pLcpCreqRetryCount
- uint8_t ∗ pIpcpCreqRetryCount
- uint8_t ∗ pAuthRetryCount
- uint8_t ∗ pAuthProtocol
- uint8_t ∗ pUserId
- uint16_t ∗ pUserIdSize
- uint8_t ∗ pAuthPassword
- uint16_t ∗ pAuthPasswordSize
- uint8_t ∗ pDataRate
- uint32_t ∗ pAppType
- uint8_t ∗ pDataMode
- uint8_t ∗ pAppPriority
- uint8_t ∗ pApnString
- uint16_t ∗ pApnStringSize
- uint8_t ∗ pPdnType
- uint8_t ∗ pIsPcscfAddressNedded
- uint32_t ∗ pPrimaryV4DnsAddress
- uint32_t ∗ pSecondaryV4DnsAddress
- uint16_t ∗ pPriV6DnsAddress
- uint16_t ∗ pSecV6DnsAddress
- uint8_t ∗ pRATType
- uint8_t ∗ pAPNEnabled3GPP2
- uint32_t ∗ pPDNInactivTimeout3GPP2
- uint8_t ∗ pAPNClass3GPP2

8.121.1 Detailed Description

This structure contains the 3GPP2 profile parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pNegoDnsSrvr-Pref	• Negotiate DNS Server Preference – 1 - (TRUE)implies request DNS addresses from the PDSN – 0 - (FALSE)implies do not request DNS addresses from the PDSN – Default value is 1 (TRUE) • Bit to check in ParamPresenceMask - **144**
pPppSessClose-TimerDO	• PPP Session Close Timer for DO – Timer value (in seconds) on DO indicating how long the PPP Session should linger before closing down • Bit to check in ParamPresenceMask - **145**
pPppSessClose-Timer1x	• PPP Session Close Timer for 1X – Timer value (in seconds) on 1X indicating how long the PPP session should linger before closing down • Bit to check in ParamPresenceMask - **146**
pAllowLinger	• Allow/disallow lingering of interface – 1 -(TRUE) implies allow lingering – 0 -(FALSE) implies do not allow lingering • Bit to check in ParamPresenceMask - **147**
pLcpAckTimeout	• LCP ACK Timeout – Value of LCP ACK Timeout in milliseconds • Bit to check in ParamPresenceMask - **148**
pIpcpAck-Timeout	• IPCP ACK Timeout – Value of IPCP ACK Timeout in milliseconds • Bit to check in ParamPresenceMask - **149**
pAuthTimeout	• AUTH Timeout – Value of Authentication Timeout in milliseconds • Bit to check in ParamPresenceMask - **150**
pLcpCreqRetry-Count	• LCP Configuration Request Retry Count • Bit to check in ParamPresenceMask - **151**
pIpcpCreqRetry-Count	• IPCP Configuration Request Retry Count • Bit to check in ParamPresenceMask - **152**
pAuthRetry-Count	• Authentication Retry Count value • Bit to check in ParamPresenceMask - **153**

pAuthProtocol	• Authentication Protocol – 1 - PAP – 2 - CHAP – 3 - PAP or CHAP • Bit to check in ParamPresenceMask - **154**
pUserId	• User ID to be used during data network authentication • maximum length allowed is 127 uint8_ts; • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value. • Bit to check in ParamPresenceMask - **155**
pUserIdSize;	• This parameter is an input parameter and should be initialised to the size of pUserId field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **155**
pAuthPassword	• Password to be used during data network authentication; • maximum length allowed is 127 uint8_ts • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value. • Bit to check in ParamPresenceMask - **156**
pAuthPassword-Size;	• This parameter is an input parameter and should be initialised to the size of pAuthPassword field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **156**
pDataRate	• Data Rate Requested – 0 - Low (Low speed Service Options (SO15) only) – 1 - Medium (SO33 + low R-SCH) – 2 - High (SO33 + high R-SCH) – Default is 2 • Bit to check in ParamPresenceMask - **157**
pAppType	• Application Type: – 0x00000001 - Default Application Type – 0x00000020 - LBS Application Type – 0x00000040 - Tethered Application Type – This parameter is not used while creating/modifying a profile • Bit to check in ParamPresenceMask - **158**

pDataMode	• Data Mode to use: – 0 - CDMA or HDR (Hybrid 1X/1xEV-DO) – 1 - CDMA Only (1X only) – 2 - HDR Only (1xEV-DO only) – Default is 0 • Bit to check in ParamPresenceMask - **159**
pAppPriority	• Application Priority – Numerical 1 uint8_t value defining the application priority; higher value implies higher priority – This parameter is not used while creating/modifying a profile • Bit to check in ParamPresenceMask - **160**
pApnString	• String representing the Access Point Name • maximum length allowed is 100 uint8_ts • QMI_ERR_ARG_TOO_LONG will be returned if the APN name is too long. • Bit to check in ParamPresenceMask - **161**
pApnStringSize;	• This parameter is an input parameter and should be initialised to the size of pApnString field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **161**
pPdnType	• Packed Data Network Type Requested: – 0 - IPv4 PDN Type – 1 - IPv6 PDN Type – 2 - IPv4 or IPv6 PDN Type – 3 - Unspecified PDN Type (implying no preference) • Bit to check in ParamPresenceMask - **162**
pIsPcscf-AddressNedded	• This boolean value is used to control if PCSCF address is requested from PDSN – 1 -(TRUE) implies request for PCSCF value from the PDSN – 0 -(FALSE) implies do not request for PCSCF value from the PDSN • Bit to check in ParamPresenceMask - **163**
pPrimaryV4Dns-Address	• IPv4 Primary DNS address – The Primary IPv4 DNS address that can be statically assigned to the UE • Bit to check in ParamPresenceMask - **164**
pSecondaryV4-DnsAddress	• IPv4 Secondary DNS address – The Secondary IPv4 DNS address that can be statically assigned to the UE • Bit to check in ParamPresenceMask - **165**

pPriV6Dns-Address	• Primary IPv6 DNS address – The Primary IPv6 DNS address that can be statically assigned to the UE • Bit to check in ParamPresenceMask - **166**
pSecV6Dns-Address	• Secondary IPv6 DNS address – The Secondary IPv6 DNS address that can be statically assigned to the UE • Bit to check in ParamPresenceMask - **167**
pRATType	• Optional 1 uint8_t Flag indicating RAT Type • Values: – 1 - HRPD – 2 - EHRPD – 3 - HRPD_EHRPD • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings(). • Bit to check in ParamPresenceMask - **168**
pAPNEnabled3-GPP2	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If disabled, the profile can not be used for making data calls • Values: – 0 - Disabled – 1 - Enabled(default value) • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings(). • Bit to check in ParamPresenceMask - **169**
pPDNInactiv-Timeout3GPP2	• Optional 4 uint8_ts indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings(). • Bit to check in ParamPresenceMask - **170**
pAPNClass3GP-P2	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings(). • Bit to check in ParamPresenceMask - **171**

8.121.2 Field Documentation

8.121.2.1 uint8_t* LibpackProfile3GPP2::pAllowLinger

8.121.2.2 uint8_t* LibpackProfile3GPP2::pAPNClass3GPP2

8.121.2.3 uint8_t* LibpackProfile3GPP2::pAPNEnabled3GPP2

8.121.2.4 uint8_t* LibpackProfile3GPP2::pApnString

8.121.2.5 uint16_t* LibpackProfile3GPP2::pApnStringSize

8.121.2.6 uint8_t* LibpackProfile3GPP2::pAppPriority

8.121.2.7 uint32_t* LibpackProfile3GPP2::pAppType

8.121.2.8 uint8_t* LibpackProfile3GPP2::pAuthPassword

8.121.2.9 uint16_t* LibpackProfile3GPP2::pAuthPasswordSize

8.121.2.10 uint8_t* LibpackProfile3GPP2::pAuthProtocol

8.121.2.11 uint8_t* LibpackProfile3GPP2::pAuthRetryCount

8.121.2.12 uint16_t* LibpackProfile3GPP2::pAuthTimeout

8.121.2.13 uint8_t* LibpackProfile3GPP2::pDataMode

8.121.2.14 uint8_t* LibpackProfile3GPP2::pDataRate

8.121.2.15 uint16_t* LibpackProfile3GPP2::pIpcpAckTimeout

8.121.2.16 uint8_t* LibpackProfile3GPP2::pIpcpCreqRetryCount

8.121.2.17 uint8_t* LibpackProfile3GPP2::pIsPcscfAddressNedded

8.121.2.18 uint16_t* LibpackProfile3GPP2::pLcpAckTimeout

8.121.2.19 uint8_t* LibpackProfile3GPP2::pLcpCreqRetryCount

8.121.2.20 uint8_t* LibpackProfile3GPP2::pNegoDnsSrvrPref

8.121.2.21 uint32_t* LibpackProfile3GPP2::pPDNInactivTimeout3GPP2

8.121.2.22 uint8_t* LibpackProfile3GPP2::pPdnType

8.121.2.23 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimer1x

8.121.2.24 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimerDO

8.121.2.25 uint32_t* LibpackProfile3GPP2::pPrimaryV4DnsAddress

8.121.2.26 uint16_t* LibpackProfile3GPP2::pPriV6DnsAddress

8.121.2.27 uint8_t* LibpackProfile3GPP2::pRATType

8.121.2.28 uint32_t* LibpackProfile3GPP2::pSecondaryV4DnsAddress

8.121.2.29 uint16_t* LibpackProfile3GPP2::pSecV6DnsAddress

8.121.2.30 uint8_t* LibpackProfile3GPP2::pUserId

8.121.2.31 uint16_t* LibpackProfile3GPP2::pUserIdSize

8.122 LibpackProfile3GPPV2 Struct Reference

Data Fields

- uint8_t * pProfilename
- uint16_t * pProfilenameSize
- uint8_t * pPDPtype
- uint8_t * pPdpHdrCompType
- uint8_t * pPdpDataCompType
- uint8_t * pAPNName
- uint16_t * pAPNnameSize
- uint32_t * pPriDNSIPv4AddPref
- uint32_t * pSecDNSIPv4AddPref
- LibPackUMTSQoS * pUMTSReqQoS
- LibPackUMTSQoS * pUMTSMinQoS
- LibPackGPRSRequestedQoS * pGPRSRequestedQos
- LibPackGPRSRequestedQoS * pGPRSMinimumQoS
- uint8_t * pUsername
- uint16_t * pUsernameSize
- uint8_t * pPassword
- uint16_t * pPasswordSize
- uint8_t * pAuthenticationPref
- uint32_t * pIPv4AddrPref
- uint8_t * pPcscfAddrUsingPCO
- uint8_t * pPdpAccessConFlag
- uint8_t * pPcscfAddrUsingDhcp
- uint8_t * pImCnFlag
- LibPackTFTIDParams * pTFTID1Params
- LibPackTFTIDParams * pTFTID2Params
- uint8_t * pPdpContext
- uint8_t * pSecondaryFlag
- uint8_t * pPrimaryID
- uint16_t * pIPv6AddPref
- LibPackUMTSReqQoSSigInd * pUMTSReqQoSSigInd
- LibPackUMTSReqQoSSigInd * pUMTSMinQosSigInd
- uint16_t * pPriDNSIPv6addpref
- uint16_t * pSecDNSIPv6addpref
- uint8_t * pAddrAllocPref
- LibPackQosClassID * pQosClassID
- uint8_t * pAPNDisabledFlag
- uint32_t * pPDNInactivTimeout
- uint8_t * pAPNClass
- uint64_t * pAPNBearer
- uint8_t * pSupportEmergencyCalls
- uint16_t * pOperatorPCOID
- uint16_t * pMcc
- LibPackProfileMnc * pMnc
- uint16_t * pMaxPDN
- uint16_t * pMaxPDNTimer
- uint16_t * pPDNWaitTimer

- uint32_t * pAppUserData
- uint8_t * pRoamDisallowFlag
- uint8_t * pPDNDisconnectWaitTimer
- uint8_t * pDnsWithDHCPFlag
- uint32_t * pLteRoamPDPType
- uint32_t * pUmtsRoamPDPType
- uint8_t * pIWLANtoLTEHandoverFlag
- uint8_t * pLTEtoIWLANHandoverFlag
- LibPackPDNThrottleTimer * pPDNThrottleTimer
- uint32_t * pOverridePDPType
- LibPackPCOIDList * pPCOIDList
- uint8_t * pMsisdnFlag
- uint8_t * pPersistFlag
- uint8_t * pClatFlag
- uint8_t * pIPV6DelegFlag

8.122.1 Detailed Description

This structure contains Input parameters of unpack_wds_SLQSGetProfileSettings_t Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pProfileName	• One or more uint8_ts describing the profile • NULL pointer - Invalid data. • Bit to check in ParamPresenceMask - **16**
pProfilename-Size;	• This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **16**
pPDPType	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0x00 - PDP-IP (IPv4) – 0x01 - PDP-PPP – 0x02 - PDP-IPV6 – 0x03 - PDP-IPV4V6 • Bit to check in ParamPresenceMask - **17**
pPdpHdrComp-Type	• PDP header compression type – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095 • Bit to check in ParamPresenceMask - **18**

pPdpDataCompType	• PDP data compression type – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression • Bit to check in ParamPresenceMask - **19**
pAPNName	• Access point name • Bit to check in ParamPresenceMask - **20**
pAPNnameSize;	• This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **20**
pPriDNSIPv4AddPref	• Primary DNS IPv4 Address Preference • Bit to check in ParamPresenceMask - **21**
pSecDNSIPv4AddPref	• Secondary DNS IPv4 Address Preference • Bit to check in ParamPresenceMask - **22**
pUMTSReqQoS	• UMTS Requested QoS • Bit to check in ParamPresenceMask - **23**
pUMTSMinQoS	• UMTS Minimum QoS • Bit to check in ParamPresenceMask - **24**
pGPRSRequestedQoS	• GPRS Requested QoS • Bit to check in ParamPresenceMask - **25**
pGPRSMinimumQoS	• GPRS Minimum QoS • Bit to check in ParamPresenceMask - **26**
pUsername	• User name • Bit to check in ParamPresenceMask - **27**
pUsernameSize;	• This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **27**

pPassword	• Password • Bit to check in ParamPresenceMask - **28**
pPasswordSize;	• This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 uint8_ts. • Bit to check in ParamPresenceMask - **28**
pAuthentication-Pref	• Authentication Preference – Bit map that indicates the authentication algorithm preference * Bit 0 - PAP preference · 0 - PAP is never performed · 1 - PAP may be performed * Bit 1 - CHAP preference · 0 - CHAP is never performed · 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism. • Bit to check in ParamPresenceMask - **29**
pIPv4AddrPref	• IPv4 Address Preference • Bit to check in ParamPresenceMask - **30**
pPcscfAddr-UsingPCO	• P-CSCF Address using PCO Flag – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0 • Bit to check in ParamPresenceMask - **31**
pPdpAccess-ConFlag	• PDP access control flag – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission • Bit to check in ParamPresenceMask - **32**
pPcscfAddr-UsingDhcp	• P-CSCF address using DHCP – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0 • Bit to check in ParamPresenceMask - **33**

pImCnFlag	• IM CN flag – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile • Bit to check in ParamPresenceMask - **34**
pTFTID1Params	• Traffic Flow Template • Bit to check in ParamPresenceMask - **35**
pTFTID2Params	• Traffic Flow Template • Bit to check in ParamPresenceMask - **36**
pPdpContext	• PDP context number • Bit to check in ParamPresenceMask - **37**
pSecondaryFlag	• PDP context secondary flag – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile • Bit to check in ParamPresenceMask - **38**
pPrimaryID	• PDP context primary ID • default value 0xFF if parameter not returned by the device • Bit to check in ParamPresenceMask - **39**
pIPv6AddPref	• IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network • Bit to check in ParamPresenceMask - **40**
pUMTSReqQoS-SigInd	• UMTS requested QoS with Signalling Indication flag • Bit to check in ParamPresenceMask - **41**
pUMTSMinQoS-SigInd	• UMTS minimum QoS with Signalling Indication flag • Bit to check in ParamPresenceMask - **42**
pPrimaryDNSI-Pv6addpref	• Primary DNS IPv6 address preference – The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP • Bit to check in ParamPresenceMask - **43**

pSecondaryDNSIPv6addpref	• Secondary DNS IPv6 address preference • Bit to check in ParamPresenceMask - **44**
paddrAllocationPref	• DHCP/NAS preference – This enumerated value may be used to indicate the address allocation preference * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation • Bit to check in ParamPresenceMask - **45**
pQosClassID	• 3GPP LTE QoS parameters • Bit to check in ParamPresenceMask - **46**
pAPNDisabledFlag	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If set, the profile can not be used for making data calls • Any data call is failed locally • Values: – 0 - FALSE(default) – 1 - True • Bit to check in ParamPresenceMask - **47**
pPDNInactivTimeout	• Optional 4 uint8_ts indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • Bit to check in ParamPresenceMask - **48**
pAPNClass	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • Bit to check in ParamPresenceMask - **49**
pAPNBearer	• Optional 8 Byte numeric APN bearer mask – 0x0000000000000001 - GSM – 0x0000000000000002 - WCDMA – 0x0000000000000004 - LTE – 0x8000000000000000 - Any • Bit to check in ParamPresenceMask - **53**

pSupport-EmergencyCalls	• Optional 1 Byte Flag indicating if the emergency call support is disabled/enabled • Can be queried to get current status • Values: – 0 - FALSE(default) – 1 - TRUE • Bit to check in ParamPresenceMask - **54**
pOperatorPCOI-D	• Optional 2 bytes value indicating container ID of this PCO • Bit to check in ParamPresenceMask - **55**
pMcc	• Optional 2 bytes value indicating Mobile country code • Range 0-999 • Bit to check in ParamPresenceMask - **56**
pMnc	• Optional 2 bytes Mobile network code and 1 byte flag to indicate if MNC includes PCS digit • range 0-999 • Bit to check in ParamPresenceMask - **57**
pMaxPDN	• Optional 2 bytes value indicating Max PDN connections per time block • Default is 20 • Range 0-1023 • Bit to check in ParamPresenceMask - **58**
pMaxPDNTimer	• Optional 2 bytes value indicating Max PDN connection timer • Default is 300 sec • Range 0-3600 sec • Bit to check in ParamPresenceMask - **59**
pPDNWaitTimer	• Optional 2 bytes value indicating PDN request wait interval • Default is 0 sec • Range 0-1023 sec • Bit to check in ParamPresenceMask - **60**
pAppUserData	• Optional 4 bytes value indicating user data ID in the profile • Bit to check in ParamPresenceMask - **61**
pRoamDisallow-Flag	• Optional 1 byte value indicating roaming disallowed flag is set or not • If flag is set as 1 UE is allowed to connect with APN while roaming • Bit to check in ParamPresenceMask - **62**

pPDN-DisconnectWait-Timer	• Optional 1 bytes value indicating PDN disconnect wait interval • Range 0-255 minutes • Bit to check in ParamPresenceMask - **63**
pDnsWithDHCP-Flag	• Optional 1 byte value indicating getting DNS address using DHCP – 0 - Dont request DNS with DHCP – 1 - Request DNS address with DHCP • Bit to check in ParamPresenceMask - **64**
pLteRoamPDP-Type	• Optional 4 bytes value indicating LTE roaming PDP type – WDS_COMMON_PDP_TYPE_PDP_IPV4 (0x00) - IPv4 – WDS_COMMON_PDP_TYPE_PDP_IPV6 (0x01) - IPv6 – WDS_COMMON_PDP_TYPE_PDP_IPV4V6 (0x02) - IPv4 and IPv6 – WDS_COMMON_PDP_TYPE_PDP_MAX (0xFF) - Nothing is configured • Bit to check in ParamPresenceMask - **65**
pUmtsRoamPD-PType	• Optional 4 bytes value indicating UMTS roaming PDP type – WDS_COMMON_PDP_TYPE_PDP_IPV4 (0x00) - IPv4 – WDS_COMMON_PDP_TYPE_PDP_IPV6 (0x01) - IPv6 – WDS_COMMON_PDP_TYPE_PDP_IPV4V6 (0x02) - IPv4 and IPv6 – WDS_COMMON_PDP_TYPE_PDP_MAX (0xFF) - Nothing is configured • Bit to check in ParamPresenceMask - **66**
pIWLANtoLTE-HandoverFlag	• This boolean value is used to indicate IWLAN to LTE handover is allowed or not – 1 -(TRUE) - Allowed – 0 -(FALSE) - Disallowed • Bit to check in ParamPresenceMask - **67**
pLTEtoIWLAN-HandoverFlag	• This boolean value is used to indicate LTE to IWLAN handover is allowed or not – 1 -(TRUE) - Allowed – 0 -(FALSE) - Disallowed • Bit to check in ParamPresenceMask - **68**
pPDNThrottle-Timer	• Optional param for the throttle timer values for Max 10 PDN connection • Bit to check in ParamPresenceMask - **69**
pOverridePDP-Type	• Optional 4 bytes value indicating overriding home PDP type – WDS_COMMON_PDP_TYPE_PDP_IPV4 (0x00) - IPv4 – WDS_COMMON_PDP_TYPE_PDP_IPV6 (0x01) - IPv6 – WDS_COMMON_PDP_TYPE_PDP_IPV4V6 (0x02) - IPv4 and IPv6 – WDS_COMMON_PDP_TYPE_PDP_MAX (0xFF) - Nothing is configured • Bit to check in ParamPresenceMask - **70**

pPCOIDList	• Optional param with 20 bytes, List of 10 PCOs • Bit to check in ParamPresenceMask - **71**
pMsisdnFlag	• This boolean value is used to indicate MSISDN flag – 1 -(TRUE) - Enabled – 0 -(FALSE) - Disabled • Bit to check in ParamPresenceMask - **72**
pPersistFlag	• This boolean value is used to indicate profile persistent flag – 1 -(TRUE) - Enabled – 0 -(FALSE) - Disabled • Bit to check in ParamPresenceMask - **143**
pClatFlag	• This boolean value is used to indicate if CLAT is enabled or not – 1 -(TRUE) - Enabled – 0 -(FALSE) - Disabled • Bit to check in ParamPresenceMask - **222**
pIPV6DelegFlag	• This boolean value is used to indicate if IPV6 prefix delegation flag is enabled or not – 1 -(TRUE) - Enabled – 0 -(FALSE) - Disabled • Bit to check in ParamPresenceMask - **223**

8.122.2 Field Documentation

8.122.2.1 uint8_t* LibpackProfile3GPPV2::pAddrAllocPref

8.122.2.2 uint64_t* LibpackProfile3GPPV2::pAPNBearer

8.122.2.3 uint8_t* LibpackProfile3GPPV2::pAPNClass

8.122.2.4 uint8_t* LibpackProfile3GPPV2::pAPNDisabledFlag

8.122.2.5 uint8_t* LibpackProfile3GPPV2::pAPNName

8.122.2.6 uint16_t* LibpackProfile3GPPV2::pAPNnameSize

8.122.2.7 uint32_t* LibpackProfile3GPPV2::pAppUserData

8.122.2.8 uint8_t* LibpackProfile3GPPV2::pAuthenticationPref

8.122.2.9 uint8_t* LibpackProfile3GPPV2::pClatFlag

8.122.2.10 uint8_t* LibpackProfile3GPPV2::pDnsWithDHCPFlag

8.122.2.11 LibPackGPRSRequestedQoS* LibpackProfile3GPPV2::pGPRSMinimumQoS

8.122.2.12 **LibPackGPRSRequestedQoS*** LibpackProfile3GPPV2::pGPRSRequestedQos

8.122.2.13 **uint8_t*** LibpackProfile3GPPV2::pImCnFlag

8.122.2.14 **uint32_t*** LibpackProfile3GPPV2::pIPv4AddrPref

8.122.2.15 **uint16_t*** LibpackProfile3GPPV2::pIPv6AddPref

8.122.2.16 **uint8_t*** LibpackProfile3GPPV2::pIPV6DelegFlag

8.122.2.17 **uint8_t*** LibpackProfile3GPPV2::pIWLANtoLTEHandoverFlag

8.122.2.18 **uint32_t*** LibpackProfile3GPPV2::pLteRoamPDPType

8.122.2.19 **uint8_t*** LibpackProfile3GPPV2::pLTEtoIWLANHandoverFlag

8.122.2.20 **uint16_t*** LibpackProfile3GPPV2::pMaxPDN

8.122.2.21 **uint16_t*** LibpackProfile3GPPV2::pMaxPDNTimer

8.122.2.22 **uint16_t*** LibpackProfile3GPPV2::pMcc

8.122.2.23 **LibPackProfileMnc*** LibpackProfile3GPPV2::pMnc

8.122.2.24 **uint8_t*** LibpackProfile3GPPV2::pMsisdnFlag

8.122.2.25 **uint16_t*** LibpackProfile3GPPV2::pOperatorPCOID

8.122.2.26 **uint32_t*** LibpackProfile3GPPV2::pOverridePDPType

8.122.2.27 **uint8_t*** LibpackProfile3GPPV2::pPassword

8.122.2.28 **uint16_t*** LibpackProfile3GPPV2::pPasswordSize

8.122.2.29 **LibPackPCOIDList*** LibpackProfile3GPPV2::pPCOIDList

8.122.2.30 **uint8_t*** LibpackProfile3GPPV2::pPcscfAddrUsingDhcp

8.122.2.31 **uint8_t*** LibpackProfile3GPPV2::pPcscfAddrUsingPCO

8.122.2.32 **uint8_t*** LibpackProfile3GPPV2::pPDNDisconnectWaitTimer

8.122.2.33 **uint32_t*** LibpackProfile3GPPV2::pPDNInactivTimeout

8.122.2.34 **LibPackPDNThrottleTimer*** LibpackProfile3GPPV2::pPDNThrottleTimer

8.122.2.35 **uint16_t*** LibpackProfile3GPPV2::pPDNWaitTimer

8.122.2.36 **uint8_t*** LibpackProfile3GPPV2::pPdpAccessConFlag

8.122.2.37 **uint8_t*** LibpackProfile3GPPV2::pPdpContext

8.122.2.38 **uint8_t*** LibpackProfile3GPPV2::pPdpDataCompType

8.122.2.39 **uint8_t*** LibpackProfile3GPPV2::pPdpHdrCompType

8.122.2.40 uint8_t* LibpackProfile3GPPV2::pPDPtype

8.122.2.41 uint8_t* LibpackProfile3GPPV2::pPersistFlag

8.122.2.42 uint32_t* LibpackProfile3GPPV2::pPriDNSIPv4AddPref

8.122.2.43 uint16_t* LibpackProfile3GPPV2::pPriDNSIPv6addpref

8.122.2.44 uint8_t* LibpackProfile3GPPV2::pPrimaryID

8.122.2.45 uint8_t* LibpackProfile3GPPV2::pProfilename

8.122.2.46 uint16_t* LibpackProfile3GPPV2::pProfilenameSize

8.122.2.47 **LibPackQosClassID*** LibpackProfile3GPPV2::pQosClassID

8.122.2.48 uint8_t* LibpackProfile3GPPV2::pRoamDisallowFlag

8.122.2.49 uint32_t* LibpackProfile3GPPV2::pSecDNSIPv4AddPref

8.122.2.50 uint16_t* LibpackProfile3GPPV2::pSecDNSIPv6addpref

8.122.2.51 uint8_t* LibpackProfile3GPPV2::pSecondaryFlag

8.122.2.52 uint8_t* LibpackProfile3GPPV2::pSupportEmergencyCalls

8.122.2.53 **LibPackTFTIDParams*** LibpackProfile3GPPV2::pTFTID1Params

8.122.2.54 **LibPackTFTIDParams*** LibpackProfile3GPPV2::pTFTID2Params

8.122.2.55 **LibPackUMTSQoS*** LibpackProfile3GPPV2::pUMTSMinQoS

8.122.2.56 **LibPackUMTSReqQoSSigInd*** LibpackProfile3GPPV2::pUMTSMinQosSigInd

8.122.2.57 **LibPackUMTSQoS*** LibpackProfile3GPPV2::pUMTSReqQoS

8.122.2.58 **LibPackUMTSReqQoSSigInd*** LibpackProfile3GPPV2::pUMTSReqQoSSigInd

8.122.2.59 uint32_t* LibpackProfile3GPPV2::pUmtsRoamPDPType

8.122.2.60 uint8_t* LibpackProfile3GPPV2::pUsername

8.122.2.61 uint16_t* LibpackProfile3GPPV2::pUsernameSize

8.123 LibPackprofile_3GPP Struct Reference

Data Fields

- uint8_t * pProfilename
- uint16_t * pProfilenameSize
- uint8_t * pPDPtype
- uint8_t * pPdpHdrCompType
- uint8_t * pPdpDataCompType
- uint8_t * pAPNName
- uint16_t * pAPNnameSize
- uint32_t * pPriDNSIPv4AddPref

- uint32_t * pSecDNSIPv4AddPref
- LibPackUMTSQoS * pUMTSReqQoS
- LibPackUMTSQoS * pUMTSMinQoS
- LibPackGPRSRequestedQoS * pGPRSRequestedQos
- LibPackGPRSRequestedQoS * pGPRSMinimumQoS
- uint8_t * pUsername
- uint16_t * pUsernameSize
- uint8_t * pPassword
- uint16_t * pPasswordSize
- uint8_t * pAuthenticationPref
- uint32_t * pIPv4AddrPref
- uint8_t * pPcscfAddrUsingPCO
- uint8_t * pPdpAccessConFlag
- uint8_t * pPcscfAddrUsingDhcp
- uint8_t * pImCnFlag
- LibPackTFTIDParams * pTFTID1Params
- LibPackTFTIDParams * pTFTID2Params
- uint8_t * pPdpContext
- uint8_t * pSecondaryFlag
- uint8_t * pPrimaryID
- uint16_t * pIPv6AddPref
- LibPackUMTSReqQoSSigInd * pUMTSReqQoSSigInd
- LibPackUMTSReqQoSSigInd * pUMTSMinQosSigInd
- uint16_t * pPriDNSIPv6addpref
- uint16_t * pSecDNSIPv6addpref
- uint8_t * pAddrAllocPref
- LibPackQosClassID * pQosClassID
- uint8_t * pAPNDisabledFlag
- uint32_t * pPDNInactivTimeout
- uint8_t * pAPNClass
- uint8_t * pSupportEmergencyCalls

8.123.1 Detailed Description

This structure contains Input/Output parameters of pack_wds_SLQSCreateProfile

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pProfileName	• One or more bytes describing the profile
pProfilename-Size;	• This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 bytes.
pPDPType	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0x00 - PDP-IP (IPv4) – 0x01 - PDP-PPP – 0x02 - PDP-IPV6 – 0x03 - PDP-IPV4V6

pPdpHdrComp-Type	• PDP header compression type – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095
pPdpDataComp-Type	• PDP data compression type – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression
pAPNName	• Access point name
pAPNnameSize;	• This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 bytes.
pPriDNSIPv4-AddPref	• Primary DNS IPv4 Address Preference
pSecDNSIPv4-AddPref	• Secondary DNS IPv4 Address Preference
pUMTSReqQoS	• UMTS Requested QoS
pUMTSMinQoS	• UMTS Minimum QoS
pGPRS-RequestedQoS	• GPRS Minimum QoS
pUsername	• User name
pUsernameSize;	• This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 bytes.
pPassword	• Password
pPasswordSize;	• This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 bytes.

pAuthentication-Pref	• Authentication Preference – Bit map that indicates the authentication algorithm preference * Bit 0 - PAP preference · 0 - PAP is never performed · 1 - PAP may be performed * Bit 1 - CHAP preference · 0 - CHAP is never performed · 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism.
pIPv4AddrPref	• IPv4 Address Preference
pPcscfAddr-UsingPCO	• P-CSCF Address using PCO Flag – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0
pPdpAccess-ConFlag	• PDP access control flag – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission
pPcscfAddr-UsingDhcp	• P-CSCF address using DHCP – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0
pImCnFlag	• IM CN flag – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile
pTFTID1Params	• Traffic Flow Template
pTFTID2Params	• Traffic Flow Template
pPdpContext	• PDP context number
pSecondaryFlag	• PDP context secondary flag – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile

pPrimaryID	• PDP context primary ID • function SLQSGetProfileSettings() returns a default value 0xFF if this parameter is not returned by the device
pIPv6AddPref	• IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network
pUMTSReqQoS-SigInd	• UMTS requested QoS with Signalling Indication flag
pUMTSMinQoS-SigInd	• UMTS minimum QoS with Signalling Indication flag
pPrimaryDNSI-Pv6addpref	• Primary DNS IPv6 address preference – The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP
pSecondaryDN-SIPv6addpref	• Secondary DNS IPv6 address preference
paddrAllocation-Pref	• DHCP/NAS preference – This enumerated value may be used to indicate the address allocation preference * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation
pQosClassID	• 3GPP LTE QoS parameters
pAPNDisabled-Flag	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If set, the profile can not be used for making data calls • Any data call is failed locally • Values: – 0 - FALSE(default) – 1 - True • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().
pPDNInactiv-Timeout	• Optional 4 Bytes indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().

pAPNClass	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().
pSupport-EmergencyCalls	• Optional 1 Byte Flag indicating if the emergency call support is disabled/enabled • If set, the profile can be used for making emergency calls • Values: – 0 - FALSE(default) – 1 - TRUE

8.123.2 Field Documentation

8.123.2.1 uint8_t* LibPackprofile_3GPP::pAddrAllocPref

8.123.2.2 uint8_t* LibPackprofile_3GPP::pAPNClass

8.123.2.3 uint8_t* LibPackprofile_3GPP::pAPNDisabledFlag

8.123.2.4 uint8_t* LibPackprofile_3GPP::pAPNName

8.123.2.5 uint16_t* LibPackprofile_3GPP::pAPNnameSize

8.123.2.6 uint8_t* LibPackprofile_3GPP::pAuthenticationPref

8.123.2.7 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSMinimumQoS

8.123.2.8 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSRequestedQos

8.123.2.9 uint8_t* LibPackprofile_3GPP::pImCnFlag

8.123.2.10 uint32_t* LibPackprofile_3GPP::pIPv4AddrPref

8.123.2.11 uint16_t* LibPackprofile_3GPP::pIPv6AddPref

8.123.2.12 uint8_t* LibPackprofile_3GPP::pPassword

8.123.2.13 uint16_t* LibPackprofile_3GPP::pPasswordSize

8.123.2.14 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingDhcp

8.123.2.15 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingPCO

8.123.2.16 uint32_t* LibPackprofile_3GPP::pPDNInactivTimeout

8.123.2.17 uint8_t* LibPackprofile_3GPP::pPdpAccessConFlag

8.123.2.18 uint8_t* LibPackprofile_3GPP::pPdpContext

8.123.2.19 uint8_t* LibPackprofile_3GPP::pPdpDataCompType

8.123.2.20 uint8_t* LibPackprofile_3GPP::pPdpHdrCompType

8.123.2.21 uint8_t* LibPackprofile_3GPP::pPDPtype

8.123.2.22 uint32_t* LibPackprofile_3GPP::pPriDNSIPv4AddPref

8.123.2.23 uint16_t* LibPackprofile_3GPP::pPriDNSIPv6addpref

8.123.2.24 uint8_t* LibPackprofile_3GPP::pPrimaryID

8.123.2.25 uint8_t* LibPackprofile_3GPP::pProfilename

8.123.2.26 uint16_t* LibPackprofile_3GPP::pProfilenameSize

8.123.2.27 **LibPackQosClassID*** LibPackprofile_3GPP::pQosClassID

8.123.2.28 uint32_t* LibPackprofile_3GPP::pSecDNSIPv4AddPref

8.123.2.29 uint16_t* LibPackprofile_3GPP::pSecDNSIPv6addpref

8.123.2.30 uint8_t* LibPackprofile_3GPP::pSecondaryFlag

8.123.2.31 uint8_t* LibPackprofile_3GPP::pSupportEmergencyCalls

8.123.2.32 **LibPackTFTIDParams*** LibPackprofile_3GPP::pTFTID1Params

8.123.2.33 **LibPackTFTIDParams*** LibPackprofile_3GPP::pTFTID2Params

8.123.2.34 **LibPackUMTSQoS*** LibPackprofile_3GPP::pUMTSMinQoS

8.123.2.35 **LibPackUMTSReqQoSSigInd*** LibPackprofile_3GPP::pUMTSMinQosSigInd

8.123.2.36 **LibPackUMTSQoS*** LibPackprofile_3GPP::pUMTSReqQoS

8.123.2.37 **LibPackUMTSReqQoSSigInd*** LibPackprofile_3GPP::pUMTSReqQoSSigInd

8.123.2.38 uint8_t* LibPackprofile_3GPP::pUsername

8.123.2.39 uint16_t* LibPackprofile_3GPP::pUsernameSize

8.124 LibPackprofile_3GPP2 Struct Reference

Data Fields

- uint8_t * pNegoDnsSrvrPref
- uint32_t * pPppSessCloseTimerDO
- uint32_t * pPppSessCloseTimer1x
- uint8_t * pAllowLinger
- uint16_t * pLcpAckTimeout
- uint16_t * pIpcpAckTimeout
- uint16_t * pAuthTimeout
- uint8_t * pLcpCreqRetryCount
- uint8_t * pIpcpCreqRetryCount
- uint8_t * pAuthRetryCount
- uint8_t * pAuthProtocol

- uint8_t * pUserId
- uint16_t * pUserIdSize
- uint8_t * pAuthPassword
- uint16_t * pAuthPassword_tSize
- uint8_t * pDataRate
- uint32_t * pAppType
- uint8_t * pDataMode
- uint8_t * pAppPriority
- uint8_t * pApnString
- uint16_t * pApnStringSize
- uint8_t * pPdnType
- uint8_t * pIsPcscfAddressNedded
- uint32_t * pPrimaryV4DnsAddress
- uint32_t * pSecondaryV4DnsAddress
- uint16_t * pPriV6DnsAddress
- uint16_t * pSecV6DnsAddress
- uint8_t * pRATType
- uint8_t * pAPNEnabled3GPP2
- uint32_t * pPDNInactivTimeout3GPP2
- uint8_t * pAPNClass3GPP2

8.124.1 Detailed Description

This structure contains the 3GPP2 profile parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pNegoDnsSrvr-Pref	• Negotiate DNS Server Preference – 1 - (TRUE)implies request DNS addresses from the PDSN – 0 - (FALSE)implies do not request DNS addresses from the PDSN – Default value is 1 (TRUE)
pPppSessClose-TimerDO	• PPP Session Close Timer for DO – Timer value (in seconds) on DO indicating how long the PPP Session should linger before closing down
pPppSessClose-Timer1x	• PPP Session Close Timer for 1X – Timer value (in seconds) on 1X indicating how long the PPP session should linger before closing down
pAllowLinger	• Allow/disallow lingering of interface – 1 -(TRUE) implies allow lingering – 0 -(FALSE) implies do not allow lingering
pLcpAckTimeout	• LCP ACK Timeout – Value of LCP ACK Timeout in milliseconds

pIpcpAck-Timeout	• IPCP ACK Timeout – Value of IPCP ACK Timeout in milliseconds
pAuthTimeout	• AUTH Timeout – Value of Authentication Timeout in milliseconds
pLcpCreqRetry-Count	• LCP Configuration Request Retry Count
pIpcpCreqRetry-Count	• IPCP Configuration Request Retry Count
pAuthRetry-Count	• Authentication Retry Count value
pAuthProtocol	• Authentication Protocol – 1 - PAP – 2 - CHAP – 3 - PAP or CHAP
pUserId	• User ID to be used during data network authentication • maximum length allowed is 127 bytes; • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
pUserIdSize;	• This parameter is an input parameter and should be initialised to the size of pUserId field. Size of this parameter is 2 bytes.
pAuthPassword	• Password to be used during data network authentication; • maximum length allowed is 127 bytes • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
pAuthPassword-Size;	• This parameter is an input parameter and should be initialised to the size of pAuthPassword field. Size of this parameter is 2 bytes.
pDataRate	• Data Rate Requested – 0 - Low (Low speed Service Options (SO15) only) – 1 - Medium (SO33 + low R-SCH) – 2 - High (SO33 + high R-SCH) – Default is 2

pAppType	• Application Type: – 0x00000001 - Default Application Type – 0x00000020 - LBS Application Type – 0x00000040 - Tethered Application Type – This parameter is not used while creating/modifying a profile
pDataMode	• Data Mode to use: – 0 - CDMA or HDR (Hybrid 1X/1xEV-DO) – 1 - CDMA Only (1X only) – 2 - HDR Only (1xEV-DO only) – Default is 0
pAppPriority	• Application Priority – Numerical 1 uint8_t value defining the application priority; higher value implies higher priority – This parameter is not used while creating/modifying a profile
pApnString	• String representing the Access Point Name • maximum length allowed is 100 bytes • QMI_ERR_ARG_TOO_LONG will be returned if the APN name is too long.
pApnStringSize;	• This parameter is an input parameter and should be initialised to the size of pApnString field. Size of this parameter is 2 bytes.
pPdnType	• Packed Data Network Type Requested: – 0 - IPv4 PDN Type – 1 - IPv6 PDN Type – 2 - IPv4 or IPv6 PDN Type – 3 - Unspecified PDN Type (implying no preference)
pIsPcscf-AddressNedded	• This boolean value is used to control if PCSCF address is requested from PDSN – 1 -(TRUE) implies request for PCSCF value from the PDSN – 0 -(FALSE) implies do not request for PCSCF value from the PDSN
pPrimaryV4Dns-Address	• IPv4 Primary DNS address – The Primary IPv4 DNS address that can be statically assigned to the UE
pSecondaryV4-DnsAddress	• IPv4 Secondary DNS address – The Secondary IPv4 DNS address that can be statically assigned to the UE

pPriV6Dns-Address	• Primary IPv6 DNS address – The Primary IPv6 DNS address that can be statically assigned to the UE
pSecV6Dns-Address	• Secondary IPv6 DNS address – The Secondary IPv6 DNS address that can be statically assigned to the UE
pRATType	• Optional 1 uint8_t Flag indicating RAT Type • Values: – 1 - HRPD – 2 - EHRPD – 3 - HRPD_EHRPD • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().
pAPNEnabled3-GPP2	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If disabled, the profile can not be used for making data calls • Values: – 0 - Disabled – 1 - Enabled(default value) • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().
pPDNInactiv-Timeout3GPP2	• Optional 4 Bytes indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().
pAPNClass3GP-P2	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().

8.124.2 Field Documentation

8.124.2.1 uint8_t* LibPackprofile_3GPP2::pAllowLinger

8.124.2.2 uint8_t* LibPackprofile_3GPP2::pAPNClass3GPP2

8.124.2.3 uint8_t* LibPackprofile_3GPP2::pAPNEnabled3GPP2

8.124.2.4 uint8_t* LibPackprofile_3GPP2::pApnString

8.124.2.5 uint16_t* LibPackprofile_3GPP2::pApnStringSize

8.124.2.6 uint8_t* LibPackprofile_3GPP2::pAppPriority

8.124.2.7 uint32_t* LibPackprofile_3GPP2::pAppType

8.124.2.8 uint8_t* LibPackprofile_3GPP2::pAuthPassword

8.124.2.9 uint16_t* LibPackprofile_3GPP2::pAuthPassword_tSize

8.124.2.10 uint8_t* LibPackprofile_3GPP2::pAuthProtocol

8.124.2.11 uint8_t* LibPackprofile_3GPP2::pAuthRetryCount

8.124.2.12 uint16_t* LibPackprofile_3GPP2::pAuthTimeout

8.124.2.13 uint8_t* LibPackprofile_3GPP2::pDataMode

8.124.2.14 uint8_t* LibPackprofile_3GPP2::pDataRate

8.124.2.15 uint16_t* LibPackprofile_3GPP2::pIpcpAckTimeout

8.124.2.16 uint8_t* LibPackprofile_3GPP2::pIpcpCreqRetryCount

8.124.2.17 uint8_t* LibPackprofile_3GPP2::pIsPcscfAddressNedded

8.124.2.18 uint16_t* LibPackprofile_3GPP2::pLcpAckTimeout

8.124.2.19 uint8_t* LibPackprofile_3GPP2::pLcpCreqRetryCount

8.124.2.20 uint8_t* LibPackprofile_3GPP2::pNegoDnsSrvrPref

8.124.2.21 uint32_t* LibPackprofile_3GPP2::pPDNInactivTimeout3GPP2

8.124.2.22 uint8_t* LibPackprofile_3GPP2::pPdnType

8.124.2.23 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimer1x

8.124.2.24 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimerDO

8.124.2.25 uint32_t* LibPackprofile_3GPP2::pPrimaryV4DnsAddress

8.124.2.26 uint16_t* LibPackprofile_3GPP2::pPriV6DnsAddress

8.124.2.27 uint8_t* LibPackprofile_3GPP2::pRATType

8.124.2.28 uint32_t* LibPackprofile_3GPP2::pSecondaryV4DnsAddress

8.124.2.29 uint16_t* LibPackprofile_3GPP2::pSecV6DnsAddress

8.124.2.30 uint8_t* LibPackprofile_3GPP2::pUserId

8.124.2.31 uint16_t* LibPackprofile_3GPP2::pUserIdSize

8.125 LibPackProfileMnc Struct Reference

Data Fields

- uint16_t MNC
- uint8_t PCSFlag

8.125.1 Detailed Description

This structure contains the MNC Information

Parameters

MNC	• Mobile Network Code • range 0-999
PCSFlag	• Indicate if PCS flag is included – 0 - FALSE – 1 - TRUE

8.125.2 Field Documentation

8.125.2.1 uint16_t LibPackProfileMnc::MNC

8.125.2.2 uint8_t LibPackProfileMnc::PCSFlag

8.126 LibPackQosClassID Struct Reference

Data Fields

- uint8_t QCI
- uint32_t gDlBitRate
- uint32_t maxDlBitRate
- uint32_t gUlBitRate
- uint32_t maxUlBitRate

8.126.1 Detailed Description

This structure contains 3GPP LTE QoS parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

QCI	• QOS specified using the QOS Class Identifier (QOS) values QCI value 0 - Requests the network to assign the appropriate QCI value QCI values 1-4 - Associated with guaranteed bit rates QCI values 5-9 - Associated with non-guaranteed bit rates
gDlBitRate	• Guaranteed DL bit rate

maxDlBitRate	• Maximum DL bit rate
gUlBitRate	• Guaranteed UL bit rate
maxUlBitRate	• Maximum UL bit rate

8.126.2 Field Documentation

8.126.2.1 uint32_t LibPackQosClassID::gDlBitRate

8.126.2.2 uint32_t LibPackQosClassID::gUlBitRate

8.126.2.3 uint32_t LibPackQosClassID::maxDlBitRate

8.126.2.4 uint32_t LibPackQosClassID::maxUlBitRate

8.126.2.5 uint8_t LibPackQosClassID::QCI

8.127 LibPackTFTIDParams Struct Reference

Data Fields

- uint8_t filterId
- uint8_t eValid
- uint8_t ipVersion
- uint16_t * pSourceIP
- uint8_t sourceIPMask
- uint8_t nextHeader
- uint16_t destPortRangeStart
- uint16_t destPortRangeEnd
- uint16_t srcPortRangeStart
- uint16_t srcPortRangeEnd
- uint32_t IPSECSPI
- uint16_t tosMask
- uint32_t flowLabel

8.127.1 Detailed Description

This structure contains traffic flow template parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

filterId	• Filter identifier

eValid	• Evaluation precedence index
ipVersion	• IP version number – 4 - IPv4 – 6 - IPv6
pSourceIP	• Source IP address – IPv4 - Fill the first 4 uint8_ts – IPv6 - Fill all the 16 uint8_ts
sourceIPMask	• Mask value for the source address
nextHeader	• Next header/protocol value
destPortRange-Start	• Start value of the destination port range
destPortRange-End	• End value of the destination port range
srcPortRange-Start	• Start value of the source port range
srcPortRange-End	• End value of the source port range
IPSECSPI	• IPSEC security parameter index
tosMask	• TOS mask (Traffic class for IPv6)
flowLabel	• Flow label

8.127.2 Field Documentation

8.127.2.1 uint16_t LibPackTFTIDParams::destPortRangeEnd

8.127.2.2 uint16_t LibPackTFTIDParams::destPortRangeStart

8.127.2.3 uint8_t LibPackTFTIDParams::eValid

8.127.2.4 uint8_t LibPackTFTIDParams::filterId

8.127.2.5 uint32_t LibPackTFTIDParams::flowLabel

8.127.2.6 uint32_t LibPackTFTIDParams::IPSECSPI

8.127.2.7 uint8_t LibPackTFTIDParams::ipVersion

8.127.2.8 uint8_t LibPackTFTIDParams::nextHeader

8.127.2.9 uint16_t* LibPackTFTIDParams::pSourceIP

8.127.2.10 uint8_t LibPackTFTIDParams::sourceIPMask

8.127.2.11 uint16_t LibPackTFTIDParams::srcPortRangeEnd

8.127.2.12 uint16_t LibPackTFTIDParams::srcPortRangeStart

8.127.2.13 uint16_t LibPackTFTIDParams::tosMask

8.128 LibPackUMTSQoS Struct Reference

Data Fields

- uint8_t trafficClass
- uint32_t maxUplinkBitrate
- uint32_t maxDownlinkBitrate
- uint32_t grntUplinkBitrate
- uint32_t grntDownlinkBitrate
- uint8_t qosDeliveryOrder
- uint32_t maxSDUSize
- uint8_t sduErrorRatio
- uint8_t resBerRatio
- uint8_t deliveryErrSDU
- uint32_t transferDelay
- uint32_t trafficPriority

8.128.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

trafficClass	• 0x00 - Subscribed • 0x01 - Conversational • 0x02 - Streaming • 0x03 - Interactive • 0x04 - Background • 0xff - Invalid UMTS Quality Of Service Information.
maxUplinkBitrate	• Maximum uplink bit rate in bits/sec

maxDownlink-Bitrate	• Maximum downlink bit rate in bits/sec
grntUplinkBitrate	• Guaranteed uplink bit rate in bits/sec
grntDownlink-Bitrate	• Guranteed downlink bit rate in bits/sec
qosDelivery-Order	- Qos delivery order • 0x00 - Subscribe • 0x01 - delivery order on • 0x02 - delivery order off
maxSDUSize	• Maximum SDU size
sduErrorRatio	- SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - 1∗10^(-2) • 0x02 - 7∗10^(-3) • 0x03 - 1∗10^(-3) • 0x04 - 1∗10^(-4) • 0x05 - 1∗10^(-5) • 0x06 - 1∗10^(-6) • 0x07 - 1∗10^(-1)
resBerRatio	- Residual bit error ratio • Target value for undetected bit error ratio in in the delivered SDUs. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)

deliveryErrSDU	- Delivery of erroneous SDUs • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)
transferDelay	- Transfer delay (ms) • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
trafficPriority	- Transfer handling priority • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

Note

Check trafficClass before use.

8.128.2 Field Documentation

8.128.2.1 uint8_t LibPackUMTSQoS::deliveryErrSDU

8.128.2.2 uint32_t LibPackUMTSQoS::grntDownlinkBitrate

8.128.2.3 uint32_t LibPackUMTSQoS::grntUplinkBitrate

8.128.2.4 uint32_t LibPackUMTSQoS::maxDownlinkBitrate

8.128.2.5 uint32_t LibPackUMTSQoS::maxSDUSize

8.128.2.6 uint32_t LibPackUMTSQoS::maxUplinkBitrate

8.128.2.7 uint8_t LibPackUMTSQoS::qosDeliveryOrder

8.128.2.8 uint8_t LibPackUMTSQoS::resBerRatio

8.128.2.9 uint8_t LibPackUMTSQoS::sduErrorRatio

8.128.2.10 uint8_t LibPackUMTSQoS::trafficClass

8.128.2.11 uint32_t LibPackUMTSQoS::trafficPriority

8.128.2.12 uint32_t LibPackUMTSQoS::transferDelay

8.129 LibPackUMTSReqQoSSigInd Struct Reference

Data Fields

- LibPackUMTSQoS UMTSReqQoS
- uint8_t SigInd

8.129.1 Detailed Description

This structure contains UMTS requested QoS with Signaling Indication flag

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

UMTSReqQoS	• Contains the UMTS Quality Of Service Information • See LibPackUMTSQoS
SigInd	- Signaling Indication flag • TRUE - Signaling indication ON • FALSE - Signaling indication OFF

8.129.2 Field Documentation

8.129.2.1 uint8_t LibPackUMTSReqQoSSigInd::SigInd

8.129.2.2 LibPackUMTSQoS LibPackUMTSReqQoSSigInd::UMTSReqQoS

8.130 loc_accelAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.130.1 Detailed Description

This structure contains Accelerometer Accept Ready Info

Parameters

injectEnable	• GNSS location engine is ready to accept data from sensor. • Values • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data

samplesPer-Batch	• number of samples per batch the GNSS location engine is to receive. • samplingFrequency = samplesPerBatch ∗ batchesPerSecond • samplesPerBatch must be a nonzero positive value.
batchPerSec	• LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.130.2 Field Documentation

8.130.2.1 uint16_t loc_accelAcceptReady::batchPerSec

8.130.2.2 uint8_t loc_accelAcceptReady::injectEnable

8.130.2.3 uint16_t loc_accelAcceptReady::samplesPerBatch

8.131 loc_accelTempAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.131.1 Detailed Description

This structure contains Accelerometer Temperature Accept Ready Info

Parameters

injectEnable	• GNSS location engine is ready to accept data from sensor. • Values • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data
samplesPer-Batch	• number of samples per batch the GNSS location engine is to receive. • samplingFrequency = samplesPerBatch ∗ batchesPerSecond • samplesPerBatch must be a nonzero positive value.
batchPerSec	• LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.131.2 Field Documentation

8.131.2.1 uint16_t loc_accelTempAcceptReady::batchPerSec

8.131.2.2 uint8_t loc_accelTempAcceptReady::injectEnable

8.131.2.3 uint16_t loc_accelTempAcceptReady::samplesPerBatch

8.132 loc_AppProviderInfoTlv Struct Reference

Data Fields

- uint8_t providerLen
- char provider [255]
- uint8_t nameLen
- char name [255]
- uint8_t verValid
- uint8_t versionLen
- char version [255]
- uint8_t TlvPresent

8.132.1 Detailed Description

Contain fields in struct loc_AppProviderInfoTlv

Parameters

providerLen	• number of sets of the following elements: – provider
provider	• Application provider.
nameLen	• number of sets of the following elements: – name
name	• Application name.
verValid	• Specifies whether the application version string contains a valid value: 0x00 (FALSE) - Application version string is invalid 0x01 (TRUE) - Application version string is valid
versionLen	• Number of sets of the following elements: version
version	• Application version.
TlvPresent	• Tlv Present or not

Note

None

8.132.2 Field Documentation

8.132.2.1 char loc_AppProviderInfoTlv::name[255]

8.132.2.2 uint8_t loc_AppProviderInfoTlv::nameLen

8.132.2.3 char loc_AppProviderInfoTlv::provider[255]

8.132.2.4 uint8_t loc_AppProviderInfoTlv::providerLen

8.132.2.5 uint8_t loc_AppProviderInfoTlv::TlvPresent

8.132.2.6 char loc_AppProviderInfoTlv::version[255]

8.132.2.7 uint8_t loc_AppProviderInfoTlv::versionLen

8.132.2.8 uint8_t loc_AppProviderInfoTlv::verValid

8.133 loc_BdsSV Struct Reference

Data Fields

- uint16_t id
- uint8_t mask

8.133.1 Detailed Description

This structure contains the BDS SV Info

Parameters

id	• SV ID of the satellite whose data is to be deleted. – Range for BDS: 201 to 237
mask	• Indicates if the ephemeris or almanac for a satellite is to be deleted • Valid values: – QMI_LOC_MASK_DELETE_EPHEMERIS (0x01) - Delete ephemeris for the satellite – QMI_LOC_MASK_DELETE_ALMANAC (0x02) - Delete almanac for the satellite

8.133.2 Field Documentation

8.133.2.1 uint16_t loc_BdsSV::id

8.133.2.2 uint8_t loc_BdsSV::mask

8.134 loc_BdsSVInfo Struct Reference

Data Fields

- uint8_t len
- loc_BdsSV * pSV

8.134.1 Detailed Description

This structure contains the number of sets of the BDS SVN Info

Parameters

len	• Number of sets of the following elements: – gnssSvId – deleteSvInfoMask
pSV	• Pointer to struct loc_BdsSV. See loc_BdsSV for more information

8.134.2 Field Documentation

8.134.2.1 uint8_t loc_BdsSVInfo::len

8.134.2.2 loc_BdsSV* loc_BdsSVInfo::pSV

8.135 loc_CellDb Struct Reference

Data Fields

- uint32_t mask

8.135.1 Detailed Description

This structure contains the cell database

Parameters

mask	• Mask for the cell database assistance data that is to be deleted • Valid values: – 0x00000001 - DELETE_CELLDB_POS – 0x00000002 - DELETE_CELLDB_LATEST_GPS_POS – 0x00000004 - DELETE_CELLDB_OTA_POS – 0x00000008 - DELETE_CELLDB_EXT_REF_POS – 0x00000010 - DELETE_CELLDB_TIMETAG – 0x00000020 - DELETE_CELLDB_CELLID – 0x00000040 - DELETE_CELLDB_CACHED_CELLID – 0x00000080 - DELETE_CELLDB_LAST_SRV_CELL – 0x00000100 - DELETE_CELLDB_CUR_SRV_CELL – 0x00000200 - DELETE_CELLDB_NEIGHBOR_INFO

8.135.2 Field Documentation

8.135.2.1 uint32_t loc_CellDb::mask

8.136 loc_ClkInfo Struct Reference

Data Fields

- uint32_t mask

8.136.1 Detailed Description

This structure contains the clock Info

Parameters

mask	• Mask for the clock information assistance data that is to be deleted • Valid bitmasks: – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_EST (0x00000001) - Mask to delete time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_FREQ_EST (0x00000002) - Mask to delete frequency estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_WEEK_NUMBER (0x00000004) - Mask to delete week number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_RTC_TIME (0x00000008) - Mask to delete RTC time from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_TRANSFER (0x00000010) - Mask to delete time transfer from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GPSTIME_EST (0x00000020) - Mask to delete GPS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLOTIME_EST (0x00000040) - Mask to delete GLONASS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLODAY_NUMBER (0x00000080) - Mask to delete GLONASS day number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO4YEAR_NUMBER (0x00000100) - Mask to delete GLONASS four year number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO_RF_GRP_DELAY (0x00000200) - Mask to delete GLONASS RF GRP delay from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_DISABLE_TT (0x00000400) - Mask to delete disable TT from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_LEAPSEC (0x00000800) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_GGTB (0x00001000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSTIME_EST (0x00002000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GB_GBTB (0x00004000) - Mask to delete Glonass-to-BDS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BG_BGTB (0x00008000) - Mask to delete BDS-to-GLONASS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSWEEK_NUMBER (0x00010000) - Mask to delete the BDS week number from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDS_RF_GRP_DELAY (0x00020000) - Mask to delete the BDS RF GRP delay from the clock information

8.136.2 Field Documentation

8.136.2.1 uint32_t loc_ClkInfo::mask

8.137 loc_FixCriteriaStatusTlv Struct Reference

Data Fields

- uint32_t status
- uint8_t TlvPresent

8.137.1 Detailed Description

Contain fields in struct loc_FixCriteriaStatusTlv

Parameters

status	• Status of the get fix criteria request • Valid values: – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out
TlvPresent	• Tlv Present or not

8.137.2 Field Documentation

8.137.2.1 uint32_t loc_FixCriteriaStatusTlv::status

8.137.2.2 uint8_t loc_FixCriteriaStatusTlv::TlvPresent

8.138 loc_GnssData Struct Reference

Data Fields

- uint64_t mask

8.138.1 Detailed Description

This structure contains the GNSS data

Parameters

mask	• Mask for the GNSS data that is to be deleted • Valid values: – QMI_LOC_MASK_DELETE_GPS_SVDIR (0x00000001) - Mask to delete GPS SVDIR – QMI_LOC_MASK_DELETE_GPS_SVSTEER (0x00000002) - Mask to delete GPS SVSTEER – QMI_LOC_MASK_DELETE_GPS_TIME (0x00000004) - Mask to delete GPS time – QMI_LOC_MASK_DELETE_GPS_ALM_CORR (0x00000008) - Mask to delete almanac correlation – QMI_LOC_MASK_DELETE_GLO_SVDIR (0x00000010) - Mask to delete GLONASS SVDIR – QMI_LOC_MASK_DELETE_GLO_SVSTEER (0x00000020) - Mask to delete GLONASS SVSTEER – QMI_LOC_MASK_DELETE_GLO_TIME (0x00000040) - Mask to delete GLONASS time – QMI_LOC_MASK_DELETE_GLO_ALM_CORR (0x00000080) - Mask to delete GLONASS almanac correlation – QMI_LOC_MASK_DELETE_SBAS_SVDIR (0x00000100) - Mask to delete SBAS SVDIR – QMI_LOC_MASK_DELETE_SBAS_SVSTEER (0x00000200) - Mask to delete SBAS SVSTEER – QMI_LOC_MASK_DELETE_POSITION (0x00000400) - Mask to delete position estimate – QMI_LOC_MASK_DELETE_TIME (0x00000800) - Mask to delete time estimate – QMI_LOC_MASK_DELETE_IONO (0x00001000) - Mask to delete IONO – QMI_LOC_MASK_DELETE_UTC (0x00002000) - Mask to delete UTC estimate – QMI_LOC_MASK_DELETE_HEALTH (0x00004000) - Mask to delete SV health record – QMI_LOC_MASK_DELETE_SADATA (0x00008000) - Mask to delete SADATA – QMI_LOC_MASK_DELETE_RTI (0x00010000) - Mask to delete RTI – QMI_LOC_MASK_DELETE_SV_NO_EXIST (0x00020000) - Mask to delete SV_NO_EXIST – QMI_LOC_MASK_DELETE_FREQ_BIAS_EST (0x00040000) - Mask to delete frequency bias estimate – QMI_LOC_MASK_DELETE_BDS_SVDIR (0x00080000) - Mask to delete BDS SVDIR – QMI_LOC_MASK_DELETE_BDS_SVSTEER (0x00100000) - Mask to delete BDS SVSTEER – QMI_LOC_MASK_DELETE_BDS_TIME (0x00200000) - Mask to delete BDS time – QMI_LOC_MASK_DELETE_BDS_ALM_CORR (0x00400000) - Mask to delete BDS almanac correlation – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GPS (0x00800000) - Mask to delete GNSS SV blacklist GPS – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GLO (0x01000000) - Mask to delete GNSS SV blacklist GLO – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_BDS (0x02000000) - Mask to delete GNSS SV blacklist BDS

8.138.2 Field Documentation

8.138.2.1 uint64_t loc_GnssData::mask

8.139 loc_gpsTime Struct Reference

Data Fields

- uint16_t gpsWeek
- uint32_t gpsTimeOfWeekMs

8.139.1 Detailed Description

This structure contains GPS Time info.

Parameters

gpsWeek	• Current GPS week as calculated from midnight, Jan. 6, 1980. • Units - Weeks
gpsTimeOfWeekMs	• Amount of time into the current GPS week. • Units - Milliseconds

8.139.2 Field Documentation

8.139.2.1 uint32_t loc_gpsTime::gpsTimeOfWeekMs

8.139.2.2 uint16_t loc_gpsTime::gpsWeek

8.140 loc_gyroAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.140.1 Detailed Description

This structure contains Gyroscope Accept Ready Info

Parameters

injectEnable	• GNSS location engine is ready to accept data from sensor. • Values • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data
samplesPerBatch	• number of samples per batch the GNSS location engine is to receive. • samplingFrequency = samplesPerBatch ∗ batchesPerSecond • samplesPerBatch must be a nonzero positive value.

batchPerSec	• LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.140.2 Field Documentation

8.140.2.1 uint16_t loc_gyroAcceptReady::batchPerSec

8.140.2.2 uint8_t loc_gyroAcceptReady::injectEnable

8.140.2.3 uint16_t loc_gyroAcceptReady::samplesPerBatch

8.141 loc_gyroTempAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.141.1 Detailed Description

This structure contains Gyroscope Temperature Accept Ready Info

Parameters

injectEnable	• GNSS location engine is ready to accept data from sensor. • Values • 0x01 - Ready to accept sensor data • 0x00 - Not ready to accept sensor data
samplesPer-Batch	• number of samples per batch the GNSS location engine is to receive. • samplingFrequency = samplesPerBatch ∗ batchesPerSecond • samplesPerBatch must be a nonzero positive value.
batchPerSec	• LTE NAS version minor • Number of sensor-data batches the GNSS location engine is to receive per second. • BatchesPerSecond must be a nonzero positive value.

8.141.2 Field Documentation

8.141.2.1 uint16_t loc_gyroTempAcceptReady::batchPerSec

8.141.2.2 uint8_t loc_gyroTempAcceptReady::injectEnable

8.141.2.3 uint16_t loc_gyroTempAcceptReady::samplesPerBatch

8.142 loc_HorAccuracyLvlTlv Struct Reference

Data Fields

- uint32_t accuracy
- uint8_t TlvPresent

8.142.1 Detailed Description

Contain fields in struct loc_HorAccuracyLvlTlv

Parameters

accuracy	• Horizontal accuracy level • Valid values: – 0x00000001 - LOW: Client requires low horizontal accuracy – 0x00000002 - MED: Client requires medium horizontal accuracy – 0x00000003 - HIGH: Client requires high horizontal accuracy
TlvPresent	• Tlv Present or not

8.142.2 Field Documentation

8.142.2.1 uint32_t loc_HorAccuracyLvlTlv::accuracy

8.142.2.2 uint8_t loc_HorAccuracyLvlTlv::TlvPresent

8.143 loc_IntermediateRptStateTlv Struct Reference

Data Fields

- uint32_t intermediate
- uint8_t TlvPresent

8.143.1 Detailed Description

Contain fields in struct loc_IntermediateRptStateTlv

Parameters

intermediate	• Intermediate Report state (ON, OFF). The client must explicitly set this field to OFF to stop receiving intermediate position reports. Intermediate position reports are generated at 1 Hz and are ON by default. If intermediate reports are turned ON, the client receives position reports even if the accuracy criteria is not met. The status in the position report is set to IN_PROGRESS for intermediate reports. • Valid values: 0x00000001 - ON: Client is interested in receiving intermediate reports 0x00000002 - OFF: Client is not interested in receiving intermediate reports

TlvPresent	• Tlv Present or not

8.143.2 Field Documentation

8.143.2.1 uint32_t loc_IntermediateRptStateTlv::intermediate

8.143.2.2 uint8_t loc_IntermediateRptStateTlv::TlvPresent

8.144 loc_IPv4Config Struct Reference

Data Fields

- uint32_t IPv4Addr
- uint16_t IPv4Port

8.144.1 Detailed Description

This structure specifies information regarding the IPv4 address and port.

Parameters

IPv4Addr	• IPv4 address.
IPv4Port	• IPv4 port.

8.144.2 Field Documentation

8.144.2.1 uint32_t loc_IPv4Config::IPv4Addr

8.144.2.2 uint16_t loc_IPv4Config::IPv4Port

8.145 loc_IPv4Info Struct Reference

Data Fields

- uint32_t address
- uint16_t port
- uint8_t TlvPresent

8.145.1 Detailed Description

This structure contains LOC Server IPV4 info field.

Parameters

address	• IPv4 address.
port	• IPv4 port.
TlvPresent	• Tlv Present or not

8.145.2 Field Documentation

8.145.2.1 uint32_t loc_IPv4Info::address

8.145.2.2 uint16_t loc_IPv4Info::port

8.145.2.3 uint8_t loc_IPv4Info::TlvPresent

8.146 loc_IPv6Config Struct Reference

Data Fields

- uint16_t IPv6Addr [8]
- uint32_t IPv6Port

8.146.1 Detailed Description

This structure specifies information regarding the IPv6 address and port.

Parameters

IPv6Addr	• IPv6 address. • Type - Array of unsigned integers • Maximum length of the array: 8
IPv6Port	• IPv6 port.

8.146.2 Field Documentation

8.146.2.1 uint16_t loc_IPv6Config::IPv6Addr[8]

8.146.2.2 uint32_t loc_IPv6Config::IPv6Port

8.147 loc_IPv6Info Struct Reference

Data Fields

- uint16_t address [8]

- uint32_t port
- uint8_t TlvPresent

8.147.1 Detailed Description

This structure contains LOC Get Server IPV6 info field.

Parameters

address	• IPv6 address. • Type - Array of unsigned integers • Maximum length of the array - 8
port	• IPv6 port.
TlvPresent	• Tlv Present or not

8.147.2 Field Documentation

8.147.2.1 uint16_t loc_IPv6Info::address[8]

8.147.2.2 uint32_t loc_IPv6Info::port

8.147.2.3 uint8_t loc_IPv6Info::TlvPresent

8.148 loc_LocApplicationInfo Struct Reference

Data Fields

- uint8_t appProviderLength
- uint8_t * pAppProvider
- uint8_t appNameLength
- uint8_t * pAppName
- uint8_t appVersionValid
- uint8_t appVersionLength
- uint8_t * pAppVersion

8.148.1 Detailed Description

This structure contains the Application Information

Parameters

appProvider-Length	• Length of the Application Provider
pAppProvider	• Application Provider • Depends upon the Length of application Provider

appNameLength	• Length of Application Name
pAppName	• Application Name • Depends upon the Length of application Name
appVersionValid	• Specifies whether the application version string contains a valid value • 0x00 (FALSE) Application version string is invalid • 0x01 (TRUE) Application version string is valid
appVersion-Length	• Length of Application Version
pAppVersion	• Application Version • Depends upon the Length of application Version

8.148.2 Field Documentation

8.148.2.1 uint8_t loc_LocApplicationInfo::appNameLength

8.148.2.2 uint8_t loc_LocApplicationInfo::appProviderLength

8.148.2.3 uint8_t loc_LocApplicationInfo::appVersionLength

8.148.2.4 uint8_t loc_LocApplicationInfo::appVersionValid

8.148.2.5 uint8_t* loc_LocApplicationInfo::pAppName

8.148.2.6 uint8_t* loc_LocApplicationInfo::pAppProvider

8.148.2.7 uint8_t* loc_LocApplicationInfo::pAppVersion

8.149 loc_MinIntervalTlv Struct Reference

Data Fields

- uint32_t interval
- uint8_t TlvPresent

8.149.1 Detailed Description

Contain fields in struct loc_MinIntervalTlv

Parameters

interval	• Minimum Interval Between fixes • Units: Milliseconds

TlvPresent	• Tlv Present or not

8.149.2 Field Documentation

8.149.2.1 uint32_t loc_MinIntervalTlv::interval

8.149.2.2 uint8_t loc_MinIntervalTlv::TlvPresent

8.150 loc_precisionDilution Struct Reference

Data Fields

- uint32_t PDOP
- uint32_t HDOP
- uint32_t VDOP

8.150.1 Detailed Description

This structure contains Dilution of precision associated with this position.

Parameters

PDOP	• Position dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy) • PDOP = square root of (Square of HDOP + Square of VDOP2)
HDOP	• Horizontal dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy)
VDOP	• Vertical dilution of precision. • Range- 1 (highest accuracy) to 50 (lowest accuracy)

8.150.2 Field Documentation

8.150.2.1 uint32_t loc_precisionDilution::HDOP

8.150.2.2 uint32_t loc_precisionDilution::PDOP

8.150.2.3 uint32_t loc_precisionDilution::VDOP

8.151 loc_satelliteInfo Struct Reference

Data Fields

- uint8_t svListLen

- uint32_t validMask
- uint32_t system
- uint16_t gnssSvId
- uint8_t healthStatus
- uint32_t svStatus
- uint8_t svInfoMask
- float elevation
- float azimuth
- float snr

8.151.1 Detailed Description

Contain fields in struct loc_satelliteInfo

Parameters

svListLen	• number of sets of the following elements: – validMask – system – gnssSvid – healthStatus – svStatus – svInfoMask – elevation – azimuth – snr
validMask	• Bitmask indicating which of the fields in this TLV are valid. Valid bitmasks: – 0x00000001 - VALID_SYSTEM – 0x00000002 - VALID_GNSS_SVID – 0x00000004 - VALID_HEALTH_STATUS – 0x00000008 - VALID_PROCESS_STATUS – 0x00000010 - VALID_SVINFO_MASK – 0x00000020 - VALID_ELEVATION – 0x00000040 - VALID_AZIMUTH – 0x00000080 - VALID_SNR
system	• Indicates to which constellation this SV belongs. Valid values: – eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite – eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite – eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite – eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite – eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite – eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite

gnssSvId	• GNSS SV ID. The GPS and GLONASS SVs can be disambiguated using the system field. Range: – FOR GPS: 1 to 32 – FOR GLONASS: 1 to 32 – FOR SBAS: 120 to 151 – for BDS: 201 to 237
healthStatus	• health status. Range: 0 - 1 – 0 - unhealthy – 1 - healthy
svStatus	• SV process status. Valid values: – eQMI_LOC_SV_STATUS_IDLE (1) - SV is not being actively processed – eQMI_LOC_SV_STATUS_SEARCH (2) - The system is searching for this SV – eQMI_LOC_SV_STATUS_TRACK (3) - SV is being tracked
svInfoMask	• Indicates whether almanac and ephemeris information is available. Valid bitmasks: – 0x01 - SVINFO_HAS_EPHEMERIS – 0x02 - SVINFO_HAS_ALMANAC
elevation	• SV elevation angle. – Units: Degrees – Range: 0 to 90
azimuth	• SV azimuth angle. – Units: Degrees – Range: 0 to 360
snr	• SV signal-to-noise ratio – Units: dB-Hz

8.151.2 Field Documentation

8.151.2.1 float loc_satelliteInfo::azimuth

8.151.2.2 float loc_satelliteInfo::elevation

8.151.2.3 uint16_t loc_satelliteInfo::gnssSvId

8.151.2.4 uint8_t loc_satelliteInfo::healthStatus

8.151.2.5 float loc_satelliteInfo::snr

8.151.2.6 uint8_t loc_satelliteInfo::svInfoMask

8.151.2.7 uint8_t loc_satelliteInfo::svListLen

8.151.2.8 uint32_t loc_satelliteInfo::svStatus

8.151.2.9 uint32_t loc_satelliteInfo::system

8.151.2.10 uint32_t loc_satelliteInfo::validMask

8.152 loc_sensorDataUsage Struct Reference

Data Fields

- uint32_t usageMask
- uint32_t aidingIndicatorMask

8.152.1 Detailed Description

This structure contains Sensor Data Usage info.

Parameters

usageMask	• Specifies which sensors were used in calculating the position in the position report. – Value * 0x00000001 - Accelerometer used * 0x00000002 - Gyroscope used
aidingIndicator-Mask	• Specifies which results were aided by sensors. – Value * 0x00000001 - AIDED_HEADING * 0x00000002 - AIDED_SPEED * 0x00000004 - AIDED_POSITION * 0x00000008 - AIDED_VELOCITY

8.152.2 Field Documentation

8.152.2.1 uint32_t loc_sensorDataUsage::aidingIndicatorMask

8.152.2.2 uint32_t loc_sensorDataUsage::usageMask

8.153 loc_SV Struct Reference

Data Fields

- uint16_t id
- uint32_t system
- uint8_t mask

8.153.1 Detailed Description

This structure contains the Delete LOC SV Info

Parameters

id	• LOC SV ID of the satellite whose data is to be deleted • Range: – For GPS: 1 to 32 – For SBAS: 33 to 64 – For GLONASS: 65 to 96
system	• Indicates to which constellation this loc_SV belongs • Valid values: – eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite – eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite – eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite – eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite – eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite – eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite
mask	• Indicates if the ephemeris or almanac for a satellite is to be deleted • Valid values: – 0x01 - DELETE_EPHEMERIS – 0x02 - DELETE_ALMANAC

8.153.2 Field Documentation

8.153.2.1 uint16_t loc_SV::id

8.153.2.2 uint8_t loc_SV::mask

8.153.2.3 uint32_t loc_SV::system

8.154 loc_SVInfo Struct Reference

Data Fields

- uint8_t len
- loc_SV * pSV

8.154.1 Detailed Description

This structure contains the elements of Delete LOC SV Info

Parameters

len	• Number of sets of the following elements in struct loc_SV: – gnssSvId – system – deleteSvInfoMask
pSV	• Pointer to struct loc_SV. See loc_SV for more information

8.154.2 Field Documentation

8.154.2.1 uint8_t loc_SVInfo::len

8.154.2.2 loc_SV* loc_SVInfo::pSV

8.155 loc_svUsedforFix Struct Reference

Data Fields

- uint8_t gnssSvUsedList_len
- uint16_t gnssSvUsedList [255]

8.155.1 Detailed Description

This structure contains SVs Used to Calculate the Fix.

Parameters

gnssSvUsedList-_len	• Number of sets of gnssSvUsedList
pGnssSvUsed-List	• Entry in the list contains the SV ID of a satellite used for calculating this position report. • Following information is associated with each SV ID: – GPS - 1 to 32 – SBAS - 33 to 64 – GLONASS - 65 to 96 – QZSS - 193 to 197 – BDS - 201 to 237

8.155.2 Field Documentation

8.155.2.1 uint16_t loc_svUsedforFix::gnssSvUsedList[255]

8.155.2.2 uint8_t loc_svUsedforFix::gnssSvUsedList_len

8.156 loc_urlAddr Struct Reference

Data Fields

- char address [256]
- uint8_t TlvPresent

8.156.1 Detailed Description

This structure contains LOC Get Server URL address field.

Parameters

address	• Uniform Resource Locator • Type - NULL terminated string • Maximum string length (including NULL terminator) 256
TlvPresent	• Tlv Present or not

8.156.2 Field Documentation

8.156.2.1 char loc_urlAddr::address[256]

8.156.2.2 uint8_t loc_urlAddr::TlvPresent

8.157 loc_URLAddrInfo Struct Reference

Data Fields

- char urlAddr [256]

8.157.1 Detailed Description

This structure specifies information regarding the URL.

Parameters

urlAddr	• URL address. • Type - NULL-terminated string • Maximum string length (including NULL terminator) - 256

8.157.2 Field Documentation

8.157.2.1 char loc_URLAddrInfo::urlAddr[256]

8.158 lteSSInfo Struct Reference

Data Fields

- int8_t rssi
- int8_t rsrq
- int16_t rsrp
- int16_t snr

8.158.1 Detailed Description

This structure contains the parameters for LTE Signal Strength Information

Parameters

rssi	• RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal. – For CDMA and UMTS, this indicates forward link pilot Ec – For GSM, this indicates received signal strength
rsrq	• RSRQ value in dB (signed integer value) as measured by L1. • Range: -3 to -20 (-3 means -3 dB, -20 means -20 dB).
rsrp	• Current RSRP in dBm as measured by L1. • Range: -44 to -140 (-44 means -44 dBm, -140 means -140 dBm).
snr	• SNR level as a scaled integer in units of 0.1 dB. e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246,

8.158.2 Field Documentation

8.158.2.1 int16_t lteSSInfo::rsrp

8.158.2.2 int8_t lteSSInfo::rsrq

8.158.2.3 int8_t lteSSInfo::rssi

8.158.2.4 int16_t lteSSInfo::snr

8.159 MBIM_CLOSE_DONE_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader
- uint32_t Status

8.159.1 Field Documentation

8.159.1.1 MBIM_MESSAGE_HEADER MBIM_CLOSE_DONE_MSG::MessageHeader

8.159.1.2 uint32_t MBIM_CLOSE_DONE_MSG::Status

8.160 MBIM_CLOSE_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader

8.160.1 Field Documentation

8.160.1.1 MBIM_MESSAGE_HEADER MBIM_CLOSE_MSG::MessageHeader

8.161 MBIM_COMMAND_DONE_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader
- MBIM_FRAGMENT_HEADER FragmentHeader
- uint8_t DeviceServiceId [16]
- uint32_t CID
- uint32_t Status
- uint32_t InformationBufferLength

8.161.1 Field Documentation

8.161.1.1 uint32_t MBIM_COMMAND_DONE_MSG::CID

8.161.1.2 uint8_t MBIM_COMMAND_DONE_MSG::DeviceServiceId[16]

8.161.1.3 MBIM_FRAGMENT_HEADER MBIM_COMMAND_DONE_MSG::FragmentHeader

8.161.1.4 uint32_t MBIM_COMMAND_DONE_MSG::InformationBufferLength

8.161.1.5 MBIM_MESSAGE_HEADER MBIM_COMMAND_DONE_MSG::MessageHeader

8.161.1.6 uint32_t MBIM_COMMAND_DONE_MSG::Status

8.162 MBIM_COMMAND_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader
- MBIM_FRAGMENT_HEADER FragmentHeader
- uint8_t DeviceServiceId [16]
- uint32_t CID
- uint32_t CommandType
- uint32_t InformationBufferLength

8.162.1 Field Documentation

8.162.1.1 uint32_t MBIM_COMMAND_MSG::CID

8.162.1.2 uint32_t MBIM_COMMAND_MSG::CommandType

8.162.1.3 uint8_t MBIM_COMMAND_MSG::DeviceServiceId[16]

8.162.1.4 MBIM_FRAGMENT_HEADER MBIM_COMMAND_MSG::FragmentHeader

8.162.1.5 uint32_t MBIM_COMMAND_MSG::InformationBufferLength

8.162.1.6 MBIM_MESSAGE_HEADER MBIM_COMMAND_MSG::MessageHeader

8.163 MBIM_FRAGMENT_HEADER Struct Reference

Data Fields

- uint32_t TotalFragments
- uint32_t CurrentFragment

8.163.1 Field Documentation

8.163.1.1 uint32_t MBIM_FRAGMENT_HEADER::CurrentFragment

8.163.1.2 uint32_t MBIM_FRAGMENT_HEADER::TotalFragments

8.164 MBIM_FRAGMENT_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader
- MBIM_FRAGMENT_HEADER FragmentHeader

8.164.1 Field Documentation

8.164.1.1 MBIM_FRAGMENT_HEADER MBIM_FRAGMENT_MSG::FragmentHeader

8.164.1.2 MBIM_MESSAGE_HEADER MBIM_FRAGMENT_MSG::MessageHeader

8.165 MBIM_INDICATE_STATUS_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader
- MBIM_FRAGMENT_HEADER FragmentHeader
- uint8_t DeviceServiceId [16]
- uint32_t CID
- uint32_t InformationBufferLength

8.165.1 Field Documentation

8.165.1.1 uint32_t MBIM_INDICATE_STATUS_MSG::CID

8.165.1.2 uint8_t MBIM_INDICATE_STATUS_MSG::DeviceServiceId[16]

8.165.1.3 MBIM_FRAGMENT_HEADER MBIM_INDICATE_STATUS_MSG::FragmentHeader

8.165.1.4 uint32_t MBIM_INDICATE_STATUS_MSG::InformationBufferLength

8.165.1.5 MBIM_MESSAGE_HEADER MBIM_INDICATE_STATUS_MSG::MessageHeader

8.166 MBIM_MESSAGE_HEADER Struct Reference

Data Fields

- uint32_t MessageType
- uint32_t MessageLength
- uint32_t TransactionId

8.166.1 Field Documentation

8.166.1.1 uint32_t MBIM_MESSAGE_HEADER::MessageLength

8.166.1.2 uint32_t MBIM_MESSAGE_HEADER::MessageType

8.166.1.3 uint32_t MBIM_MESSAGE_HEADER::TransactionId

8.167 MBIM_OPEN_DONE_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader
- uint32_t Status

8.167.1 Field Documentation

8.167.1.1 MBIM_MESSAGE_HEADER MBIM_OPEN_DONE_MSG::MessageHeader

8.167.1.2 uint32_t MBIM_OPEN_DONE_MSG::Status

8.168 MBIM_OPEN_MSG Struct Reference

Data Fields

- MBIM_MESSAGE_HEADER MessageHeader
- uint32_t MaxControlTransfer

8.168.1 Field Documentation

8.168.1.1 uint32_t MBIM_OPEN_MSG::MaxControlTransfer

8.168.1.2 MBIM_MESSAGE_HEADER MBIM_OPEN_MSG::MessageHeader

8.169 MbimEventEntry Struct Reference

Data Fields

- uint8_t DeviceServiceId [16]
- uint32_t CidCount
- uint32_t Cids [64]

8.169.1 Detailed Description

This structure represents the command IDs (CIDs) of indications related to a specific MBIM device service for which a user will be informed.

Parameters

DeviceServiceId	• The MBIM device service's UUID which relates to the indications.
CidCount	• Number of CIDs for which indications are to be generated. 0 means all CIDs of that MBIM device service.
Cids	• Fixed size array of CIDs. The first 'CidCount' elements of the array will be passed to the device, upto a maximum of MBIM_MAX_CIDS.

8.169.2 Field Documentation

8.169.2.1 uint32_t MbimEventEntry::CidCount

8.169.2.2 uint32_t MbimEventEntry::Cids[64]

8.169.2.3 uint8_t MbimEventEntry::DeviceServiceId[16]

8.170 MbimIndicator Struct Reference

Data Fields

- struct MbimIndicator * pPrev
- struct MbimIndicator * pNext
- MBIM_INDICATION_CALLBACK pIndicationCallback
- void * pIndicationCallbackContext

8.170.1 Detailed Description

This structure abstracts a MBIM indication callback. Users who wish to be notified of MBIM indications can register one of these objects with the MbimTransport object using MbimTransport_AttachIndicator/MbimTransport_DetachIndicator calls.

Parameters

pPrev	• A link so that this object can be stored in a linked list by the transport layer. This points to the previous object or NULL if this is the first object in the list.
pNext	• A link so that this object can be stored in a linked list by the transport layer. This points to the next object or NULL if this is the last object in the list.
pIndication-Callback	• User callback for MBIM indication.
pIndication-CallbackContext	• User context for user callback.

8.170.2 Field Documentation

8.170.2.1 MBIM_INDICATION_CALLBACK MbimIndicator::pIndicationCallback

8.170.2.2 void* MbimIndicator::pIndicationCallbackContext

8.170.2.3 struct MbimIndicator* MbimIndicator::pNext

8.170.2.4 struct MbimIndicator* MbimIndicator::pPrev

8.171 MbimSyncObject Struct Reference

Data Fields

- pthread_mutex_t mutex
- pthread_cond_t condition
- uint32_t status
- uint8_t * informationBuffer
- uint32_t informationBufferLength

8.171.1 Detailed Description

This structure is a wrapper around a mutex and condition object and an external buffer. It is a convenience object which may be used to create synchronous MBIM APIs.

Parameters

mutex	• For thread safety.
condition	• Triggered on completion.
status	• The returned MBIM transaction status.

informationBuffer	• Buffer to which the transaction response's informationBuffer will be copied.
information-BufferLength	• Set to maximum expected size in bytes of transaction response's informationBuffer. On request completion, it is set to size of transaction response's informationBuffer.

8.171.2 Field Documentation

8.171.2.1 pthread_cond_t MbimSyncObject::condition

8.171.2.2 uint8_t* MbimSyncObject::informationBuffer

8.171.2.3 uint32_t MbimSyncObject::informationBufferLength

8.171.2.4 pthread_mutex_t MbimSyncObject::mutex

8.171.2.5 uint32_t MbimSyncObject::status

8.172 MbimTransaction Struct Reference

Data Fields

- struct MbimTransaction * pPrev
- struct MbimTransaction * pNext
- uint32_t transactionId
- MBIM_DONE_CALLBACK pDoneCallback
- void * pDoneCallbackContext
- uint32_t status

8.172.1 Detailed Description

This structure abstracts a MBIM transaction. It is responsible for tracking a MBIM request to its response.

Parameters

pPrev	• A link so that this object can be stored in a linked list by the transport layer. This points to the previous object or NULL if this is the first object in the list.
pNext	• A link so that this object can be stored in a linked list by the transport layer. This points to the next object or NULL if this is the last object in the list.
transactionId	• MBIM transaction ID. Can never be 0.
pDoneCallback	• User callback for MBIM response.
pDoneCallback-Context	• User context for user callback.

status	• MBIM status of response, typically contained in first response fragment.

8.172.2 Field Documentation

8.172.2.1 MBIM_DONE_CALLBACK MbimTransaction::pDoneCallback

8.172.2.2 void* MbimTransaction::pDoneCallbackContext

8.172.2.3 struct MbimTransaction* MbimTransaction::pNext

8.172.2.4 struct MbimTransaction* MbimTransaction::pPrev

8.172.2.5 uint32_t MbimTransaction::status

8.172.2.6 uint32_t MbimTransaction::transactionId

8.173 MbimTransport Struct Reference

Data Fields

- int deviceFd
- int shutdownFd
- pthread_mutex_t writeLock
- pthread_t readThread
- uint32_t transactionId
- pthread_mutex_t transactionIdLock
- MbimTransaction * transactionList
- pthread_mutex_t transactionListLock
- struct MultiFragmentMessage * pIncomingMessage
- MbimIndicator * indicatorList
- pthread_mutex_t indicatorListLock

8.173.1 Detailed Description

This structure abstracts a MBIM transport layer. It is responsible for applying MBIM headers to packets destined for MBIM device services and forwarding them to the device. It is responsible for receiving MBIM packets from the device. On the basis of MBIM headers, it assembles multi-fragment responses and indications and delivers them back to user.

Parameters

deviceFd	• File descriptor of device.
shutdownFd	• Event descriptor. Used to signal read thread to terminate.
writeLock	• Provides mutual exclusion for write operations.

readThread	• Thread responsible for reading from device.
transactionId	• MBIM transaction ID. Upper limit 0xffffffff. Never 0. Increments for each transaction.
transactionId-Lock	• Mutex to make transactionId access thread-safe.
transactionList	• Head of linked list of outstanding transactions.
transactionList-Lock	• Provides thread safety for the transaction list.
pIncoming-Message	• Private workspace to assemble incoming information fragments.
indicatorList	• Head of linked list of indicators which are called when MBIM indications are received.
indicatorListLock	• Provides thread safety for the indicators list.

8.173.2 Field Documentation

8.173.2.1 int MbimTransport::deviceFd

8.173.2.2 MbimIndicator* MbimTransport::indicatorList

8.173.2.3 pthread_mutex_t MbimTransport::indicatorListLock

8.173.2.4 struct MultiFragmentMessage* MbimTransport::pIncomingMessage

8.173.2.5 pthread_t MbimTransport::readThread

8.173.2.6 int MbimTransport::shutdownFd

8.173.2.7 uint32_t MbimTransport::transactionId

8.173.2.8 pthread_mutex_t MbimTransport::transactionIdLock

8.173.2.9 MbimTransaction* MbimTransport::transactionList

8.173.2.10 pthread_mutex_t MbimTransport::transactionListLock

8.173.2.11 pthread_mutex_t MbimTransport::writeLock

8.174 messageModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMessageModeInfo MessageModeInfo

8.174.1 Detailed Description

This structure contains message mode TLV information.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
MessageMode-Info	• Message Mode • See sMSMessageModeInfo for more information

8.174.2 Field Documentation

8.174.2.1 sMSMessageModeInfo messageModeTlv::MessageModeInfo

8.174.2.2 uint8_t messageModeTlv::TlvPresent

8.175 nas_acqOrderPref Struct Reference

Data Fields

- uint8_t acqOrdeLen
- uint8_t * pAcqOrder

8.175.1 Detailed Description

Contain the Acquisition Order Preference.

Parameters

acqOrdeLen	• Number of sets of the following elements.
pAcqOrder	• Acquisition order preference to be set. Values: – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA

8.175.2 Field Documentation

8.175.2.1 uint8_t nas_acqOrderPref::acqOrdeLen

8.175.2.2 uint8_t* nas_acqOrderPref::pAcqOrder

8.176 nas_AcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t acqOrdeLen
- uint8_t * pAcqOrder
- uint8_t TlvPresent

8.176.1 Detailed Description

Contain the Acquisition Order Preference.

Parameters

acqOrdeLen	• Number of sets of the following elements.
pAcqOrder	• Acquisition order preference to be set. Values: – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA – 0x0C - NAS_RADIO_IF_NR5G - NR5G
TlvPresent	• Tlv Present

8.176.2 Field Documentation

8.176.2.1 uint8_t nas_AcqOrderPrefTlv::acqOrdeLen

8.176.2.2 uint8_t* nas_AcqOrderPrefTlv::pAcqOrder

8.176.2.3 uint8_t nas_AcqOrderPrefTlv::TlvPresent

8.177 nas_ActPilotPNElement Struct Reference

Data Fields

- uint16_t ActSetPilotPN
- uint8_t ActSetPilotPNStrength

8.177.1 Detailed Description

This structure contains pilot PN Element.

Parameters

ActSetPilotPN	• Active Pilot PN
ActSetPilotPN-Strength	• Active Pilot PN strength

8.177.2 Field Documentation

8.177.2.1 uint16_t nas_ActPilotPNElement::ActSetPilotPN

8.177.2.2 uint8_t nas_ActPilotPNElement::ActSetPilotPNStrength

8.178 nas_AddCDMASysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx
- uint16_t regPrd

8.178.1 Detailed Description

Structure for storing the Additional CDMA System Information.

Parameters

geoSysIdx	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
regPrd	• Registration period after the CDMA system is acquired. • When the CDMA registration period is not valid, 0xFFFF is used.

8.178.2 Field Documentation

8.178.2.1 uint16_t nas_AddCDMASysInfo::geoSysIdx

8.178.2.2 uint16_t nas_AddCDMASysInfo::regPrd

8.179 nas_AddSysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx

- uint32_t cellBroadcastCap

8.179.1 Detailed Description

Structure for storing the Additional GSM and WCDMA System Information.

Parameters

geoSysIdx	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
cellBroadcast-Cap	• Cell broadcast capability of the serving system. • When the CDMA registration period is not valid, 0xFFFF is used. – NAS_CELL_BROADCAST_CAP_UNKNOWN - Cell broadcast support is unknown – NAS_CELL_BROADCAST_CAP_OFF - Cell broadcast is not supported – NAS_CELL_BROADCAST_CAP_ON - Cell broadcast is supported

8.179.2 Field Documentation

8.179.2.1 uint32_t nas_AddSysInfo::cellBroadcastCap

8.179.2.2 uint16_t nas_AddSysInfo::geoSysIdx

8.180 nas_BandPrefInfoTlv Struct Reference

Data Fields

- uint64_t bits_1_64
- uint64_t bits_65_128
- uint64_t bits_129_192
- uint64_t bits_193_256
- uint8_t TlvPresent

8.180.1 Detailed Description

Contain the Band Preference.

Parameters

bits_1_64	• Bits 1 to 64 of the 256-bit Operating Band bitmask.
bits_65_128	• Bits 65 to 128 of the 256-bit Operating Band bitmask.
bits_129_192	• Bits 129 to 192 of the 256-bit Operating Band bitmask.

bits_193_256	• Bits 193 to 256 of the 256-bit Operating Band bitmask.
TlvPresent	• Tlv Present

8.180.2 Field Documentation

8.180.2.1 uint64_t nas_BandPrefInfoTlv::bits_129_192

8.180.2.2 uint64_t nas_BandPrefInfoTlv::bits_193_256

8.180.2.3 uint64_t nas_BandPrefInfoTlv::bits_1_64

8.180.2.4 uint64_t nas_BandPrefInfoTlv::bits_65_128

8.180.2.5 uint8_t nas_BandPrefInfoTlv::TlvPresent

8.181 nas_BandPrefTlv Struct Reference

Data Fields

- uint64_t BandPref
- uint8_t TlvPresent

8.181.1 Detailed Description

Contain the band preference for system selection preferences.

Parameters

BandPref	• Bitmask representing the radio technologies that are disabled. • Bit values: • Bit mask representing the band preference • Bit values: – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved

TlvPresent	• Tlv Present

8.181.2 Field Documentation

8.181.2.1 uint64_t nas_BandPrefTlv::BandPref

8.181.2.2 uint8_t nas_BandPrefTlv::TlvPresent

8.182 nas_CallBarringSysInfo Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.182.1 Detailed Description

Structure for storing the GSM and WCDMA Call Barring System Information.

Parameters

csBarStatus	• Call barring status for circuit-switched calls. – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown
psBarStatus	• Call barring status for packet-switched calls. – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.182.2 Field Documentation

8.182.2.1 uint32_t nas_CallBarringSysInfo::csBarStatus

8.182.2.2 uint32_t nas_CallBarringSysInfo::psBarStatus

8.183 nas_callBarStatus Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.183.1 Detailed Description

This structure contains Call Barring Status.

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

csBarStatus	• Call Barring Status for circuit-switched calls. • Values: • NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only • NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only • NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type • NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types • NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown
psBarStatus	• Call Barring Status for packet-switched calls. • Values: – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.183.2 Field Documentation

8.183.2.1 uint32_t nas_callBarStatus::csBarStatus

8.183.2.2 uint32_t nas_callBarStatus::psBarStatus

8.184 nas_CDMAChannel Struct Reference

Data Fields

- uint16_t priChA
- uint16_t priChB
- uint16_t secChA
- uint16_t secChB

8.184.1 Detailed Description

This structure contains CDMA channel.

Parameters

priChA	• A Channel number for the primary carrier. – 0xFFFF - Not Available
priChB	• B Channel number for the primary carrier. – 0xFFFF - Not Available
secChA	• A Channel number for the secondary carrier. – 0xFFFF - Not Available
secChB	• B Channel number for the secondary carrier. – 0xFFFF - Not Available

8.184.2 Field Documentation

8.184.2.1 uint16_t nas_CDMAChannel::priChA

8.184.2.2 uint16_t nas_CDMAChannel::priChB

8.184.2.3 uint16_t nas_CDMAChannel::secChA

8.184.2.4 uint16_t nas_CDMAChannel::secChB

8.185 nas_CDMAECIOThresh Struct Reference

Data Fields

- uint8_t CDMAECIOThreshListLen
- int16_t * pCDMAECIOThreshList

8.185.1 Detailed Description

This structure contains CDMA ECIO threshold related parameters.

Parameters

CDMAECIO-ThreshListLen	• Length of the CDMA ECIO threshold list parameter to follow
pCDMAECIO-ThreshList	• Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values. Range for ECIO values: -31.5 to 0 (in dB)

8.185.2 Field Documentation

8.185.2.1 uint8_t nas_CDMAECIOThresh::CDMAECIOThreshListLen

8.185.2.2 int16_t* nas_CDMAECIOThresh::pCDMAECIOThreshList

8.186 nas_CDMAInfo Struct Reference

Data Fields

- uint16_t sid
- uint16_t nid
- uint16_t baseId
- uint16_t refpn
- uint32_t baseLat
- uint32_t baseLong

8.186.1 Detailed Description

This structure contains information about the CDMA Network.

Parameters

sid	• System ID. – 0xFFFF - Not Available
nid	• Network ID. – 0xFFFF - Not Available
baseId	• Base station ID. – 0xFFFF - Not Available
refpn	• Reference PN. – 0xFFFF - Not Available
baseLat	• Latitude of the current base station in units of 0.25 sec. – 0xFFFFFFFF - Not Available
baseLong	• Longitude of the current base station in units of 0.25 sec. – 0xFFFFFFFF - Not Available

8.186.2 Field Documentation

8.186.2.1 uint16_t nas_CDMAInfo::baseId

8.186.2.2 uint32_t nas_CDMAInfo::baseLat

8.186.2.3 uint32_t nas_CDMAInfo::baseLong

8.186.2.4 uint16_t nas_CDMAInfo::nid

8.186.2.5 uint16_t nas_CDMAInfo::refpn

8.186.2.6 uint16_t nas_CDMAInfo::sid

8.187 nas_CDMARSSIThresh Struct Reference

Data Fields

- uint8_t CDMARSSIThreshListLen
- int16_t * pCDMARSSIThreshList

8.187.1 Detailed Description

This structure contains CDMA RSSI threshold related parameters.

Parameters

CDMARSSI-ThreshListLen	• Length of the CDMARSSI threshold list parameter to follow
pCDMARSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • maximum of 32 values. • Range for RSSI values:-105 to -21 (in dBm).

8.187.2 Field Documentation

8.187.2.1 uint8_t nas_CDMARSSIThresh::CDMARSSIThreshListLen

8.187.2.2 int16_t* nas_CDMARSSIThresh::pCDMARSSIThreshList

8.188 nas_CDMASysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoCDMA
- uint8_t isSysPrlMatchValid
- uint8_t isSysPrlMatch
- uint8_t pRevInUseValid
- uint8_t pRevInUse
- uint8_t bsPRevValid
- uint8_t bsPRev
- uint8_t ccsSupportedValid
- uint8_t ccsSupported
- uint8_t cdmaSysIdValid
- uint16_t systemID
- uint16_t networkID
- uint8_t bsInfoValid
- uint16_t baseId
- uint32_t baseLat

- uint32_t baseLong
- uint8_t packetZoneValid
- uint16_t packetZone
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]

8.188.1 Detailed Description

Structure for storing the CDMA System Information.

Parameters

sysInfoCDMA	• See nas_sysInfoCommon for more information.
isSysPrlMatch-Valid	• Indicates whether the system PRL match is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
isSysPrlMatch	• Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, roam_status carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, roam_status is set to the value based on the standard specification.
pRevInUseValid	• Indicates whether the P_Rev in use is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
pRevInUse	• The lesser of the base station P_Rev and mobile P_Rev • Only applicable for CDMA. – 0xFF - Not Available
bsPRevValid	• Indicates whether the base station P_Rev is valid – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
bsPRev	• Base station P_Rev. • Only applicable for CDMA. – 0xFF - Not Available

ccsSupported-Valid	• Indicates whether the supported concurrent service is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
ccsSupported	• Whether concurrent service is supported. • Only applicable for CDMA. – 0x00 - Not supported – 0x01 - Supported – 0xFF - Not Available
cdmaSysIdValid	• Indicates whether the CDMA system ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
systemID	• System ID. – 0xFFFF - Not Available
networkID	• Network ID. – 0xFFFF - Not Available
bsInfoValid	• Indicates whether the base station information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
baseLat	• Base station latitude in units of 0.25 sec. • Expressed as a two's complement signed number with positive numbers signifying North latitudes. – 0xFFFFFFFF - Not Available
baseLong	• Base station longitude in units of 0.25 sec. • Expressed as a two's complement signed number with positive numbers signifying East latitudes. – 0xFFFFFFFF - Not Available
packetZoneValid	• Indicates whether the packet zone is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

packetZone	• Packet zone (8-bit). – 0xFFFF indicates no packet zone. • Only applicable for CDMA.
networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
MCC[NAS_PLMN_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters • MCC wildcard value is returned as {'3', 0xFF, 0xFF}.
MNC[NAS_PLMN_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • MNC wildcard value is returned as {'7', 0xFF, 0xFF}.

8.188.2 Field Documentation

8.188.2.1 uint16_t nas_CDMASysInfo::baseId

8.188.2.2 uint32_t nas_CDMASysInfo::baseLat

8.188.2.3 uint32_t nas_CDMASysInfo::baseLong

8.188.2.4 uint8_t nas_CDMASysInfo::bsInfoValid

8.188.2.5 uint8_t nas_CDMASysInfo::bsPRev

8.188.2.6 uint8_t nas_CDMASysInfo::bsPRevValid

8.188.2.7 uint8_t nas_CDMASysInfo::ccsSupported

8.188.2.8 uint8_t nas_CDMASysInfo::ccsSupportedValid

8.188.2.9 uint8_t nas_CDMASysInfo::cdmaSysIdValid

8.188.2.10 uint8_t nas_CDMASysInfo::isSysPrlMatch

8.188.2.11 uint8_t nas_CDMASysInfo::isSysPrlMatchValid

8.188.2.12 uint8_t nas_CDMASysInfo::MCC[3]

8.188.2.13 uint8_t nas_CDMASysInfo::MNC[3]

8.188.2.14 uint16_t nas_CDMASysInfo::networkID

8.188.2.15 uint8_t nas_CDMASysInfo::networkIdValid

8.188.2.16 uint16_t nas_CDMASysInfo::packetZone

8.188.2.17 uint8_t nas_CDMASysInfo::packetZoneValid

8.188.2.18 uint8_t nas_CDMASysInfo::pRevInUse

8.188.2.19 uint8_t nas_CDMASysInfo::pRevInUseValid

8.188.2.20 nas_sysInfoCommon nas_CDMASysInfo::sysInfoCDMA

8.188.2.21 uint16_t nas_CDMASysInfo::systemID

8.189 nas_CDMASysInfoExt Struct Reference

Data Fields

- uint16_t MCC
- uint8_t imsi_11_12

8.189.1 Detailed Description

This structure contains CDMA system information extension

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

MCC	• Mobile Country Code
imsi_11_12	• IMSI_11_12

8.189.2 Field Documentation

8.189.2.1 uint8_t nas_CDMASysInfoExt::imsi_11_12

8.189.2.2 uint16_t nas_CDMASysInfoExt::MCC

8.190 nas_cellParams Struct Reference

Data Fields

- uint16_t pci
- int16_t rsrq
- int16_t rsrp
- int16_t rssi
- int16_t srxlev

8.190.1 Detailed Description

This structure contains information about the Cell parameters.

Parameters

pci	• Physical cell ID. • Range: 0 to 503.
rsrq	• Current RSRQ in 1/10 dB as measured by L1. • Range: -20.0 dB to -3.0 dB.
rsrp	• Current RSRP in 1/10 dBm as measured by L1. • Range: -140.0 dBm to -44.0 dBm.
rssi	• Current RSSI in 1/10 dBm as measured by L1. • Range: -120.0 dBm to 0.
srxlev	• Cell selection Rx level (Srxlev) value. • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.190.2 Field Documentation

8.190.2.1 uint16_t nas_cellParams::pci

8.190.2.2 int16_t nas_cellParams::rsrp

8.190.2.3 int16_t nas_cellParams::rsrq

8.190.2.4 int16_t nas_cellParams::rssi

8.190.2.5 int16_t nas_cellParams::srxlev

8.191 nas_ciotAcqOrderPref Struct Reference

Data Fields

- uint8_t ciotAcqOrderLen
- uint32_t * pCiotAcqOrder

8.191.1 Detailed Description

Contain the CIOT Acquisition Order Preference.

Parameters

ciotAcqOrderLen	• Number of sets of the following elements.
pCiotAcqOrder	• CIOT Acquisition order preference to be set. • Values – 0x01 - cdma2000 ®1X – 0x02 - cdma2000 ® HRPD (1xEV-DO) – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA – 0x0a - LTE_M1 – 0x0b - LTE_NB1

8.191.2 Field Documentation

8.191.2.1 uint8_t nas_ciotAcqOrderPref::ciotAcqOrderLen

8.191.2.2 uint32_t* nas_ciotAcqOrderPref::pCiotAcqOrder

8.192 nas_CiotAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t ciotAcqOrderLen
- uint32_t * pCiotAcqOrder
- uint8_t TlvPresent

8.192.1 Detailed Description

Contain the CIOT Acquisition Order Preference.

Parameters

ciotAcqOrderLen	• Number of sets of the following elements.
pCiotAcqOrder	• CIOT Acquisition order preference to be set. • Values – 0x01 - cdma2000 ®1X – 0x02 - cdma2000 ® HRPD (1xEV-DO) – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA – 0x0a - LTE_M1 – 0x0b - LTE_NB1

TlvPresent	• Tlv Present

8.192.2 Field Documentation

8.192.2.1 uint8_t nas_CiotAcqOrderPrefTlv::ciotAcqOrderLen

8.192.2.2 uint32_t* nas_CiotAcqOrderPrefTlv::pCiotAcqOrder

8.192.2.3 uint8_t nas_CiotAcqOrderPrefTlv::TlvPresent

8.193 nas_CiotLteOpModePrefTlv Struct Reference

Data Fields

- uint64_t ciotLteOpModePref
- uint8_t TlvPresent

8.193.1 Detailed Description

Contain the CIOT LTE Operational Mode Preference.

Parameters

ciotLteOpMode-Pref	• CIOT LTE Operational Mode Preference • Values: – Bit 0 (0x01) - LTE wideband – Bit 1 (0x02) - LTE M1 – Bit 2 (0x04) - LTE NB1 All unlisted bits are reserved for future use and the service point ignores them if used.
TlvPresent	• Tlv Present

8.193.2 Field Documentation

8.193.2.1 uint64_t nas_CiotLteOpModePrefTlv::ciotLteOpModePref

8.193.2.2 uint8_t nas_CiotLteOpModePrefTlv::TlvPresent

8.194 nas_CommInfo Struct Reference

Data Fields

- int8_t temperature
- uint8_t modemMode
- uint8_t systemMode
- uint8_t imsRegState

- uint8_t psState

8.194.1 Detailed Description

Structure for storing the common information for the device.

Parameters

temperature	• Temperature. – 8-bit signed integer – 0xFF - Not Available.
modemMode	• Modem Operating Mode. – 0x00 - POWERING OFF – 0x01 - FACTORY TEST – 0x02 - OFFLINE – 0x03 - OFFLINE_AMPS – 0x04 - OFFLINE_CDMA – 0x05 - ONLINE – 0x06 - LOW POWER MODE – 0x07 - RESETTING – 0x08 - NETWORK TEST – 0x09 - OFFLINE REQUEST – 0x0A - PSEUDO ONLINE – 0x0B - RESETTING MODEM – 0xFF - Unknown
systemMode	• System Acquisition Mode. – 0x00 - No service – 0x01 - AMPS – 0x02 - CDMA – 0x03 - GSM – 0x04 - HDR – 0x05 - WCDMA – 0x06 - GPS – 0x08 - WLAN – 0x09 - LTE – 0xFF - Unknown
imsRegState	• IMS Registration State. – 0x00 - NO SRV – 0x01 - IN PROG – 0x02 - FAILED – 0x03 - LIMITED – 0x04 - FULL SRV – 0xFF - Unknown

psState	• PS Attach State. – 0x00 - Attached – 0x01 - Detached – 0xFF - Unknown

8.194.2 Field Documentation

8.194.2.1 uint8_t nas_CommInfo::imsRegState

8.194.2.2 uint8_t nas_CommInfo::modemMode

8.194.2.3 uint8_t nas_CommInfo::psState

8.194.2.4 uint8_t nas_CommInfo::systemMode

8.194.2.5 int8_t nas_CommInfo::temperature

8.195 nas_CsgId Struct Reference

Data Fields

- uint32_t csgId
- uint8_t TlvPresent

8.195.1 Detailed Description

This structure contains csg ID.

Parameters

csgId	• Closed subscriber group identifier.
TlvPresent	• Tlv Present or not

8.195.2 Field Documentation

8.195.2.1 uint32_t nas_CsgId::csgId

8.195.2.2 uint8_t nas_CsgId::TlvPresent

8.196 nas_CSGID Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc

- uint8_t mncPcsDigits
- uint32_t id
- uint8_t rat

8.196.1 Detailed Description

Contain the CSGID.

Parameters

mcc	• MCC value. Range 0 to 999
mnc	• MNC value. Range 0 to 999
mncPcsDigits	• TRUE - MNC is a three-digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 • FALSE - MNC is a two-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
id	• Closed subscriber group identifier.
rat	• Radio interface technology of the CSG network. Values: – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE – 0x09 - RADIO_IF_TDSCDMA - TDS

8.196.2 Field Documentation

8.196.2.1 uint32_t nas_CSGID::id

8.196.2.2 uint16_t nas_CSGID::mcc

8.196.2.3 uint16_t nas_CSGID::mnc

8.196.2.4 uint8_t nas_CSGID::mncPcsDigits

8.196.2.5 uint8_t nas_CSGID::rat

8.197 nas_CSGInfo Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint32_t csg_list_cat
- uint32_t id
- uint8_t name_len

- uint16_t name [255]

8.197.1 Detailed Description

Contain fields in struct nas_CSGInfo

Parameters

mcc	• A 16-bit integer representation of MCC. Range: 0 to 999.
mnc	• A 16-bit integer representation of MCC. Range: 0 to 999.
csg_list_cat	• Closed subscriber group category. • Values: – 0 - NAS_CSG_LIST_CAT_UNKNOWN – 1 - NAS_CSG_LIST_CAT_ALLOWED – 2 - NAS_CSG_LIST_CAT_OPERATOR
id	• Closed subscriber group identifier.
name_len	• Number of sets of the following elements: – name
name	• Home Node B (HNB) or Home eNode B (HeNB) name in UTF-16.

Note

None

8.197.2 Field Documentation

8.197.2.1 uint32_t nas_CSGInfo::csg_list_cat

8.197.2.2 uint32_t nas_CSGInfo::id

8.197.2.3 uint16_t nas_CSGInfo::mcc

8.197.2.4 uint16_t nas_CSGInfo::mnc

8.197.2.5 uint16_t nas_CSGInfo::name[255]

8.197.2.6 uint8_t nas_CSGInfo::name_len

8.198 nas_CSGInfoTlv Struct Reference

Data Fields

- uint8_t csg_info_len
- nas_CSGInfo csgInfo [255]
- uint8_t TlvPresent

8.198.1 Detailed Description

Contain fields in struct CSGInfoTlv

Parameters

csg_info_len	• Number of sets of the following elements: – mobile_country_code – mobile_network_code – csg_list_cat – id – name_len – name
csgInfo	• See nas_CSGInfo for more information.
TlvPresent	• Tlv Present.

Note

None

8.198.2 Field Documentation

8.198.2.1 uint8_t nas_CSGInfoTlv::csg_info_len

8.198.2.2 nas_CSGInfo nas_CSGInfoTlv::csgInfo[255]

8.198.2.3 uint8_t nas_CSGInfoTlv::TlvPresent

8.199 nas_currentPLMN Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t netDescrLength
- uint8_t netDescr [255]

8.199.1 Detailed Description

This structure contains the current PLMN parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

MCC	• mobile country code – A 16 bit representation of MCC – Range 0 to 999
MNC	• mobile network code – A 16 bit representation of MNC – Range 0 to 999
netDescrLength	• Length of Network description field • Defaults to zero
netDescr	• Network Description – optional string containing network name or description

8.199.2 Field Documentation

8.199.2.1 uint16_t nas_currentPLMN::MCC

8.199.2.2 uint16_t nas_currentPLMN::MNC

8.199.2.3 uint8_t nas_currentPLMN::netDescr[255]

8.199.2.4 uint8_t nas_currentPLMN::netDescrLength

8.200 nas_dataSrvCapabilities Struct Reference

Data Fields

- uint8_t dataCapabilitiesLen
- uint8_t dataCapabilities [32]

8.200.1 Detailed Description

This structure contains the data services capability

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

dataCapabilities-Len	• Length of data capabilities list • Defaults to zero

dataCapabilities	• List of data capabilities • Values: – 0x01 - GPRS – 0x02 - EDGE – 0x03 - HSDPA – 0x04 - HSUPA – 0x05 - WCDMA – 0x06 - CDMA – 0x07 - EV-DO Rev0 – 0x08 - EV-DO RevA – 0x09 - GSM – 0x0A - EV-DO Rev B – 0x0B - LTE – 0x0C - HSDPA+ – 0x0D - DC-HSDPA+

8.200.2 Field Documentation

8.200.2.1 uint8_t nas_dataSrvCapabilities::dataCapabilities[32]

8.200.2.2 uint8_t nas_dataSrvCapabilities::dataCapabilitiesLen

8.201 nas_DataStatusDetail Struct Reference

Data Fields

- uint32_t IPAddress
- uint8_t LastErrCode

8.201.1 Detailed Description

This structure contains data status detail

Parameters

IPAddress	• IP Address • 0xABCDEFGH - AB.CD.EF.GH • Example: – 0x12345678 - 18.52.86.120 0x12=18 0x34=52 0x56=86 0x78=120 • 0xFFFFFFFF - NA

LastErrCode	• MIP Error code – 0x00 - MIP_RRP_CODE_SUCCESS – 0x01 - MIP_RRP_CODE_SUCCESS_NO_SIM_BINDINGS – 0x40 - MIP_RRP_CODE_FAILURE_FA_REASON_UNSPECIFIED – 0x41 - MIP_RRP_CODE_FAILURE_FA_ADMIN_PROHIBITED – 0x42 - MIP_RRP_CODE_FAILURE_FA_INSUFFICIENT_RESOURCES – 0x43 - MIP_RRP_CODE_FAILURE_FA_MOBILE_NODE_FAILED_AUTH – 0x44 - MIP_RRP_CODE_FAILURE_FA_HA_FAILED_AUTH – 0x45 - MIP_RRP_CODE_FAILURE_FA_REQUESTED_LIFETIME_TOO_LONG – 0x46 - MIP_RRP_CODE_FAILURE_FA_MALFORMED_REQUEST – 0x47 - MIP_RRP_CODE_FAILURE_FA_MALFORMED_REPLY – 0x48 - MIP_RRP_CODE_FAILURE_FA_ENCAPSULATION_UNAVAILABLE – 0x49 - MIP_RRP_CODE_FAILURE_FA_VJHC_UNAVAILABLE – 0x4A - MIP_RRP_CODE_FAILURE_FA_CANT_REV_TUN – 0x4B - MIP_RRP_CODE_FAILURE_FA_MUST_REV_TUN – 0x4C - MIP_RRP_CODE_FAILURE_FA_BAD_TTL – 0x4D - MIP_RRP_CODE_FAILURE_INVALID_COA – 0x4F - MIP_RRP_CODE_FAILURE_FA_DELIVERY_STYLE_NOT_SUPPORTED – 0x59 - MIP_RRP_CODE_FAILURE_FA_VS_REASON – 0x61 - MIP_RRP_CODE_FAILURE_MISSING_NAI – 0x62 - MIP_RRP_CODE_FAILURE_MISSING_HA_ADDR – 0x63 - MIP_RRP_CODE_FAILURE_MISSING_HOMEADDR – 0x68 - MIP_RRP_CODE_FAILURE_UNKNOWN_CHALLENGE – 0x69 - MIP_RRP_CODE_FAILURE_MISSING_CHALLENGE – 0x6A - MIP_RRP_CODE_FAILURE_STALE_CHALLENGE – 0x6B - MIP_RRP_CODE_FAILURE_MISSING_MN_FA – 0x80 - MIP_RRP_CODE_FAILURE_HA_REASON_UNSPECIFIED – 0x81 - MIP_RRP_CODE_FAILURE_HA_ADMIN_PROHIBITED – 0x82 - MIP_RRP_CODE_FAILURE_HA_INSUFFICIENT_RESOURCES – 0x83 - MIP_RRP_CODE_FAILURE_HA_MOBILE_NODE_FAILED_AUTH – 0x84 - MIP_RRP_CODE_FAILURE_HA_FA_FAILED_AUTH – 0x85 - MIP_RRP_CODE_FAILURE_HA_REG_ID_MISMATCH – 0x86 - MIP_RRP_CODE_FAILURE_HA_MALFORMED_REQUEST – 0x88 - MIP_RRP_CODE_FAILURE_UNKNOWN_HA – 0x89 - MIP_RRP_CODE_FAILURE_HA_CANT_REV_TUN – 0x8A - MIP_RRP_CODE_FAILURE_HA_MUST_REV_TUN – 0x8B - MIP_RRP_CODE_FAILURE_HA_ENCAPSULATION_UNAVAILABLE – 0x8F - MIP_RRP_CODE_FAILURE_REDIRECTED_HA – 0x90 - MIP_RRP_CODE_FAILURE_HA_BAD_AAA_AUTH – 0xFF - NA

8.201.2 Field Documentation

8.201.2.1 uint32_t nas_DataStatusDetail::IPAddress

8.201.2.2 uint8_t nas_DataStatusDetail::LastErrCode

8.202 nas_DDTMSettings Struct Reference

Data Fields

- uint8_t CurrentDDTMStatus
- uint8_t DDTMPrefSetting
- uint16_t DDTMAction
- uint8_t SOListAction
- uint8_t SOLen
- uint16_t ServiceOption [255]

8.202.1 Detailed Description

This structure contains the DDTM Settings 0x01 tlv parameters.

Parameters

CurrentDDTM-Status[Out]	• Current DDTM Status, • 0x00 - CURRENT_DDTM_STATUS_DISABLED • 0x01 - CURRENT_DDTM_STATUS_ENABLED
DDTMPref-Setting[Out]	• DDTM preference. The value should be one of following, • 0x00 Disable DDTM • 0x01 Enable DDTM • 0x02 Do not change DDTM preference
DDTMAction[-Out]	• DDTM action Bitmask of following represents what combined DDTM actions take place. To enable all masks, a value of 0x3FFF must be sent in this field. • Bit 0 - QMI_NAS_DDTM_ACTION_SUPPRESS_L2ACK_BIT - Do not send L2 ACK on 1X • Bit 1 - QMI_NAS_DDTM_ACTION_SUPPRESS_REG_BIT - Suppress 1X registrations • Bit 2 - QMI_NAS_DDTM_ACTION_IGNORE_SO_PAGES_BIT - Ignore 1X pages with specified service options • Bit 3 - QMI_NAS_DDTM_ACTION_SUPPRESS_MO_DBM_BIT - Block MO SMS and DBM
SOListAction[-Out]	• Action to be taken with the specified SO list in the SO field. Value should be one of following, • 0x00 Add the specified SOs to the current DDTM SO list • 0x01 Replace the current DDTM SO list • 0x02 Delete the specified SOs from the DDTM SO list • 0x03 No change in the DDTM SO list
SOLen[Out]	• Number of sets of the following elements: ServiceOption
ServiceOption[-Out]	• Service option for which SO pages are ignored when DDTM status is ON. To ignore all SO pages, a value of 0xFFFF must be specified.

8.202.2 Field Documentation

8.202.2.1 uint8_t nas_DDTMSettings::CurrentDDTMStatus

8.202.2.2 uint16_t nas_DDTMSettings::DDTMAction

8.202.2.3 uint8_t nas_DDTMSettings::DDTMPrefSetting

8.202.2.4 uint16_t nas_DDTMSettings::ServiceOption[255]

8.202.2.5 uint8_t nas_DDTMSettings::SOLen

8.202.2.6 uint8_t nas_DDTMSettings::SOListAction

8.203 nas_detailSvcInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t srvCapability
- uint8_t hdrSrvStatus
- uint8_t hdrHybrid
- uint8_t isSysForbidden

8.203.1 Detailed Description

This structure contains Detailed Service information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

srvStatus	• Service status • Values: – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep
srvCapability	• System's service capability • Values: – 0x00 - No Service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - MS found the right system but not yet registered/attached

hdrSrvStatus	• HDR service status • Values: – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep
hdrHybrid	• HDR hybrid information • Values: – 0x00 - System is not hybrid – 0x01 - System is hybrid
isSysForbidden	• Forbidden system information • Values: – 0x00 - System is not a forbidden system – 0x01 - System is a forbidden system

8.203.2 Field Documentation

8.203.2.1 uint8_t nas_detailSvcInfo::hdrHybrid

8.203.2.2 uint8_t nas_detailSvcInfo::hdrSrvStatus

8.203.2.3 uint8_t nas_detailSvcInfo::isSysForbidden

8.203.2.4 uint8_t nas_detailSvcInfo::srvCapability

8.203.2.5 uint8_t nas_detailSvcInfo::srvStatus

8.204 nas_DeviceConfigDetail Struct Reference

Data Fields

- uint8_t Technology
- uint8_t QLIC
- uint8_t Chipset
- uint8_t HWVersion

8.204.1 Detailed Description

This structure contains device configure detail.

Parameters

Technology	• Current technology in use – 0x00 - 1x RTT – 0x01 - EVDO Rev 0 – 0x02 - EVDO Rev A – 0xFF - NA
QLIC	• Quasi Linear Interference Cancellation – 0x00 - Not supported – 0x01 - Supported
Chipset	• Qualcomm platform – 0x4E - MDM6200 – 0x4F - MDM6600 – 0xFF - NA
HWVersion	• Hardware version – 0x00 - BSHWREV0 – 0x01 - BSHWREV1 – 0x02 - BSHWREV2 – 0x03 - BSHWREV3 – 0x04 - BSHWREVMAX – 0xFF - BSHWREVUNKNOWN

8.204.2 Field Documentation

8.204.2.1 uint8_t nas_DeviceConfigDetail::Chipset

8.204.2.2 uint8_t nas_DeviceConfigDetail::HWVersion

8.204.2.3 uint8_t nas_DeviceConfigDetail::QLIC

8.204.2.4 uint8_t nas_DeviceConfigDetail::Technology

8.205 nas_dirNum Struct Reference

Data Fields

- uint8_t dirNumLen
- uint8_t dirNum [255]

8.205.1 Detailed Description

This structure contains the parameters for Directory Number Information

Parameters

dirNumLen	• Number of sets of the following elements: – dir_num • If zero(0), then no information follows.
dirNum	• Directory number in ASCII characters.

8.205.2 Field Documentation

8.205.2.1 uint8_t nas_dirNum::dirNum[255]

8.205.2.2 uint8_t nas_dirNum::dirNumLen

8.206 nas_DRCParams Struct Reference

Data Fields

- uint8_t DRCValue
- uint8_t DRCCover

8.206.1 Detailed Description

This structure contains DRC information.

Parameters

DRCValue	• Current Data Rate Channel
DRCCover	• Current Data Rate Channel cover

8.206.2 Field Documentation

8.206.2.1 uint8_t nas_DRCParams::DRCCover

8.206.2.2 uint8_t nas_DRCParams::DRCValue

8.207 nas_ecioListElement Struct Reference

Data Fields

- int16_t ecio
- uint8_t radioIf

8.207.1 Detailed Description

This structure contains the ECIO Information

Parameters

ecio	• ECIO value in dBm
radioIf	• Radio interface technology of the signal being measured – 0x00 - RADIO_IF_NO_SVC - None (no service) – 0x01 - RADIO_IF_CDMA_1X - cdma2000® 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS

8.207.2 Field Documentation

8.207.2.1 int16_t nas_ecioListElement::ecio

8.207.2.2 uint8_t nas_ecioListElement::radioIf

8.208 nas_ECIOThresh Struct Reference

Data Fields

- uint8_t ECIOThresListLen
- int16_t * pECIOThresList

8.208.1 Detailed Description

This sturcture contain ECIO threshold list.

Parameters

ECIOThresList-Len	• Length of the ECIO threshold list parameter to follow
pECIOThresList	• Sequence of thresholds delimiting ECIO event reporting bands • Every time a new ECIO value crosses a threshold value, an event report indication message with the new ECIO value is sent to the requesting control point. For this field – Each ECIO threshold value is a signed 2 byte value – Each ECIO threshold value increments in negative 0.5 dB, e.g., an ECIO threshold value of 2 means -1dB – Maximum number of threshold values is 16 – At least one value must be specified – Threshold values specified above are used for all RATs

8.208.2 Field Documentation

8.208.2.1 uint8_t nas_ECIOThresh::ECIOThresListLen

8.208.2.2 int16_t* nas_ECIOThresh::pECIOThresList

8.209 nas_EdrxCiotLteMode Struct Reference

Data Fields

- uint32_t IteOpMode
- uint8_t TlvPresent

8.209.1 Detailed Description

This structure contains the parameters for LTE Operational Mode Indicator Type.

Parameters

IteOpMode	• CIOT LTE mode. • Values – NAS_CIOT_SYS_MODE_NO_SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_NB1 (0x03) - Camped on LTE NB1
TlvPresent	• Tlv Present.

8.209.2 Field Documentation

8.209.2.1 uint32_t nas_EdrxCiotLteMode::IteOpMode

8.209.2.2 uint8_t nas_EdrxCiotLteMode::TlvPresent

8.210 nas_EdrxCycleLength Struct Reference

Data Fields

- uint8_t cycleLength
- uint8_t TlvPresent

8.210.1 Detailed Description

This structure contains the parameters for eDRX Cycle Length Indicator Type.

Parameters

cycleLength	• eDRX cycle length. Value is per 3GPP TS 24.008 Rel-13 Section 10.5.5.32.
TlvPresent	• Tlv Present.

8.210.2 Field Documentation

8.210.2.1 uint8_t nas_EdrxCycleLength::cycleLength

8.210.2.2 uint8_t nas_EdrxCycleLength::TlvPresent

8.211 nas_EdrxEnableType Struct Reference

Data Fields

- uint8_t edrxEnabled
- uint8_t TlvPresent

8.211.1 Detailed Description

This structure contains the parameters for eDRX Enabled Indicator Type.

Parameters

edrxEnabled	• Values – 0x00 - Disabled – 0x01 - Enabled
TlvPresent	• Tlv Present.

8.211.2 Field Documentation

8.211.2.1 uint8_t nas_EdrxEnableType::edrxEnabled

8.211.2.2 uint8_t nas_EdrxEnableType::TlvPresent

8.212 nas_EdrxPagingTimeWindow Struct Reference

Data Fields

- uint8_t edrxPtw
- uint8_t TlvPresent

8.212.1 Detailed Description

This structure contains the parameters for eDRX Paging Time Window Indicator Type.

Parameters

edrxPtw	• eDRX paging time window. Value is per 3GPP TS 24.008 Rel-13 Section 10.5.5.32
TlvPresent	• Tlv Present.

8.212.2 Field Documentation

8.212.2.1 uint8_t nas_EdrxPagingTimeWindow::edrxPtw

8.212.2.2 uint8_t nas_EdrxPagingTimeWindow::TlvPresent

8.213 nas_EdrxRatType Struct Reference

Data Fields

- uint8_t edrxRatType
- uint8_t TlvPresent

8.213.1 Detailed Description

This structure contains the parameters for eDRX RAT type Indicator Type.

Parameters

edrxRatType	• Radio access technology • Values – NAS_RADIO_IF_NO_SVC (0x00) - None (no service) – NAS_RADIO_IF_CDMA_1X (0x01) - cdma2000 ® 1X – NAS_RADIO_IF_CDMA_1XEVDO (0x02) - cdma2000 ® HRPD (1xEV-DO) – NAS_RADIO_IF_AMPS (0x03) - AMPS – NAS_RADIO_IF_GSM (0x04) - GSM – NAS_RADIO_IF_UMTS (0x05) - UMTS – NAS_RADIO_IF_WLAN (0x06) - WLAN – NAS_RADIO_IF_GPS (0x07) - GPS – NAS_RADIO_IF_LTE (0x08) - LTE – NAS_RADIO_IF_TDSCDMA (0x09) - TD-SCDMA – NAS_RADIO_IF_LTE_M1 (0x0a) - LTE-M1 – NAS_RADIO_IF_LTE_NB1 (0x0b) - LTE-NB1 – NAS_RADIO_IF_NO_CHANGE (-1) - No change
TlvPresent	• Tlv Present.

8.213.2 Field Documentation

8.213.2.1 uint8_t nas_EdrxRatType::edrxRatType

8.213.2.2 uint8_t nas_EdrxRatType::TlvPresent

8.214 nas_EmerModeTlv Struct Reference

Data Fields

- uint8_t EmerMode
- uint8_t TlvPresent

8.214.1 Detailed Description

Contain the emergency Mode parameters for system selection preferences.

Parameters

EmerMode	• Emergency Mode • Values: – 0 - OFF (normal) – 1 - ON (Emergency)
TlvPresent	• Tlv Present

8.214.2 Field Documentation

8.214.2.1 uint8_t nas_EmerModeTlv::EmerMode

8.214.2.2 uint8_t nas_EmerModeTlv::TlvPresent

8.215 nas_errorRateListElement Struct Reference

Data Fields

- uint16_t errorRate
- uint8_t radioIf

8.215.1 Detailed Description

This structure contains the Error Rate Information

Parameters

errorRate	• Error rate value corresponds to the RAT that is currently registered. – For CDMA, the error rate reported is Frame Error Rate: * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For HDR, the error rate reported is Packet Error Rate: * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For GSM, the error rate reported is Bit Error Rate: * Valid values are 0, 100, 200, 300, 400, 500, 600, and 700 The reported value divided by 100 gives the error rate as an RxQual value, e.g.,a value of 300 represents an RxQual value of 3. * A value of 25500 indicates No Data – For WCDMA, the error rate reported is Block Error Rate (BLER): * Valid values are 1 to 10000 * The reported value divided by 100 provides the error rate in percentages, e.g., a value of 300 represents a BLER of 3%. * A value of 0 indicates No Data
radioIf	• Radio interface technology of the signal being measured – 0x00 - RADIO_IF_NO_SVC - None (no service) – 0x01 - RADIO_IF_CDMA_1X - cdma2000® 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS

8.215.2 Field Documentation

8.215.2.1 uint16_t nas_errorRateListElement::errorRate

8.215.2.2 uint8_t nas_errorRateListElement::radioIf

8.216 nas_ForbiddenNetworks3GPP Struct Reference

Data Fields

- uint16_t forbiddenNwInstLen
- uint16_t MCC [255]
- uint16_t MNC [255]
- uint8_t TlvPresent

8.216.1 Detailed Description

This structure contains the ForbiddenNetworks3GPP response parameters.

Parameters

forbiddenNwInst-Len	• Number of sets of the following elements – MCC – MNC
MCC	• Mobile Country Code • A 16-bit integer representation of MCC • Range: 0 to 999
MNC	• Mobile Network Code • A 16-bit integer representation of MNC • Range: 0 to 999
TlvPresent	• Tlv Present

8.216.2 Field Documentation

8.216.2.1 uint16_t nas_ForbiddenNetworks3GPP::forbiddenNwInstLen

8.216.2.2 uint16_t nas_ForbiddenNetworks3GPP::MCC[255]

8.216.2.3 uint16_t nas_ForbiddenNetworks3GPP::MNC[255]

8.216.2.4 uint8_t nas_ForbiddenNetworks3GPP::TlvPresent

8.217 nas_GERANInfo Struct Reference

Data Fields

- uint32_t cellID
- uint8_t plmn [3]
- uint16_t lac
- uint16_t arfcn
- uint8_t bsic
- uint32_t timingAdvance
- uint16_t rxLev
- uint8_t nmrInst
- nas_nmrCellInfo insNmrCellInfo [255]

8.217.1 Detailed Description

This structure contains information about the GERAN Network.

Parameters

cellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.

plmn[NAS_PLMN_LENGTH]	• MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when nmrCellID is not present.
lac	• Location area code. • This field is ignored when nmrCellID is not present. – 0xFFFF - Not Available
arfcn	• Absolute RF channel number. – 0xFFFF - Not Available
bsic	• Base station identity code. – 0xFF - Not Available
timingAdvance	• Measured delay (in bit periods; 1 bit period = 48/13 microsecond) of access burst transmission on RACH or PRACH to the expected signal from an MS at zero distance under static channel conditions. – 0xFFFFFFFF - Not Available
rxLev	• Serving Cell Rx measurement. • Values range between 0 and 63. • Mapped to a measured signal level: – Rxlev 0 is a signal strength less than -110 dBm – Rxlev 1 is -110 dBm to -109 dBm – Rxlev 2 is -109 dBm to -108 dBm – ... – Rxlev 62 is -49 dBm to -48 dBm – Rxlev 63 is greater than -48 dBm – 0xFFFF - Not Available
nmrInst	• Provides the number of set of instances which follow. • If 0(zero), then no information follows it.
insNmrCellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_nmrCellInfo for more information.

8.217.2 Field Documentation

8.217.2.1 uint16_t nas_GERANInfo::arfcn

8.217.2.2 uint8_t nas_GERANInfo::bsic

8.217.2.3 uint32_t nas_GERANInfo::cellID

8.217.2.4 nas_nmrCellInfo nas_GERANInfo::insNmrCellInfo[255]

8.217.2.5 uint16_t nas_GERANInfo::lac

8.217.2.6 uint8_t nas_GERANInfo::nmrInst

8.217.2.7 uint8_t nas_GERANInfo::plmn[3]

8.217.2.8 uint16_t nas_GERANInfo::rxLev

8.217.2.9 uint32_t nas_GERANInfo::timingAdvance

8.218 nas_geranInstArr Struct Reference

Data Fields

- uint16_t geranArfcn
- uint8_t geranBsicNcc
- uint8_t geranBsicBcc
- int16_t geranRssi
- int16_t geranRank

8.218.1 Detailed Description

This structure contains information about the GERAN Info.

Parameters

geranArfcn	• Absolute RF channel number.
geranBsicNcc	• Base station identity code network color code • 0xFF indicates information is not present.
geranBsicBcc	• Base station identity code base station color code • 0xFF indicates information is not present.
geranRssi	• Received signal strength indicator.
geranRank	• Rank of the cell.

8.218.2 Field Documentation

8.218.2.1 uint16_t nas_geranInstArr::geranArfcn

8.218.2.2 uint8_t nas_geranInstArr::geranBsicBcc

8.218.2.3 uint8_t nas_geranInstArr::geranBsicNcc

8.218.2.4 int16_t nas_geranInstArr::geranRank

8.218.2.5 int16_t nas_geranInstArr::geranRssi

8.219 nas_geranInstInfo Struct Reference

Data Fields

- uint16_t geranArfcn
- uint8_t geranBsicNcc
- uint8_t geranBsicBcc
- int16_t geranRssi

8.219.1 Detailed Description

This structure contains information about the GERAN Instances in UMTS Network.

Parameters

geranArfcn	• Absolute RF channel number.
geranBsicNcc	• Base station identity code network color code. • 0xFF indicates information is not present.
geranBsicBcc	• Base station identity code base station color code. • 0xFF indicates information is not present.
geranRssi	• Received signal strength indicator.

8.219.2 Field Documentation

8.219.2.1 uint16_t nas_geranInstInfo::geranArfcn

8.219.2.2 uint8_t nas_geranInstInfo::geranBsicBcc

8.219.2.3 uint8_t nas_geranInstInfo::geranBsicNcc

8.219.2.4 int16_t nas_geranInstInfo::geranRssi

8.220 nas_gsmCellInfo Struct Reference

Data Fields

- uint16_t arfcn
- uint8_t band1900
- uint8_t cellIdValid
- uint8_t bsicId
- int16_t rssi

- int16_t srxlev

8.220.1 Detailed Description

This structure contains information about the GSM Cell.

Parameters

arfcn	• GSM frequency being reported. • Range: 0 to 1023.
band1900	• Band indicator for the GSM ARFCN • This field is only valid if arfcn is in the overlapping region. • If TRUE and the cell is in the overlapping region, the ARFCN is on the 1900 band. • If FALSE, it is on the 1800 band.
cellIdValid	• Flag indicating whether the base station identity code ID is valid.
bsicId	• Base station identity code ID, including base station color code and network color code. • The lower 6 bits can be set to any value.
rssi	• Measured RSSI value in 1/10 dB. • Range: -200.0 dB to 0
srxlev	• Cell selection Rx level (Srxlev) value. • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.220.2 Field Documentation

8.220.2.1 uint16_t nas_gsmCellInfo::arfcn

8.220.2.2 uint8_t nas_gsmCellInfo::band1900

8.220.2.3 uint8_t nas_gsmCellInfo::bsicId

8.220.2.4 uint8_t nas_gsmCellInfo::cellIdValid

8.220.2.5 int16_t nas_gsmCellInfo::rssi

8.220.2.6 int16_t nas_gsmCellInfo::srxlev

8.221 nas_GSMRSSIThresh Struct Reference

Data Fields

- uint8_t GSMRSSIThreshListLen
- int16_t * pGSMRSSIThreshList

8.221.1 Detailed Description

This structure contains GSM RSSI threshold related parameters.

Parameters

GSMRSSI-ThreshListLen	• Length of the GSM RSSI threshold list parameter to follow
pGSMRSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSSI values: -111 to -48 (in dBm)

8.221.2 Field Documentation

8.221.2.1 uint8_t nas_GSMRSSIThresh::GSMRSSIThreshListLen

8.221.2.2 int16_t* nas_GSMRSSIThresh::pGSMRSSIThreshList

8.222 nas_GSMSrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t trueSrvStatus
- uint8_t isPrefDataPath

8.222.1 Detailed Description

Structure for storing the service status information for GSM, WCDMA and LTE networks.

Parameters

srvStatus	• Service status of the system. – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available

trueSrvStatus	• True service status of the system. • Not applicable to CDMA/HDR. – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
isPrefDataPath	• Whether the RAT is the preferred data path. – 0x00 - Not preferred – 0x01 - Preferred – 0xFF - Not Available

8.222.2 Field Documentation

8.222.2.1 uint8_t nas_GSMSrvStatusInfo::isPrefDataPath

8.222.2.2 uint8_t nas_GSMSrvStatusInfo::srvStatus

8.222.2.3 uint8_t nas_GSMSrvStatusInfo::trueSrvStatus

8.223 nas_GSMSysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoGSM
- uint8_t lacValid
- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejCause
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]
- uint8_t egprsSuppValid
- uint8_t egprsSupp
- uint8_t dtmSuppValid
- uint8_t dtmSupp

8.223.1 Detailed Description

Structure for storing the GSM System Information.

Parameters

sysInfoGSM	• See nas_sysInfoCommon for more information.
lacValid	• Indicates whether the location area code is valid.. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
lac	• Location area code. • Only applies to 3GPP. – 0xFFFF - Not Available
cellIdValid	• Indicates whether the cell ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
cellId	• Cell ID. – 0xFFFFFFFF - Not Available
regRejectInfo-Valid	• Indicates whether the registration reject information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
rejectSrvDomain	• Type of service domain in which the registration is rejected. – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
rejCause	• Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6 and 10.5.5.14] – 0xFF - Not Available
networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

MCC[NAS_PLMN_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters
MNC[NAS_PLMN_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
egprsSuppValid	• Indicates whether the EGPRS support is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
egprsSupp	• EGPRS support indication. • Only applicable for GSM. – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available
dtmSuppValid	• Indicates whether Dual Transfer mode support is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
dtmSupp	• Dual Transfer mode support indication. • Only applicable for GSM. – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available

8.223.2 Field Documentation

8.223.2.1 uint32_t nas_GSMSysInfo::cellId

8.223.2.2 uint8_t nas_GSMSysInfo::cellIdValid

8.223.2.3 uint8_t nas_GSMSysInfo::dtmSupp

8.223.2.4 uint8_t nas_GSMSysInfo::dtmSuppValid

8.223.2.5 uint8_t nas_GSMSysInfo::egprsSupp

8.223.2.6 uint8_t nas_GSMSysInfo::egprsSuppValid

8.223.2.7 uint16_t nas_GSMSysInfo::lac

8.223.2.8 uint8_t nas_GSMSysInfo::lacValid

8.223.2.9 uint8_t nas_GSMSysInfo::MCC[3]

8.223.2.10 uint8_t nas_GSMSysInfo::MNC[3]

8.223.2.11 uint8_t nas_GSMSysInfo::networkIdValid

8.223.2.12 uint8_t nas_GSMSysInfo::regRejectInfoValid

8.223.2.13 uint8_t nas_GSMSysInfo::rejCause

8.223.2.14 uint8_t nas_GSMSysInfo::rejectSrvDomain

8.223.2.15 nas_sysInfoCommon nas_GSMSysInfo::sysInfoGSM

8.224 nas_GWAcqOrderPrefTlv Struct Reference

Data Fields

- uint32_t GWAcqOrderPref
- uint8_t TlvPresent

8.224.1 Detailed Description

Contain the GSM/WCDMA Acquisition order preference for system selection preferences.

Parameters

GWAcqOrder-Pref	• Parameter indicating GSM/WCDMA Acquisition order Preference • Values: – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM
TlvPresent	• Tlv Present

8.224.2 Field Documentation

8.224.2.1 uint32_t nas_GWAcqOrderPrefTlv::GWAcqOrderPref

8.224.2.2 uint8_t nas_GWAcqOrderPrefTlv::TlvPresent

8.225 nas_HDRECIOThresh Struct Reference

Data Fields

- uint8_t HDRECIOThreshListLen

- int16_t * pHDRECIOThreshList

8.225.1 Detailed Description

This structure contains HDR ECIO threshold related parameters.

Parameters

HDRECIO-ThreshListLen	• Length of the HDR ECIO threshold list parameter to follow
pHDRECIO-ThreshList	• Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for ECIO values: -31.5 to 0 (in dB).

8.225.2 Field Documentation

8.225.2.1 uint8_t nas_HDRECIOThresh::HDRECIOThreshListLen

8.225.2.2 int16_t* nas_HDRECIOThresh::pHDRECIOThreshList

8.226 nas_HDRIOThresh Struct Reference

Data Fields

- uint8_t HDRIOThreshListLen
- int16_t * pHDRIOThreshList

8.226.1 Detailed Description

This structure contains HDR IO threshold related parameters.

Parameters

HDRIOThresh-ListLen	• Length of the HDR IO threshold list parameter to follow
pHDRIOThresh-List	• Array of IO thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for IO values: -128 to -13 (in dBm).

8.226.2 Field Documentation

8.226.2.1 uint8_t nas_HDRIOThresh::HDRIOThreshListLen

8.226.2.2 int16_t* nas_HDRIOThresh::pHDRIOThreshList

8.227 nas_HDRPersonality_Ind_Data Struct Reference

Data Fields

- uint16_t * pCurrentPersonality
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtocolSubtypeElement

8.227.1 Detailed Description

Elements for HDR Personality indication.

Parameters

pCurrent-Personality	• Current active personality index. • Bit to check in ParamPresenceMask - **16**
pPersonalityList-Length	• Number of Personality Protocol Subtype contains in this response. • Bit to check in ParamPresenceMask - **17**
pProtocol-SubtypeElement	• See nas_protocolSubtypeElement for more information. • Bit to check in ParamPresenceMask - **17**

8.227.2 Field Documentation

8.227.2.1 uint16_t* nas_HDRPersonality_Ind_Data::pCurrentPersonality

8.227.2.2 uint8_t* nas_HDRPersonality_Ind_Data::pPersonalityListLength

8.227.2.3 nas_protocolSubtypeElement* nas_HDRPersonality_Ind_Data::pProtocolSubtypeElement

8.228 nas_HDRRSSIThresh Struct Reference

Data Fields

- uint8_t HDRRSSIThreshListLen
- int16_t * pHDRRSSIThreshList

8.228.1 Detailed Description

This structure contains HDR RSSI threshold related parameters.

Parameters

HDRRSSI-ThreshListLen	• Length of the HDR RSSI threshold list parameter to follow

pHDRRSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -118 to -13 (in dBm).

8.228.2 Field Documentation

8.228.2.1 uint8_t nas_HDRRSSIThresh::HDRRSSIThreshListLen

8.228.2.2 int16_t* nas_HDRRSSIThresh::pHDRRSSIThreshList

8.229 nas_HDRSINRThresh Struct Reference

Data Fields

- uint8_t HDRSINRThresListLen
- uint8_t * pHDRSINRThresList

8.229.1 Detailed Description

This structure contains HDR SINR threshold list.

Parameters

HDRSINRThres-ListLen	• Length of the HDR SINR threshold list parameter to follow
pHDRSINR-ThresList	• Sequence of thresholds delimiting SINR event reporting bands • Every time a new SINR value crosses a threshold value, an event report indication message with the new SINR value is sent to the requesting control point. For this field – SINR is reported only for HDR – Each SINR threshold value is an unsigned 1 byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.229.2 Field Documentation

8.229.2.1 uint8_t nas_HDRSINRThresh::HDRSINRThresListLen

8.229.2.2 uint8_t* nas_HDRSINRThresh::pHDRSINRThresList

8.230 nas_HDRSINRThreshold Struct Reference

Data Fields

- uint8_t HDRSINRThreshListLen
- uint16_t * pHDRSINRThreshList

8.230.1 Detailed Description

This structure contains HDR SINR threshold related parameters.

Parameters

HDRSINR-ThreshListLen	• Length of the HDR ECIO threshold list parameter to follow
pHDRSINR-ThreshList	• Array of SINR level thresholds (in units of 1) • maximum of 32 values. • Valid levels are 0 to 8 – 0x00 - SINR_LEVEL_0 is -9 dB – 0x01 - SINR_LEVEL_1 is -6 dB – 0x02 - SINR_LEVEL_2 is -4.5 dB – 0x03 - SINR_LEVEL_3 is -3 dB – 0x04 - SINR_LEVEL_4 is -2 dB – 0x05 - SINR_LEVEL_5 is +1 dB – 0x06 - SINR_LEVEL_6 is +3 dB – 0x07 - SINR_LEVEL_7 is +6 dB – 0x08 - SINR_LEVEL_8 is +9 dB

8.230.2 Field Documentation

8.230.2.1 uint8_t nas_HDRSINRThreshold::HDRSINRThreshListLen

8.230.2.2 uint16_t* nas_HDRSINRThreshold::pHDRSINRThreshList

8.231 nas_HDRSysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoHDR
- uint8_t isSysPrlMatchValid
- uint8_t isSysPrlMatch
- uint8_t hdrPersonalityValid
- uint8_t hdrPersonality
- uint8_t hdrActiveProtValid
- uint8_t hdrActiveProt
- uint8_t is856SysIdValid
- uint8_t is856SysId [16]

8.231.1 Detailed Description

Structure for storing the HDR System Information.

Parameters

sysInfoHDR	• See nas_sysInfoCommon for more information.

isSysPrlMatch-Valid	• Indicates whether the system PRL match is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
isSysPrlMatch	• Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, roam_status carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, roam_status is set to the value based on the standard specification.
hdrPersonality-Valid	• Indicates whether the HDR personality is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
hdrPersonality	• HDR personality information. • Only applicable for HDR. – 0x00 - None – 0x02 - HRPD – 0x03 - eHRPD – 0xFF - Not Available
hdrActiveProt-Valid	• Indicates whether the HDR active protocol revision information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
hdrActiveProt	• HDR active protocol revision information . • Only applicable for HDR. – 0x00 - None – 0x02 - HDR Rel 0 – 0x03 - HDR Rel A – 0x04 - HDR Rel B – 0xFF - Not Available
is856SysIdValid	• Indicates whether the IS-856 system ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

is856SysId[SLQS_SYSTEM_ID_SIZE]	• IS-856 system ID. • Only applicable for HDR.

8.231.2 Field Documentation

8.231.2.1 uint8_t nas_HDRSysInfo::hdrActiveProt

8.231.2.2 uint8_t nas_HDRSysInfo::hdrActiveProtValid

8.231.2.3 uint8_t nas_HDRSysInfo::hdrPersonality

8.231.2.4 uint8_t nas_HDRSysInfo::hdrPersonalityValid

8.231.2.5 uint8_t nas_HDRSysInfo::is856SysId[16]

8.231.2.6 uint8_t nas_HDRSysInfo::is856SysIdValid

8.231.2.7 uint8_t nas_HDRSysInfo::isSysPrlMatch

8.231.2.8 uint8_t nas_HDRSysInfo::isSysPrlMatchValid

8.231.2.9 nas_sysInfoCommon nas_HDRSysInfo::sysInfoHDR

8.232 nas_homeNwMNC3GppTlv Struct Reference

Data Fields

- uint8_t is3GppNw
- uint8_t mccIncPcsDigit
- uint8_t TlvPresent

8.232.1 Detailed Description

This structure contains the 3GPP Home Network MNC parameters.

Parameters

is3GppNw[OUT]	• Value – True - if TLV 0x01 corresponds to a 3GPP network – False - In all other cases.
mccIncPcsDigit[OUT]	• This field is used to interpret the length of the mobile_network_code reported in TLV 0x01. • Values – TRUE - MNC is a three-digit value if a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a two-digit value if a reported value of 90 corresponds to an MNC value of 90

TlvPresent[OUT]	• Tlv Present

Note

mccIncPcsDigit value is ignored if is3GppNw is FALSE.

8.232.2 Field Documentation

8.232.2.1 uint8_t nas_homeNwMNC3GppTlv::is3GppNw

8.232.2.2 uint8_t nas_homeNwMNC3GppTlv::mccIncPcsDigit

8.232.2.3 uint8_t nas_homeNwMNC3GppTlv::TlvPresent

8.233 nas_homeSIDNID Struct Reference

Data Fields

- uint8_t numInstances
- nas_sidNid SidNid [255]

8.233.1 Detailed Description

This structure contains Home SID/NID

Parameters

numInstances	• Number of sets of the following elements: – sid – nid • If zero(0), then no information follows.
SidNid	• See nas_sidNid for more information

8.233.2 Field Documentation

8.233.2.1 uint8_t nas_homeSIDNID::numInstances

8.233.2.2 nas_sidNid nas_homeSIDNID::SidNid[255]

8.234 nas_ImsVoiceSupportLteTlv Struct Reference

Data Fields

- uint8_t ImsVoiceSupportLte
- uint8_t TLVPresent

8.234.1 Detailed Description

Structure for storing the IMS Voice Support Status on LTE

Parameters

ImsVoice-SupportLte	• IMS Voice Support Status on LTE • Values – 0x00 - Support is not available – 0x01 - Support is available
TLVPresent	• TLV Present

8.234.2 Field Documentation

8.234.2.1 uint8_t nas_ImsVoiceSupportLteTlv::ImsVoiceSupportLte

8.234.2.2 uint8_t nas_ImsVoiceSupportLteTlv::TLVPresent

8.235 nas_infoInterFreq Struct Reference

Data Fields

- uint16_t earfcn
- uint8_t threshXLow
- uint8_t threshXHigh
- uint8_t cell_resel_priority
- uint8_t cells_len
- nas_cellParams cellInterFreqParams [255]

8.235.1 Detailed Description

This structure contains information about the inter-frequency.

Parameters

earfcn	• E-UTRA absolute radio frequency channel number of the serving cell. • Range: 0 to 65535.
threshXLow	• Cell Srxlev low threshold. • Range: 0 to 31. • When the serving cell does not exceed thresh_serving_low, the value of an evaluated cell must be smaller than this value to be considered for re-selection.
threshXHigh	• Cell Srxlev high threshold. • Range: 0 to 31. • When the serving cell exceeds thresh_serving_low, the value of an evaluated cell must be greater than this value to be considered for re-selection.

cell_resel_priority	• Cell re-selection priority • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE.
cells_len	• Provides the number of set of cell params.
cellInterFreqParams[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_cellParams for more information.

8.235.2 Field Documentation

8.235.2.1 uint8_t nas_infoInterFreq::cell_resel_priority

8.235.2.2 nas_cellParams nas_infoInterFreq::cellInterFreqParams[255]

8.235.2.3 uint8_t nas_infoInterFreq::cells_len

8.235.2.4 uint16_t nas_infoInterFreq::earfcn

8.235.2.5 uint8_t nas_infoInterFreq::threshXHigh

8.235.2.6 uint8_t nas_infoInterFreq::threshXLow

8.236 nas_IOThresh Struct Reference

Data Fields

- uint8_t IOThresListLen
- int32_t * pIOThresList

8.236.1 Detailed Description

This structure contains IO threshold list.

Parameters

IOThresListLen	• Length of the IO threshold list parameter to follow
pIOThresList	• Sequence of thresholds delimiting IO event reporting bands • Every time a new IO value crosses a threshold value, an event report indication message with the new IO value is sent to the requesting control point. For this field – IO is applicable only for HDR – Each IO threshold value is a signed 4 byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.236.2 Field Documentation

8.236.2.1 uint8_t nas_IOThresh::IOThresListLen

8.236.2.2 int32_t* nas_IOThresh::pIOThresList

8.237 nas_LTEARFCNTlv Struct Reference

Data Fields

- uint8_t numIteArfcn
- uint32_t IteArfcn [255]

8.237.1 Detailed Description

This structure contains the parameters for LTE ARFCN.

Parameters

numIteArfcn	• Number of sets of the following elements: • Ite_arfcn
IteArfcn	• List of LTE RF channel numbers to be scanned. Only applicable to the following scan types: – NAS_SCAN_TYPE_PLMN – NAS_SCAN_TYPE_MODE_PREF

8.237.2 Field Documentation

8.237.2.1 uint32_t nas_LTEARFCNTlv::IteArfcn[255]

8.237.2.2 uint8_t nas_LTEARFCNTlv::numIteArfcn

8.238 nas_IteBandPrefExt Struct Reference

Data Fields

- uint64_t bits_1_64
- uint64_t bits_65_128
- uint64_t bits_129_192
- uint64_t bits_193_256

8.238.1 Detailed Description

Contain the LTE Band Preference Extended parameters

Parameters

bits_1_64	• Bits 1 to 64 of the 256-bit LTE E-UTRA Operating Band bitmask.

bits_65_128	• Bits 65 to 128 of the 256-bit LTE E-UTRA Operating Band bitmask.
bits_129_192	• Bits 129 to 192 of the 256-bit LTE E-UTRA Operating Band bitmask.
bits_193_256	• Bits 193 to 256 of the 256-bit LTE E-UTRA Operating Band bitmask.

8.238.2 Field Documentation

8.238.2.1 uint64_t nas_lteBandPrefExt::bits_129_192

8.238.2.2 uint64_t nas_lteBandPrefExt::bits_193_256

8.238.2.3 uint64_t nas_lteBandPrefExt::bits_1_64

8.238.2.4 uint64_t nas_lteBandPrefExt::bits_65_128

8.239 nas_LTEBandPrefTlv Struct Reference

Data Fields

- uint64_t LTEBandPref
- uint8_t TlvPresent

8.239.1 Detailed Description

Contain the LTE Band Preference for system selection preferences.

Parameters

LTEBandPref	• Bit mask representing the LTE band preference • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – All other bits are reserved
TlvPresent	• Tlv Present

8.239.2 Field Documentation

8.239.2.1 uint64_t nas_LTEBandPrefTlv::LTEBandPref

8.239.2.2 uint8_t nas_LTEBandPrefTlv::TlvPresent

8.240 nas_LteCiotOpModeTlv Struct Reference

Data Fields

- uint32_t campedCiotLteOpMode
- uint8_t TlvPresent

8.240.1 Detailed Description

This structure contains Camped CIOT LTE Operational Mode parameters.

Parameters

campedCiotLte-OpMode	• Indicates the camped CIoT LTE mode of operation. • Values: – NAS_CIOT_SYS_MODE_NO_SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_ NB1 (0x03) - Camped on LTE NB1
TlvPresent	• Tlv Present or not

8.240.2 Field Documentation

8.240.2.1 uint32_t nas_LteCiotOpModeTlv::campedCiotLteOpMode

8.240.2.2 uint8_t nas_LteCiotOpModeTlv::TlvPresent

8.241 nas_lteEARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn0
- uint32_t earfcn1

8.241.1 Detailed Description

This structure contains LTE EARFCN infromation.

Parameters

status	• 0 - Disable • 1 - Enable

earfcn0	• Primary DL EARFCN to which the UE is locked
earfcn1	• Secondary DL EARFCN to which the UE is locked • Note : Make earfcn1 value equal to earfcn0 if only one EARFCN is desired.

8.241.2 Field Documentation

8.241.2.1 uint32_t nas_lteEARFCN::earfcn0

8.241.2.2 uint32_t nas_lteEARFCN::earfcn1

8.241.2.3 uint8_t nas_lteEARFCN::status

8.242 nas_LteEarfcnInfo Struct Reference

Data Fields

- uint8_t lteInterEarfcnlen
- uint32_t lteEarfcn [255]
- uint8_t TlvPresent

8.242.1 Detailed Description

This structure contains information about the LTE EARFCN Info.

Parameters

lteInterEarfcnlen	• Number of sets of the lteEarfcn
lteEarfcn	• LTE neighbor cell information EARFCN.
TlvPresent	• Tlv Present

8.242.2 Field Documentation

8.242.2.1 uint32_t nas_LteEarfcnInfo::lteEarfcn[255]

8.242.2.2 uint8_t nas_LteEarfcnInfo::lteInterEarfcnlen

8.242.2.3 uint8_t nas_LteEarfcnInfo::TlvPresent

8.243 nas_LteEmbmsCoverageTlv Struct Reference

Data Fields

- uint8_t LteEmbmsCoverage
- uint8_t TLVPresent

8.243.1 Detailed Description

Structure for storing the LTE eMBMS Coverage Information.

Parameters

LteEmbms-Coverage	• Values – TRUE - Current LTE system supports eMBMS – FALSE - Current LTE system does not support eMBMS
TLVPresent	• TLV Present

8.243.2 Field Documentation

8.243.2.1 uint8_t nas_LteEmbmsCoverageTlv::LteEmbmsCoverage

8.243.2.2 uint8_t nas_LteEmbmsCoverageTlv::TLVPresent

8.244 nas_LteEmbmsTraceIdTlv Struct Reference

Data Fields

- int16_t LteEmbmsTraceId
- uint8_t TLVPresent

8.244.1 Detailed Description

Structure for storing the LTE eMBMS Coverage Info Trace ID.

Parameters

LteEmbmsTrace-Id	• LTE eMBMS coverage information trace ID. • Values – 0 to 32768 - Valid trace ID – -1 - Trace ID is not used
TLVPresent	• TLV Present

8.244.2 Field Documentation

8.244.2.1 int16_t nas_LteEmbmsTraceIdTlv::LteEmbmsTraceId

8.244.2.2 uint8_t nas_LteEmbmsTraceIdTlv::TLVPresent

8.245 nas_lteGsmCellInfo Struct Reference

Data Fields

- uint8_t cellReselPriority
- uint8_t threshGsmHigh
- uint8_t threshGsmLow
- uint8_t nccPermitted
- uint8_t cells_len
- nas_gsmCellInfo GsmCellInfo [255]

8.245.1 Detailed Description

This structure contains information about the LTE GSM Cell.

Parameters

cellReselPriority	• Priority of this frequency group. • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE.
threshGsmHigh	• Reselection threshold for high priority layers. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
threshGsmLow	• Reselection threshold for low priority layers. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
nccPermitted	• Bitmask specifying whether a neighbor with a specific network color code is to be reported. • Range: 0 to 255. • Bit n set to 1 means a neighbor with NCC n must be included in the report. This flag is synonymous with a blacklist in other RATs. • This field is only valid when ue_in_idle is TRUE.
cells_len	• Provides the number of set of gsm cells.
GsmCellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_gsmCellInfo for more information.

8.245.2 Field Documentation

8.245.2.1 uint8_t nas_lteGsmCellInfo::cellReselPriority

8.245.2.2 uint8_t nas_lteGsmCellInfo::cells_len

8.245.2.3 nas_gsmCellInfo nas_lteGsmCellInfo::GsmCellInfo[255]

8.245.2.4 uint8_t nas_lteGsmCellInfo::nccPermitted

8.245.2.5 uint8_t nas_lteGsmCellInfo::threshGsmHigh

8.245.2.6 uint8_t nas_lteGsmCellInfo::threshGsmLow

8.246 nas_LTEInfo Struct Reference

Data Fields

- uint8_t band
- uint8_t bandwidth
- uint16_t RXChan
- uint16_t TXChan
- uint8_t emmState
- uint8_t emmSubState
- uint8_t emmConnState

8.246.1 Detailed Description

Structure for storing the LTE information for the device.

Parameters

band	• LTE Band – 1 – 40 (Band in decimal) – 0xFF - Invalid
bandwidth	• BandWidth. – 0x00 - 1.4 MHz – 0x01 - 3 MHz – 0x02 - 5 MHz – 0x03 - 10 MHz – 0x04 - 15 MHz – 0x05 - 20 MHz – 0x06 - Invalid – 0xFF - Unknown
RXChan	• RX channel number in decimal – 0xFFFF - Not Available
TXChan	• TX channel number in decimal – 0xFFFF - Not Available

emmState	• EMM State. – 0x00 - Deregistered – 0x01 - Reg Initiated – 0x02 - Registered – 0x03 - TAU Initiated – 0x04 - SR Initiated – 0x05 - Dereg Initiated – 0x06 - Invalid – 0xFF - Unknown
emmSubState	• EMM Sub State. – 0xFF - NOT Applicable • When EMM_state is 0x00: – 0x00 - No IMSI – 0x01 - PLMN Search – 0x02 - Attach Needed – 0x03 - No Cell – 0x04 - Attaching – 0x05 - Normal Service – 0x06 - Limited Service – 0x07 - Waiting for PDN • When EMM_state is 0x01: – 0x00 - Waiting for NW – 0x01 - Waiting for ESM • When EMM_state is 0x02: – 0x00 - Normal Service – 0x01 - Update Needed – 0x02 - Attempt Update – 0x03 - No Cell – 0x04 - PLMN Search – 0x05 - Limited Service – 0x06 - MM Update – 0x07 - IMSI Detach – 0x08 - Waiting for ESM
emmConnState	• EMM Connected Mode State. – 0x00 - RRC Idle – 0x01 - Waiting RRC Cfm – 0x02 - RRC Connected – 0x03 - RRC Releasing – 0xFF - Unknown

8.246.2 Field Documentation

8.246.2.1 uint8_t nas_LTEInfo::band

8.246.2.2 uint8_t nas_LTEInfo::bandwidth

8.246.2.3 uint8_t nas_LTEInfo::emmConnState

8.246.2.4 uint8_t nas_LTEInfo::emmState

8.246.2.5 uint8_t nas_LTEInfo::emmSubState

8.246.2.6 uint16_t nas_LTEInfo::RXChan

8.246.2.7 uint16_t nas_LTEInfo::TXChan

8.247 nas_LTEInfoInterfreq Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t freqsLen
- nas_infoInterFreq InfoInterfreq [255]

8.247.1 Detailed Description

This structure contains information about the LTE Inter-Frequency Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available
freqsLen	• Provides the number of set of inter frequency information. • If 0(zero), then no information follows it.
InfoInterfreq[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_infoInterFreq for more information.

8.247.2 Field Documentation

8.247.2.1 uint8_t nas_LTEInfoInterfreq::freqsLen

8.247.2.2 nas_infoInterFreq nas_LTEInfoInterfreq::InfoInterfreq[255]

8.247.2.3 uint8_t nas_LTEInfoInterfreq::ueInIdle

8.248 nas_LTEInfoIntrafreq Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t plmn [3]

- uint16_t tac
- uint32_t globalCellId
- uint16_t earfcn
- uint16_t servingCellId
- uint8_t cellReselPriority
- uint8_t sNonIntraSearch
- uint8_t threshServingLow
- uint8_t sIntraSearch
- uint8_t cellsLen
- nas_cellParams CellParams [255]

8.248.1 Detailed Description

This structure contains information about the LTE Intra-Frequency Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available
plmn[NAS_PLMN_LENGTH]	• PLMN ID coded as octet 3, 4, and 5.
tac	• Tracking area code. – 0xFFFF - Not Available
globalCellId	• Global cell ID in the system information block. – 0xFFFFFFFF - Not Available
earfcn	• E-UTRA absolute radio frequency channel number of the serving cell. • Range: 0 to 65535. – 0xFFFF - Not Available
servingCellId	• LTE serving cell ID. • Range: 0 to 503. • This is the cell ID of the serving cell and can be found in the cell list. – 0xFFFF - Not Available
cellReselPriority	• Priority for serving frequency. • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available
sNonIntraSearch	• S non-intra search threshold to control non-intrafrequency searches. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available

threshServing-Low	• Serving cell low threshold. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available
sIntraSearch	• S Intra search threshold. • Range: 0 to 31. • The current cell measurement must fall below this threshold to consider intrafrequency for reselection. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available
cellsLen	• Provides the number of set of cell params. • If 0(zero), then no information follows it.
CellParams[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_cellParams for more information.

8.248.2 Field Documentation

8.248.2.1 nas_cellParams nas_LTEInfoIntrafreq::CellParams[255]

8.248.2.2 uint8_t nas_LTEInfoIntrafreq::cellReselPriority

8.248.2.3 uint8_t nas_LTEInfoIntrafreq::cellsLen

8.248.2.4 uint16_t nas_LTEInfoIntrafreq::earfcn

8.248.2.5 uint32_t nas_LTEInfoIntrafreq::globalCellId

8.248.2.6 uint8_t nas_LTEInfoIntrafreq::plmn[3]

8.248.2.7 uint16_t nas_LTEInfoIntrafreq::servingCellId

8.248.2.8 uint8_t nas_LTEInfoIntrafreq::sIntraSearch

8.248.2.9 uint8_t nas_LTEInfoIntrafreq::sNonIntraSearch

8.248.2.10 uint16_t nas_LTEInfoIntrafreq::tac

8.248.2.11 uint8_t nas_LTEInfoIntrafreq::threshServingLow

8.248.2.12 uint8_t nas_LTEInfoIntrafreq::ueInIdle

8.249 nas_LTEInfoNeighboringGSM Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t freqsLen
- nas_lteGsmCellInfo LteGsmCellInfo [255]

8.249.1 Detailed Description

This structure contains information about the LTE Neighboring GSM Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available
freqsLen	• Provides the number of set of LTE GSM cell information. • If 0(zero), then no information follows it.
LteGsmCellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_lteGsmCellInfo for more information.

8.249.2 Field Documentation

8.249.2.1 uint8_t nas_LTEInfoNeighboringGSM::freqsLen

8.249.2.2 nas_lteGsmCellInfo nas_LTEInfoNeighboringGSM::LteGsmCellInfo[255]

8.249.2.3 uint8_t nas_LTEInfoNeighboringGSM::ueInIdle

8.250 nas_LTEInfoNeighboringWCDMA Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t freqsLen
- nas_lteWcdmaCellInfo LTEWCDMACellInfo [255]

8.250.1 Detailed Description

This structure contains information about the LTE Neighboring WCDMA Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available

freqsLen	• Provides the number of set of LTE WCDMA cell information. • If 0(zero), then no information follows it.
LTEWCDMA-CellInfo[NAS_M-AX_DESCRIPTI-ON_LENGTH]	• See nas_lteWcdmaCellInfo for more information.

8.250.2 Field Documentation

8.250.2.1 uint8_t nas_LTEInfoNeighboringWCDMA::freqsLen

8.250.2.2 nas_lteWcdmaCellInfo nas_LTEInfoNeighboringWCDMA::LTEWCDMACellInfo[255]

8.250.2.3 uint8_t nas_LTEInfoNeighboringWCDMA::ueInIdle

8.251 nas_LteM1BandPrefTlv Struct Reference

Data Fields

- uint64_t lteM1BandPref
- uint8_t tlvPresent

8.251.1 Detailed Description

Contain the LTE M1 band Preference.

Parameters

lteM1BandPref	• Bitmask representing the LTE M1 band preference • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – Bit 61 - E-UTRA Operating Band 126 – Bit 62 - E-UTRA Operating Band 127
TlvPresent	• Tlv Present

8.251.2 Field Documentation

8.251.2.1 uint64_t nas_LteM1BandPrefTlv::lteM1BandPref

8.251.2.2 uint8_t nas_LteM1BandPrefTlv::TlvPresent

8.252 nas_LteNb1BandPrefTlv Struct Reference

Data Fields

- uint64_t lteNb1BandPref
- uint8_t TlvPresent

8.252.1 Detailed Description

Contain the LTE NB1 band Preference.

Parameters

lteNb1BandPref	• Bitmask representing the LTE NB1 band preference. • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – Bit 61 - E-UTRA Operating Band 126 – Bit 62 - E-UTRA Operating Band 127
TlvPresent	• Tlv Present

8.252.2 Field Documentation

8.252.2.1 uint64_t nas_LteNb1BandPrefTlv::lteNb1BandPref

8.252.2.2 uint8_t nas_LteNb1BandPrefTlv::TlvPresent

8.253 nas_LTEOperationalModeTlv Struct Reference

Data Fields

- uint32_t ciotLteOpMode
- uint8_t TlvPresent

8.253.1 Detailed Description

Contain fields in struct nas_LTEOperationalModeTlv

Parameters

ciotLteOpMode	• CIOT LTE operational mode. • Values – 0x00 - No service – 0x01 - Camped on LTE wideband – 0x02 - Camped on LTE M1 – 0x03 - Camped on LTE NB1
TlvPresent	• Tlv Present.

Note

None

8.253.2 Field Documentation

8.253.2.1 uint32_t nas_LTEOperationalModeTlv::ciotLteOpMode

8.253.2.2 uint8_t nas_LTEOperationalModeTlv::TlvPresent

8.254 nas_LTEOperationMode Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t * pLTEOperationMode

8.254.1 Detailed Description

This structure contains the nas_LTEOperationMode response parameters.

Parameters

TlvPresent	• Tlv Present
pLTEOperation-Mode	• LTE Operational Mode

8.254.2 Field Documentation

8.254.2.1 uint32_t* nas_LTEOperationMode::pLTEOperationMode

8.254.2.2 uint8_t nas_LTEOperationMode::TlvPresent

8.255 nas_LteOpMode Struct Reference

Data Fields

- uint32_t lteOpMode
- uint8_t TlvPresent

8.255.1 Detailed Description

This structure contains LTE Operational Mode

Parameters

lteOpMode	• LTE Operational Mode • CIOT LTE mode on which reject indication is received. • Values – NAS_CIOT_SYS_MODE_NO_SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_NB1 (0x03) - Camped on LTE NB1
TlvPresent	• Tlv Present or not

8.255.2 Field Documentation

8.255.2.1 uint32_t nas_LteOpMode::lteOpMode

8.255.2.2 uint8_t nas_LteOpMode::TlvPresent

8.256 nas_lteOpModeTlv Struct Reference

Data Fields

- uint8_t lteOpModeLen
- uint16_t MCC [255]

- uint16_t MNC [255]
- uint32_t lteOpMode [255]
- uint8_t TlvPresent

8.256.1 Detailed Description

This structure contains the LTE Operational Mode response parameters.

Parameters

lteOpModeLen	• Number of sets of the following elements – MCC – MNC – lteOpMode
MCC	• Mobile Country Code • A 16-bit integer representation of MCC • Range: 0 to 999
MNC	• Mobile Network Code • A 16-bit integer representation of MNC • Range: 0 to 999
lteOpMode	• Indicates the LTE mode of operation. • Values: – NAS_CIOT_SYS_MODE_NO_SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_NB1 (0x03) - Camped on LTE NB1
TlvPresent	• Tlv Present

8.256.2 Field Documentation

8.256.2.1 uint32_t nas_lteOpModeTlv::lteOpMode[255]

8.256.2.2 uint8_t nas_lteOpModeTlv::lteOpModeLen

8.256.2.3 uint16_t nas_lteOpModeTlv::MCC[255]

8.256.2.4 uint16_t nas_lteOpModeTlv::MNC[255]

8.256.2.5 uint8_t nas_lteOpModeTlv::TlvPresent

8.257 nas_ltePCI Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn
- uint32_t pci

8.257.1 Detailed Description

This structure contains LTE PCI infromation.

Parameters

status	• 0 - Disable • 1 - Enable
earfcn	• UARFCN to which UE is locked
pci	• PCI to which the UE is locked

8.257.2 Field Documentation

8.257.2.1 uint32_t nas_ltePCI::earfcn

8.257.2.2 uint32_t nas_ltePCI::pci

8.257.2.3 uint8_t nas_ltePCI::status

8.258 nas_LteRegDomainTlv Struct Reference

Data Fields

- uint32_t LteRegDomain
- uint8_t TLVPresent

8.258.1 Detailed Description

Structure for storing the LTE Registration Domain

Parameters

LteRegDomain	• LTE registration domain. • Values – 0 - Not applicable because the UE is not in Camp Only mode – 1 - UE is in Camp Only mode and the PLMN can provide CS service only – 2 - UE is in Camp Only mode and the PLMN can provide PS service only – 3 - UE is in Camp Only mode and the PLMN can provide CS and PS service – 4 - UE is in Camp Only

TLVPresent	• TLV Present

8.258.2 Field Documentation

8.258.2.1 uint32_t nas_LteRegDomainTlv::LteRegDomain

8.258.2.2 uint8_t nas_LteRegDomainTlv::TLVPresent

8.259 nas_lteRsrpinformation Struct Reference

Data Fields

- int16_t rsrplevel

8.259.1 Detailed Description

This structure contains the LTE RSRP Information

Parameters

rsrplevel	• LTE RSRP in dBm as a mesaured by L1. Range: -44 to -140(-44 means -44dBm, -140 means -140dBm).

8.259.2 Field Documentation

8.259.2.1 int16_t nas_lteRsrpinformation::rsrplevel

8.260 nas_LTERSRPThresh Struct Reference

Data Fields

- uint8_t LTERSRPThreshListLen
- int16_t * pLTERSRPThreshList

8.260.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

LTERSRP-ThreshListLen	• Length of the LTE RSRP threshold list parameter to follow
pLTERSRP-ThreshList	• Array of RSRP thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSRP values: -140 to -44 (in dBm).

8.260.2 Field Documentation

8.260.2.1 uint8_t nas_LTERSRPThresh::LTERSRPThreshListLen

8.260.2.2 int16_t* nas_LTERSRPThresh::pLTERSRPThreshList

8.261 nas_LTERSRQThresh Struct Reference

Data Fields

- uint8_t LTERSRQThreshListLen
- int16_t * pLTERSRQThreshList

8.261.1 Detailed Description

This structure contains LTE RSRQ threshold related parameters.

Parameters

LTERSRQ-ThreshListLen	• Length of the LTE RSRQ threshold list parameter to follow
pLTERSRQ-ThreshList	• Array of RSRQ thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSRQ values: -20 to -3 (in dBm)

8.261.2 Field Documentation

8.261.2.1 uint8_t nas_LTERSRQThresh::LTERSRQThreshListLen

8.261.2.2 int16_t* nas_LTERSRQThresh::pLTERSRQThreshList

8.262 nas_LTERSSIThresh Struct Reference

Data Fields

- uint8_t LTERSSIThreshListLen
- int16_t * pLTERSSIThreshList

8.262.1 Detailed Description

This structure contains LTE RSSI threshold related parameters.

Parameters

LTERSSI-ThreshListLen	• Length of the LTE RSSI threshold list parameter to follow
pLTERSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -120 to 0 (in dBm)

8.262.2 Field Documentation

8.262.2.1 uint8_t nas_LTERSSIThresh::LTERSSIThreshListLen

8.262.2.2 int16_t* nas_LTERSSIThresh::pLTERSSIThreshList

8.263 nas_LTESigRptCfg Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.263.1 Detailed Description

This structure contains LTE signal report config.

Parameters

rptRate	• Rate on how often the LTE signal must be checked for reporting • Values: – 0 - Report using the default configuration – 1 - Report every 1 sec – 2 - Report every 2 sec – 3 - Report every 3 sec – 4 - Report every 4 sec – 5 - Report every 5 sec

avgPeriod	• Averaging period to be used for the LTE signal • Values: – 0 - Average using the default configuration – 1 - Average over 1 sec – 2 - Average over 2 sec – 3 - Average over 3 sec – 4 - Average over 4 sec – 5 - Average over 5 sec – 6 - Average over 6 sec – 7 - Average over 7 sec – 8 - Average over 8 sec – 9 - Average over 9 sec – 10 - Average over 10 sec

8.263.2 Field Documentation

8.263.2.1 uint8_t nas_LTESigRptCfg::avgPeriod

8.263.2.2 uint8_t nas_LTESigRptCfg::rptRate

8.264 nas_LTESigRptConfig Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.264.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

rptRate	• Rate on how often the LTE signal must be checked for reporting Values • 0 - Report using the default configuration • 1 - Report every 1 sec • 2 - Report every 2 sec • 3 - Report every 3 sec • 4 - Report every 4 sec • 5 - Report every 5 sec

avgPeriod	• Averaging period to be used for the LTE signal. • Values – 0 - Average using the default configuration – 1 - Average over 1 sec – 2 - Average over 2 sec – 3 - Average over 3 sec – 4 - Average over 4 sec – 5 - Average over 5 sec – 6 - Average over 6 sec – 7 - Average over 7 sec – 8 - Average over 8 sec – 9 - Average over 9 sec – 10 - Average over 10 sec

8.264.2 Field Documentation

8.264.2.1 uint8_t nas_LTESigRptConfig::avgPeriod

8.264.2.2 uint8_t nas_LTESigRptConfig::rptRate

8.265 nas_lteSnrinformation Struct Reference

Data Fields

- int16_t snrlevel

8.265.1 Detailed Description

This structure contains the LTE SNR Information

Parameters

snrlevel	• LTE SNR level as a scaled integer in units of 0.1dB e.g. -16dB has a value of -160 and 24.6dB has value of 246.

8.265.2 Field Documentation

8.265.2.1 int16_t nas_lteSnrinformation::snrlevel

8.266 nas_LTESNRThresh Struct Reference

Data Fields

- uint8_t LTESNRThresListLen
- int16_t * pLTESNRThresList

8.266.1 Detailed Description

This structure contains LTE SNR threshold list.

Parameters

LTESNRThres-ListLen	• Length of the LTE SNR threshold list parameter to follow
pLTESNRThres-List	• Sequence of thresholds delimiting SNR event reporting bands • Every time a SNR value crosses a threshold value, an event report indication message with the new SNR value is sent to the requesting control point. For this field – For LTE, each SNR threshold value is a signed 2 Byte value – Maximum number of threshold values is 16 – At least one value must be specified – SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246

8.266.2 Field Documentation

8.266.2.1 uint8_t nas_LTESNRThresh::LTESNRThresListLen

8.266.2.2 int16_t* nas_LTESNRThresh::pLTESNRThresList

8.267 nas_LTESNRThreshold Struct Reference

Data Fields

- uint8_t LTESNRThreshListLen
- int16_t * pLTESNRThreshList

8.267.1 Detailed Description

This structure contains LTE SNR threshold related parameters.

Parameters

LTESNRThresh-ListLen	• Length of the LTE SNR threshold list parameter to follow
pLTESNR-ThreshList	• Array of SNR thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for SNR values: -20 to 30 (in dB).

8.267.2 Field Documentation

8.267.2.1 uint8_t nas_LTESNRThreshold::LTESNRThreshListLen

8.267.2.2 int16_t* nas_LTESNRThreshold::pLTESNRThreshList

8.268 nas_LTESysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoLTE
- uint8_t lacValid
- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejCause
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]
- uint8_t tacValid
- uint16_t tac

8.268.1 Detailed Description

Structure for storing the LTE System Information.

Parameters

sysInfoLTE	• See nas_sysInfoCommon for more information.
lacValid	• Indicates whether the location area code is valid.. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
lac	• Location area code. • Only applies to 3GPP. – 0xFFFF - Not Available
cellIdValid	• Indicates whether the cell ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
cellId	• Cell ID. – 0xFFFFFFFF - Not Available

regRejectInfo-Valid	• Indicates whether the registration reject information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
rejectSrvDomain	• Type of service domain in which the registration is rejected. – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
rejCause	• Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. and [3GPP TS 24.301, Section 9.9.3.9] – 0xFF - Not Available
networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
MCC[NAS_PLM-N_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters
MNC[NAS_PLM-N_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
tacValid	• Indicates whether tracking area code is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
tac	• Tracking area code. • Only applicable for LTE. – 0xFFFF - Not Available

8.268.2 Field Documentation

8.268.2.1 uint32_t nas_LTESysInfo::cellId

8.268.2.2 uint8_t nas_LTESysInfo::cellIdValid

8.268.2.3 uint16_t nas_LTESysInfo::lac

8.268.2.4 uint8_t nas_LTESysInfo::lacValid

8.268.2.5 uint8_t nas_LTESysInfo::MCC[3]

8.268.2.6 uint8_t nas_LTESysInfo::MNC[3]

8.268.2.7 uint8_t nas_LTESysInfo::networkIdValid

8.268.2.8 uint8_t nas_LTESysInfo::regRejectInfoValid

8.268.2.9 uint8_t nas_LTESysInfo::rejCause

8.268.2.10 uint8_t nas_LTESysInfo::rejectSrvDomain

8.268.2.11 nas_sysInfoCommon nas_LTESysInfo::sysInfoLTE

8.268.2.12 uint16_t nas_LTESysInfo::tac

8.268.2.13 uint8_t nas_LTESysInfo::tacValid

8.269 nas_LteVoiceDomainTlv Struct Reference

Data Fields

- uint32_t LteVoiceDomain
- uint8_t TLVPresent

8.269.1 Detailed Description

Structure for storing the LTE Voice Domain

Parameters

LteVoiceDomain	• LTE voice domain. – Values * 0 - Data-centric devices: No voice, stay on LTE * 1 - Voice is supported over the IMS network * 2 - Voice is supported over the 1X network * 3 - Voice is supported over the 3GPP network
TLVPresent	• TLV Present

8.269.2 Field Documentation

8.269.2.1 uint32_t nas_LteVoiceDomainTlv::LteVoiceDomain

8.269.2.2 uint8_t nas_LteVoiceDomainTlv::TLVPresent

8.270 nas_lteWcdmaCellInfo Struct Reference

Data Fields

- uint16_t uarfcn
- uint8_t cellReselPriority
- uint16_t threshXhigh
- uint16_t threshXlow
- uint8_t cellsLen
- nas_wcdmaCellInfo WCDMACellInfo [255]

8.270.1 Detailed Description

This structure contains information about the LTE WCDMA Cell.

Parameters

uarfcn	• WCDMA layer frequency. • Range: 0 to 16383.
cellReselPriority	• Cell re-selection priority. • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE.
threshXhigh	• Re-selection low threshold. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
threshXlow	• Re-selection high threshold. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
cellsLen	• Provides the number of set of WCDMA cells.
WCDMACellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_wcdmaCellInfo for more information.

8.270.2 Field Documentation

8.270.2.1 uint8_t nas_lteWcdmaCellInfo::cellReselPriority

8.270.2.2 uint8_t nas_lteWcdmaCellInfo::cellsLen

8.270.2.3 uint16_t nas_lteWcdmaCellInfo::threshXhigh

8.270.2.4 uint16_t nas_lteWcdmaCellInfo::threshXlow

8.270.2.5 uint16_t nas_lteWcdmaCellInfo::uarfcn

8.270.2.6 nas_wcdmaCellInfo nas_lteWcdmaCellInfo::WCDMACellInfo[255]

8.271 nas_Mdn Struct Reference

Data Fields

- uint8_t mdnLen
- uint8_t mdn [15]

8.271.1 Detailed Description

This structure contains the parameters for CDMA MDN Information

Parameters

mdnLen	• Length of MDN
MDN	• MDN in ASCII format, max length is 15

8.271.2 Field Documentation

8.271.2.1 uint8_t nas_Mdn::mdn[15]

8.271.2.2 uint8_t nas_Mdn::mdnLen

8.272 nas_minBasedIMSI Struct Reference

Data Fields

- uint8_t mccM [3]
- uint16_t imsiM1112
- uint8_t imsiMS1 [7]
- uint8_t imsiMS2 [3]

8.272.1 Detailed Description

This structure contains MIN-based IMSI.

Parameters

mccM	• ASCII character representation of MCC_M

imsiM1112	• ASCII character representation of IMSI_M_11_12 value – 0xFFFF - Not Available
imsiMS1	• ASCII character representation of IMSI_M_S1 value
imsiMS2	• ASCII character representation of IMSI_M_S2 value

8.272.2 Field Documentation

8.272.2.1 uint16_t nas_minBasedIMSI::imsiM1112

8.272.2.2 uint8_t nas_minBasedIMSI::imsiMS1[7]

8.272.2.3 uint8_t nas_minBasedIMSI::imsiMS2[3]

8.272.2.4 uint8_t nas_minBasedIMSI::mccM[3]

8.273 nas_MNCPCSDigitStatus Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t MNCIncPCSDigit
- uint8_t TLVPresent

8.273.1 Detailed Description

This structure contains MNC PCS Digit Include Status.

Parameters

MCC	• A 16-bit integer representation of MCC. • Range - 0 to 999.
MNC	• A 16-bit integer representation of MNC. • Range - 0 to 999.
MNCIncPCS-Digit	• This field is used to interpret the length of the corresponding MNC reported in the TLVs with an MNC. • Values – TRUE - MNC is a three-digit value. e.g., a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a two-digit value. e.g., a reported value of 90 corresponds to an MNC value of 90

TLVPresent	• TLV Present.

8.273.2 Field Documentation

8.273.2.1 uint16_t nas_MNCPCSDigitStatus::MCC

8.273.2.2 uint16_t nas_MNCPCSDigitStatus::MNC

8.273.2.3 uint8_t nas_MNCPCSDigitStatus::MNCIncPCSDigit

8.273.2.4 uint8_t nas_MNCPCSDigitStatus::TLVPresent

8.274 nas_MNRInfo Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint32_t rat

8.274.1 Detailed Description

Structure contains Manual Network Register Information parameters

Parameters

mcc	• A 16-bit integer representation of Mobile Country Code. Range - 0 to 999.
mnc	• A 16-bit integer representation of Mobile Network Code. Range - 0 to 999.
rat	• Radio access technology for which to register. – 0x04 - RADIO_IF_GSM – 0x05 - RADIO_IF_UMTS – 0x08 - RADIO_IF_LTE

8.274.2 Field Documentation

8.274.2.1 uint16_t nas_MNRInfo::mcc

8.274.2.2 uint16_t nas_MNRInfo::mnc

8.274.2.3 uint32_t nas_MNRInfo::rat

8.275 nas_ModePrefTlv Struct Reference

Data Fields

- uint16_t ModePref
- uint8_t TlvPresent

8.275.1 Detailed Description

Contain the radio technology mode preference

Parameters

ModePref	• Bit Mask indicating the radio technology mode preference • Bit values: – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE – Bit 5 - TD-SCDMA – Bit 6 - NR5G
TlvPresent	• Tlv Present

8.275.2 Field Documentation

8.275.2.1 uint16_t nas_ModePrefTlv::ModePref

8.275.2.2 uint8_t nas_ModePrefTlv::TlvPresent

8.276 nas_namName Struct Reference

Data Fields

- uint8_t namNameLen
- uint8_t namName [12]

8.276.1 Detailed Description

This structure contains NAM Name.

Parameters

namNameLen	• Number of sets of the following elements: – nam_name • If zero(0), then no information follows.
namName	• Name information in ASCII. The maximum length of nam_name is 12.

8.276.2 Field Documentation

8.276.2.1 uint8_t nas_namName::namName[12]

8.276.2.2 uint8_t nas_namName::namNameLen

8.277 nas_netSelectionPref Struct Reference

Data Fields

- uint8_t netReg
- uint16_t mcc
- uint16_t mnc

8.277.1 Detailed Description

Contain the network selection preference.

Parameters

netReg	• specifies one of the following actions: – 0x00 - Automatic registration * Device registers according to its provisioning; mcc and mnc fields are ignored – 0x01 - Manual Registration * Device registers to specified network; mcc and mnc must contain valid values
mcc	• MCC value. Range 0 to 999
mnc	• MNC value. Range 0 to 999

8.277.2 Field Documentation

8.277.2.1 uint16_t nas_netSelectionPref::mcc

8.277.2.2 uint16_t nas_netSelectionPref::mnc

8.277.2.3 uint8_t nas_netSelectionPref::netReg

8.278 nas_NetSelPrefTlv Struct Reference

Data Fields

- uint8_t NetSelPref
- uint8_t TlvPresent

8.278.1 Detailed Description

Contain the Network selection Preference for system selection preferences.

Parameters

NetSelPref	• Parameter indicating network selection preference • Values: – 0x00 - Automatic network selection – 0x01 - Manual network selection
TlvPresent	• Tlv Present

8.278.2 Field Documentation

8.278.2.1 uint8_t nas_NetSelPrefTlv::NetSelPref

8.278.2.2 uint8_t nas_NetSelPrefTlv::TlvPresent

8.279 nas_NetworkNameSrcTlv Struct Reference

Data Fields

- uint8_t nw_name_src_len
- uint32_t nw_name_src [255]
- uint8_t TlvPresent

8.279.1 Detailed Description

Contain fields in struct NetworkNameSrcTlv

Parameters

nw_name_src_len	• Number of sets of the following elements: – nw_name_src
nw_name_src	• Network name source. • Values: – 0x00 - NAS_NW_NAME_SOURCE_UNKNOWN – 0x01 - NAS_NW_NAME_SOURCE_OPL_PNN – 0x02 - NAS_NW_NAME_SOURCE_CPHS_ONS – 0x03 - NAS_NW_NAME_SOURCE_NITZ – 0x04 - NAS_NW_NAME_SOURCE_SE13 – 0x05 - NAS_NW_NAME_SOURCE_MCC_MNC – 0x06 - NAS_NW_NAME_SOURCE_SPN
TlvPresent	• Tlv Present.

Note

None

8.279.2 Field Documentation

8.279.2.1 uint32_t nas_NetworkNameSrcTlv::nw_name_src[255]

8.279.2.2 uint8_t nas_NetworkNameSrcTlv::nw_name_src_len

8.279.2.3 uint8_t nas_NetworkNameSrcTlv::TlvPresent

8.280 nas_networkNameSrcTlv Struct Reference

Data Fields

- uint8_t nwNameSrcLen
- uint32_t nwNameSrc [255]
- uint8_t TlvPresent

8.280.1 Detailed Description

This structure contains the Network Name Source parameters.

Parameters

nwNameSrcLen	• Number of sets of the following elements – nwNameSrc
nwNameSrc	• Network name source. • Values – 0x00 - Unknown – 0x01 - Operator PLMN list and PLMN network name – 0x02 - Common PCN handset specification and operator name string – 0x03 - Network identity and time zone – 0x04 - GSMA SE13 table – 0x05 - Mobile country code and mobile network code – 0x06 - Service provider name
TlvPresent	• Tlv Present

8.280.2 Field Documentation

8.280.2.1 uint32_t nas_networkNameSrcTlv::nwNameSrc[255]

8.280.2.2 uint8_t nas_networkNameSrcTlv::nwNameSrcLen

8.280.2.3 uint8_t nas_networkNameSrcTlv::TlvPresent

8.281 nas_NetworkScan3GPPInfo Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t network_status
- uint8_t rat
- uint8_t mnc_includes_pcs_digit
- uint8_t network_desc_len
- uint8_t network_desc [255]

8.281.1 Detailed Description

Contain fields in struct nas_NetworkScan3GPPInfo

Parameters

mcc	• A 16-bit integer representation of MCC. Range: 0 to 999.
mnc	• A 16-bit integer representation of MCC. Range: 0 to 999.
network_status	• Status of the network identified by MCC and MNC. • Values – Bits 0-1 - QMI_NAS_NETWORK_IN_USE_STATUS_BITS * 0 - QMI_NAS_NETWORK_IN_USE_STATUS_UNKNOWN * 1 - QMI_NAS_NETWORK_IN_USE_STATUS_CURRENT_SERVING * 2 - QMI_NAS_NETWORK_IN_USE_STATUS_AVAILABLE – Bits 2-3 - QMI_NAS_NETWORK_ROAMING_STATUS_BITS * 0 - QMI_NAS_NETWORK_ROAMING_STATUS_UNKNOWN * 1 - QMI_NAS_NETWORK_ROAMING_STATUS_HOME * 2 - QMI_NAS_NETWORK_ROAMING_STATUS_ROAM – Bits 4-5 - QMI_NAS_NETWORK_FORBIDDEN_STATUS_BITS * 0 - QMI_NAS_NETWORK_FORBIDDEN_STATUS_UNKNOWN * 1 - QMI_NAS_NETWORK_FORBIDDEN_STATUS_FORBIDDEN * 2 - QMI_NAS_NETWORK_FORBIDDEN_STATUS_NOT_FORBIDDEN – Bits 6-7 - QMI_NAS_NETWORK_PREFERRED_STATUS_BITS * 0 - QMI_NAS_NETWORK_PREFERRED_STATUS_UNKNOWN * 1 - QMI_NAS_NETWORK_PREFERRED_STATUS_PREFERRED * 2 - QMI_NAS_NETWORK_PREFERRED_STATUS_NOT_PREFERRED
rat	• Radio access technology. Values: – 0x04 - GERAN – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA

mnc_inlcudes_pcs_digit	• This field is used to interpret the length of the corresponding MNC reported in the TLVs. Values: – TRUE – FALSE
network_desc_len	• Number of sets of the following elements: – network_description
network_desc	• Optional string containing network name or description.

Note

None

8.281.2 Field Documentation

8.281.2.1 uint16_t nas_NetworkScan3GPPInfo::mcc

8.281.2.2 uint16_t nas_NetworkScan3GPPInfo::mnc

8.281.2.3 uint8_t nas_NetworkScan3GPPInfo::mnc_includes_pcs_digit

8.281.2.4 uint8_t nas_NetworkScan3GPPInfo::network_desc[255]

8.281.2.5 uint8_t nas_NetworkScan3GPPInfo::network_desc_len

8.281.2.6 uint8_t nas_NetworkScan3GPPInfo::network_status

8.281.2.7 uint8_t nas_NetworkScan3GPPInfo::rat

8.282 nas_NetworkScan3GPPInfoTlv Struct Reference

Data Fields

- uint16_t num_inst
- nas_NetworkScan3GPPInfo netScan3GPPInfo [255]
- uint8_t TlvPresent

8.282.1 Detailed Description

Contain fields in struct 3GPPNetworkScanInfoTlv

Parameters

num_inst	• Number of sets of the following elements: – mobile_country_code – mobile_network_code – network_status – rat – mnc_icludes_pcs_digit – network_desc_len – network_desc
netScan3GPP-Info	• See nas_NetworkScan3GPPInfo for more information.
TlvPresent	• Tlv Present.

Note

None

8.282.2 Field Documentation

8.282.2.1 nas_NetworkScan3GPPInfo nas_NetworkScan3GPPInfoTlv::netScan3GPPInfo[255]

8.282.2.2 uint16_t nas_NetworkScan3GPPInfoTlv::num_inst

8.282.2.3 uint8_t nas_NetworkScan3GPPInfoTlv::TlvPresent

8.283 nas_NetworkScanStatusTlv Struct Reference

Data Fields

- uint32_t scanStatus
- uint8_t TlvPresent

8.283.1 Detailed Description

Contain fields in struct NetworkScanStatusTlv

Parameters

scanStatus	• Indicates the status of the network scan. • Values – 0x00 - NAS_SCAN_STATUS_COMPLETE – 0x01 - NAS_SCAN_STATUS_PARTIAL – 0x02 - NAS_SCAN_STATUS_ABORT – 0x03 - NAS_SCAN_STATUS_REJ_IN_RLF – 0x04 - NAS_SCAN_STATUS_INCREMENT_ERROR – 0x05 - NAS_SCAN_STATUS_PARTIAL_PERIODIC

TlvPresent	• Tlv Present.

Note

None

8.283.2 Field Documentation

8.283.2.1 uint32_t nas_NetworkScanStatusTlv::scanStatus

8.283.2.2 uint8_t nas_NetworkScanStatusTlv::TlvPresent

8.284 nas_NetworkStat1x Struct Reference

Data Fields

- uint8_t State
- uint16_t SO
- uint32_t RX_PWR
- uint16_t RX_EC_IO
- uint32_t TX_PWR
- uint8_t ActSetCnt
- nas_ActPilotPNElement * pActPilotPNElements
- uint8_t NeighborSetCnt
- uint16_t * pNeighborSetPilotPN

8.284.1 Detailed Description

This structure contains network stat 1x.

Parameters

State	• CDMA current access state – 0x01 - Searching – 0x02 - Idle – 0x03 - Traffic – Others - NA
SO	• CDMA service option – 0xFFFF - Not in a call – 0x0001 - Basic Variable Rate Voice Service(8kbps) – 0x0002 - Mobile Station Loopback(8kbps) – 0x0003 - Enhanced Variable Rate Codec(EVRC) Voice Service(8kbps) – 0x0006 - Short message Services(Rate Set 1) – 0x0009 - Mobile Station Loopback(13kbps) – 0x000E - Short Message Service (Rate Set 2) – 0x0011 - High Rate Voice Service(13kbps) – 0x0020 - Test Data Service Option(TDSO) – 0x0021 - cdma2000 High Speed Packet Data Service, Internet or ISO Protocol Stack – 0x0044 - EVRC-B Voice Service(8 kbps) – 0x0046 - EVRC-WB Voice Service(8 kbps) – 0x0049 - Voice Echo mode supports smart blanking(EVRC-NW) – 0x004B - Enhanced loopback – 0x8000 - Proprietary Service Option (Qualcomm Inc.)
RX_PWR	• RX Pwr(dBm) – 0xABCD00EF - -ABCD.EF dBm – ABCD00EF should be transferred to decimal while displaying – Example: 0x12340056 - -4660.86dBm 0x1234 = 4660, 0x0056 = 86 – 0xFFFFFFFF - NA
RX_EC_IO	• RX EC/IO(dB) – 0xABCD - -AB.CD dB – ABCD should be transferred to decimal while displaying – Example: 0x1234 - -18.52dB 0x12 = 18, 0x34 = 52 – 0xFFFF - NA
TX_PWR	• TX PWR(dBm) – 0xFFFFFFFF - NA – Others - display actual value in decimal – Example: 0x1234 - -4660dBm 0x1234 = 4660

ActSetCnt	• Count of active pilot PN elements • As input specifies number of sets of parameter pActPilotElements for which memory has been assigned • As output specifies the actual number of sets of parameter pActPilotElements returned by device
pActPilotPN-Elements	• See nas_ActPilotPNElement for more information
NeighborSetCnt	• Count of neighbor pilot PN elements • As input specifies number of sets of parameter pNeighborSetPilotPN for which memory has been assigned • As output specifies the actual number of sets of parameter pNeighborSetPilotPN returned by device
pNeighborSet-PilotPN	• Neighbor pilot PN

8.284.2 Field Documentation

8.284.2.1 uint8_t nas_NetworkStat1x::ActSetCnt

8.284.2.2 uint8_t nas_NetworkStat1x::NeighborSetCnt

8.284.2.3 nas_ActPilotPNElement* nas_NetworkStat1x::pActPilotPNElements

8.284.2.4 uint16_t* nas_NetworkStat1x::pNeighborSetPilotPN

8.284.2.5 uint16_t nas_NetworkStat1x::RX_EC_IO

8.284.2.6 uint32_t nas_NetworkStat1x::RX_PWR

8.284.2.7 uint16_t nas_NetworkStat1x::SO

8.284.2.8 uint8_t nas_NetworkStat1x::State

8.284.2.9 uint32_t nas_NetworkStat1x::TX_PWR

8.285 nas_NetworkStatEVDO Struct Reference

Data Fields

- uint8_t State
- uint8_t MACIndex
- uint8_t SectorIDLen
- uint16_t * pSectorID
- uint16_t RX_PWR
- uint16_t PER
- uint16_t PilotEnergy
- uint8_t SNR

8.285.1 Detailed Description

This structure contains network stat EVDO.

Parameters

State	• EVDO network access state – 0x00 - Sleep – 0x01 - Searching – 0x02 - Idle – 0x03 - Active – 0xFF - NA
MACIndex	• HDR Mac index – 0xFF - NA – Others - Display the actual value in decimal – Example: 0x12 - 18 0x12 = 18
SectorIDLen	(IN/OUT) • Sector ID length • As input specifies length of parameter pSectorID for which memory has been assigned • As output specifies the actual length of parameter pSectorID returned by device
pSectorID	• Sector ID
RX_PWR	• TX PWR(dBm) – 0xABCD - -ABCD dBm – ABCD should be transferred to decimal while displaying – Example: 0x1234 - -4660dBm 0x1234 = 4660 – 0xFFFF - NA
PER	• HDR Packet Error Rate – 0xFFFF - Unknown – Others - display the actual value in decimal – Example: 0x1234 - -4660dBm 0x1234 = 4660
PilotEnergy	• Pilt Energy (dB) – 0xFFFF - NA – 0xABCD should be transferred to decimal while displaying – Example: 0x1234 - -4660dBm 0x1234 = 4660
SNR	• Signal to Noise ratio (dB)

8.285.2 Field Documentation

8.285.2.1 uint8_t nas_NetworkStatEVDO::MACIndex

8.285.2.2 uint16_t nas_NetworkStatEVDO::PER

8.285.2.3 uint16_t nas_NetworkStatEVDO::PilotEnergy

8.285.2.4 uint16_t* nas_NetworkStatEVDO::pSectorID

8.285.2.5 uint16_t nas_NetworkStatEVDO::RX_PWR

8.285.2.6 uint8_t nas_NetworkStatEVDO::SectorIDLen

8.285.2.7 uint8_t nas_NetworkStatEVDO::SNR

8.285.2.8 uint8_t nas_NetworkStatEVDO::State

8.286 nas_nmrCellInfo Struct Reference

Data Fields

- uint32_t nmrCellID
- uint8_t nmrPlmn [3]
- uint16_t nmrLac
- uint16_t nmrArfcn
- uint8_t nmrBsic
- uint16_t nmrRxLev

8.286.1 Detailed Description

This structure contains information about the Network Measurement Report (NMR) Cell Information.

Parameters

nmrCellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
nmrPlmn[NAS_PLMN_LENGTH]	• MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when nmrCellID is not present.
nmrLac	• Location area code. • This field is ignored when nmrCellID is not present. – 0xFFFF - Not Available
nmrArfcn	• Absolute RF channel number. – 0xFFFF - Not Available

nmrBsic	• Base station identity code. – 0xFF - Not Available
nmrRxLev	• Cell Rx measurement. • Values range between 0 and 63. • Mapped to a measured signal level: – Rxlev 0 is a signal strength less than -110 dBm – Rxlev 1 is -110 dBm to -109 dBm – Rxlev 2 is -109 dBm to -108 dBm – ... – Rxlev 62 is -49 dBm to -48 dBm – Rxlev 63 is greater than -48 dBm – 0xFFFF - Not Available

8.286.2 Field Documentation

8.286.2.1 uint16_t nas_nmrCellInfo::nmrArfcn

8.286.2.2 uint8_t nas_nmrCellInfo::nmrBsic

8.286.2.3 uint32_t nas_nmrCellInfo::nmrCellID

8.286.2.4 uint16_t nas_nmrCellInfo::nmrLac

8.286.2.5 uint8_t nas_nmrCellInfo::nmrPlmn[3]

8.286.2.6 uint16_t nas_nmrCellInfo::nmrRxLev

8.287 nas_nr5gBandPref Struct Reference

Data Fields

- uint64_t bits_1_64
- uint64_t bits_65_128
- uint64_t bits_129_192
- uint64_t bits_193_256

8.287.1 Detailed Description

Contain the NR5G Band Preference.

Parameters

bits_1_64	• Bits 1 to 64 of the 256-bit NR5G Operating Band bitmask.
bits_65_128	• Bits 65 to 128 of the 256-bit NR5G Operating Band bitmask.

bits_129_192	• Bits 129 to 192 of the 256-bit NR5G Operating Band bitmask.
bits_193_256	• Bits 193 to 256 of the 256-bit NR5G Operating Band bitmask.

8.287.2 Field Documentation

8.287.2.1 uint64_t nas_nr5gBandPref::bits_129_192

8.287.2.2 uint64_t nas_nr5gBandPref::bits_193_256

8.287.2.3 uint64_t nas_nr5gBandPref::bits_1_64

8.287.2.4 uint64_t nas_nr5gBandPref::bits_65_128

8.288 nas_NR5GCellStatusInfoTlv Struct Reference

Data Fields

- uint32_t nr5gCellStatus
- uint8_t TLVPresent

8.288.1 Detailed Description

Structure for storing the NR5G Cell Access Status Info

Parameters

nr5gCellStatus	• Cell access status for NR5G calls. – Values – 0x00 - Cell access is allowed for normal calls only – 0x01 - Cell access is allowed for emergency calls only – 0x02 - Cell access is not allowed for any call type – 0x03 - Cell access is allowed for all call types – -1 - Cell access type is unknown
TLVPresent	• TLV Present

8.288.2 Field Documentation

8.288.2.1 uint32_t nas_NR5GCellStatusInfoTlv::nr5gCellStatus

8.288.2.2 uint8_t nas_NR5GCellStatusInfoTlv::TLVPresent

8.289 nas_NR5GCellStatusTlv Struct Reference

Data Fields

- uint32_t nr5gCellStatus
- uint8_t TlvPresent

8.289.1 Detailed Description

Structure for storing the NR5G Cell Access Status Info parameters

Parameters

nr5gCellStatus	• Cell access status for NR5G calls. – Values: * NAS_CELL_ACCESS_NORMAL_ ONLY (0x00) - Cell access is allowed for normal calls only * NAS_CELL_ACCESS_EMERGENCY_ ONLY (0x01) - Cell access is allowed for emergency calls only * NAS_CELL_ACCESS_NO_CALLS (0x02) - Cell access is not allowed for any call type * NAS_CELL_ACCESS_ALL_CALLS (0x03) - Cell access is allowed for all call types * NAS_CELL_ACCESS_UNKNOWN (-1) - Cell access type is unknown
TlvPresent	• Tlv Present or not

8.289.2 Field Documentation

8.289.2.1 uint32_t nas_NR5GCellStatusTlv::nr5gCellStatus

8.289.2.2 uint8_t nas_NR5GCellStatusTlv::TlvPresent

8.290 nas_NR5GSerStatTlv Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t trueSrvStatus
- uint8_t isPrefDataPath
- uint8_t TlvPresent

8.290.1 Detailed Description

This structure contains NR5G Service Status Info parameters.

Parameters

srvStatus	• Service status of the system. • Values: – 0x00 - SYS_SRV_STATUS_NO_SRV - No service – 0x01 - SYS_SRV_STATUS_LIMITED - Limited service – 0x02 - SYS_SRV_STATUS_SRV - Service – 0x03 - SYS_SRV_STATUS_LIMITED_REGIONAL - Limited regional service – 0x04 - SYS_SRV_STATUS_PWR_SAVE - Power save
trueSrvStatus	• True service status of the system (not applicable to CDMA/HDR). • Values: – 0x00 - SYS_SRV_STATUS_NO_SRV - No service – 0x01 - SYS_SRV_STATUS_LIMITED - Limited service – 0x02 - SYS_SRV_STATUS_SRV - Service – 0x03 - SYS_SRV_STATUS_LIMITED_REGIONAL - Limited regional service – 0x04 - SYS_SRV_STATUS_PWR_SAVE - Power save
isPrefDataPath	• Whether the RAT is the preferred data path: • Values – 0x00 - Not preferred – 0x01 - Preferred
TlvPresent	• Tlv Present or not

8.290.2 Field Documentation

8.290.2.1 uint8_t nas_NR5GSerStatTlv::isPrefDataPath

8.290.2.2 uint8_t nas_NR5GSerStatTlv::srvStatus

8.290.2.3 uint8_t nas_NR5GSerStatTlv::TlvPresent

8.290.2.4 uint8_t nas_NR5GSerStatTlv::trueSrvStatus

8.291 nas_NR5GSrvStatusTlv Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t trueSrvStatus
- uint8_t isPrefDataPath
- uint8_t TLVPresent

8.291.1 Detailed Description

Structure for storing the NR5G Service Status Info.

Parameters

srvStatus	• Service status of the system. • Values – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save
trueSrvStatus	• True service status of the system (not applicable to CDMA/HDR). • Values • 0x00 - No service • 0x01 - Limited service • 0x02 - Service • 0x03 - Limited regional service • 0x04 - Power save
isPrefDataPath	• Whether the RAT is the preferred data path • Values – 0x00 - Not preferred – 0x01 - Preferred
TLVPresent	• TLV Present

8.291.2 Field Documentation

8.291.2.1 uint8_t nas_NR5GSrvStatusTlv::isPrefDataPath

8.291.2.2 uint8_t nas_NR5GSrvStatusTlv::srvStatus

8.291.2.3 uint8_t nas_NR5GSrvStatusTlv::TLVPresent

8.291.2.4 uint8_t nas_NR5GSrvStatusTlv::trueSrvStatus

8.292 nas_NR5GSysInfoTlv Struct Reference

Data Fields

- uint8_t srvDomainValid
- uint8_t srvDomain
- uint8_t srvCapabilityValid
- uint8_t srvCapability
- uint8_t roamStatusValid
- uint8_t roamStatus
- uint8_t isSysForbiddenValid
- uint8_t isSysForbidden
- uint8_t lacValid

- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejectCause
- uint8_t nwIdValid
- uint8_t mcc [3]
- uint8_t mnc [3]
- uint8_t tacValid
- uint16_t tac
- uint8_t TLVPresent

8.292.1 Detailed Description

Structure for storing the NR5G System Info.

Parameters

srvDomainValid	• Indicates whether the service domain is valid.
srvDomain	• Service domain registered on the system. – Values * 0x00 - No service * 0x01 - Circuit-switched only * 0x02 - Packet-switched only * 0x03 - Circuit-switched and packet-switched * 0x04 - Camped
srvCapability-Valid	• Indicates whether the service capability is valid.
srvCapability	• Current system's service capability. • Values – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped
roamStatusValid	• Indicates whether the roaming status is valid.

roamStatus	• Current roaming status. • Values • 0x00 - Off • 0x01 - On • 0x02 - Blinking • 0x03 - Out of the neighborhood • 0x04 - Out of the building • 0x05 - Preferred system • 0x06 - Available system • 0x07 - Alliance partner • 0x08 - Premium partner • 0x09 - Full service • 0x0A - Partial service • 0x0B - Banner is on • 0x0C - Banner is off • Remainder of the values are per 3GPP2 C.R1001-F. Values from 0x02 onward are only applicable for 3GPP2.
isSysForbidden-Valid	• Indicates whether the forbidden system is valid.
isSysForbidden	• Whether the system is forbidden – 0x00 - Not forbidden – 0x01 - Forbidden
lacValid	• Indicates whether the location area code is valid.
lac	• Location area code (only applicable for 3GPP).
cellIdValid	• Indicates whether the cell ID is valid.
cellId	• Cell ID.
regRejectInfo-Valid	• Indicates whether the registration reject information is valid.
rejectSrvDomain	• Type of service domain in which the registration is rejected. • Values – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped

rejectCause	• Reject cause values sent are specified in 3GPP TS 24.008 Sections 10.5.3.6 and 10.5.5.14, and 3GPP TS 24.301 Section 9.9.3.9.
nwIdValid	• Indicates whether the network ID is valid.
mcc	• MCC digits in ASCII characters. • For CDMA, the MCC wildcard value is returned as {'3', 0xFF, 0xFF}.
mnc	• MNC digits in ASCII characters. • For this field – Unused byte is set to 0xFF – In the case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x35 0x31 0xFF. – For CDMA, the MNC wildcard value is returned as '7', 0xFF, 0xFF}.
tacValid	• Indicates whether the tracking area code is valid.
tac	• Tracking area code (only applicable for NR5G).
TLVPresent	• TLV Present

8.292.2 Field Documentation

8.292.2.1 uint32_t nas_NR5GSysInfoTlv::cellId

8.292.2.2 uint8_t nas_NR5GSysInfoTlv::cellIdValid

8.292.2.3 uint8_t nas_NR5GSysInfoTlv::isSysForbidden

8.292.2.4 uint8_t nas_NR5GSysInfoTlv::isSysForbiddenValid

8.292.2.5 uint16_t nas_NR5GSysInfoTlv::lac

8.292.2.6 uint8_t nas_NR5GSysInfoTlv::lacValid

8.292.2.7 uint8_t nas_NR5GSysInfoTlv::mcc[3]

8.292.2.8 uint8_t nas_NR5GSysInfoTlv::mnc[3]

8.292.2.9 uint8_t nas_NR5GSysInfoTlv::nwIdValid

8.292.2.10 uint8_t nas_NR5GSysInfoTlv::regRejectInfoValid

8.292.2.11 uint8_t nas_NR5GSysInfoTlv::rejectCause

8.292.2.12 uint8_t nas_NR5GSysInfoTlv::rejectSrvDomain

8.292.2.13 uint8_t nas_NR5GSysInfoTlv::roamStatus

8.292.2.14 uint8_t nas_NR5GSysInfoTlv::roamStatusValid

8.292.2.15 uint8_t nas_NR5GSysInfoTlv::srvCapability

8.292.2.16 uint8_t nas_NR5GSysInfoTlv::srvCapabilityValid

8.292.2.17 uint8_t nas_NR5GSysInfoTlv::srvDomain

8.292.2.18 uint8_t nas_NR5GSysInfoTlv::srvDomainValid

8.292.2.19 uint16_t nas_NR5GSysInfoTlv::tac

8.292.2.20 uint8_t nas_NR5GSysInfoTlv::tacValid

8.292.2.21 uint8_t nas_NR5GSysInfoTlv::TLVPresent

8.293 nas_NR5GSystemInfoTlv Struct Reference

Data Fields

- uint8_t srvDomainValid
- uint8_t srvDomain
- uint8_t srvCapValid
- uint8_t srvcapability
- uint8_t roamStatusValid
- uint8_t roamStatus
- uint8_t sysForbiddenValid
- uint8_t sysForbidden
- uint8_t lacValid
- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejCause
- uint8_t nwIdValid
- int8_t MCC [3]
- int8_t MNC [3]
- uint8_t tacValid
- uint16_t tac
- uint8_t TlvPresent

8.293.1 Detailed Description

This structure contains NR5G System Info parameters.

Parameters

srvDomainValid	• Indicates whether the service domain is valid.
srvDomain	• Service domain registered on the system. – Values: * 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service * 0x01 - SYS_SRV_DOMAIN_CS_ONLY - Circuit-switched only * 0x02 - SYS_SRV_DOMAIN_PS_ONLY - Packet-switched only * 0x03 - SYS_SRV_DOMAIN_CS_PS - Circuit-switched and packet-switched * 0x04 - SYS_SRV_DOMAIN_CAMPED -Camped
srvCapValid	• Indicates whether the service capability is valid.
srvCapability	• Current system service capability. – Values: * 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service * 0x01 - SYS_SRV_DOMAIN_CS_ONLY - Circuit-switched only * 0x02 - SYS_SRV_DOMAIN_PS_ONLY - Packet-switched only * 0x03 - SYS_SRV_DOMAIN_CS_PS - Circuit-switched and packet-switched * 0x04 - SYS_SRV_DOMAIN_CAMPED - Camped
roamStatusValid	• Indicates whether the roaming status is valid.
roamStatus	• Current roaming status. – Values: * 0x00 - SYS_ROAM_STATUS_OFF - Off * 0x01 - SYS_ROAM_STATUS_ON - On * 0x02 - SYS_ROAM_STATUS_BLINK - Blinking * 0x03 - SYS_ROAM_STATUS_OUT_OF_NEIGHBORHOOD - Out of the neighborhood * 0x04 - SYS_ROAM_STATUS_OUT_OF_BLDG - Out of the building * 0x05 - SYS_ROAM_STATUS_PREF_SYS - Preferred system * 0x06 - SYS_ROAM_STATUS_AVAIL_SYS - Available system * 0x07 - SYS_ROAM_STATUS_ALLIANCE_PARTNER - Alliance partner * 0x08 - SYS_ROAM_STATUS_PREMIUM_PARTNER - Premium partner * 0x09 - SYS_ROAM_STATUS_FULL_SVC - Full service * 0x0A - SYS_ROAM_STATUS_PARTIAL_SVC -Partial service * 0x0B - SYS_ROAM_STATUS_BANNER_ON -Banner is on * 0x0C - SYS_ROAM_STATUS_BANNER_OFF -Banner is off Remainder of the values are per 3GPP2 C.R1001-F. Values from 0x02 onward are only applicable for 3GPP2.
sysForbidden-Valid	• Indicates whether the forbidden system is valid.

sysForbidden	• Whether the system is forbidden: • 0x00 Not forbidden • 0x01 Forbidden
lacValid	• Indicates whether the location area code is valid.
lac	• Location area code (only applicable for 3GPP).
cellIdValid	• Indicates whether the cell ID is valid.
cellId	• Cell ID.
rejectSrvDomain	• Type of service domain in which the registration is rejected. • Values: • 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service • 0x01 - SYS_SRV_DOMAIN_CS_ONLY - Circuit-switched only • 0x02 - SYS_SRV_DOMAIN_PS_ONLY - Packet-switched only • 0x03 - SYS_SRV_DOMAIN_CS_PS - Circuit-switched and packet-switched • 0x04 - SYS_SRV_DOMAIN_CAMPED - Camped
rejCause	• Reject cause values sent are specified in 3GPP TS • 24.008 Sections 10.5.3.6 and 10.5.5.14, and 3GPP • TS 24.301 Section 9.9.3.9.
nwIdValid	• Indicates whether the network ID is valid.
MCC	• MCC digits in ASCII characters.For CDMA, the MCC wildcard value is returned as{3, 0xFF, 0xFF}..
MNC	• MNC digits in ASCII characters. For this field: • Unused byte is set to 0xFF • In the case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using thebyte stream 0x35 0x31 0xFF.For CDMA, the MNC wildcard value is returned as {7, 0xFF, 0xFF}.
tacValid	• Indicates whether the tracking area code is valid.
tac	• Tracking area code (only applicable for NR5G).

TlvPresent	• Tlv Present or not

8.293.2 Field Documentation

8.293.2.1 uint32_t nas_NR5GSystemInfoTlv::cellId

8.293.2.2 uint8_t nas_NR5GSystemInfoTlv::cellIdValid

8.293.2.3 uint16_t nas_NR5GSystemInfoTlv::lac

8.293.2.4 uint8_t nas_NR5GSystemInfoTlv::lacValid

8.293.2.5 int8_t nas_NR5GSystemInfoTlv::MCC[3]

8.293.2.6 int8_t nas_NR5GSystemInfoTlv::MNC[3]

8.293.2.7 uint8_t nas_NR5GSystemInfoTlv::nwIdValid

8.293.2.8 uint8_t nas_NR5GSystemInfoTlv::regRejectInfoValid

8.293.2.9 uint8_t nas_NR5GSystemInfoTlv::rejCause

8.293.2.10 uint8_t nas_NR5GSystemInfoTlv::rejectSrvDomain

8.293.2.11 uint8_t nas_NR5GSystemInfoTlv::roamStatus

8.293.2.12 uint8_t nas_NR5GSystemInfoTlv::roamStatusValid

8.293.2.13 uint8_t nas_NR5GSystemInfoTlv::srvcapability

8.293.2.14 uint8_t nas_NR5GSystemInfoTlv::srvCapValid

8.293.2.15 uint8_t nas_NR5GSystemInfoTlv::srvDomain

8.293.2.16 uint8_t nas_NR5GSystemInfoTlv::srvDomainValid

8.293.2.17 uint8_t nas_NR5GSystemInfoTlv::sysForbidden

8.293.2.18 uint8_t nas_NR5GSystemInfoTlv::sysForbiddenValid

8.293.2.19 uint16_t nas_NR5GSystemInfoTlv::tac

8.293.2.20 uint8_t nas_NR5GSystemInfoTlv::tacValid

8.293.2.21 uint8_t nas_NR5GSystemInfoTlv::TlvPresent

8.294 nas_NumScellsConfig Struct Reference

Data Fields

- uint16_t numScellsCfg
- uint8_t TlvPresent

8.294.1 Detailed Description

This structure contains the parameters for Number of Scells Configured

Parameters

numScellsCfg	• Number of Scells Configured.
TlvPresent	• Tlv Present.

8.294.2 Field Documentation

8.294.2.1 uint16_t nas_NumScellsConfig::numScellsCfg

8.294.2.2 uint8_t nas_NumScellsConfig::TlvPresent

8.295 nas_nwNameSrc3GppTlv Struct Reference

Data Fields

- uint32_t NwNameSrc3Gpp
- uint8_t TlvPresent

8.295.1 Detailed Description

This structure contains the 3GPP Network Name Source parameters.

Parameters

NwNameSrc3-Gpp[OUT]	• 3GPP Network Name Source • Values – 0x00 - Unknown – 0x01 - Operator PLMN list and PLMN network name – 0x02 - Common PCN handset specification and operator name string – 0x03 - Network identity and time zone – 0x04 - GSMA SE13 table – 0x05 - Mobile country code and mobile network code – 0x06 - Service provider name
TlvPresent[OUT]	• Tlv Present

8.295.2 Field Documentation

8.295.2.1 uint32_t nas_nwNameSrc3GppTlv::NwNameSrc3Gpp

8.295.2.2 uint8_t nas_nwNameSrc3GppTlv::TlvPresent

8.296 nas_operatorNameString Struct Reference

Data Fields

- uint8_t PLMNName [255]

8.296.1 Detailed Description

This structure contains PLMN name.

Parameters

PLMNName	• PLMN name must be coded in a default 7-bit alphabet with b8 set to 0.

8.296.2 Field Documentation

8.296.2.1 uint8_t nas_operatorNameString::PLMNName[255]

8.297 nas_OperatorPLMNData Struct Reference

Data Fields

- uint8_t mcc [3]
- uint8_t mnc [3]
- uint16_t lac1
- uint16_t lac2
- uint8_t PLMNRecID

8.297.1 Detailed Description

This structure contains Operator PLMN.

Parameters

mcc	• MCC in ASCII string (a value of D in any of the digits is to be used to indicate a "wild" value for that corresponding digit).
mnc	• MNC in ASCII string (a value of D in any of the digits is to be used to indicate a "wild" value for that corresponding digit; digit 3 in MNC is optional and when not present, will be set as ASCII F).
lac1	• Location area code 1.
lac2	• Location area code 1.
PLMNRecID	• PLMN network name record identifier.

8.297.2 Field Documentation

8.297.2.1 uint16_t nas_OperatorPLMNData::lac1

8.297.2.2 uint16_t nas_OperatorPLMNData::lac2

8.297.2.3 uint8_t nas_OperatorPLMNData::mcc[3]

8.297.2.4 uint8_t nas_OperatorPLMNData::mnc[3]

8.297.2.5 uint8_t nas_OperatorPLMNData::PLMNRecID

8.298 nas_operatorPLMNList Struct Reference

Data Fields

- uint16_t numInstance
- nas_OperatorPLMNData PLMNData [255]

8.298.1 Detailed Description

This structure contains Operator PLMN List.

Parameters

numInstance	• Number of sets of the elements.
PLMNData	• Refer OperatorPLMNLData for details (Optional).

8.298.2 Field Documentation

8.298.2.1 uint16_t nas_operatorPLMNList::numInstance

8.298.2.2 nas_OperatorPLMNData nas_operatorPLMNList::PLMNData[255]

8.299 nas_PhyCaAggDlBW Struct Reference

Data Fields

- uint16_t aggDlBW
- uint8_t TlvPresent

8.299.1 Detailed Description

This structure contains the parameters for Aggregated DL Bandwidth.

Parameters

aggDlBW	• Aggregated DL bandwidth. • Units: MHz
TlvPresent	• Tlv Present.

8.299.2 Field Documentation

8.299.2.1 uint16_t nas_PhyCaAggDlBW::aggDlBW

8.299.2.2 uint8_t nas_PhyCaAggDlBW::TlvPresent

8.300 nas_PhyCaAggPcellInfo Struct Reference

Data Fields

- int pci
- int freq
- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- int iLTEbandValue
- uint8_t TlvPresent

8.300.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Pcell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
dl_bw_value	• Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.
TlvPresent	• Tlv Present.

8.300.2 Field Documentation

8.300.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggPcellInfo::dl_bw_value

8.300.2.2 int nas_PhyCaAggPcellInfo::freq

8.300.2.3 int nas_PhyCaAggPcellInfo::iLTEbandValue

8.300.2.4 int nas_PhyCaAggPcellInfo::pci

8.300.2.5 uint8_t nas_PhyCaAggPcellInfo::TlvPresent

8.301 nas_PhyCaAggScellDlBw Struct Reference

Data Fields

- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- uint8_t TlvPresent

8.301.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation Downlink Bandwidth of Scell.

Parameters

dl_bw_value	• Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.

8.301.2 Field Documentation

8.301.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellDlBw::dl_bw_value

8.301.2.2 uint8_t nas_PhyCaAggScellDlBw::TlvPresent

8.302 nas_PhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.302.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

scell_idx	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.

TlvPresent	• Tlv Present.

8.302.2 Field Documentation

8.302.2.1 uint8_t nas_PhyCaAggScellIndex::scell_idx

8.302.2.2 uint8_t nas_PhyCaAggScellIndex::TlvPresent

8.303 nas_PhyCaAggScellIndType Struct Reference

Data Fields

- int pci
- int freq
- NAS_LTE_CPHY_SCELL_STATE_LITE scell_state
- uint8_t TlvPresent

8.303.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Indicator Type.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.
TlvPresent	• Tlv Present.

8.303.2 Field Documentation

8.303.2.1 int nas_PhyCaAggScellIndType::freq

8.303.2.2 int nas_PhyCaAggScellIndType::pci

8.303.2.3 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellIndType::scell_state

8.303.2.4 uint8_t nas_PhyCaAggScellIndType::TlvPresent

8.304 nas_PhyCaAggScellInfo Struct Reference

Data Fields

- int pci
- int freq
- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- uint16_t iLTEbandValue
- NAS_LTE_CPHY_SCELL_STATE_LITE scell_state
- uint8_t TlvPresent

8.304.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
dl_bw_value	• Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.

iLTEbandValue	• Band value. • Range for LTE Band class 120 to 160. – 120 - LTE E-UTRA Operating Band 1 – 121 - LTE E-UTRA Operating Band 2 – 122 - LTE E-UTRA Operating Band 3 – 123 - LTE E-UTRA Operating Band 4 – 124 - LTE E-UTRA Operating Band 5 – 125 - LTE E-UTRA Operating Band 6 – 126 - LTE E-UTRA Operating Band 7 – 127 - LTE E-UTRA Operating Band 8 – 128 - LTE E-UTRA Operating Band 9 – 129 - LTE E-UTRA Operating Band 10 – 130 - LTE E-UTRA Operating Band 11 – 131 - LTE E-UTRA Operating Band 12 – 132 - LTE E-UTRA Operating Band 13 – 133 - LTE E-UTRA Operating Band 14 – 134 - LTE E-UTRA Operating Band 17 – 135 - LTE E-UTRA Operating Band 33 – 136 - LTE E-UTRA Operating Band 34 – 137 - LTE E-UTRA Operating Band 35 – 138 - LTE E-UTRA Operating Band 36 – 139 - LTE E-UTRA Operating Band 37 – 140 - LTE E-UTRA Operating Band 38 – 141 - LTE E-UTRA Operating Band 39 – 142 - LTE E-UTRA Operating Band 40 – 143 - LTE E-UTRA Operating Band 18 – 144 - LTE E-UTRA Operating Band 19 – 145 - LTE E-UTRA Operating Band 20 – 146 - LTE E-UTRA Operating Band 21 – 147 - LTE E-UTRA Operating Band 24 – 148 - LTE E-UTRA Operating Band 25 – 149 - LTE E-UTRA Operating Band 41 – 150 - LTE E-UTRA Operating Band 42 – 151 - LTE E-UTRA Operating Band 43 – 152 - LTE E-UTRA Operating Band 23 – 153 - LTE E-UTRA Operating Band 26 – 154 - LTE E-UTRA Operating Band 32 – 155 - LTE E-UTRA Operating Band 125 – 156 - LTE E-UTRA Operating Band 126 – 157 - LTE E-UTRA Operating Band 127 – 158 - LTE E-UTRA Operating Band 28 – 159 - LTE E-UTRA Operating Band 29 – 160 - LTE E-UTRA Operating Band 30
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.

TlvPresent	• Tlv Present.

8.304.2 Field Documentation

8.304.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellInfo::dl_bw_value

8.304.2.2 int nas_PhyCaAggScellInfo::freq

8.304.2.3 uint16_t nas_PhyCaAggScellInfo::iLTEbandValue

8.304.2.4 int nas_PhyCaAggScellInfo::pci

8.304.2.5 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellInfo::scell_state

8.304.2.6 uint8_t nas_PhyCaAggScellInfo::TlvPresent

8.305 nas_PilotSetData Struct Reference

Data Fields

- uint8_t NumPilots
- nas_PilotSetParams * pPilotSetInfo

8.305.1 Detailed Description

This structure contains pilot sets data.

Parameters

NumPilots	• Number of Pilot Sets • As input specifies number of sets of parameter pPilotSetInfo for which memory has been assigned • As output specifies the actual number of sets of parameter pPilotSetInfo returned by device
pPilotSetInfo	• Pilot Set Parameters • See nas_PilotSetParams for more information.

note A buffer under sized error is returned if the number of sets of pPilotSetInfo returned by the device is greater than the value in NumPilots input parameter.

8.305.2 Field Documentation

8.305.2.1 uint8_t nas_PilotSetData::NumPilots

8.305.2.2 nas_PilotSetParams* nas_PilotSetData::pPilotSetInfo

8.306 nas_PilotSetParams Struct Reference

Data Fields

- uint32_t PilotType
- uint16_t PilotPN
- uint16_t PilotStrength

8.306.1 Detailed Description

This structure contains pilot sets parameters.

Parameters

PilotType	• 0x00 - NAS_HRPD_PILOT_CURR_ACT_PLT Current Active Pilot • 0x01 - NAS_HRPD_PILOT_NEIGHBOR_PLT Neighbor pilot information
PilotPN	• Pilot PN sequence offset index
PilotStrength	• Strength of the pilot (in dB)

8.306.2 Field Documentation

8.306.2.1 uint16_t nas_PilotSetParams::PilotPN

8.306.2.2 uint16_t nas_PilotSetParams::PilotStrength

8.306.2.3 uint32_t nas_PilotSetParams::PilotType

8.307 nas_PlmnID Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t pcsDigit
- uint8_t TlvPresent

8.307.1 Detailed Description

This structure contains PLMN ID information

Parameters

mcc	• A 16-bit integer representation of MCC. Range: 0 to 999
mnc	• A 16-bit integer representation of MNC. Range: 0 to 999

pcsDigit	• This field is used to interpret the length of the corresponding MNC reported in the TLVs (in this table) with an mnc or mobile_network_code field. Values: – TRUE - MNC is a three-digit value; for example, a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a two-digit value; for example, a reported value of 90 corresponds to an MNC value of 90
TlvPresent	• Tlv Present or not

8.307.2 Field Documentation

8.307.2.1 uint16_t nas_PlmnID::mcc

8.307.2.2 uint16_t nas_PlmnID::mnc

8.307.2.3 uint8_t nas_PlmnID::pcsDigit

8.307.2.4 uint8_t nas_PlmnID::TlvPresent

8.308 nas_PLMNNetworkName Struct Reference

Data Fields

- uint8_t numInstance
- nas_PLMNNetworkNameData PLMNNetName [255]

8.308.1 Detailed Description

This structure contains NITZ Information list.

Parameters

numInstance	• Number of sets of the elements.
PLMNNetName	• Refer nas_PLMNNetworkNameData for details (Optional).

8.308.2 Field Documentation

8.308.2.1 uint8_t nas_PLMNNetworkName::numInstance

8.308.2.2 nas_PLMNNetworkNameData nas_PLMNNetworkName::PLMNNetName[255]

8.309 nas_PLMNNetworkNameData Struct Reference

Data Fields

- uint8_t codingScheme
- uint8_t countryInitials
- uint8_t longNameSpareBits
- uint8_t shortNameSpareBits
- uint8_t longNameLen
- uint8_t longName [255]
- uint8_t shortNameLen
- uint8_t shortName [255]

8.309.1 Detailed Description

This structure contains NITZ Information.

Parameters

codingScheme	• Coding scheme: – 0 - CODING_SCHEME_CELL_BROADCAST_GSM - Cell broadcast data coding scheme, GSM default alphabet, language unspecified;defined in 3GPP TS 23.038. – 1 - CODING_SCHEME_UCS2 - UCS2 (16 bit);defined in ISO/IEC 10646
countryInitials	• Country's initials: – 0 - COUNTRY_INITIALS_DO_NOT_ADD - MS should not add the letters for the country's initials to the text string. – 1 - COUNTRY_INITIALS_ADD - MS should add the letters for the country's initials and a separator, e.g., a space, to the text string.
longNameSpare-Bits	• Long Name Spare Bits: – 1 - SPARE_BITS_8 - Bit 8 is spare and set to 0 in octet n – 2 - SPARE_BITS_7_TO_8 - Bits 7 and 8 are spare and set to 0 in octet n. – 3 - SPARE_BITS_6_TO_8 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n. – 4 - SPARE_BITS_5_TO_8 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n. – 5 - SPARE_BITS_4_TO_8 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n. – 6 - SPARE_BITS_3_TO_8 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n. – 7 - SPARE_BITS_2_TO_8 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n. – 0 - SPARE_BITS_UNKNOWN - Carries no information about the number of spare bits in octet n.
shortName-SpareBits	• Short Name Spare Bits: – 1 - SPARE_BITS_8 - Bit 8 is spare and set to 0 in octet n. – 2 - SPARE_BITS_7_TO_8 - Bits 7 and 8 are spare and set to 0 in octet n. – 3 - SPARE_BITS_6_TO_8 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n. – 4 - SPARE_BITS_5_TO_8 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n. – 5 - SPARE_BITS_4_TO_8 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n. – 6 - SPARE_BITS_3_TO_8 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n. – 7 - SPARE_BITS_2_TO_8 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n. – 0 - SPARE_BITS_UNKNOWN - Carries no information about the number of spare bits in octet n.

longNameLen	• It provides the length of long name.
longName	• Long name string in coding_scheme.
shortNameLen	• It provides the length of short name.
shortName	• Short name string in coding_scheme.

8.309.2 Field Documentation

8.309.2.1 uint8_t nas_PLMNNetworkNameData::codingScheme

8.309.2.2 uint8_t nas_PLMNNetworkNameData::countryInitials

8.309.2.3 uint8_t nas_PLMNNetworkNameData::longName[255]

8.309.2.4 uint8_t nas_PLMNNetworkNameData::longNameLen

8.309.2.5 uint8_t nas_PLMNNetworkNameData::longNameSpareBits

8.309.2.6 uint8_t nas_PLMNNetworkNameData::shortName[255]

8.309.2.7 uint8_t nas_PLMNNetworkNameData::shortNameLen

8.309.2.8 uint8_t nas_PLMNNetworkNameData::shortNameSpareBits

8.310 nas_PrefPLMNListTlv Struct Reference

Data Fields

- uint8_t plmnListLen
- uint16_t mcc [255]
- uint16_t mnc [255]
- uint8_t mnc_pcs_digit [255]

8.310.1 Detailed Description

This structure contains the parameters for preferred PLMN list.

Parameters

PLMNListLen	• Preffered PLMN list provided to the modem. Number of sets of the following elements: – mcc – mnc – mnc_includes_pcs_digit

mcc	• mobile country code – A 16 bit representation of MCC – Range 0 to 999
mnc	• mobile network code – A 16 bit representation of MNC – Range 0 to 999
mnc_pcs_digit	• This field is used to interpret the length of the corresponding MNC reported in the TLV present. Vaues: – TRUE – FALSE

8.310.2 Field Documentation

8.310.2.1 uint16_t nas_PrefPLMNListTlv::mcc[255]

8.310.2.2 uint16_t nas_PrefPLMNListTlv::mnc[255]

8.310.2.3 uint8_t nas_PrefPLMNListTlv::mnc_pcs_digit[255]

8.310.2.4 uint8_t nas_PrefPLMNListTlv::plmnListLen

8.311 nas_PRLPrefTlv Struct Reference

Data Fields

- uint16_t PRLPref
- uint8_t TlvPresent

8.311.1 Detailed Description

Contain the PRL preference for system selection preferences.

Parameters

PRLPref	• Parameter indicating the CDMA PRL Preference • Values: – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems
TlvPresent	• Tlv Present

8.311.2 Field Documentation

8.311.2.1 uint16_t nas_PRLPrefTlv::PRLPref

8.311.2.2 uint8_t nas_PRLPrefTlv::TlvPresent

8.312 nas_protocolSubtypeElement Struct Reference

Data Fields

- uint16_t PhysicalLayer
- uint16_t ControlMac
- uint16_t AccessMac
- uint16_t ForwardMac
- uint16_t ReverseMac
- uint16_t KeyExchange
- uint16_t AuthProt
- uint16_t EncryptProt
- uint16_t SecProt
- uint16_t IdleState
- uint16_t MultDisc
- uint16_t VirtStream

8.312.1 Detailed Description

This structure contains protocol subtype element.

Parameters

PhysicalLayer	• Specifies Physical Layer Protocol subtype
ControlMac	• Specifies Control Channel MAC Protocol subtype
AccessMac	• Specifies Access Channel MAC Protocol subtype
ForwardMac	• Specifies Forward Traffic Channel MAC Protocol subtype
ReverseMac	• Specifies Reverse Traffic Channel MAC Protocol subtype
KeyExchange	• Specifies Key exchange Protocol subtype
AuthProt	• Specifies Authentication Protocol subtype
EncryptProt	• Specifies Encryption Protocol subtype

SecProt	• Specifies Security Protocol subtype
IdleState	• Specifies Idle state Protocol subtype
MultDisc	• Specifies Generic multimode capability discovery Protocol subtype
VirtStream	• Specifies Generic Virtual Stream Protocol subtype

8.312.2 Field Documentation

8.312.2.1 uint16_t nas_protocolSubtypeElement::AccessMac

8.312.2.2 uint16_t nas_protocolSubtypeElement::AuthProt

8.312.2.3 uint16_t nas_protocolSubtypeElement::ControlMac

8.312.2.4 uint16_t nas_protocolSubtypeElement::EncryptProt

8.312.2.5 uint16_t nas_protocolSubtypeElement::ForwardMac

8.312.2.6 uint16_t nas_protocolSubtypeElement::IdleState

8.312.2.7 uint16_t nas_protocolSubtypeElement::KeyExchange

8.312.2.8 uint16_t nas_protocolSubtypeElement::MultDisc

8.312.2.9 uint16_t nas_protocolSubtypeElement::PhysicalLayer

8.312.2.10 uint16_t nas_protocolSubtypeElement::ReverseMac

8.312.2.11 uint16_t nas_protocolSubtypeElement::SecProt

8.312.2.12 uint16_t nas_protocolSubtypeElement::VirtStream

8.313 nas_qaQmi3Gpp2TimeZone Struct Reference

Data Fields

- uint8_t leapSeconds
- uint8_t localTimeOffset
- uint8_t daylightSavings

8.313.1 Detailed Description

This structure contains the 3GPP2TimeZone parameters

Parameters

leapSeconds	• leap seconds - Number of leap seconds since the start of CDMA system time.
localTimeOffset	• Local Time Offset - Offset of system time in units of 30 minutes; the value in this field conveys as 8 bit 2's compliment number.
daylightSavings	• Day Light Savings Indicator – 0x00 - OFF (daylight savings not in effect) – 0x01 - ON (daylight savings in effect)

8.313.2 Field Documentation

8.313.2.1 uint8_t nas_qaQmi3Gpp2TimeZone::daylightSavings

8.313.2.2 uint8_t nas_qaQmi3Gpp2TimeZone::leapSeconds

8.313.2.3 uint8_t nas_qaQmi3Gpp2TimeZone::localTimeOffset

8.314 nas_QmiNas3GppNetworkInfo Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint32_t InUse
- uint32_t Roaming
- uint32_t Forbidden
- uint32_t Preferred
- char Desription [255]

8.314.1 Detailed Description

Contain the 3GPP network information.

Parameters

MCC	• Mobile Country Code
MNC	• Mobile Network Code
InUse	• Is the Network the current serving Network – 0 - Unknown – 1 - Current serving network – 2 - Not current serving network, available

Roaming	• Home/Roam Status of the Network – 0 - Unknown – 1 - Home – 2 - Roam
Forbidden	• Is the Network in the forbidden network list – 0 - Unknown – 1 - Forbidden – 2 - Not Forbidden
Preferred	• Is the Network in the Preferred network list – 0 - Unknown – 1 - Preferred – 2 - Not Preferred
Description	• Network Name/Description • This is a NULL terminated string.

8.314.2 Field Documentation

8.314.2.1 char nas_QmiNas3GppNetworkInfo::Desription[255]

8.314.2.2 uint32_t nas_QmiNas3GppNetworkInfo::Forbidden

8.314.2.3 uint32_t nas_QmiNas3GppNetworkInfo::InUse

8.314.2.4 uint16_t nas_QmiNas3GppNetworkInfo::MCC

8.314.2.5 uint16_t nas_QmiNas3GppNetworkInfo::MNC

8.314.2.6 uint32_t nas_QmiNas3GppNetworkInfo::Preferred

8.314.2.7 uint32_t nas_QmiNas3GppNetworkInfo::Roaming

8.315 nas_QmiNas3GppNetworkRAT Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t RAT

8.315.1 Detailed Description

Contain the 3GPP radio access technology information.

Parameters

MCC	• Mobile Country Code
MNC	• Mobile Network Code
RAT	• Radio Access Technology – 0x04 - GERAN – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA

8.315.2 Field Documentation

8.315.2.1 uint16_t nas_QmiNas3GppNetworkRAT::MCC

8.315.2.2 uint16_t nas_QmiNas3GppNetworkRAT::MNC

8.315.2.3 uint8_t nas_QmiNas3GppNetworkRAT::RAT

8.316 nas_QmisNasPcsDigit Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t includes_pcs_digit

8.316.1 Detailed Description

Contain the PCS Digit information

Parameters

MCC	• Mobile Country Code
MNC	• Mobile Network Code
includes_pcs_digit	• this field is use to interpret the length of corresponding MNC reported • 0x01 - MNC is a three-digit value • 0x00 - MNC is a two-digit value

8.316.2 Field Documentation

8.316.2.1 uint8_t nas_QmisNasPcsDigit::includes_pcs_digit

8.316.2.2 uint16_t nas_QmisNasPcsDigit::MCC

8.316.2.3 uint16_t nas_QmisNasPcsDigit::MNC

8.317 nas_QmisNasSlqsNasPCICellInfo Struct Reference

Data Fields

- uint32_t freq
- uint16_t cellID
- uint32_t GlobalCellID
- uint8_t PlmnLen
- nas_QmisNasPcsDigit nasQmisNasPcsDigit [255]

8.317.1 Detailed Description

Contain the PCI Cell Info.

Parameters

freq	• Aboslute cell's frequency. Range 0 to 65535.
cellID	• Cell ID.
GlobalCellID	• Global Cell ID.
PlmnLen	• Number of Plmn(nasQmisNasPcsDigit) information sets.
nasQmisNas-PcsDigit	• See nas_QmisNasPcsDigit for more information.

8.317.2 Field Documentation

8.317.2.1 uint16_t nas_QmisNasSlqsNasPCICellInfo::cellID

8.317.2.2 uint32_t nas_QmisNasSlqsNasPCICellInfo::freq

8.317.2.3 uint32_t nas_QmisNasSlqsNasPCICellInfo::GlobalCellID

8.317.2.4 nas_QmisNasPcsDigit nas_QmisNasSlqsNasPCICellInfo::nasQmisNasPcsDigit[255]

8.317.2.5 uint8_t nas_QmisNasSlqsNasPCICellInfo::PlmnLen

8.318 nas_QmisNasSlqsNasPCIInfo Struct Reference

Data Fields

- uint8_t PCICellInfoLen
- nas_QmisNasSlqsNasPCICellInfo nasQmisNasSlqsNasPCICellInfo [255]
- uint16_t rsrp
- uint16_t rsrpRx0
- uint16_t rsrpRx1
- uint16_t rsrq
- uint16_t rsrqRx0
- uint16_t rsrqRx1

8.318.1 Detailed Description

Contain the NAS Perform network scan PCI information

Parameters

PCICellInfoLen	• Number of PCI Cell Info(PCICellInfo) information sets.
nasQmisNasSlqsNasPCICellInfo	• See nas_QmisNasSlqsNasPCICellInfo for more information.
rsrp	• Combined RSRP.
rsrpRx0	• Rx0 RSRP.
rsrpRx1	• Rx1 RSRP.
rsrq	• Combined RSRQ.
rsrqRx0	• Rx0 RSRQ.
rsrpRx1	• Rx1 RSRQ.

8.318.2 Field Documentation

8.318.2.1 nas_QmisNasSlqsNasPCICellInfo nas_QmisNasSlqsNasPCIInfo::nasQmisNasSlqsNasPCICellInfo[255]

8.318.2.2 uint8_t nas_QmisNasSlqsNasPCIInfo::PCICellInfoLen

8.318.2.3 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrp

8.318.2.4 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrpRx0

8.318.2.5 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrpRx1

8.318.2.6 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrq

8.318.2.7 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrqRx0

8.318.2.8 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrqRx1

8.319 nas_RankIndicatorTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t count0
- uint32_t count1

8.319.1 Detailed Description

Elements for Rank indicator TLV.

Parameters

TlvPresent	• value is 1 if TLV is present, otherwise 0
count0	• count of Rank1
count1	• count of Rank2

8.319.2 Field Documentation

8.319.2.1 uint32_t nas_RankIndicatorTlv::count0

8.319.2.2 uint32_t nas_RankIndicatorTlv::count1

8.319.2.3 uint8_t nas_RankIndicatorTlv::TlvPresent

8.320 nas_RatDisabledMaskTlv Struct Reference

Data Fields

- uint16_t ratDisabledMask
- uint8_t TlvPresent

8.320.1 Detailed Description

Contain the RAT disabled mask

Parameters

ratDisabledMask	• Bitmask representing the radio technologies that are disabled. • Bit values: – Bit 0 (0x01) - cdma2000 ® 1X – Bit 1 (0x02) - cdma2000 ® HRPD(1xEV-DO) – Bit 2 (0x04) - GSM – Bit 3 (0x08) - UMTS – Bit 4 (0x10) - LTE – Bit 5 (0x20) - TD-SCDMA – Bit 6 (0x40) - NR5G All unlisted bits are reserved for future use and the service point ignores them if used.
TlvPresent	• Tlv Present

8.320.2 Field Documentation

8.320.2.1 uint16_t nas_RatDisabledMaskTlv::ratDisabledMask

8.320.2.2 uint8_t nas_RatDisabledMaskTlv::TlvPresent

8.321 nas_RejectReasonTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t serviceDomain
- uint32_t rejectCause

8.321.1 Detailed Description

This structure contains paramaters reject reason.

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
serviceDomain	service domain
rejectCause	cause of reject

8.321.2 Field Documentation

8.321.2.1 uint32_t nas_RejectReasonTlv::rejectCause

8.321.2.2 uint32_t nas_RejectReasonTlv::serviceDomain

8.321.2.3 uint8_t nas_RejectReasonTlv::TlvPresent

8.322 nas_RFBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBandClass
- uint16_t activeChannel

8.322.1 Detailed Description

This structure contains the RFBandInfo response parameters.

Parameters

radioInterface	• Radio interface technology – See Tables for Radio Interface
activeBandClass	• Active Band Class – See Tables for Band Classes
activeChannel	• Active channel (0 if channel is not relevant to the reported technology)

8.322.2 Field Documentation

8.322.2.1 uint16_t nas_RFBandInfoElements::activeBandClass

8.322.2.2 uint16_t nas_RFBandInfoElements::activeChannel

8.322.2.3 uint8_t nas_RFBandInfoElements::radioInterface

8.323 nas_RfBandInfoExtFormat Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t * pInstancesSize
- struct nas_RfBandInfoExtFormatElements * pRfBandInfoExtFormatParam

8.323.1 Detailed Description

This structure contains the nas_RfBandInfoExtFormat response parameters.

Parameters

TlvPresent	• Tlv Present
pInstancesSize[-IN/OUT]	• Number of sets of the RFBandInfoElements • Upon input, maximum number of elements that the RF info instances array can contain. • Upon successful output, actual number of elements in RF info instances array.
pRfBandInfoExt-FormatParam	• See nas_RfBandInfoExtFormatElements for more information

8.323.2 Field Documentation

8.323.2.1 uint8_t* nas_RfBandInfoExtFormat::pInstancesSize

8.323.2.2 struct nas_RfBandInfoExtFormatElements* nas_RfBandInfoExtFormat::pRfBandInfoExtFormatParam

8.323.2.3 uint8_t nas_RfBandInfoExtFormat::TlvPresent

8.324 nas_RfBandInfoExtFormatElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBand
- uint32_t activeChannel

8.324.1 Detailed Description

This structure contains the RF Band Information, Extended Format.

Parameters

radioInterface	• Radio interface technology – See Tables for Radio Interface
activeBand	• Active Band Class – See Tables for Band Classes
activeChannel	• Active channel (0 if channel is not relevant to the reported technology)

8.324.2 Field Documentation

8.324.2.1 uint16_t nas_RfBandInfoExtFormatElements::activeBand

8.324.2.2 uint32_t nas_RfBandInfoExtFormatElements::activeChannel

8.324.2.3 uint8_t nas_RfBandInfoExtFormatElements::radioInterface

8.325 nas_RFBandInfoExtTlv Struct Reference

Data Fields

- uint8_t radioIf
- uint16_t activeBand
- uint32_t activeChannel
- uint8_t TlvPresent

8.325.1 Detailed Description

Contain fields in struct nas_RFBandInfoExtTlv

Parameters

radioIf	• Radio interface currently in use. • Values – 0x01 - cdma2000® 1X – 0x02 - cdma2000® HRPD (1xEV-DO) – 0x03 - AMPS – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA – 0x0C - NR5G
activeBand	• Active band class. • Values – 00 to 39 - CDMA band classes – 40 to 79 - GSM band classes – 80 to 91 - WCDMA band classes – 120 to 168 - LTE band classes – 200 to 205 - TD-SCDMA band classes – 250 to 251 - NR-5G band classes
activeChannel	• Active channel. • If the channel is not relevant to the technology, a value of 0 is returned.
TlvPresent	• Tlv Present.

Note

None

8.325.2 Field Documentation

8.325.2.1 uint16_t nas_RFBandInfoExtTlv::activeBand

8.325.2.2 uint32_t nas_RFBandInfoExtTlv::activeChannel

8.325.2.3 uint8_t nas_RFBandInfoExtTlv::radioIf

8.325.2.4 uint8_t nas_RFBandInfoExtTlv::TlvPresent

8.326 nas_RfBandInfoList Struct Reference

Data Fields

- uint8_t * pInstanceSize
- struct nas_RFBandInfoElements * pRFBandInfoParam

8.326.1 Detailed Description

This structure contains the RfBandInfoList response parameters.

Parameters

pInstanceSize[IN/OUT]	• Number of sets of the RFBandInfoElements • Upon input, maximum number of elements that the RF info instances array can contain. • Upon successful output, actual number of elements in RF info instances array.
pRFBandInfoParam	• See nas_RFBandInfoElements for more information

8.326.2 Field Documentation

8.326.2.1 uint8_t* nas_RfBandInfoList::pInstanceSize

8.326.2.2 struct nas_RFBandInfoElements* nas_RfBandInfoList::pRFBandInfoParam

8.327 nas_RFBandInfoTlv Struct Reference

Data Fields

- uint8_t radioIf
- uint16_t activeBand
- uint16_t activeChannel
- uint8_t TlvPresent

8.327.1 Detailed Description

Contain fields in struct nas_RFBandInfoTlv

Parameters

radioIf	• Radio interface currently in use. • Values – 0x01 - cdma2000® 1X – 0x02 - cdma2000® HRPD (1xEV-DO) – 0x03 - AMPS – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA – 0x0C - NR5G
activeBand	• Active band class. • Values – 00 to 39 - CDMA band classes – 40 to 79 - GSM band classes – 80 to 91 - WCDMA band classes – 120 to 168 - LTE band classes – 200 to 205 - TD-SCDMA band classes – 250 to 251 - NR-5G band classes
activeChannel	• Active channel. • If the channel is not relevant to the technology, a value of 0 is returned.
TlvPresent	• Tlv Present.

Note

None

8.327.2 Field Documentation

8.327.2.1 uint16_t nas_RFBandInfoTlv::activeBand

8.327.2.2 uint16_t nas_RFBandInfoTlv::activeChannel

8.327.2.3 uint8_t nas_RFBandInfoTlv::radioIf

8.327.2.4 uint8_t nas_RFBandInfoTlv::TlvPresent

8.328 nas_RfBandwidthInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t * pInstancesSize
- struct nas_RfBandwidthInfoElements * pRfBandwidthInfoParam

8.328.1 Detailed Description

This structure contains the nas_RfBandwidthInfo response parameters.

Parameters

TlvPresent	• Tlv Present
pInstancesSize[-IN/OUT]	• Number of sets of the RFBandInfoElements • Upon input, maximum number of elements that the RF info instances array can contain. • Upon successful output, actual number of elements in RF info instances array.
pRfBandwidth-InfoParam	• See nas_RfBandwidthInfoElements for more information

8.328.2 Field Documentation

8.328.2.1 uint8_t* nas_RfBandwidthInfo::pInstancesSize

8.328.2.2 struct nas_RfBandwidthInfoElements* nas_RfBandwidthInfo::pRfBandwidthInfoParam

8.328.2.3 uint8_t nas_RfBandwidthInfo::TlvPresent

8.329 nas_RfBandwidthInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint32_t bandwidth

8.329.1 Detailed Description

This structure contains the RF Bandwidth Information List parameters.

Parameters

radioInterface	• Radio interface technology – See Tables for Radio Interface
bandwidth	• Bandwidth – Values * NAS_LTE_BW_NRB_6 (0) - LTE 1.4 MHz bandwidth * NAS_LTE_BW_NRB_15 (1) - LTE 3 MHz bandwidth * NAS_LTE_BW_NRB_25 (2) - LTE 5 MHz bandwidth * NAS_LTE_BW_NRB_50 (3) - LTE 10 MHz bandwidth * NAS_LTE_BW_NRB_75 (4) - LTE 15 MHz bandwidth * NAS_LTE_BW_NRB_100 (5) - LTE 20 MHz bandwidth * NAS_NR5G_BW_NRB_6 (6) - NR5G 1.4 MHz bandwidth * NAS_NR5G_BW_NRB_15 (7) - NR5G 3 MHz bandwidth * NAS_NR5G_BW_NRB_25 (8) - NR5G 5 MHz bandwidth * NAS_NR5G_BW_NRB_50 (9) - NR5G 10 MHz bandwidth * NAS_NR5G_BW_NRB_75 (10) - NR5G 15 MHz bandwidth * NAS_NR5G_BW_NRB_100 (11) - NR5G 20 MHz bandwidth

8.329.2 Field Documentation

8.329.2.1 uint32_t nas_RfBandwidthInfoElements::bandwidth

8.329.2.2 uint8_t nas_RfBandwidthInfoElements::radioInterface

8.330 nas_RFBandwidthInfoTlv Struct Reference

Data Fields

- uint8_t radioIf
- uint32_t bandwidth
- uint8_t TlvPresent

8.330.1 Detailed Description

Contain fields in struct nas_RFBandwidthInfoTlv

Parameters

radioIf	• Radio interface currently in use. • Values – 0x00 - no service – 0x01 - cdma2000® 1X – 0x02 - cdma2000® HRPD (1xEV-DO) – 0x03 - AMPS – 0x04 - GSM – 0x05 - UMTS – 0x06 - WLAN – 0x07 - GPS – 0x08 - LTE – 0x09 - TD-SCDMA – 0x0a - LTE-M1 – 0x0b - LTE-NB1 – -1 - No change
bandwidth	• Bandwidth. • Values – 0 - LTE 1.4 MHz bandwidth – 1 - LTE 3 MHz bandwidth – 2 - LTE 5 MHz bandwidth – 3 - LTE 10 MHz bandwidth – 4 - LTE 15 MHz bandwidth – 5 - LTE 20 MHz bandwidth – 6 - NR5G 1.4 MHz bandwidth – 7 - NR5G 3 MHz bandwidth – 8 - NR5G 5 MHz bandwidth – 9 - NR5G 10 MHz bandwidth – 10 - NR5G 15 MHz bandwidth – 11 - NR5G 20 MHz bandwidth
TlvPresent	• Tlv Present.

Note

None

8.330.2 Field Documentation

8.330.2.1 uint32_t nas_RFBandwidthInfoTlv::bandwidth

8.330.2.2 uint8_t nas_RFBandwidthInfoTlv::radioIf

8.330.2.3 uint8_t nas_RFBandwidthInfoTlv::TlvPresent

8.331 nas_RfDedicatedBandInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t * pInstancesSize
- struct nas_RfDedicatedBandInfoElements * pRfDedicatedBandInfoParam

8.331.1 Detailed Description

This structure contains the nas_RfDedicatedBandInfo response parameters.

Parameters

TlvPresent	• Tlv Present
pInstancesSize[-IN/OUT]	• Number of sets of the RFBandInfoElements • Upon input, maximum number of elements that the RF info instances array can contain. • Upon successful output, actual number of elements in RF info instances array.
pRfDedicated-BandInfoParam	• See nas_RfDedicatedBandInfoElements for more information

8.331.2 Field Documentation

8.331.2.1 uint8_t* nas_RfDedicatedBandInfo::pInstancesSize

8.331.2.2 struct nas_RfDedicatedBandInfoElements* nas_RfDedicatedBandInfo::pRfDedicatedBandInfoParam

8.331.2.3 uint8_t nas_RfDedicatedBandInfo::TlvPresent

8.332 nas_RfDedicatedBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t dedicatedBand

8.332.1 Detailed Description

This structure contains the RF Dedicated Band Info.

Parameters

radioInterface	• Radio interface technology – See Tables for Radio Interface
dedicatedBand	• Dedicated Band Class – See Tables for Band Classes

8.332.2 Field Documentation

8.332.2.1 uint16_t nas_RfDedicatedBandInfoElements::dedicatedBand

8.332.2.2 uint8_t nas_RfDedicatedBandInfoElements::radioInterface

8.333 nas_RFDedicatedBandInfoTlv Struct Reference

Data Fields

- uint8_t radioIf
- uint16_t dedicatedBand
- uint8_t TlvPresent

8.333.1 Detailed Description

Contain fields in struct nas_RFDedicatedBandInfoTlv

Parameters

radioIf	• Radio interface currently in use. • Values – 0x01 - cdma2000® 1X – 0x02 - cdma2000® HRPD (1xEV-DO) – 0x03 - AMPS – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA
dedicatedBand	• Dedicated band class. • Values – 00 to 39 - CDMA band classes – 40 to 79 - GSM band classes – 80 to 91 - WCDMA band classes – 120 to 168 - LTE band classes – 200 to 205 - TD-SCDMA band classes – 0xFFFF - invalid - indicates that the UE moved out from the dedicated band
TlvPresent	• Tlv Present.

Note

None

8.333.2 Field Documentation

8.333.2.1 uint16_t nas_RFDedicatedBandInfoTlv::dedicatedBand

8.333.2.2 uint8_t nas_RFDedicatedBandInfoTlv::radioIf

8.333.2.3 uint8_t nas_RFDedicatedBandInfoTlv::TlvPresent

8.334 nas_RFInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t radioInterfaceSize
- uint32_t radioInterface [255]
- uint32_t activeBandClass [255]
- uint32_t activeChannel [255]

8.334.1 Detailed Description

This structure contains paramaters RF Band Information List.

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
radioInterface-Size	Number of sets of the following elements • radioInterface • activeBandClass • activeChannel
radioInterface	radio interface technology of the signal being measured • 0x01 - cdma2000 1X • 0x02 - cdma2000 HRPD (1xEV-DO) • 0x03 - AMPS • 0x04 - GSM • 0x05 - UMTS • 0x08 - LTE • 0x09 - TD-SCDMA
activeBandClass	active band class • 00 to 39 - CDMA band classes • 40 to 79 - GSM band classes • 80 to 91 - WCDMA band classes • 120 to 160 - LTE band classes • 200 to 205 - TD-SCDMA band classes
activeChannel	active channel • Active channel. If the channel is not relevant to the technology, a value of 0 is returned.

8.334.2 Field Documentation

8.334.2.1 uint32_t nas_RFInfoTlv::activeBandClass[255]

8.334.2.2 uint32_t nas_RFInfoTlv::activeChannel[255]

8.334.2.3 uint32_t nas_RFInfoTlv::radioInterface[255]

8.334.2.4 uint8_t nas_RFInfoTlv::radioInterfaceSize

8.334.2.5 uint8_t nas_RFInfoTlv::TlvPresent

8.335 nas_roamIndList Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t radioInterface [32]
- uint8_t roamIndicator [32]

8.335.1 Detailed Description

This structure contains the Roaming Indicator List

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

numInstances	• number of sets of radio interface currently in use and roaming indicator – defaults to zero
radioInterface	• Radio Interface currently in use • Values: – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE
roamIndicator	• Roaming Indicator • Values: – 0x00 - Roaming – 0x01 - Home

8.335.2 Field Documentation

8.335.2.1 uint8_t nas_roamIndList::numInstances

8.335.2.2 uint8_t nas_roamIndList::radioInterface[32]

8.335.2.3 uint8_t nas_roamIndList::roamIndicator[32]

8.336 nas_RoamPrefTlv Struct Reference

Data Fields

- uint16_t RoamPref
- uint8_t TlvPresent

8.336.1 Detailed Description

Contain the roaming preference for system selection preferences.

Parameters

RoamPref	• Parameter indicating the roaming Preference • Values: – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator
TlvPresent	• Tlv Present

8.336.2 Field Documentation

8.336.2.1 uint16_t nas_RoamPrefTlv::RoamPref

8.336.2.2 uint8_t nas_RoamPrefTlv::TlvPresent

8.337 nas_RSRPThresh Struct Reference

Data Fields

- uint8_t RSRPThresListLen
- int16_t * pRSRPThresList

8.337.1 Detailed Description

This structure contains RSRP threshold list.

Parameters

RSRPThresList-Len	• Length of the RSRP threshold list parameter to follow
pRSRPThresList	• Sequence of thresholds delimiting current RSRP event reporting bands • Every time a new RSRP value crosses a specified threshold value, an event report indication message with the new RSRQ value is sent to the requesting control point. For this field – RSRP values are applicable only for LTE – RSRP values are measured in dBm, with a range of -44 dBm to -140 dBm – Each RSRP threshold value is a signed byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.337.2 Field Documentation

8.337.2.1 int16_t* nas_RSRPThresh::pRSRPThresList

8.337.2.2 uint8_t nas_RSRPThresh::RSRPThresListLen

8.338 nas_rsrqInformation Struct Reference

Data Fields

- int8_t rsrq
- uint8_t radioIf

8.338.1 Detailed Description

This structure contains the RSRQ Information

Parameters

rsrq	• RSRQ value in dB (signed integer value); valid range is -3 to -20 (-3 means -3 dB, -20 means -20 dB)
radioIf	• Radio interface technology of the signal being measured – 0x08 - LTE

8.338.2 Field Documentation

8.338.2.1 uint8_t nas_rsrqInformation::radioIf

8.338.2.2 int8_t nas_rsrqInformation::rsrq

8.339 nas_RSRQThresh Struct Reference

Data Fields

- uint8_t RSRQThresListLen
- int8_t * pRSRQThresList

8.339.1 Detailed Description

This structure contains RSRQ threshold list.

Parameters

RSRQThresList-Len	• Length of the RSRQ threshold list parameter to follow
pRSRQThresList	• Sequence of thresholds delimiting current RSRQ event reporting bands • Every time a new RSRQ value crosses a threshold value, an event report indication message with the new RSRQ value is sent to the requesting control point. For this field – RSRQ values are applicable only for LTE – RSRQ values are measured in dBm, with a range of -20 dBm to -3 dBm – Each RSRQ threshold value is a signed byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.339.2 Field Documentation

8.339.2.1 int8_t* nas_RSRQThresh::pRSRQThresList

8.339.2.2 uint8_t nas_RSRQThresh::RSRQThresListLen

8.340 nas_RSSIThresh Struct Reference

Data Fields

- uint8_t RSSIThresListLen
- int8_t * pRSSIThresList

8.340.1 Detailed Description

This sturcture contain RSSI threshold list.

Parameters

RSSIThresList-Len	• Length of the RSSI threshold list parameter to follow
pRSSIThresList	• RSSI in dBm(signed bytes) • A value of -125 dBm or lower is used to indicate No Signal • RSSI values have the following ranges (in dBm) – CDMA is -105 to -21 – HDR is -118 to -13 – GSM is -111 to -48 – WCDMA is -121 to 0 – LTE is -120 to 0 • Threshold values specified above are used for all RATs • The maximum number of threshold values is 16, each a signed byte value.

8.340.2 Field Documentation

8.340.2.1 int8_t* nas_RSSIThresh::pRSSIThresList

8.340.2.2 uint8_t nas_RSSIThresh::RSSIThresListLen

8.341 nas_rxInfo Struct Reference

Data Fields

- uint8_t isRadioTuned
- int32_t rxPower
- int32_t ecio
- int32_t rscp
- int32_t rsrp
- uint32_t phase

8.341.1 Detailed Description

This structure contains rx radio infomation.

Parameters

isRadioTuned	• Whether Rx is tuned to a channel: – 0x00 - Not tuned – 0x01 - Tuned – 0xFF - Not Available • If the radio is tuned, instantaneous values are set for the signal information fields below. • If the radio is not tuned, or is delayed or invalid, the values are set depending on each technology.
rx_pwr	• Rx power value in 1/10 dbm resolution.

ecio	• ECIO in 1/10 dB; valid for CDMA, HDR, GSM, WCDMA, and LTE.
rscp	• Received signal code power in 1/10 dbm. • Valid for WCDMA.
rsrp	• Current reference signal received power in 1/10 dbm valid for LTE.
phase	• Phase in 1/100 degrees; valid for LTE. • When the phase is unknown, 0xFFFFFFFF is used.

8.341.2 Field Documentation

8.341.2.1 int32_t nas_rxInfo::ecio

8.341.2.2 uint8_t nas_rxInfo::isRadioTuned

8.341.2.3 uint32_t nas_rxInfo::phase

8.341.2.4 int32_t nas_rxInfo::rscp

8.341.2.5 int32_t nas_rxInfo::rsrp

8.341.2.6 int32_t nas_rxInfo::rxPower

8.342 nas_RxSigInfo Struct Reference

Data Fields

- uint8_t rxChainIndex
- uint8_t isRadioTuned
- int32_t rxPower
- int32_t rsrp

8.342.1 Detailed Description

This structure contains the parameters for Rx Signal Info.

Parameters

rxChainIndex	• Rx antenna path • Valid Values – 0 - Primary Rx – 1 - Diversity Rx

isRadioTuned	• Rx path is tuned to a channel or Not • Values – 0x00 - Not tuned – 0x01 - Tuned

Note

If the radio is tuned, the instantaneous values are set for the fields below. If the radio is not tuned, the values set below may be invalid.

Parameters

rxPower	• Rx power value in 1/10 dBm resolution
rsrp	• Current reference signal received power in 1/10 dBm resolution

8.342.2 Field Documentation

8.342.2.1 uint8_t nas_RxSigInfo::isRadioTuned

8.342.2.2 int32_t nas_RxSigInfo::rsrp

8.342.2.3 uint8_t nas_RxSigInfo::rxChainIndex

8.342.2.4 int32_t nas_RxSigInfo::rxPower

8.343 nas_rxSignalStrengthListElement Struct Reference

Data Fields

- int16_t rxSignalStrength
- uint8_t radioIf

8.343.1 Detailed Description

This structure contains the Received Signal Strength Information

Parameters

rxSignalStrength	• Received signal strength in dBm – For CDMA and UMTS, this indicates forward link pilotEc. – For GSM, the received signal strength. – For LTE, this indicates the total received wideband power observed by UE.

radioIf	• Radio interface technology of the signal being radio_if measured – 0x00 - RADIO_IF_NO_SVC - None (no service) – 0x01 - RADIO_IF_CDMA_1X - cdma2000® 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE

Note

First elemnet of the RSSI list always contains the current Signal strength and Radio Interface.

8.343.2 Field Documentation

8.343.2.1 uint8_t nas_rxSignalStrengthListElement::radioIf

8.343.2.2 int16_t nas_rxSignalStrengthListElement::rxSignalStrength

8.344 nas_SccRxInfo Struct Reference

Data Fields

- int32_t rsrq
- int16_t snr
- uint8_t numInstances
- nas_RxSigInfo sigInfo [255]
- uint8_t TlvPresent

8.344.1 Detailed Description

This structure contains information about the SccRxInfo parameters.

Parameters

rsrq	• Current reference signal • Receive quality in 1/10 dB resolution
snr	• Reference signal signal-to-noise ratio in dB. • Range -10 to 30
numInstances	• Number of sets of the following – rxChainIndex – isRadioTuned – rxPower – rsrp

sigInfo	• See nas_RxSigInfo for more information
TlvPresent	• Tlv Present.

8.344.2 Field Documentation

8.344.2.1 uint8_t nas_SccRxInfo::numInstances

8.344.2.2 int32_t nas_SccRxInfo::rsrq

8.344.2.3 nas_RxSigInfo nas_SccRxInfo::sigInfo[255]

8.344.2.4 int16_t nas_SccRxInfo::snr

8.344.2.5 uint8_t nas_SccRxInfo::TlvPresent

8.345 nas_serviceProviderName Struct Reference

Data Fields

- uint8_t displayCondition
- uint8_t spnLength
- uint8_t spn [255]

8.345.1 Detailed Description

This structure contains service provider name.

Parameters

displayCondition	• Display condition.
spnLength	• It provides length of spn.
spn	• Service provider name string must use: The SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 9.

8.345.2 Field Documentation

8.345.2.1 uint8_t nas_serviceProviderName::displayCondition

8.345.2.2 uint8_t nas_serviceProviderName::spn[255]

8.345.2.3 uint8_t nas_serviceProviderName::spnLength

8.346 nas_servSystem Struct Reference

Data Fields

- uint8_t regState
- uint8_t csAttachState
- uint8_t psAttachState
- uint8_t selNetwork
- uint8_t numRadioInterfaces
- uint8_t radioInterface [32]

8.346.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

regState	• Registration state - Registration state of the mobile • Values: – 0 - Not Registered; mobile is not currently searching for a new network to provide service – 1 - Registered with a network – 2 - Not registered, but mobile is currently searching for a new network to provide service – 3 - Registration denied by visible network – 4 - Registration state is unknown
csAttachState	• CS Attach State - Circuit-switched domain attach state of the mobile • Values: – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached
psAttachState	• PS Attach State - Packet-switched domain attach state of the mobile • Values: – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached
selNetwork	• Selected Network - Type of selected radio access network • Values: – 0 - Unknown – 1 - 3GPP2 network – 2 - 3GPP network
numRadio-Interfaces	• In Use Radio Interfaces Number – Number of radio interfaces currently in use – defaults to zero

radioInterface	• Radio Interface currently in use • Values: – 0x00 - RADIO_IF_NO_SVC - None(no service) – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE

8.346.2 Field Documentation

8.346.2.1 uint8_t nas_servSystem::csAttachState

8.346.2.2 uint8_t nas_servSystem::numRadioInterfaces

8.346.2.3 uint8_t nas_servSystem::psAttachState

8.346.2.4 uint8_t nas_servSystem::radioInterface[32]

8.346.2.5 uint8_t nas_servSystem::regState

8.346.2.6 uint8_t nas_servSystem::selNetwork

8.347 nas_sidNid Struct Reference

Data Fields

- uint16_t nid
- uint16_t sid

8.347.1 Detailed Description

This structure contains the parameters for SidNid Information

Parameters

nid	• Network ID
sid	• System ID

8.347.2 Field Documentation

8.347.2.1 uint16_t nas_sidNid::nid

8.347.2.2 uint16_t nas_sidNid::sid

8.348 nas_SignalQualityTlv Struct Reference

Data Fields

- uint8_t signal_quality_len
- uint32_t signal_quality [255]
- uint8_t TlvPresent

8.348.1 Detailed Description

Contain fields in struct SignalQualityTlv

Parameters

signal_quality_len	• Number of sets of the following elements: – signal_quality
signal_quality	• Network name source. • Values: – 0x00 - SYS_SIGNAL_QUALITY_LOW – 0x01 - SYS_SIGNAL_QUALITY_HIGH
TlvPresent	• Tlv Present.

Note

None

8.348.2 Field Documentation

8.348.2.1 uint32_t nas_SignalQualityTlv::signal_quality[255]

8.348.2.2 uint8_t nas_SignalQualityTlv::signal_quality_len

8.348.2.3 uint8_t nas_SignalQualityTlv::TlvPresent

8.349 nas_SignalStrengthTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- int8_t signalStrength
- uint32_t radioInterface

8.349.1 Detailed Description

This structure contains paramaters Signal Strength.

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
signalStrength	signal strength
radioInterface	radio interface technology of the signal being measured

8.349.2 Field Documentation

8.349.2.1 uint32_t nas_SignalStrengthTlv::radioInterface

8.349.2.2 int8_t nas_SignalStrengthTlv::signalStrength

8.349.2.3 uint8_t nas_SignalStrengthTlv::TlvPresent

8.350 nas_SignalStrngthTlv Struct Reference

Data Fields

- uint8_t signal_strength_len
- uint32_t signal_strength [255]
- uint8_t TlvPresent

8.350.1 Detailed Description

Contain fields in struct SignalStrengthTlv

Parameters

signal_strength-_len	• Number of sets of the following elements: – signal_quality
signal_strength	• PLMN's signal strength infomation (in dBm).
TlvPresent	• Tlv Present.

Note

None

8.350.2 Field Documentation

8.350.2.1 uint32_t nas_SignalStrngthTlv::signal_strength[255]

8.350.2.2 uint8_t nas_SignalStrngthTlv::signal_strength_len

8.350.2.3 uint8_t nas_SignalStrngthTlv::TlvPresent

8.351 nas_SimRejInfoTlv Struct Reference

Data Fields

- uint32_t SimRejInfo
- uint8_t TLVPresent

8.351.1 Detailed Description

Structure for storing the SIM Reject Information.

Parameters

SimRejInfo	• Current reject state information of the SIM. • Values – 0 - SIM is not available – 1 - SIM is available – 2 - SIM has been marked by the network as invalid for circuit-switched services – 3 - SIM has been marked by the network as invalid for packet-switched services – 4 - SIM has been marked by the network as invalid for circuit-switched and packet-switched services
TLVPresent	• TLV Present

8.351.2 Field Documentation

8.351.2.1 uint32_t nas_SimRejInfoTlv::SimRejInfo

8.351.2.2 uint8_t nas_SimRejInfoTlv::TLVPresent

8.352 nas_SLQSSignalStrengthsIndReq Struct Reference

Data Fields

- uint8_t rxSignalStrengthDelta
- uint8_t ecioDelta
- uint8_t ioDelta
- uint8_t sinrDelta
- uint8_t rsrqDelta
- uint8_t ecioThresholdListLen
- int16_t ecioThresholdList [10]
- uint8_t sinrThresholdListLen
- uint8_t sinrThresholdList [5]
- uint16_t lteSnrDelta
- uint8_t lteRsrpDelta

8.352.1 Detailed Description

Structure for storing the input parameters passed for SLQSSetSignalStrengthsCallback by the user.

Parameters

rxSignal-StrengthDelta	• RSSI delta(in dBm) at which an event report indication, including the current RSSI, will be sent to the requesting control point.
ecioDelta	• ECIO delta at which an event report indication, ecioDelta including the current ECIO, will be sent to the requesting control point. • ECIO delta is an unsigned 1 byte value that increments in negative 0.5 dBm, e.g., ecio_delta of 2 means a change of -1 dBm.
ioDelta	• IO delta (in dBm) at which an event report indication, ioDelta including the current IO, will be sent to the requesting control point.
sinrDelta	• SINR delta level at which an event report indication, sinrDelta including the current SINR, will be sent to the requesting control point.
rsrqDelta	• RSRQ delta level at which an event report indication, including the current RSRQ, will be sent to the requesting control point.
ecioThreshold-ListLen	• Number of elements in the ECIO threshold list.
ecioThreshold-List	• A sequence of thresholds delimiting EcIo event reporting bands. Every time a new EcIo value crosses a threshold value, an event report indication message with the new ECIO value is sent to the requesting control point. For this field: – Maximum number of threshold values is 10 – At least one value must be specified.
sinrThreshold-ListLen	• Number of elements in the SINR threshold list.
sinrThreshold-List	• A sequence of thresholds delimiting SINR event reporting bands. Every time a new SINR value crosses a threshold value, an event report indication message with the new sinr value is sent to the requesting control point. For this field: – Maximum number of threshold values is 5 – At least one value must be specified.
ltesnrdelta	• LTE SNR delta level at which an event report indication, including the current SNR, will be sent to the requesting control point. LTE SNR delta level is an unsigned 2 byte value, representing the delta in units of 0.1 dB, e.g., lte_snr_delta of 3 means a change 0.3dB.
ltersrpdelta	-LTE RSRP delta level at which an event report -indication, including the current RSRP, will be sent -to the requesting control point. LTE RSRP delta -level is an unsigned 1 byte value, representing the -delta in dB.

8.352.2 Field Documentation

8.352.2.1 uint8_t nas_SLQSSignalStrengthsIndReq::ecioDelta

8.352.2.2 int16_t nas_SLQSSignalStrengthsIndReq::ecioThresholdList[10]

8.352.2.3 uint8_t nas_SLQSSignalStrengthsIndReq::ecioThresholdListLen

8.352.2.4 uint8_t nas_SLQSSignalStrengthsIndReq::ioDelta

8.352.2.5 uint8_t nas_SLQSSignalStrengthsIndReq::lteRsrpDelta

8.352.2.6 uint16_t nas_SLQSSignalStrengthsIndReq::lteSnrDelta

8.352.2.7 uint8_t nas_SLQSSignalStrengthsIndReq::rsrqDelta

8.352.2.8 uint8_t nas_SLQSSignalStrengthsIndReq::rxSignalStrengthDelta

8.352.2.9 uint8_t nas_SLQSSignalStrengthsIndReq::sinrDelta

8.352.2.10 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdList[5]

8.352.2.11 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdListLen

8.353 nas_SLQSSignalStrengthsInformation Struct Reference

Data Fields

- nas_rxSignalStrengthListElement rxSignalStrengthInfo
- nas_ecioListElement ecioInfo
- uint32_t io
- uint8_t sinr
- nas_errorRateListElement errorRateInfo
- nas_rsrqInformation rsrqInfo
- nas_lteSnrinformation lteSnrinfo
- nas_lteRsrpinformation lteRsrpinfo

8.353.1 Detailed Description

This structure contains paramaters signal strengths information.

Parameters

rxSignal-StrengthInfo	• signal strength info list • Bit to check in ParamPresenceMask - **19**
ecioInfo	• ecio info list • Bit to check in ParamPresenceMask - **20**
io	• received IO in dBm; IO is only applicable for 1xEV-DO • Bit to check in ParamPresenceMask - **21**

sinr	• SINR level • Bit to check in ParamPresenceMask - **22**
errorRateInfo	• error rate info • Bit to check in ParamPresenceMask - **23**
rsrqInfo	• rsrq info • Bit to check in ParamPresenceMask - **24**
lteSnrinfo	• lte Snr information • Bit to check in ParamPresenceMask - **25**
lteRsrpinfo	• lte rsrp info • Bit to check in ParamPresenceMask - **26**

8.353.2 Field Documentation

8.353.2.1 nas_ecioListElement nas_SLQSSignalStrengthsInformation::ecioInfo

8.353.2.2 nas_errorRateListElement nas_SLQSSignalStrengthsInformation::errorRateInfo

8.353.2.3 uint32_t nas_SLQSSignalStrengthsInformation::io

8.353.2.4 nas_lteRsrpinformation nas_SLQSSignalStrengthsInformation::lteRsrpinfo

8.353.2.5 nas_lteSnrinformation nas_SLQSSignalStrengthsInformation::lteSnrinfo

8.353.2.6 nas_rsrqInformation nas_SLQSSignalStrengthsInformation::rsrqInfo

8.353.2.7 nas_rxSignalStrengthListElement nas_SLQSSignalStrengthsInformation::rxSignalStrengthInfo

8.353.2.8 uint8_t nas_SLQSSignalStrengthsInformation::sinr

8.354 nas_SLQSSignalStrengthsTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- nas_SLQSSignalStrengthsInformation sSLQSSignalStrengthsInfo

8.354.1 Detailed Description

This structure contains paramaters SLQS Signal Strength.

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
sSLQSSignal-StrengthsInfo	signal strength info

8.354.2 Field Documentation

8.354.2.1 nas_SLQSSignalStrengthsInformation nas_SLQSSignalStrengthsTlv::sSLQSSignalStrengthsInfo

8.354.2.2 uint8_t nas_SLQSSignalStrengthsTlv::TlvPresent

8.355 nas_SrvDomainPrefTlv Struct Reference

Data Fields

- uint32_t SrvDomainPref
- uint8_t TlvPresent

8.355.1 Detailed Description

Contain the Service domain Preference for system selection preferences.

Parameters

SrvDomainPref	• Parameter indicating Service domain preference • Values: – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach
TlvPresent	• Tlv Present

8.355.2 Field Documentation

8.355.2.1 uint32_t nas_SrvDomainPrefTlv::SrvDomainPref

8.355.2.2 uint8_t nas_SrvDomainPrefTlv::TlvPresent

8.356 nas_SrvRegRestrictionTlv Struct Reference

Data Fields

- uint32_t SrvRegRestriction
- uint8_t TLVPresent

8.356.1 Detailed Description

Structure for storing the Network Selection Registration Restriction

Parameters

SrvReg-Restriction	• Registration restriction. – Values * 0x00 - Device follows the normal registration process * 0x01 - Device follows the camp-only registration process * All other values are reserved.
TLVPresent	• TLV Present

8.356.2 Field Documentation

8.356.2.1 uint32_t nas_SrvRegRestrictionTlv::SrvRegRestriction

8.356.2.2 uint8_t nas_SrvRegRestrictionTlv::TLVPresent

8.357 nas_SrvRegRestricTlv Struct Reference

Data Fields

- uint32_t SrvRegRestriction
- uint8_t TlvPresent

8.357.1 Detailed Description

Contain the Network Selection Registration Restriction Preference.

Parameters

SrvReg-Restriction	• Registration restriction preference. • Values – 0x00 - Device follows the normal registration process – 0x01 - Device camps on the network according to its provisioning, but does not register – 0x02 - Device selects the network for limited service – All other values are reserved.
TlvPresent	• Tlv Present

8.357.2 Field Documentation

8.357.2.1 uint32_t nas_SrvRegRestricTlv::SrvRegRestriction

8.357.2.2 uint8_t nas_SrvRegRestricTlv::TlvPresent

8.358 nas_SrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t isPrefDataPath

8.358.1 Detailed Description

Structure for storing the service status information for CDMA and HDR networks.

Parameters

srvStatus	• Service status of the system. – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
isPrefDataPath	• Whether the RAT is the preferred data path. – 0x00 - Not preferred – 0x01 - Preferred – 0xFF - Not Available

8.358.2 Field Documentation

8.358.2.1 uint8_t nas_SrvStatusInfo::isPrefDataPath

8.358.2.2 uint8_t nas_SrvStatusInfo::srvStatus

8.359 nas_sysInfoCommon Struct Reference

Data Fields

- uint8_t srvDomainValid
- uint8_t srvDomain
- uint8_t srvCapabilityValid
- uint8_t srvCapability
- uint8_t roamStatusValid
- uint8_t roamStatus
- uint8_t isSysForbiddenValid
- uint8_t isSysForbidden

8.359.1 Detailed Description

Structure for storing the System Information common to CDMA, HDR, GSM, WCDMA and LTE networks.

Parameters

srvDomainValid	• Indicates whether the service domain is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
srvDomain	• Service domain registered on the system. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
srvCapability-Valid	• Indicates whether the service capability is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
srvCapability	• Current system's service capability. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
roamStatusValid	• Indicates whether the roaming status is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

roamStatus	• Current roaming status. – 0x00 - Off – 0x01 - On – 0x02 - Blinking – 0x03 - Out of the neighborhood – 0x04 - Out of the building – 0x05 - Preferred system – 0x06 - Available system – 0x07 - Alliance partner – 0x08 - Premium partner – 0x09 - Full service – 0x0A - Partial service – 0x0B - Banner is on – 0x0C - Banner is off – 0x0D to 0x3F - Reserved for Standard Enhanced Roaming Indicator Numbers – 0x40 to 0x7F - Reserved for Non-Standard Enhanced Roaming Indicator Numbers – 0x40 to 0xFF - Reserved. – 0xFF - Not Available • Values from 0x02 onward are only applicable for 3GPP2
isSysForbidden-Valid	• Indicates whether the forbidden system is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
isSysForbidden	• Whether the system is forbidden. – 0x00 - Not forbidden – 0x01 - Forbidden – 0xFF - Not Available

8.359.2 Field Documentation

8.359.2.1 uint8_t nas_sysInfoCommon::isSysForbidden

8.359.2.2 uint8_t nas_sysInfoCommon::isSysForbiddenValid

8.359.2.3 uint8_t nas_sysInfoCommon::roamStatus

8.359.2.4 uint8_t nas_sysInfoCommon::roamStatusValid

8.359.2.5 uint8_t nas_sysInfoCommon::srvCapability

8.359.2.6 uint8_t nas_sysInfoCommon::srvCapabilityValid

8.359.2.7 uint8_t nas_sysInfoCommon::srvDomain

8.359.2.8 uint8_t nas_sysInfoCommon::srvDomainValid

8.360 nas_TDSCDMABandPrefTlv Struct Reference

Data Fields

- uint64_t TdscdmaBandPref
- uint8_t TlvPresent

8.360.1 Detailed Description

Contain the TDSCDMA Band Preference.

Parameters

TdscdmaBand-Pref	• Bitmask representing the TD-SCDMA band preference to be set. • Values – 0x01 - NAS_TDSCDMA_BAND_A - TD-SCDMA Band A – 0x02 - NAS_TDSCDMA_BAND_B - TD-SCDMA Band B – 0x04 - NAS_TDSCDMA_BAND_C - TD-SCDMA Band C – 0x08 - NAS_TDSCDMA_BAND_D - TD-SCDMA Band D – 0x10 - NAS_TDSCDMA_BAND_E - TD-SCDMA Band E – 0x20 - NAS_TDSCDMA_BAND_F - TD-SCDMA Band F – All other bits are reserved.
TlvPresent	• Tlv Present

8.360.2 Field Documentation

8.360.2.1 uint64_t nas_TDSCDMABandPrefTlv::TdscdmaBandPref

8.360.2.2 uint8_t nas_TDSCDMABandPrefTlv::TlvPresent

8.361 nas_TDSCDMAECIOThresh Struct Reference

Data Fields

- uint8_t TDSCDMAECIOThreshListLen
- float * pTDSCDMAECIOThreshList

8.361.1 Detailed Description

This structure contains TDSCDMA ECIO threshold related parameters.

Parameters

TDSCDMAECI-OThreshListLen	• Length of the TDSCDMA ECIO threshold list parameter to follow

pTDSCDMAECIOThreshList	• Array of ECIO thresholds (in dB) used by TD-SCDMA • Maximum of 32 values.

8.361.2 Field Documentation

8.361.2.1 float* nas_TDSCDMAECIOThresh::pTDSCDMAECIOThreshList

8.361.2.2 uint8_t nas_TDSCDMAECIOThresh::TDSCDMAECIOThreshListLen

8.362 nas_TDSCDMARSCPThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSCPThreshListLen
- int16_t * pTDSCDMARSCPThreshList

8.362.1 Detailed Description

This structure contains TDSCDMA RSCP threshold related parameters.

Parameters

TDSCDMARSCPThreshListLen	• Length of the TDSCDMA RSCP threshold list parameter to follow
pTDSCDMARSCPThreshList	• Array of RSCP thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSCP values: -120 to -25 (in dBm).

8.362.2 Field Documentation

8.362.2.1 int16_t* nas_TDSCDMARSCPThresh::pTDSCDMARSCPThreshList

8.362.2.2 uint8_t nas_TDSCDMARSCPThresh::TDSCDMARSCPThreshListLen

8.363 nas_TDSCDMARSSIThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSSIThreshListLen
- float * pTDSCDMARSSIThreshList

8.363.1 Detailed Description

This structure contains TDSCDMA RSSI threshold related parameters.

Parameters

TDSCDMARSSIThreshListLen	• Length of the TDSCDMA RSSI threshold list parameter to follow
pTDSCDMARSSIThreshList	• Array of RSSI thresholds (in dBm) used by TD-SCDMA • Maximum of 32 values.

8.363.2 Field Documentation

8.363.2.1 float* nas_TDSCDMARSSIThresh::pTDSCDMARSSIThreshList

8.363.2.2 uint8_t nas_TDSCDMARSSIThresh::TDSCDMARSSIThreshListLen

8.364 nas_TDSCDMASINRCONFThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRCONFThreshListLen
- float * pTDSCDMASINRCONFThreshList

8.364.1 Detailed Description

This structure contains TDSCDMA SINR threshold list.

Parameters

TDSCDMASINRCONFThreshListLen	• Length of the TDSCDMA SINR threshold list parameter to follow
pTDSCDMASINRCONFThreshList	• Array of SINR thresholds (in dB) used by TD-SCDMA • Maximum of 32 values

8.364.2 Field Documentation

8.364.2.1 float* nas_TDSCDMASINRCONFThresh::pTDSCDMASINRCONFThreshList

8.364.2.2 uint8_t nas_TDSCDMASINRCONFThresh::TDSCDMASINRCONFThreshListLen

8.365 nas_TDSCDMASINRThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRThreshListLen
- float * pTDSCDMASINRThreshList

8.365.1 Detailed Description

This structure contains TDSCDMA SINR threshold related parameters.

Parameters

TDSCDMASINRThreshListLen	• Length of the TDSCDMA SINR threshold list parameter to follow
pTDSCDMASINRThreshList	• Array of SINR thresholds (in dB) used by TD-SCDMA • Maximum of 32 values

8.365.2 Field Documentation

8.365.2.1 float* nas_TDSCDMASINRThresh::pTDSCDMASINRThreshList

8.365.2.2 uint8_t nas_TDSCDMASINRThresh::TDSCDMASINRThreshListLen

8.366 nas_timeInfo Struct Reference

Data Fields

- uint16_t year
- uint8_t month
- uint8_t day
- uint8_t hour
- uint8_t minute
- uint8_t second
- uint8_t dayOfWeek
- int8_t timeZone
- uint8_t dayLtSavingAdj
- uint8_t radioInterface
- uint8_t TlvPresent

8.366.1 Detailed Description

This structure contains the parameters for Network Time.

Parameters

year	• Year
month	• Month • 1 is January and 12 is December
day	• Day • Range - 1 to 31
hour	• Hour • Range - 0 to 59

minute	• Minute • Range - 0 to 59
second	• Second • Range - 0 to 59
dayOfWeek	• Day of the week • 0 is Monday and 6 is Sunday
timeZone	• Offset from Universal time • The difference between local time and Universal time, in increments of 15 min • Signed Value
dayLtSavingAdj	• Daylight saving adjustment in hours • Possible values - 0, 1, and 2. • This field is ignored if radio_if is NAS_RADIO_IF_CDMA_1XEVDO
radioInterface	• Radio interface from which the information comes • Values – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA -TD-SCDMA
TlvPresent	• Tlv Present.

8.366.2 Field Documentation

8.366.2.1 uint8_t nas_timeInfo::day

8.366.2.2 uint8_t nas_timeInfo::dayLtSavingAdj

8.366.2.3 uint8_t nas_timeInfo::dayOfWeek

8.366.2.4 uint8_t nas_timeInfo::hour

8.366.2.5 uint8_t nas_timeInfo::minute

8.366.2.6 uint8_t nas_timeInfo::month

8.366.2.7 uint8_t nas_timeInfo::radioInterface

8.366.2.8 uint8_t nas_timeInfo::second

8.366.2.9 int8_t nas_timeInfo::timeZone

8.366.2.10 uint8_t nas_timeInfo::TlvPresent

8.366.2.11 uint16_t nas_timeInfo::year

8.367 nas_trueIMSI Struct Reference

Data Fields

- uint8_t mccT [3]
- uint16_t imsiT1112
- uint8_t imsiTS1 [7]
- uint8_t imsiTS2 [3]
- uint8_t imsiTaddrNum

8.367.1 Detailed Description

This structure contains True IMSI.

Parameters

mccT	• ASCII character representation of MCC_T
imsiT1112	• ASCII character representation of IMSI_T_11_12 value – 0xFFFF - Not Available
imsiTS1	• ASCII character representation of IMSI_T_S1 value
imsiTS2	• ASCII character representation of IMSI_T_S2 value
imsiTaddrNum	• Value of IMSI_T_ADDR_NUM – 0xFF - Not Available

8.367.2 Field Documentation

8.367.2.1 uint16_t nas_trueIMSI::imsiT1112

8.367.2.2 uint8_t nas_trueIMSI::imsiTaddrNum

8.367.2.3 uint8_t nas_trueIMSI::imsiTS1[7]

8.367.2.4 uint8_t nas_trueIMSI::imsiTS2[3]

8.367.2.5 uint8_t nas_trueIMSI::mccT[3]

8.368 nas_txInfo Struct Reference

Data Fields

- uint8_t isInTraffic
- int32_t txPower

8.368.1 Detailed Description

This structure contains tx radio infomation.

Parameters

isInTraffic	• Whether the device is in traffic. – 0x00 - not in traffic – 0x01 - in traffic • The txPower field is only meaningful when in the device is in traffic.
txPower	• Tx power value in 1/10 dbm.

8.368.2 Field Documentation

8.368.2.1 uint8_t nas_txInfo::isInTraffic

8.368.2.2 int32_t nas_txInfo::txPower

8.369 nas_UMTSExtInfo Struct Reference

Data Fields

- uint16_t cellId
- uint8_t plmn [3]
- uint16_t lac
- uint16_t uarfcn
- uint16_t psc
- int16_t rscp
- int16_t ecio
- int16_t squal
- int16_t srxLvl
- uint8_t umtsInst
- nas_umtsInstArr umtsInstElement [255]
- uint8_t geranInst
- nas_geranInstArr geranInstElement [255]
- uint8_t TlvPresent

8.369.1 Detailed Description

This structure contains information about the UMTS Extended Info.

Parameters

cellId	• Cell ID • 0xFFFFFFFF indicates cell ID information is not present.
plmn	• MCC/MNC information
lac	• Location area code.
uarfcn	• UTRA absolute RF channel number.
psc	• Primary scrambling code.
rscp	• Received signal code power • The received power on one code measured in dBm on the primary CPICH channel of the serving cell.
ecio	• The received energy per chip divided by the power density in the band measured in dBm on the primary CPICH channel of the serving cell.
squal	• Cell selection quality value in dB.
srxLvl	• Cell selection Rx level value in dB.
umtsInst	• Number of sets of the umtsInstElement
umtsInstElement	• See nas_umtsInstArr for more information.
geranInst	• Number of sets of the geranInstElement
geranInst-Element	• See nas_geranInstArr for more information.
TlvPresent	• Tlv Present

8.369.2 Field Documentation

8.369.2.1 uint16_t nas_UMTSExtInfo::cellId

8.369.2.2 int16_t nas_UMTSExtInfo::ecio

8.369.2.3 uint8_t nas_UMTSExtInfo::geranInst

8.369.2.4 nas_geranInstArr nas_UMTSExtInfo::geranInstElement[255]

8.369.2.5 uint16_t nas_UMTSExtInfo::lac

8.369.2.6 uint8_t nas_UMTSExtInfo::plmn[3]

8.369.2.7 uint16_t nas_UMTSExtInfo::psc

8.369.2.8 int16_t nas_UMTSExtInfo::rscp

8.369.2.9 int16_t nas_UMTSExtInfo::squal

8.369.2.10 int16_t nas_UMTSExtInfo::srxLvl

8.369.2.11 uint8_t nas_UMTSExtInfo::TlvPresent

8.369.2.12 uint16_t nas_UMTSExtInfo::uarfcn

8.369.2.13 uint8_t nas_UMTSExtInfo::umtsInst

8.369.2.14 nas_umtsInstArr nas_UMTSExtInfo::umtsInstElement[255]

8.370 nas_UMTSInfo Struct Reference

Data Fields

- uint16_t cellID
- uint8_t plmn [3]
- uint16_t lac
- uint16_t uarfcn
- uint16_t psc
- int16_t rscp
- int16_t ecio
- uint8_t umtsInst
- nas_UMTSinstInfo UMTSInstInfo [255]
- uint8_t geranInst
- nas_geranInstInfo GeranInstInfo [255]

8.370.1 Detailed Description

This structure contains information about the UMTS Network.

Parameters

cellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
plmn[NAS_PLMN_LENGTH]	• MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when nmrCellID is not present.

lac	• Location area code. • This field is ignored when nmrCellID is not present. – 0xFFFF - Not Available
uarfcn	• UTRA absolute RF channel number. – 0xFFFF - Not Available
psc	• Primary scrambling code. – 0xFFFF - Not Available
rscp	• Received signal code power. – 0xFFFF - Not Available
ecio	• ECIO(Signal-to-Interference-ratio). – 0xFFFF - Not Available
umtsInst	• Provides the number of set of UMTS info instances. • If 0(zero), then no information follows it.
UMTSInstInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_UMTSinstInfo for more information.
geranInst	• Provides the number of set of GERAN info instances. • If 0(zero), then no information follows it.
GeranInstInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_geranInstInfo for more information.

8.370.2 Field Documentation

8.370.2.1 uint16_t nas_UMTSInfo::cellID

8.370.2.2 int16_t nas_UMTSInfo::ecio

8.370.2.3 uint8_t nas_UMTSInfo::geranInst

8.370.2.4 nas_geranInstInfo nas_UMTSInfo::GeranInstInfo[255]

8.370.2.5 uint16_t nas_UMTSInfo::lac

8.370.2.6 uint8_t nas_UMTSInfo::plmn[3]

8.370.2.7 uint16_t nas_UMTSInfo::psc

8.370.2.8 int16_t nas_UMTSInfo::rscp

8.370.2.9 uint16_t nas_UMTSInfo::uarfcn

8.370.2.10 uint8_t nas_UMTSInfo::umtsInst

8.370.2.11 nas_UMTSinstInfo nas_UMTSInfo::UMTSInstInfo[255]

8.371 nas_umtsInstArr Struct Reference

Data Fields

- uint16_t umtsUarfcn
- uint16_t umtsPsc
- int16_t umtsRscp
- int16_t umtsEcio
- int16_t umtsSqual
- int16_t umtsSrxLvl
- int16_t umtsRank
- uint8_t umtsSet

8.371.1 Detailed Description

This structure contains information about the UMTS Info.

Parameters

umtsUarfcn	• UTRA absolute RF channel number.
umtsPsc	• Primary scrambling code.
umtsRscp	• Received signal code power • The received power on one code measured in dBm on the primary CPICH channel of the neighbor/monitored cell.
umtsEcio	• ECIO • The received energy per chip divided by the power density in the band measured in dBm on the primary CPICH channel of the neighbor/monitored cell.
umtsSqual	• Squal • cell selection quality value in dB.
umtsSrxLvl	• Srxlev • cell selection Rx level value in dB.

umtsRank	• Rank of the cell.
umtsSet	• Set of the cell.

8.371.2 Field Documentation

8.371.2.1 int16_t nas_umtsInstArr::umtsEcio

8.371.2.2 uint16_t nas_umtsInstArr::umtsPsc

8.371.2.3 int16_t nas_umtsInstArr::umtsRank

8.371.2.4 int16_t nas_umtsInstArr::umtsRscp

8.371.2.5 uint8_t nas_umtsInstArr::umtsSet

8.371.2.6 int16_t nas_umtsInstArr::umtsSqual

8.371.2.7 int16_t nas_umtsInstArr::umtsSrxLvl

8.371.2.8 uint16_t nas_umtsInstArr::umtsUarfcn

8.372 nas_UMTSinstInfo Struct Reference

Data Fields

- uint16_t umtsUarfcn
- uint16_t umtsPsc
- int16_t umtsRscp
- int16_t umtsEcio

8.372.1 Detailed Description

This structure contains information about the UMTS Instances in UMTS Network.

Parameters

umtsUarfcn	• UTRA absolute RF channel number.
umtsPsc	• Primary scrambling code.
umtsRscp	• Received signal code power.
umtsEcio	• ECIO(Signal-to-Interference-ratio).

8.372.2 Field Documentation

8.372.2.1 int16_t nas_UMTSinstInfo::umtsEcio

8.372.2.2 uint16_t nas_UMTSinstInfo::umtsPsc

8.372.2.3 int16_t nas_UMTSinstInfo::umtsRscp

8.372.2.4 uint16_t nas_UMTSinstInfo::umtsUarfcn

8.373 nas_umtsLTENbrCell Struct Reference

Data Fields

- uint16_t earfcn
- uint16_t pci
- uint32_t rsrp
- uint32_t rsrq
- int16_t srxlev
- uint8_t cellIsTDD

8.373.1 Detailed Description

This structure contains information about the UMTS LTE neighbour Cell.

Parameters

earfcn	• E-UTRA absolute RF channel number of the detected cell.
pci	• Physical cell ID of the detected cell. • Range is defined in 3GPP TS 36.211
rsrp	• Current received signal strength indication (in dBm) of the detected cell.
rsrq	• Current reference signal received quality (in dB) of the detected cell.
srxlev	• Cell selection Rx level (Srxlev) value of the detected cell in linear scale. • This field is only valid when wcdma_rrc_state is not NAS_WCDMA_RRC_STATE_CEL_FACH or NAS_WCDMA_RRC_STATE_CELL_DCH.
cellIsTDD	• TRUE if the cell is TDD; FALSE if the cell is FDD.

8.373.2 Field Documentation

8.373.2.1 uint8_t nas_umtsLTENbrCell::cellIsTDD

8.373.2.2 uint16_t nas_umtsLTENbrCell::earfcn

8.373.2.3 uint16_t nas_umtsLTENbrCell::pci

8.373.2.4 uint32_t nas_umtsLTENbrCell::rsrp

8.373.2.5 uint32_t nas_umtsLTENbrCell::rsrq

8.373.2.6 int16_t nas_umtsLTENbrCell::srxlev

8.374 nas_UniversalTime Struct Reference

Data Fields

- uint16_t year
- uint8_t month
- uint8_t day
- uint8_t hour
- uint8_t minute
- uint8_t second
- uint8_t dayOfWeek

8.374.1 Detailed Description

This structure contains the parameters for Universal Time Information.

Parameters

year	• Year.
month	• Month. – 1 is January and 12 is December.
day	• Day. – Range 1 to 31.
hour	• Hour. – Range 0 to 59.
minute	• Minute. – Range 0 to 59.
second	• Second. – Range 0 to 59.
dayOfWeek	• Day of the Week. – 0 is Monday and 6 is Sunday.

8.374.2 Field Documentation

8.374.2.1 uint8_t nas_UniversalTime::day

8.374.2.2 uint8_t nas_UniversalTime::dayOfWeek

8.374.2.3 uint8_t nas_UniversalTime::hour

8.374.2.4 uint8_t nas_UniversalTime::minute

8.374.2.5 uint8_t nas_UniversalTime::month

8.374.2.6 uint8_t nas_UniversalTime::second

8.374.2.7 uint16_t nas_UniversalTime::year

8.375 nas_UsageSettingTlv Struct Reference

Data Fields

- uint32_t UsageSetting
- uint8_t TlvPresent

8.375.1 Detailed Description

Contain the Usage Preference info.

Parameters

UsageSetting	• Modem usage preference to be set. • Values – 0 - Unknown – 1 - Voice centric – 2 - Data centric
TlvPresent	• Tlv Present

8.375.2 Field Documentation

8.375.2.1 uint8_t nas_UsageSettingTlv::TlvPresent

8.375.2.2 uint32_t nas_UsageSettingTlv::UsageSetting

8.376 nas_VoiceDomainPrefTlv Struct Reference

Data Fields

- uint32_t VoiceDomainPref
- uint8_t TlvPresent

8.376.1 Detailed Description

Contain the Voice Domain Preference Info.

Parameters

VoiceDomainPref	• Voice domain preference. • Values – 0x00 - Circuit-switched (CS) voice only – 0x01 - Packet-switched (PS) voice only – 0x02 - CS is preferred; PS is secondary – 0x03 - PS is preferred; CS is secondary
TlvPresent	• Tlv Present

8.376.2 Field Documentation

8.376.2.1 uint8_t nas_VoiceDomainPrefTlv::TlvPresent

8.376.2.2 uint32_t nas_VoiceDomainPrefTlv::VoiceDomainPref

8.377 nas_WCDMAARFCNTlv Struct Reference

Data Fields

- uint8_t numWcdmaArfcn
- uint16_t wcdmaArfcn [255]

8.377.1 Detailed Description

This structure contains the parameters for WCDMA ARFCN.

Parameters

numWcdmaArfcn	• Number of sets of the following elements: • wcdma_arfcn
wcdmaArfcn	• List of WCDMA RF channel numbers to be scanned. Only applicable to the following scan types: – NAS_SCAN_TYPE_PLMN – NAS_SCAN_TYPE_MODE_PREF

8.377.2 Field Documentation

8.377.2.1 uint8_t nas_WCDMAARFCNTlv::numWcdmaArfcn

8.377.2.2 uint16_t nas_WCDMAARFCNTlv::wcdmaArfcn[255]

8.378 nas_wcdmaCellInfo Struct Reference

Data Fields

- uint16_t psc
- int16_t cpich_rscp
- int16_t cpich_ecno
- int16_t srxlev

8.378.1 Detailed Description

This structure contains information about the WCDMA Cell.

Parameters

psc	• Primary scrambling code. • Range: 0 to 511.
cpich_rscp	• Absolute power level (in 1/10 dBm) of the common pilot channel as received by the UE. • Range: -120.0 dBm to -25.0 dBm
cpich_ecno	• CPICH Ec/No; ratio (in 1/10 dB) of the received energy per PN chip for the CPICH to the total received power spectral density at the UE antenna connector. • Range: -50.0 dB to 0.
srxlev	• Cell selection Rx level (Srxlev) value. • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.378.2 Field Documentation

8.378.2.1 int16_t nas_wcdmaCellInfo::cpich_ecno

8.378.2.2 int16_t nas_wcdmaCellInfo::cpich_rscp

8.378.2.3 uint16_t nas_wcdmaCellInfo::psc

8.378.2.4 int16_t nas_wcdmaCellInfo::srxlev

8.379 nas_WCDMACellInfoExt Struct Reference

Data Fields

- float wAgc
- float wTxAgc
- uint16_t wDlBler
- uint8_t TlvPresent

8.379.1 Detailed Description

This structure contains information about the WCDMA Cell Info Ext.

Parameters

wAgc	• Power in dB
wTxAgc	• Tx power in dB.
wDlBler	• Downlink block error rate percentage.
TlvPresent	• Tlv Present

8.379.2 Field Documentation

8.379.2.1 uint8_t nas_WCDMACellInfoExt::TlvPresent

8.379.2.2 float nas_WCDMACellInfoExt::wAgc

8.379.2.3 uint16_t nas_WCDMACellInfoExt::wDlBler

8.379.2.4 float nas_WCDMACellInfoExt::wTxAgc

8.380 nas_WCDMAECIOThresh Struct Reference

Data Fields

- uint8_t WCDMAECIOThreshListLen
- int16_t * pWCDMAECIOThreshList

8.380.1 Detailed Description

This structure contains WCDMA ECIO threshold related parameters.

Parameters

WCDMAECIO-ThreshListLen	• Length of the WCDMA ECIO threshold list parameter to follow
pWCDMAECIO-ThreshList	• Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for ECIO values: -31.5 to 0 (in dB)

8.380.2 Field Documentation

8.380.2.1 int16_t* nas_WCDMAECIOThresh::pWCDMAECIOThreshList

8.380.2.2 uint8_t nas_WCDMAECIOThresh::WCDMAECIOThreshListLen

8.381 nas_WCDMAInfoLTENeighborCell Struct Reference

Data Fields

- uint32_t wcdmaRRCState
- uint8_t umtsLTENbrCellLen
- nas_umtsLTENbrCell UMTSLTENbrCell [255]

8.381.1 Detailed Description

This structure contains information about the WCDMA - LTE Neighboring Cell Info Set.

Parameters

wcdmaRRC-State	• WCDMA RRC states. • Defined in 3GPP TS 25.331 • Values: – 0x00 - NAS_WCDMA_RRC_STATE_DISCONNECTED * WCDMA RRC State is IDLE – 0x01 - NAS_WCDMA_RRC_STATE_CELL_PCH * WCDMA RRC state is CELL_PCH – 0x02 - NAS_WCDMA_RRC_STATE_URA_PCH * WCDMA RRC state is URA_PCH – 0x03 - NAS_WCDMA_RRC_STATE_CELL_FACH * WCDMA RRC state is CELL_FACH – 0x04 - NAS_WCDMA_RRC_STATE_CELL_DCH * WCDMA RRC state is CELL_DCH
umtsLTENbr-CellLen	• Number of sets of UMTS LTE Neighbors.
UMTSLTENbr-Cell	• See nas_umtsLTENbrCell for more information.

8.381.2 Field Documentation

8.381.2.1 nas_umtsLTENbrCell nas_WCDMAInfoLTENeighborCell::UMTSLTENbrCell[255]

8.381.2.2 uint8_t nas_WCDMAInfoLTENeighborCell::umtsLTENbrCellLen

8.381.2.3 uint32_t nas_WCDMAInfoLTENeighborCell::wcdmaRRCState

8.382 nas_WCDMARSSIThresh Struct Reference

Data Fields

- uint8_t WCDMARSSIThreshListLen
- int16_t * pWCDMARSSIThreshList

8.382.1 Detailed Description

This structure contains WCDMA RSSI threshold related parameters.

Parameters

WCDMARSSI-ThreshListLen	• Length of the WCDMA RSSI threshold list parameter to follow
pWCDMARSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -121 to 0 (in dBm)

8.382.2 Field Documentation

8.382.2.1 int16_t* nas_WCDMARSSIThresh::pWCDMARSSIThreshList

8.382.2.2 uint8_t nas_WCDMARSSIThresh::WCDMARSSIThreshListLen

8.383 nas_WCDMASysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoWCDMA
- uint8_t lacValid
- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejCause
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]
- uint8_t hsCallStatusValid
- uint8_t hsCallStatus
- uint8_t hsIndValid
- uint8_t hsInd
- uint8_t pscValid
- uint16_t psc

8.383.1 Detailed Description

Structure for storing the WCDMA System Information.

Parameters

sysInfoWCDMA	• See nas_sysInfoCommon for more information.
lacValid	• Indicates whether the location area code is valid.. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
lac	• Location area code. • Only applies to 3GPP. – 0xFFFF - Not Available
cellIdValid	• Indicates whether the cell ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
cellId	• Cell ID. – 0xFFFFFFFF - Not Available
regRejectInfo-Valid	• Indicates whether the registration reject information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
rejectSrvDomain	• Type of service domain in which the registration is rejected. – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
rejCause	• Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. – 0xFF - Not Available
networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

MCC[NAS_PLMN_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters
MNC[NAS_PLMN_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
hsCallStatusValid	• Indicates whether the high-speed call status is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
hsCallStatus	• Call status on high speed. • Only applicable for WCDMA. – 0x00 - HSDPA and HSUPA are unsupported – 0x01 - HSDPA is supported – 0x02 - HSUPA is supported – 0x03 - HSDPA and HSUPA are supported – 0x04 - HSDPA+ is supported – 0x05 - HSDPA+ and HSUPA are supported – 0x06 - Dual-cell HSDPA+ is supported – 0x07 - Dual-cell HSDPA+ and HSUPA are supported – 0xFF - Not Available
hsIndValid	• Indicates whether high-speed service indication is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
hsInd	• High-speed service indication • Only applicable for WCDMA. – 0x00 - HSDPA and HSUPA are unsupported – 0x01 - HSDPA is supported – 0x02 - HSUPA is supported – 0x03 - HSDPA and HSUPA are supported – 0x04 - HSDPA+ is supported – 0x05 - HSDPA+ and HSUPA are supported – 0x06 - Dual-cell HSDPA+ is supported – 0x07 - Dual-cell HSDPA+ and HSUPA are supported – 0xFF - Not Available

pscValid	• Indicates whether primary scrambling code is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
psc	• Primary scrambling code. – 0xFFFF - Not Available

8.383.2 Field Documentation

8.383.2.1 uint32_t nas_WCDMASysInfo::cellId

8.383.2.2 uint8_t nas_WCDMASysInfo::cellIdValid

8.383.2.3 uint8_t nas_WCDMASysInfo::hsCallStatus

8.383.2.4 uint8_t nas_WCDMASysInfo::hsCallStatusValid

8.383.2.5 uint8_t nas_WCDMASysInfo::hsInd

8.383.2.6 uint8_t nas_WCDMASysInfo::hsIndValid

8.383.2.7 uint16_t nas_WCDMASysInfo::lac

8.383.2.8 uint8_t nas_WCDMASysInfo::lacValid

8.383.2.9 uint8_t nas_WCDMASysInfo::MCC[3]

8.383.2.10 uint8_t nas_WCDMASysInfo::MNC[3]

8.383.2.11 uint8_t nas_WCDMASysInfo::networkIdValid

8.383.2.12 uint16_t nas_WCDMASysInfo::psc

8.383.2.13 uint8_t nas_WCDMASysInfo::pscValid

8.383.2.14 uint8_t nas_WCDMASysInfo::regRejectInfoValid

8.383.2.15 uint8_t nas_WCDMASysInfo::rejCause

8.383.2.16 uint8_t nas_WCDMASysInfo::rejectSrvDomain

8.383.2.17 nas_sysInfoCommon nas_WCDMASysInfo::sysInfoWCDMA

8.384 nas_wcdmaUARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t uarfcn

8.384.1 Detailed Description

This structure contains WCDMA UARFCN infromation.

Parameters

status	• 0 - Disable • 1 - Enable
uarfcn	• UARFCN to which UE is locked

8.384.2 Field Documentation

8.384.2.1 uint8_t nas_wcdmaUARFCN::status

8.384.2.2 uint32_t nas_wcdmaUARFCN::uarfcn

8.385 NASAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t AcqOrderLen
- uint8_t AcqOrderPref [255]

8.385.1 Detailed Description

Structure used to store acquisition order preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
AcqOrderPref	- Acquisition order Preference Value

8.385.2 Field Documentation

8.385.2.1 uint8_t NASAcqOrderPrefTlv::AcqOrderLen

8.385.2.2 uint8_t NASAcqOrderPrefTlv::AcqOrderPref[255]

8.385.2.3 uint8_t NASAcqOrderPrefTlv::TlvPresent

8.386 NASBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t band_pref

8.386.1 Detailed Description

Structure used to store all Band Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
band_pref	- Roaming Indication Value

8.386.2 Field Documentation

8.386.2.1 uint64_t NASBandPreferenceTlv::band_pref

8.386.2.2 uint8_t NASBandPreferenceTlv::TlvPresent

8.387 NASCiotAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t CiotAcqOrderLen
- uint32_t CiotAcqOrderPref [255]

8.387.1 Detailed Description

Structure used to store CIOT acquisition order preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
CiotAcqOrder-Pref	- CIOT Acquisition order Preference Value

8.387.2 Field Documentation

8.387.2.1 uint8_t NASCiotAcqOrderPrefTlv::CiotAcqOrderLen

8.387.2.2 uint32_t NASCiotAcqOrderPrefTlv::CiotAcqOrderPref[255]

8.387.2.3 uint8_t NASCiotAcqOrderPrefTlv::TlvPresent

8.388 NASCiotLteOpModePrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t CiotLteOpModePref

8.388.1 Detailed Description

Structure used to store CIOT LTE Operational Mode Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
CiotLteOpMode-Pref	- Bitmask representing the CIoT LTE operational mode preference

8.388.2 Field Documentation

8.388.2.1 uint64_t NASCiotLteOpModePrefTlv::CiotLteOpModePref

8.388.2.2 uint8_t NASCiotLteOpModePrefTlv::TlvPresent

8.389 NASEmergencyModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EmerMode

8.389.1 Detailed Description

Structure used to store Emergency Mode TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
EmerMode	- Emergency Mode Value

8.389.2 Field Documentation

8.389.2.1 uint8_t NASEmergencyModeTlv::EmerMode

8.389.2.2 uint8_t NASEmergencyModeTlv::TlvPresent

8.390 NasGetLTECphyCaInfo Struct Reference

Data Fields

- NASPhyCaAggScellIndType PhyCaAggScellIndType
- NASPhyCaAggScellDlBw PhyCaAggScellDlBw
- NASPhyCaAggScellInfo PhyCaAggScellInfo
- NASPhyCaAggPcellInfo PhyCaAggPcellInfo
- NASPhyCaAggScellIndex PhyCaAggScellIndex
- NASPhyCaAggScellArray PhyCaAggScellArray

8.390.1 Detailed Description

This structure contains the parameters for aggregation event information.

Parameters

PhyCaAggScell-IndType	• see NASPhyCaAggScellIndType • Bit to check in ParamPresenceMask - **16**
PhyCaAggScell-DlBw	• see NASPhyCaAggScellDlBw • Bit to check in ParamPresenceMask - **17**
PhyCaAggScell-Info	• see NASPhyCaAggScellInfo • Bit to check in ParamPresenceMask - **18**
PhyCaAggPcell-Info	• see NASPhyCaAggPcellInfo • Bit to check in ParamPresenceMask - **19**
PhyCaAggScell-Index	• see NASPhyCaAggScellIndex • Bit to check in ParamPresenceMask - **20**
PhyCaAggScell-Array	• see NASPhyCaAggScellArray • Bit to check in ParamPresenceMask - **21**

8.390.2 Field Documentation

8.390.2.1 NASPhyCaAggPcellInfo NasGetLTECphyCaInfo::PhyCaAggPcellInfo

8.390.2.2 NASPhyCaAggScellArray NasGetLTECphyCaInfo::PhyCaAggScellArray

8.390.2.3 NASPhyCaAggScellDlBw NasGetLTECphyCaInfo::PhyCaAggScellDlBw

8.390.2.4 NASPhyCaAggScellIndex NasGetLTECphyCaInfo::PhyCaAggScellIndex

8.390.2.5 NASPhyCaAggScellIndType NasGetLTECphyCaInfo::PhyCaAggScellIndType

8.390.2.6 NASPhyCaAggScellInfo NasGetLTECphyCaInfo::PhyCaAggScellInfo

8.391 NASGWAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t GWAcqOrderPref

8.391.1 Detailed Description

Structure used to store GSM/WCDMA acquisition order preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
GWAcqOrder-Pref	- GSM/WCDMA acquisition order Preference Value

8.391.2 Field Documentation

8.391.2.1 uint32_t NASGWAcqOrderPrefTlv::GWAcqOrderPref

8.391.2.2 uint8_t NASGWAcqOrderPrefTlv::TlvPresent

8.392 NASLTEBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LTEBandPref

8.392.1 Detailed Description

Structure used to store LTE Band Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
LTEBandPref	- LTE Band Preference Value

8.392.2 Field Documentation

8.392.2.1 uint64_t NASLTEBandPreferenceTlv::LTEBandPref

8.392.2.2 uint8_t NASLTEBandPreferenceTlv::TlvPresent

8.393 NASLteM1BandPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LteM1BandPref

8.393.1 Detailed Description

Purpose: Structure used to store LTE M1 Band Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
LteM1BandPref	- Bitmask representing the LTE M1 band preference

8.393.2 Field Documentation

8.393.2.1 uint64_t NASLteM1BandPrefTlv::LteM1BandPref

8.393.2.2 uint8_t NASLteM1BandPrefTlv::TlvPresent

8.394 NASLteNasReleaseInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t nas_release
- uint8_t nas_major
- uint8_t nas_minor

8.394.1 Detailed Description

This structure contains LTE Nas Release Infomation

Parameters

Tlvresult	• unpack result
nas_release	• LTE NAS release
nas_major	• LTE NAS version major
nas_minor	• LTE NAS version minor

8.394.2 Field Documentation

8.394.2.1 uint8_t NASLteNasReleaseInfoTlv::nas_major

8.394.2.2 uint8_t NASLteNasReleaseInfoTlv::nas_minor

8.394.2.3 uint8_t NASLteNasReleaseInfoTlv::nas_release

8.394.2.4 uint8_t NASLteNasReleaseInfoTlv::TlvPresent

8.395 NASLteNB1BandPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LteNB1BandPref

8.395.1 Detailed Description

Structure used to store LTE NB1 Band Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
LteNB1BandPref	- Bitmask representing the LTE NB1 band preference

8.395.2 Field Documentation

8.395.2.1 uint64_t NASLteNB1BandPrefTlv::LteNB1BandPref

8.395.2.2 uint8_t NASLteNB1BandPrefTlv::TlvPresent

8.396 NASModePreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ModePref

8.396.1 Detailed Description

Structure used to store Mode Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
ModePref	- Mode Preference Value

8.396.2 Field Documentation

8.396.2.1 uint16_t NASModePreferenceTlv::ModePref

8.396.2.2 uint8_t NASModePreferenceTlv::TlvPresent

8.397 NASNetSelPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t NetSelPref

8.397.1 Detailed Description

Structure used to store Network Selection Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
NetSelPref	- Network Selection Preference Value

8.397.2 Field Documentation

8.397.2.1 uint8_t NASNetSelPreferenceTlv::NetSelPref

8.397.2.2 uint8_t NASNetSelPreferenceTlv::TlvPresent

8.398 NASNr5gBandPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t Nr5gBandPrefbits1_64
- uint64_t Nr5gBandPrefbits_65_128
- uint64_t Nr5gBandPrefbits_129_192
- uint64_t Nr5gBandPrefbits_193_256

8.398.1 Detailed Description

Structure used to store NR5G Band Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
Nr5gBandPref	- NR5G Band Preference

8.398.2 Field Documentation

8.398.2.1 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits1_64

8.398.2.2 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits_129_192

8.398.2.3 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits_193_256

8.398.2.4 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits_65_128

8.398.2.5 uint8_t NASNr5gBandPrefTlv::TlvPresent

8.399 NASOTAMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t message_type
- uint16_t data_len
- uint8_t data_buf [2048]

8.399.1 Detailed Description

This structure contains OTA message

Parameters

Tlvresult	• unpack result

message_type	• message type – 0 - LTE ESM uplink – 1 - LTE ESM downlink – 2 - LTE EMM uplink – 3 - LTE EMM downlink – 4 - GSM/UMTS uplink – 5 - GSM/UMTS downlink
data_len	• OTA Message Content Length
data_buf	• OTA Message Content

8.399.2 Field Documentation

8.399.2.1 uint8_t NASOTAMessageTlv::data_buf[2048]

8.399.2.2 uint16_t NASOTAMessageTlv::data_len

8.399.2.3 uint32_t NASOTAMessageTlv::message_type

8.399.2.4 uint8_t NASOTAMessageTlv::TlvPresent

8.400 NASPhyCaAggPcellInfo Struct Reference

Data Fields

- uint32_t pci
- uint32_t freq
- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint32_t iLTEbandValue
- uint8_t TlvPresent

8.400.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Pcell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.

dl_bw_value	• Downlink Bandwidth Values. • See LITEQMI_NAS_LTE_CPHY_CA_BW_NRB for more information.
scell_state	• Scell state Values. • See LITEQMI_NAS_LTE_CPHY_SCELL_STATE for more information.
TlvPresent	• Tlv Present.

8.400.2 Field Documentation

8.400.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggPcellInfo::dl_bw_value

8.400.2.2 uint32_t NASPhyCaAggPcellInfo::freq

8.400.2.3 uint32_t NASPhyCaAggPcellInfo::iLTEbandValue

8.400.2.4 uint32_t NASPhyCaAggPcellInfo::pci

8.400.2.5 uint8_t NASPhyCaAggPcellInfo::TlvPresent

8.401 NASPhyCaAggScellArray Struct Reference

Data Fields

- uint8_t cphy_scell_info_list_len
- uint16_t pci [255]
- uint16_t freq [255]
- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB cphy_ca_dl_bandwidth [255]
- uint16_t band [255]
- LITEQMI_NAS_LTE_CPHY_SCELL_STATE scell_state [255]
- uint16_t scell_idx [255]
- uint8_t TlvPresent

8.401.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

cphy_scell_info_list_len	• Number of sets of the following elements: – pci – freq – cphy_ca_dl_bandwidth – band – scell_state – scell_idx

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
cphy_ca_dl_-bandwidth	• Downlink Bandwidth Values. • See LITEQMI_NAS_LTE_CPHY_CA_BW_NRB for more information.
band	• Band value. • See LITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND for more information.
scell_state	• Scell state Values. • See LITEQMI_NAS_LTE_CPHY_SCELL_STATE for more information.
scell_idx	• Scell index Values. • Range for index values: 0 to 7.
TlvPresent	• Tlv Present.

8.401.2 Field Documentation

8.401.2.1 uint16_t NASPhyCaAggScellArray::band[255]

8.401.2.2 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellArray::cphy_ca_dl_bandwidth[255]

8.401.2.3 uint8_t NASPhyCaAggScellArray::cphy_scell_info_list_len

8.401.2.4 uint16_t NASPhyCaAggScellArray::freq[255]

8.401.2.5 uint16_t NASPhyCaAggScellArray::pci[255]

8.401.2.6 uint16_t NASPhyCaAggScellArray::scell_idx[255]

8.401.2.7 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellArray::scell_state[255]

8.401.2.8 uint8_t NASPhyCaAggScellArray::TlvPresent

8.402 NASPhyCaAggScellDlBw Struct Reference

Data Fields

- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint8_t TlvPresent

8.402.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation Downlink Bandwidth of Scell.

Parameters

dl_bw_value	• Downlink Bandwidth Values. • See LITEQMI_NAS_LTE_CPHY_CA_BW_NRB for more information.

8.402.2 Field Documentation

8.402.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellDlBw::dl_bw_value

8.402.2.2 uint8_t NASPhyCaAggScellDlBw::TlvPresent

8.403 NASPhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.403.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

scell_idx	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
TlvPresent	• Tlv Present.

8.403.2 Field Documentation

8.403.2.1 uint8_t NASPhyCaAggScellIndex::scell_idx

8.403.2.2 uint8_t NASPhyCaAggScellIndex::TlvPresent

8.404 NASPhyCaAggScellIndType Struct Reference

Data Fields

- uint32_t pci
- uint32_t freq
- LITEQMI_NAS_LTE_CPHY_SCELL_STATE scell_state
- uint8_t TlvPresent

8.404.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Indeicator Type.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
scell_state	• Scell state Values. • See LITEQMI_NAS_LTE_CPHY_SCELL_STATE for more information.
TlvPresent	• Tlv Present.

8.404.2 Field Documentation

8.404.2.1 uint32_t NASPhyCaAggScellIndType::freq

8.404.2.2 uint32_t NASPhyCaAggScellIndType::pci

8.404.2.3 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellIndType::scell_state

8.404.2.4 uint8_t NASPhyCaAggScellIndType::TlvPresent

8.405 NASPhyCaAggScellInfo Struct Reference

Data Fields

- uint32_t pci
- uint32_t freq
- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint32_t iLTEbandValue
- LITEQMI_NAS_LTE_CPHY_SCELL_STATE scell_state
- uint8_t TlvPresent

8.405.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.

freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
dl_bw_value	• Downlink Bandwidth Values. • See LITEQMI_NAS_LTE_CPHY_CA_BW_NRB for more information.
iLTEbandValue	• Band value. • Range for LTE Band class 120 to 160.
scell_state	• Scell state Values. • See LITEQMI_NAS_LTE_CPHY_SCELL_STATE for more information.
TlvPresent	• Tlv Present.

8.405.2 Field Documentation

8.405.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellInfo::dl_bw_value

8.405.2.2 uint32_t NASPhyCaAggScellInfo::freq

8.405.2.3 uint32_t NASPhyCaAggScellInfo::iLTEbandValue

8.405.2.4 uint32_t NASPhyCaAggScellInfo::pci

8.405.2.5 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellInfo::scell_state

8.405.2.6 uint8_t NASPhyCaAggScellInfo::TlvPresent

8.406 NASPRLPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t PRLPref

8.406.1 Detailed Description

Structure used to store CDMA PRL Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
PRLPref	- CDMA PRL Preference Value

8.406.2 Field Documentation

8.406.2.1 uint16_t NASPRLPreferenceTlv::PRLPref

8.406.2.2 uint8_t NASPRLPreferenceTlv::TlvPresent

8.407 NASQmiCbkNasSwiOTAMessageInd Struct Reference

Data Fields

- NASOTAMessageTlv otaMsgTlv
- NASLteNasReleaseInfoTlv nasRelInfoTlv
- NASTimeInfoTlv timeTlv

8.407.1 Detailed Description

This structure contains OTA message

Parameters

otaMsgTlv	• See NASOTAMessageTlv for more details • Bit to check in ParamPresenceMask - **1**
nasRelInfoTlv	• See NASLteNasReleaseInfoTlv for more details • Bit to check in ParamPresenceMask - **16**
timeTlv	• See NASTimeInfoTlv for more details • Bit to check in ParamPresenceMask - **17**

8.407.2 Field Documentation

8.407.2.1 NASLteNasReleaseInfoTlv NASQmiCbkNasSwiOTAMessageInd::nasRelInfoTlv

8.407.2.2 NASOTAMessageTlv NASQmiCbkNasSwiOTAMessageInd::otaMsgTlv

8.407.2.3 NASTimeInfoTlv NASQmiCbkNasSwiOTAMessageInd::timeTlv

8.408 NASQmiCbkNasSystemSelPrefInd Struct Reference

Data Fields

- NASEmergencyModeTlv EMTlv
- NASModePreferenceTlv MPTlv
- NASBandPreferenceTlv BPTlv
- NASPRLPreferenceTlv PRLPTlv
- NASRoamPreferenceTlv RPTlv
- NASLTEBandPreferenceTlv LBPTlv
- NASNetSelPreferenceTlv NSPTlv
- NASServDomainPrefTlv SDPTlv
- NASGWAcqOrderPrefTlv GWAOPTlv
- NASAcqOrderPrefTlv AOPTlv

- NASRatDisabledMaskTlv RatDMTlv
- NASCiotLteOpModePrefTlv CiotOpMPTlv
- NASLteM1BandPrefTlv M1BandPTlv
- NASLteNB1BandPrefTlv NB1BandPTlv
- NASCiotAcqOrderPrefTlv CiotAOPTlv
- NASNr5gBandPrefTlv NR5gBandPTlv

8.408.1 Detailed Description

Structure used to store all QMI Notification parameters.

Parameters

EMTlv	- Emergency mode Tlv • See NASEmergencyModeTlv for more details • Bit to check in ParamPresenceMask - **16**
MPTlv	- Mode preference Tlv • See NASModePreferenceTlv for more details • Bit to check in ParamPresenceMask - **17**
BPTlv	- Band preference Tlv • See NASBandPreferenceTlv for more details • Bit to check in ParamPresenceMask - **18**
PRLPTlv	- PRL preference Tlv • See NASPRLPreferenceTlv for more details • Bit to check in ParamPresenceMask - **19**
RPTlv	- Roaming preference Tlv • See NASRoamPreferenceTlv for more details • Bit to check in ParamPresenceMask - **20**
LBPTlv	- LTE band preference Tlv • See NASLTEBandPreferenceTlv for more details • Bit to check in ParamPresenceMask - **21**
NSPTlv	- Network selection preference Tlv • See NASNetSelPreferenceTlv for more details • Bit to check in ParamPresenceMask - **22**
SDPTlv	- Service domain preference Tlv • See NASServDomainPrefTlv for more details • Bit to check in ParamPresenceMask - **24**
GWAOPTlv	- GSM WCDMA Acquisition Order Preference Tlv • See NASGWAcqOrderPrefTlv for more details • Bit to check in ParamPresenceMask - **25**
AOPTlv	- Acquisition Order Preference Tlv • See NASAcqOrderPrefTlv for more details • Bit to check in ParamPresenceMask - **28**

RatDMTlv	- Disabled RAT Bitmask Tlv • See NASRatDisabledMaskTlv for more details • Bit to check in ParamPresenceMask - **34**
CiotOpMPTlv	- CIOT LTE Operational Mode Preference Tlv • See NASCiotLteOpModePrefTlv for more details • Bit to check in ParamPresenceMask - **36**
M1BandPTlv	- LTE M1 Band Preference Tlv • See NASLteM1BandPrefTlv for more details • Bit to check in ParamPresenceMask - **37**
NB1BandPTlv	- LTE NB1 Band Preference Tlv • See NASLteNB1BandPrefTlv for more details • Bit to check in ParamPresenceMask - **38**
CiotAOPTlv	- CIOT acquisition order preference Tlv • See NASCiotAcqOrderPrefTlv for more details • Bit to check in ParamPresenceMask - **39**
NR5gBandPTlv	- NR5G Band Preference Tlv • See NASNr5gBandPrefTlv for more details • Bit to check in ParamPresenceMask - **40**

Note: None

8.408.2 Field Documentation

8.408.2.1 NASAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::AOPTlv

8.408.2.2 NASBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::BPTlv

8.408.2.3 NASCiotAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::CiotAOPTlv

8.408.2.4 NASCiotLteOpModePrefTlv NASQmiCbkNasSystemSelPrefInd::CiotOpMPTlv

8.408.2.5 NASEmergencyModeTlv NASQmiCbkNasSystemSelPrefInd::EMTlv

8.408.2.6 NASGWAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::GWAOPTlv

8.408.2.7 NASLTEBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::LBPTlv

8.408.2.8 NASLteM1BandPrefTlv NASQmiCbkNasSystemSelPrefInd::M1BandPTlv

8.408.2.9 NASModePreferenceTlv NASQmiCbkNasSystemSelPrefInd::MPTlv

8.408.2.10 NASLteNB1BandPrefTlv NASQmiCbkNasSystemSelPrefInd::NB1BandPTlv

8.408.2.11 NASNr5gBandPrefTlv NASQmiCbkNasSystemSelPrefInd::NR5gBandPTlv

8.408.2.12 NASNetSelPreferenceTlv NASQmiCbkNasSystemSelPrefInd::NSPTlv

8.408.2.13 NASPRLPreferenceTlv NASQmiCbkNasSystemSelPrefInd::PRLPTlv

8.408.2.14 NASRatDisabledMaskTlv NASQmiCbkNasSystemSelPrefInd::RatDMTlv

8.408.2.15 NASRoamPreferenceTlv NASQmiCbkNasSystemSelPrefInd::RPTlv

8.408.2.16 NASServDomainPrefTlv NASQmiCbkNasSystemSelPrefInd::SDPTlv

8.409 NASRatDisabledMaskTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RatDisabledMask

8.409.1 Detailed Description

Structure used to store Disabled RAT Bitmask TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
RatDisabled-Mask	- Bitmask representing the radio technologies that are disabled

8.409.2 Field Documentation

8.409.2.1 uint16_t NASRatDisabledMaskTlv::RatDisabledMask

8.409.2.2 uint8_t NASRatDisabledMaskTlv::TlvPresent

8.410 NASRoamPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RoamPref

8.410.1 Detailed Description

Structure used to store Roaming Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
RoamPref	- Roaming Preference Value

8.410.2 Field Documentation

8.410.2.1 uint16_t NASRoamPreferenceTlv::RoamPref

8.410.2.2 uint8_t NASRoamPreferenceTlv::TlvPresent

8.411 NASServDomainPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SrvDomainPref

8.411.1 Detailed Description

Structure used to store Service domain preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
SrvDomainPref	- Service Domain Preference Value

8.411.2 Field Documentation

8.411.2.1 uint32_t NASServDomainPrefTlv::SrvDomainPref

8.411.2.2 uint8_t NASServDomainPrefTlv::TlvPresent

8.412 NASServingSystemInfo Struct Reference

Data Fields

- uint8_t registrationState
- uint8_t csAttachState
- uint8_t psAttachState
- uint8_t selectedNetwork
- uint8_t radioInterfaceNo
- uint8_t radioInterfaceList [255]
- uint8_t hdrPersonality

8.412.1 Detailed Description

This structure will hold the serving system parameters information

Parameters

registrationState	- Registration state of the mobile • 0 - QMI_NAS_NOT_REGISTERED Not registered;mobile is not currently searching for a new network to provide service • 1 - QMI_NAS_REGISTERED Registered with a network • 2 - QMI_NAS_NOT_REGISTERED_SEARCHING Not registered, but mobile is currently searching for a new network to provide service • 3 - QMI_NAS_REGISTRATION_DENIED Registration denied by the visible network • 4 - QMI_NAS_REGISTRATION_UNKNOWN Registration state is unknown
csAttachState	- Circuit Switch domain attach state of the mobile • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached

psAttachState	- Packet domain attach state of the mobile • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached
selectedNetwork	- Type of selected radio access network • 0x00 - Unknown • 0x01 - 3GPP2 network • 0x02 - 3GPP network
radioInterfaceNo	- Number of radio interfaces currently in use; this indicates how many radio_if identifiers follow this field
radioInterface-List	- Radio interface currently in use (each is 1 byte) • 0x00 - None (no service) • 0x01 - cdma2000 1X • 0x02 - cdma2000 HRPD (1xEV-DO) • 0x03 - AMPS • 0x04 - GSM • 0x05 - UMTS • 0x08 - LTE
hdrPersonality	- HDR personality information (valid only for EVDO) • 0x00 - Unknown • 0x01 - HRPD • 0x02 - eHRPD

Note: None

8.412.2 Field Documentation

8.412.2.1 uint8_t NASServingSystemInfo::csAttachState

8.412.2.2 uint8_t NASServingSystemInfo::hdrPersonality

8.412.2.3 uint8_t NASServingSystemInfo::psAttachState

8.412.2.4 uint8_t NASServingSystemInfo::radioInterfaceList[255]

8.412.2.5 uint8_t NASServingSystemInfo::radioInterfaceNo

8.412.2.6 uint8_t NASServingSystemInfo::registrationState

8.412.2.7 uint8_t NASServingSystemInfo::selectedNetwork

8.413 NASTimeInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t time

8.413.1 Detailed Description

This structure contains LTE Nas time Infomation

Parameters

Tlvresult	• unpack result
time	• Seconds in local time since Jan. 6th 1980 00:00:00 UTC

8.413.2 Field Documentation

8.413.2.1 uint64_t NASTimeInfoTlv::time

8.413.2.2 uint8_t NASTimeInfoTlv::TlvPresent

8.414 newMTMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMTMessageInfo MTMessageInfo

8.414.1 Detailed Description

This structure contains MT message TLV information.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
MTMessageInfo	• MT Message • See sMSMTMessageInfo for more information

8.414.2 Field Documentation

8.414.2.1 sMSMTMessageInfo newMTMessageTlv::MTMessageInfo

8.414.2.2 uint8_t newMTMessageTlv::TlvPresent

8.415 pack_audio_SLQSGetAudioPathConfig_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Item

8.415.1 Detailed Description

This structure contains the pack parameters for Get Audio Path Config.

Parameters

Profile	• Audio Profile – 0-9
Item	• Item – 0 - AV_EC – 1 - AV_NS – 2 - AV_TXVOL – 3 - AV_DTMFTXG – 4 - AV_CODECSTG – 5 - AV_TXPCMIIRFLTR – 6 - AV_RXPCMIIRFLTR – 7 - AV_MICGAIN – 8 - AV_RXAGC – 9 - AV_TXAGC – 10 - AV_RXAGCLIST – 11 - AV_RXAVCLIST – 12 - AV_TXAGCLIST

8.415.2 Field Documentation

8.415.2.1 uint8_t pack_audio_SLQSGetAudioPathConfig_t::Item

8.415.2.2 uint8_t pack_audio_SLQSGetAudioPathConfig_t::Profile

8.416 pack_audio_SLQSGetAudioProfile_t Struct Reference

Data Fields

- uint8_t Generator

8.416.1 Detailed Description

This structure contains the pack parameters to Get Audio Profile.

Parameters

Generator	• Audio Generator – 0 - Voice – 1 - Key Beep – 2 - MIDI

8.416.2 Field Documentation

8.416.2.1 uint8_t pack_audio_SLQSGetAudioProfile_t::Generator

8.417 pack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator
- uint8_t Volume
- uint8_t Item

8.417.1 Detailed Description

This structure contains the pack parameters for SLQSGetAudioVolTLBConfig.

Parameters

Profile	• Audio Profile – 0-9
Generator	• Audio Generator – 0-2
Volume	• Audio Volume Level – 0-7
Item	• Item – 13 - AV_RXVOLDB – 14 - AV_DTMFVOLDB – 15 - AV_PAD

8.417.2 Field Documentation

8.417.2.1 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Generator

8.417.2.2 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Item

8.417.2.3 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Profile

8.417.2.4 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Volume

8.418 pack_audio_SLQSSetAudioPathConfig_t Struct Reference

Data Fields

- uint8_t Profile

- uint8_t * pECMode
- uint8_t * pNSEnable
- uint16_t * pTXGain
- uint16_t * pDTMFTXGain
- uint16_t * pCodecSTGain
- audio_TXPCMIIRFltr * pTXPCMIIRFltr
- audio_RXPCMIIRFltr * pRXPCMIIRFltr
- uint8_t * pRXAVCAGCSwitch
- uint8_t * pTXAVCSwitch
- audio_RXAGCList * pRXAGCList
- audio_RXAVCList * pRXAVCList
- audio_TXAGCList * pTXAGCList

8.418.1 Detailed Description

This structure contains the pack parameters for SLQSSetAudioPathConfig .

Parameters

Profile	[Mandatory] • Audio Profile – 0-9
pECMode	[Optional] • AV_EC – 0 - Echo cancellation off – 1 - Handset echo mode – 2 - Headset mode – 3 - Car kit mode – 4 - Speaker Mode
pNSEnable	[Optional] • Noise Suppression – 0 - Noise suppression off – 1 - Noise suppression on
pTXGain	[Optional] • TX Voice volume – 0x0000 - 0xffff
pDTMFTXGain	[Optional] • AV_DTMFTXG – 0x0000 - 0xffff
pCodecSTGain	[Optional] • AV_CODECSTG – 0x0000 - 0xffff
pTXPCMIIRFltr	[Optional] • See audio_TXPCMIIRFltr for more information
pRXPCMIIRFltr	[Optional] • See audio_RXPCMIIRFltr for more information

pRXAVCAGC-Switch	[Optional] • RX AVC/AGC Switch
pTXAVCSwitch	[Optional] • TX AVC Switch
pRXAGCList	[Optional] • See audio_RXAGCList for more information
pRXAVCList	[Optional] • See audio_RXAVCList for more information
pTXAGCList	[Optional] • See audio_TXAGCList for more information

8.418.2 Field Documentation

8.418.2.1 uint16_t* pack_audio_SLQSSetAudioPathConfig_t::pCodecSTGain

8.418.2.2 uint16_t* pack_audio_SLQSSetAudioPathConfig_t::pDTMFTXGain

8.418.2.3 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pECMode

8.418.2.4 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pNSEnable

8.418.2.5 uint8_t pack_audio_SLQSSetAudioPathConfig_t::Profile

8.418.2.6 audio_RXAGCList* pack_audio_SLQSSetAudioPathConfig_t::pRXAGCList

8.418.2.7 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pRXAVCAGCSwitch

8.418.2.8 audio_RXAVCList* pack_audio_SLQSSetAudioPathConfig_t::pRXAVCList

8.418.2.9 audio_RXPCMIIRFltr* pack_audio_SLQSSetAudioPathConfig_t::pRXPCMIIRFltr

8.418.2.10 audio_TXAGCList* pack_audio_SLQSSetAudioPathConfig_t::pTXAGCList

8.418.2.11 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pTXAVCSwitch

8.418.2.12 uint16_t* pack_audio_SLQSSetAudioPathConfig_t::pTXGain

8.418.2.13 audio_TXPCMIIRFltr* pack_audio_SLQSSetAudioPathConfig_t::pTXPCMIIRFltr

8.419 pack_audio_SLQSSetAudioProfile_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t EarMute
- uint8_t MicMute
- uint8_t Generator
- uint8_t Volume

8.419.1 Detailed Description

This structure contains pack parameters to Set Audio Profile.

Parameters

Profile	• Audio Profile – 0 - Handset – 1 - Headset – 2 - Car Kit – 3 - Speaker phone – 4 - Auxiliary – 5 - TTY – 6 - Auxiliary external PCM – 7 - Primary external PCM – 8 - External slave PCM – 9 - I2S
EarMute	• Ear Mute Setting – 0 - unmuted – 1 - muted
MicMute	• MIC Mute Setting – 0 - unmuted – 1 - muted
Generator	• Audio Generator – 0 - Voice – 1 - Key Beep – 2 - MIDI
Volume	• Audio Volume Level – 0 to 7

8.419.2 Field Documentation

8.419.2.1 uint8_t pack_audio_SLQSSetAudioProfile_t::EarMute

8.419.2.2 uint8_t pack_audio_SLQSSetAudioProfile_t::Generator

8.419.2.3 uint8_t pack_audio_SLQSSetAudioProfile_t::MicMute

8.419.2.4 uint8_t pack_audio_SLQSSetAudioProfile_t::Profile

8.419.2.5 uint8_t pack_audio_SLQSSetAudioProfile_t::Volume

8.420 pack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator
- uint8_t Volume
- uint8_t Item
- uint16_t VolValue

8.420.1 Detailed Description

This structure contains the pack parameters for SLQSSetAudioVolTLBConfig.

Parameters

Profile	• Audio Profile – 0-9
Generator	• Audio Generator – 0-2
Volume	• Audio Volume Level – 0-7
Item	• Item – 13 - AV_RXVOLDB – 14 - AV_DTMFVOLDB – 15 - AV_PAD
Value	• Value to be set to the volume table

8.420.2 Field Documentation

8.420.2.1 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Generator

8.420.2.2 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Item

8.420.2.3 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Profile

8.420.2.4 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Volume

8.420.2.5 uint16_t pack_audio_SLQSSetAudioVolTLBConfig_t::VolValue

8.421 pack_cat_CATSendEnvelopeCommand_t Struct Reference

Data Fields

- uint32_t cmdID
- uint32_t dataLen
- uint8_t * pData

8.421.1 Detailed Description

Structure used to store envelope command data.

Parameters

cmdID	• Envelope command type – 0x01 - Menu Selection – 0x02 - Event DL User activity – 0x03 - Event DL Idle Screen Available – 0x04 - Event DL Language Selection
dataLen	• Length of pData in bytes
pData	• Encoded envelope data as defined in ETSI TS 102 223, section 7 [Smart Cards: Card Application Toolkit (CAT) - Release 4]

8.421.2 Field Documentation

8.421.2.1 uint32_t pack_cat_CATSendEnvelopeCommand_t::cmdID

8.421.2.2 uint32_t pack_cat_CATSendEnvelopeCommand_t::dataLen

8.421.2.3 uint8_t* pack_cat_CATSendEnvelopeCommand_t::pData

8.422 pack_cat_CATSendTerminalResponse_t Struct Reference

Data Fields

- uint32_t refID
- uint32_t dataLen
- uint8_t * pData

8.422.1 Detailed Description

Structure used to terminal response data.

Parameters

refID	• Proactive command reference ID. The value should be the same as indicated in the CAT event callback data for the relevant proactive command.

dataLen	• Terminal response data length
pData	• Terminal response for the relevant proactive command encoded as per ETSI TS 102 223, section 6.8 [Smart Cards: Card Application Toolkit (CAT) - Release 4]

8.422.2 Field Documentation

8.422.2.1 uint32_t pack_cat_CATSendTerminalResponse_t::dataLen

8.422.2.2 uint8_t* pack_cat_CATSendTerminalResponse_t::pData

8.422.2.3 uint32_t pack_cat_CATSendTerminalResponse_t::refID

8.423 pack_cat_SetCATEventCallback_t Struct Reference

Data Fields

- uint32_t eventMask

8.423.1 Detailed Description

This structure contains sEnables/disables the CAT event callback pack variable.

Parameters

eventMask	• bitmask of CAT events to register for – 0x00000001 - Display Text – 0x00000002 - Get In-Key – 0x00000004 - Get Input – 0x00000008 - Setup Menu – 0x00000010 - Select Item – 0x00000020 - Send SMS - Alpha Identifier – 0x00000040 - Setup Event: User Activity – 0x00000080 - Setup Event: Idle Screen Notify – 0x00000100 - Setup Event: Language Sel Notify – 0x00000200 - Setup Idle Mode Text – 0x00000400 - Language Notification – 0x00000800 - Refresh – 0x00001000 - End Proactive Session

8.423.2 Field Documentation

8.423.2.1 uint32_t pack_cat_SetCATEventCallback_t::eventMask

8.424 pack_ctl_GetClientId_t Struct Reference

Data Fields

- uint8_t serviceType

8.424.1 Detailed Description

Structure to store get QMI client ID for service.

Parameters

serviceType	• Selected service from which a client ID is to be obtained.

8.424.2 Field Documentation

8.424.2.1 uint8_t pack_ctl_GetClientId_t::serviceType

8.425 pack_ctl_ReleaseClientId_t Struct Reference

Data Fields

- uint8_t serviceType
- uint8_t clientId

8.425.1 Detailed Description

Structure to store release QMI client ID for service.

Parameters

serviceType	• Selected service from which a client ID is to be released.
clientId	• Previously assigned client ID is to be released.

8.425.2 Field Documentation

8.425.2.1 uint8_t pack_ctl_ReleaseClientId_t::clientId

8.425.2.2 uint8_t pack_ctl_ReleaseClientId_t::serviceType

8.426 pack_dms_ActivateAutomatic_t Struct Reference

Data Fields

- uint8_t actCode [81]

8.426.1 Detailed Description

This structure contains UIM activate automation information

Parameters

actCode	• NULL-terminated string representing activation code (maximum string length of 12). • Specific carrier requirements may dictate actual activation code that is applicable, e.g., "∗22899"

8.426.2 Field Documentation

8.426.2.1 uint8_t pack_dms_ActivateAutomatic_t::actCode[81]

8.427 pack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint8_t list_type
- uint16_t Tlvresult

8.427.1 Detailed Description

This structure contains which customization id or the list type want to retrieve from modem. This TLV is only applicable for 9x30 modules so far

Parameters

cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.
list_type	• list type requested
Tlvresult	• Pack Result

8.427.2 Field Documentation

8.427.2.1 uint8_t pack_dms_GetCustFeaturesV2_t::cust_id[64+1]

8.427.2.2 uint8_t pack_dms_GetCustFeaturesV2_t::list_type

8.427.2.3 uint16_t pack_dms_GetCustFeaturesV2_t::Tlvresult

8.428 pack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint8_t spc [6]

8.428.1 Detailed Description

This structure contains UIM reset to factory default information

Parameters

spc[IN]	• Service programming code in ASCII format (digits 0 to 9 only)

8.428.2 Field Documentation

8.428.2.1 uint8_t pack_dms_ResetToFactoryDefaults_t::spc[6]

8.429 pack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint8_t activationState

8.429.1 Detailed Description

This structure is used to store Set Service Activation Status callback parameter pack.

Parameters

activationState	• Service activation state. • Values – 0 - Do not report – 1 - Report activation state changes

8.429.2 Field Documentation

8.429.2.1 uint8_t pack_dms_SetActivationStatusCallback_t::activationState

8.430 pack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t crashAction

8.430.1 Detailed Description

This structure is used to store pack_dms_SetCrashAction parameters.

Parameters

crashAction	• 0 - USB Memory Download. Modem will reset after a crash and will stay in USB download mode with only DM port enumerated. • 1 - Reset. Modem will reset and come back in ONLINE mode. Minimal crash data will be available and can be extracted with at!gcdump? AT command • 2 - No Action

8.430.2 Field Documentation

8.430.2.1 uint8_t pack_dms_SetCrashAction_t::crashAction

8.431 pack_dms_SetCustFeature_t Struct Reference

Data Fields

- uint32_t GpsEnable
- uint8_t DisableIMSI
- uint16_t IPFamSupport
- uint8_t RMAutoConnect
- uint8_t GPSSel
- uint8_t SMSSupport
- uint8_t IsVoiceEnabled
- uint8_t DHCPRelayEnabled
- uint8_t GPSLPM

8.431.1 Detailed Description

This structure contains current settings of custom features pack parameters

Parameters

GpsEnable	• describes if GPS is enabled or disabled • values: – 0x00 - GPS is disabled – 0x01 - GPS is enabled
DisableIMSI	• optional 1 byte parameter • describes if IMSI display is enabled or disabled • values: – 0x00 - Allow display of IMSI – 0x01 - Do not display IMSI

IPFamSupport	• optional 2 byte BitMask • bitmask representing the IP families supported • values: – 0x01 - IPv4 – 0x02 - IPv6 – 0x04 - IPv4v6
RMAutoConnect	• optional 1 byte parameter • QMI Mode RM Net Auto Connect Support • values: – 0x00 - Not Supported – 0x01 - Supported
GPSSel	• optional 1 byte parameter • GPS Antenna Select • values: – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied
SMSSupport	• optional 1 byte parameter • SMS support • values: – 0x00 - Not supported – 0x01 - supported • Used to determine whether or not to hide SMS from user
IsVoiceEnabled	• optional 1 byte parameter • Voice support • values: – 0x00 - Enable voice on both AT and QMI interface (default) – 0x01 - Reserved – 0x02 - Disable voice on both AT and QMI interface
DHCPRelay-Enabled	• optional 1 byte parameter • DHCP Relay support • values: – 0x00 - Disable DHCP relay – 0x01 - Enable DHCP relay

GPSLPM	• optional 1 byte parameter • GPSLPM support • values: – 0x00 - Enable GPS in Low Power Mode – 0x01 - Disable GPS in Low Power Mode

8.431.2 Field Documentation

8.431.2.1 uint8_t pack_dms_SetCustFeature_t::DHCPRelayEnabled

8.431.2.2 uint8_t pack_dms_SetCustFeature_t::DisableIMSI

8.431.2.3 uint32_t pack_dms_SetCustFeature_t::GpsEnable

8.431.2.4 uint8_t pack_dms_SetCustFeature_t::GPSLPM

8.431.2.5 uint8_t pack_dms_SetCustFeature_t::GPSSel

8.431.2.6 uint16_t pack_dms_SetCustFeature_t::IPFamSupport

8.431.2.7 uint8_t pack_dms_SetCustFeature_t::IsVoiceEnabled

8.431.2.8 uint8_t pack_dms_SetCustFeature_t::RMAutoConnect

8.431.2.9 uint8_t pack_dms_SetCustFeature_t::SMSSupport

8.432 pack_dms_SetCustFeaturesV2_t Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint16_t value_length
- uint8_t cust_value [8+1]
- uint16_t Tlvresult

8.432.1 Detailed Description

This structure contains customization settings set to modem pack

Parameters

cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.
value_length	• length of cust_value field
cust_value	• Customization Setting Value (Maximum 8 bytes)

Tlvresult	• Pack Result

8.432.2 Field Documentation

8.432.2.1 uint8_t pack_dms_SetCustFeaturesV2_t::cust_id[64+1]

8.432.2.2 uint8_t pack_dms_SetCustFeaturesV2_t::cust_value[8+1]

8.432.2.3 uint16_t pack_dms_SetCustFeaturesV2_t::Tlvresult

8.432.2.4 uint16_t pack_dms_SetCustFeaturesV2_t::value_length

8.433 pack_dms_SetEventReport_t Struct Reference

Data Fields

- uint8_t mode

8.433.1 Detailed Description

This structure is used to store pack_dms_SetEventReport parameters.

Parameters

mode	• Operating Mode • 0 - reset • 1 - set • 2 - no change

8.433.2 Field Documentation

8.433.2.1 uint8_t pack_dms_SetEventReport_t::mode

8.434 pack_dms_SetIndicationRegister_t Struct Reference

Data Fields

- uint8_t PSMStatus
- uint8_t PSMCfgChangeInfo
- uint8_t RptIMSCapability

8.434.1 Detailed Description

This structure contains the DMS Register Indication request parameters

Parameters

PSMStatus	• Power Save Mode Status • Values – 0 - Do not report – 1 - Report PSM status – 2 - No Change
PSMCfgChange-Info	• Power Save Mode Configuration Change Information • Values – 0 - Do not report – 1 - Report Power Save Mode configuration changes – 2 - No Change
RptIMS-Capability	• Report IMS Capability • Values – 0 - Do not report – 1 - Report IMS capability – 2 - No Change

8.434.2 Field Documentation

8.434.2.1 uint8_t pack_dms_SetIndicationRegister_t::PSMCfgChangeInfo

8.434.2.2 uint8_t pack_dms_SetIndicationRegister_t::PSMStatus

8.434.2.3 uint8_t pack_dms_SetIndicationRegister_t::RptIMSCapability

8.435 pack_dms_SetPower_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t Tlvresult

8.435.1 Detailed Description

Structure to store pack the operating mode of the device.

Parameters

mode	• Selected operating mode – See qaGobiApiTablePowerModes.h for power modes
Tlvresult	• pack Tlv Result.

8.435.2 Field Documentation

8.435.2.1 uint32_t pack_dms_SetPower_t::mode

8.435.2.2 uint16_t pack_dms_SetPower_t::Tlvresult

8.436 pack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint8_t USBComp
- uint16_t Tlvresult

8.436.1 Detailed Description

This structure is used to store USB composition information pack paramters

Parameters

USBComp	• Current USB Composition • Values: – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM
Tlvresult	• pack Tlv Result.

8.436.2 Field Documentation

8.436.2.1 uint16_t pack_dms_SetUSBComp_t::Tlvresult

8.436.2.2 uint8_t pack_dms_SetUSBComp_t::USBComp

8.437 pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint8_t resetInfoInd

8.437.1 Detailed Description

This structure contains set registration state for different indication pack

Parameters

resetInfoInd[IN]	• Values – 0 - Disable – 1 - Enable

8.437.2 Field Documentation

8.437.2.1 uint8_t pack_dms_SLQSDmsSwiIndicationRegister_t::resetInfoInd

8.438 pack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference

Data Fields

- uint8_t * pPsmEnableState
- uint32_t * pActiveTimer
- uint32_t * pPeriodicUpdateTimer

8.438.1 Detailed Description

This structure contains the Power Save Mode (PSM) configuration parameter.

Parameters

pPsmEnable-State	[Optional] • PSM Enable State • Values – 0 - PSM is not enabled – 1 - PSM is enabled
pActiveTimer	[Optional] • Active Timer • PSM active timer value (in seconds). • ActiveTimer encoding is the GPRS timer 2 encoding specified in Table 10.5.172 of 3GPP TS 24.008.

pPeriodic-UpdateTimer	[Optional] • Periodic Update Timer • PSM periodic update timer value (in seconds) • Periodic Update Timer encoding is the GPRS timer 3 encoding specified in Table 10.5.163a of 3GPP TS 24.008.

8.438.2 Field Documentation

8.438.2.1 uint32_t* pack_dms_SLQSSetPowerSaveModeConfig_t::pActiveTimer

8.438.2.2 uint32_t* pack_dms_SLQSSetPowerSaveModeConfig_t::pPeriodicUpdateTimer

8.438.2.3 uint8_t* pack_dms_SLQSSetPowerSaveModeConfig_t::pPsmEnableState

8.439 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint8_t clear

8.439.1 Detailed Description

This structure contains pack_dms_SLQSSwiGetCrashInfo parameters

Parameters

clear	• 0 - do not clear crash data after response • 1 - clear crash data after response

8.439.2 Field Documentation

8.439.2.1 uint8_t pack_dms_SLQSSwiGetCrashInfo_t::clear

8.440 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.440.1 Detailed Description

This structure is used to store pack_dms_SLQSSwiSetDyingGaspCfg parameters

Parameters

pDestSMSNum[IN]	• SMS Destination Number as string of 8 bit ASCII Characters Max 20 chars. • Optional parameter.
pDestSMSContent[IN]	• SMS COntent as a string of 8 bit ASCII text characters Max 160 chars. • Optional parameter.

8.440.2 Field Documentation

8.440.2.1 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSContent

8.440.2.2 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSNum

8.441 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- char manString [255]
- char modelString [255]
- char swVerString [255]
- char plasmaIDString [255]
- char hostID [255]

8.441.1 Detailed Description

This structure contains SWI set host device info unpack information

Parameters

manString[IN]	• Host Device Manufacturer Name • Null terminated ASCII String
modelString[IN]	• Host Device Model String • Null terminated ASCII string.
swVerString[IN]	• Host Device Software Version String • Null terminated ASCII string
plasmaIDString[IN]	• Host Device Plasma ID String • Null terminated alphanumeric ASCII string
HostID[IN]	• Device Host ID String • Null terminated alphanumeric ASCII string

8.441.2 Field Documentation

8.441.2.1 char pack_dms_SLQSSwiSetHostDevInfo_t::hostID[255]

8.441.2.2 char pack_dms_SLQSSwiSetHostDevInfo_t::manString[255]

8.441.2.3 char pack_dms_SLQSSwiSetHostDevInfo_t::modelString[255]

8.441.2.4 char pack_dms_SLQSSwiSetHostDevInfo_t::plasmaIDString[255]

8.441.2.5 char pack_dms_SLQSSwiSetHostDevInfo_t::swVerString[255]

8.442 pack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- char nameString [255]
- char versionString [255]

8.442.1 Detailed Description

This structure contains SWI set host OS info pack information

Parameters

nameString[IN]	Host device manufacture String • NULL terminated ASCII string.
VersionString[IN]	Host device model String • NULL terminated ASCII string.

8.442.2 Field Documentation

8.442.2.1 char pack_dms_SLQSSwiSetOSInfo_t::nameString[255]

8.442.2.2 char pack_dms_SLQSSwiSetOSInfo_t::versionString[255]

8.443 pack_dms_SwiSetEventReport_t Struct Reference

Data Fields

- uint8_t * pTempReport
- uint8_t * pVoltReport
- uint8_t * pUIMStatusReport

8.443.1 Detailed Description

This structure is used to store pack_dms_SwiSetEventReport parameters.

Parameters

pTempReport	(optional) • Temperature indicator. – 0 - Do not report – 1 - Report when state changes
pVoltReport	(optional) • Voltage Indicator. – 0 - Do not report – 1 - Report when state changes
pUIMStatus-Report	(optional) • UIM status Indicator. – 0 - Do not report – 1 - Report when state changes

8.443.2 Field Documentation

8.443.2.1 uint8_t* pack_dms_SwiSetEventReport_t::pTempReport

8.443.2.2 uint8_t* pack_dms_SwiSetEventReport_t::pUIMStatusReport

8.443.2.3 uint8_t* pack_dms_SwiSetEventReport_t::pVoltReport

8.444 pack_dms_SwiUimSelect_t Struct Reference

Data Fields

- uint8_t uim_select

8.444.1 Detailed Description

This structure is used to store pack_dms_SwiUimSelect parameters.

Parameters

uim_select	• Selection of UIM Card – 0 - External UIM Interface – 1 - Embedded UIM – 2 - Remote UIM – 3 - Auto switch

Note

Auto switch is applicable only if UIMAUTOSWITCH feature is enabled. This feature will activate the auto-switch mechanism if it is inactive.

8.444.2 Field Documentation

8.444.2.1 uint8_t pack_dms_SwiUimSelect_t::uim_select

8.445 pack_dms_UIMChangePIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t oldValue [255]
- uint8_t newValue [255]

8.445.1 Detailed Description

This structure contains UIM Unblock PIN Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
oldValue[IN]	• Old PIN value of PIN to change
newValue[IN]	• New PIN value of PIN to change

8.445.2 Field Documentation

8.445.2.1 uint8_t pack_dms_UIMChangePIN_t::id

8.445.2.2 uint8_t pack_dms_UIMChangePIN_t::newValue[255]

8.445.2.3 uint8_t pack_dms_UIMChangePIN_t::oldValue[255]

8.446 pack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facility

8.446.1 Detailed Description

This structure contains UIM get control key status information

Parameters

facility[IN]	• MT or network facility – 0 - Network Personalization (PN) – 1 - Network Subset Personalization (PU) – 2 - Service Provider Personalization (PP) – 3 - Corporate Personalization (PC) – 4 - UIM Personalization (PF)

8.446.2 Field Documentation

8.446.2.1 uint8_t pack_dms_UIMGetControlKeyStatus_t::facility

8.447 pack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.447.1 Detailed Description

This structure contains UIM Get ICCID pack

Parameters

Tlvresult	• Pack result.

8.447.2 Field Documentation

8.447.2.1 swi_uint256_t pack_dms_UIMGetICCID_t::ParamPresenceMask

8.447.2.2 uint16_t pack_dms_UIMGetICCID_t::Tlvresult

8.448 pack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t facility
- uint8_t facilityState
- uint8_t facilityCk [8]

8.448.1 Detailed Description

This structure contains UIM Set control key protection information

Parameters

facility[IN]	• MT or network facility – 0 - Network Personalization (PN) – 1 - Network Subset Personalization (PU) – 2 - Service Provider Personalization (PP) – 3 - Corporate Personalization (PC) – 4 - UIM Personalization (PF)
facilityState[IN]	• UIM facility state – 0 - Deactivated – 1 - Activated – 2 - Blocked
facilityCk[IN]	• Facility depersonalization control key, string in ASCII text (maximum 8 bytes)

8.448.2 Field Documentation

8.448.2.1 uint8_t pack_dms_UIMSetControlKeyProtection_t::facility

8.448.2.2 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityCk[8]

8.448.2.3 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityState

8.449 pack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t id
- uint8_t bEnable
- uint8_t value [255]

8.449.1 Detailed Description

This structure contains PIN Protection Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
bEnable[IN]	• Enable/disable PIN protection, 0 = Disable
pValue[IN]	• PIN value of the PIN to be enabled/disabled

8.449.2 Field Documentation

8.449.2.1 uint8_t pack_dms_UIMSetPINProtection_t::bEnable

8.449.2.2 uint8_t pack_dms_UIMSetPINProtection_t::id

8.449.2.3 uint8_t pack_dms_UIMSetPINProtection_t::value[255]

8.450 pack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t facility
- uint8_t facilityCk [8]

8.450.1 Detailed Description

This structure contains UIM unblock Control Key information

Parameters

facility[IN]	• MT or network facility – 0 - Network Personalization (PN) – 1 - Network Subset Personalization (PU) – 2 - Service Provider Personalization (PP) – 3 - Corporate Personalization (PC) – 4 - UIM Personalization (PF)
facilityCk[IN]	• Facility depersonalization control key, string in ASCII text (maximum 8 bytes)

8.450.2 Field Documentation

8.450.2.1 uint8_t pack_dms_UIMUnblockControlKey_t::facility

8.450.2.2 uint8_t pack_dms_UIMUnblockControlKey_t::facilityCk[8]

8.451 pack_dms_UIMUnblockPIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t pukValue [255]
- uint8_t newPin [255]

8.451.1 Detailed Description

This structure contains UIM Unblock PIN Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
pukValue[IN]	• PUK value of PIN to be unblocked
newPin[IN]	• New PIN value for the PIN to be unblocked

8.451.2 Field Documentation

8.451.2.1 uint8_t pack_dms_UIMUnblockPIN_t::id

8.451.2.2 uint8_t pack_dms_UIMUnblockPIN_t::newPin[255]

8.451.2.3 uint8_t pack_dms_UIMUnblockPIN_t::pukValue[255]

8.452 pack_dms_UIMVerifyPIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t value [255]

8.452.1 Detailed Description

This structure contains PIN Value Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
value[IN]	• PIN value of the PIN to be enabled/disabled

8.452.2 Field Documentation

8.452.2.1 uint8_t pack_dms_UIMVerifyPIN_t::id

8.452.2.2 uint8_t pack_dms_UIMVerifyPIN_t::value[255]

8.453 pack_dsd_register_ui_info_changes_t Struct Reference

Data Fields

- uint8_t * pReport_ui_changes
- uint8_t * pSuppress_so_change
- uint8_t * pSuppress_null_bearer_reason

8.453.1 Detailed Description

This structure contains the DSD Register ui changes request parameters.

Parameters

pReport_ui_changes	• Register/Deregister for UI information changes. Values: 0 - Do not report 1 - Report UI change (default).
pSuppress_so_change	• Suppress SO mask. Values: – 0 - Do not suppress only SO mask changes (Default). – 1 - Suppress SO mask change. The indication is not generated only if SO mask changes from the last indication. Ex: Indiation is sent when technology, rat_value, or ui_mask change from the last indication.
PSuppress_null_bearer_reason	• Suppress null bearer reason. Values: – 0 - Do not suppress only null bearer reason changes (Default). – 1 - Suppress null bearer reason change. The indication is not generated only if null bearer reason changes from the last indication.

Note

FOR_5G

8.453.2 Field Documentation

8.453.2.1 uint8_t* pack_dsd_register_ui_info_changes_t::pReport_ui_changes

8.453.2.2 uint8_t* pack_dsd_register_ui_info_changes_t::pSuppress_null_bearer_reason

8.453.2.3 uint8_t* pack_dsd_register_ui_info_changes_t::pSuppress_so_change

8.454 pack_dsd_system_status_changes_t Struct Reference

Data Fields

- uint8_t * pLimit_so_mask_change_ind
- uint8_t * pReport_data_system_status_changes
- uint8_t * pReport_pref_tech_change_only
- uint8_t * pReport_null_bearer_reason

8.454.1 Detailed Description

This structure contains SYSTEM Status Change Req info

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pLimit_so_mask_change_ind	• If this TLV is provided in the request message, the indication is generated only if the rat_value changes from the last indication.
pReport_data_system_status_changes	• Register/Deregister for Data System Status Changes Values: 0 - Do not report 1 - Report data system status change(default).
pReport_pref_tech_change_only	• Report only Preferred technology changes If this TLV is provided and set to true in the request message, the indication is generated only if the Preferred Technology value (i.e. only technology field of the Preferred System) changes from the last indication.
pReport_null_bearer_reason	• Report NULL Bearer Reason Report NULL bearer reason with global preferred system status.

Note

For_5G

8.454.2 Field Documentation

8.454.2.1 uint8_t* pack_dsd_system_status_changes_t::pLimit_so_mask_change_ind

8.454.2.2 uint8_t* pack_dsd_system_status_changes_t::pReport_data_system_status_changes

8.454.2.3 uint8_t* pack_dsd_system_status_changes_t::pReport_null_bearer_reason

8.454.2.4 uint8_t* pack_dsd_system_status_changes_t::pReport_pref_tech_change_only

8.455 pack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.455.1 Detailed Description

This structure contains the Get Image Preference information pack

Parameters

Tlvresult	• Pack result

8.455.2 Field Documentation

8.455.2.1 uint16_t pack_fms_GetImagesPreference_t::Tlvresult

8.456 pack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.456.1 Detailed Description

This structure contains the Get Stored Images pack

Parameters

Tlvresult	• Pack result

8.456.2 Field Documentation

8.456.2.1 uint16_t pack_fms_GetStoredImages_t::Tlvresult

8.457 pack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t imageListSize
- FMSPrefImageList * pImageList
- uint32_t bForceDownload
- uint8_t modemindex
- uint16_t Tlvresult

8.457.1 Detailed Description

This structure contains the Set Images Preference pack

Parameters

imageListSize	• Image List Size
pImageList	• Image List • See FMSPrefImageList
bForceDownload	• 0 - Not Force Donwload. • 1 - Focrce Download.
modemindex	• Modem Index.

Tlvresult	• Unpack result

8.457.2 Field Documentation

8.457.2.1 uint32_t pack_fms_SetImagesPreference_t::bForceDownload

8.457.2.2 uint32_t pack_fms_SetImagesPreference_t::imageListSize

8.457.2.3 uint8_t pack_fms_SetImagesPreference_t::modemindex

8.457.2.4 FMSPrefImageList* pack_fms_SetImagesPreference_t::pImageList

8.457.2.5 uint16_t pack_fms_SetImagesPreference_t::Tlvresult

8.458 pack_ims_SLQSImsConfigIndicationRegister_t Struct Reference

Data Fields

- uint8_t * pSIPConfigEvents
- uint8_t * pRegMgrConfigEvents
- uint8_t * pSMSConfigEvents
- uint8_t * pUserConfigEvents
- uint8_t * pVoIPConfigEvents

8.458.1 Detailed Description

This structure contains parameters of IMS Config Indication Register

Parameters

pSIPConfig-Events(optional)	• Registration Indication For SIP Configuration Events. • When this registration is enabled, the device learns of SIP config events via the QMI_IMS_SIP-_CONFIG_IND indication. – 0x00 - Disable – 0x01 - Enable
pRegMgrConfig-Events(optional)	• Registration Indication For Registration Manager Configuration Events. • When this registration is enabled, the device learns of Reg Mgr config events via the QMI_IMS-_REG_MGR_CONFIG_IND indication. – 0x00 - Disable – 0x01 - Enable
pSMSConfig-Events(optional)	• Registration Indication For SMS Configuration Events. • When this registration is enabled, the device learns of SMS config events via the QMI_IMS_S-MS_CONFIG_IND indication. – 0x00 - Disable – 0x01 - Enable

pUserConfig-Events(optional)	• Registration Indication For User Configuration Events. • When this registration is enabled, the device learns of user config events via the QMI_IMS_USER_CONFIG_IND indication. – 0x00 - Disable – 0x01 - Enable
pVoIPConfig-Events(optional)	• Registration Indication For VoIP Configuration Events. • When this registration is enabled, the device learns of VOIP config events via the QMI_IMS_VOIP_CONFIG_IND indication. – 0x00 - Disable – 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.458.2 Field Documentation

8.458.2.1 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pRegMgrConfigEvents

8.458.2.2 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pSIPConfigEvents

8.458.2.3 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pSMSConfigEvents

8.458.2.4 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pUserConfigEvents

8.458.2.5 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pVoIPConfigEvents

8.459 pack_ims_SLQSSetIMSSMSConfig_t Struct Reference

Data Fields

- uint8_t * pSMSFormat
- uint8_t * pSMSOverIPNwInd
- uint8_t * pPhoneCtxtURILen
- uint8_t * pPhoneCtxtURI

8.459.1 Detailed Description

This structure contains the SLQSSetIMSSMSConfig pack parameters.

Parameters

pSMSFormat	• SMS format – 0 - 3GPP – 1 - 3GPP2

pSMSOverIPNwInd	• SMS over IP Network Indication Flag – TRUE - Turn on mobile-originated SMS – FALSE - Turn off mobile-originated SMS
pPhoneCtxtURILen	• Length of Phone context Universal Resource Identifier to follow
pPhoneCtxtURI	• Phone context universal resource identifier • Length of this string must be specified in pPhoneCtxtURILen parameter

8.459.2 Field Documentation

8.459.2.1 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pPhoneCtxtURI

8.459.2.2 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pPhoneCtxtURILen

8.459.2.3 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pSMSFormat

8.459.2.4 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pSMSOverIPNwInd

8.460 pack_ims_SLQSSetIMSUserConfig_t Struct Reference

Data Fields

- uint8_t * pIMSDomainLen
- uint8_t * pIMSDomain

8.460.1 Detailed Description

This structure contains the SLQSSetIMSUserConfig pack parameters.

Parameters

pIMSDomainLen	• Length of IMS Domain Name to follow
pIMSDomain	• IMS domain name

8.460.2 Field Documentation

8.460.2.1 uint8_t* pack_ims_SLQSSetIMSUserConfig_t::pIMSDomain

8.460.2.2 uint8_t* pack_ims_SLQSSetIMSUserConfig_t::pIMSDomainLen

8.461 pack_ims_SLQSSetIMSVoIPConfig_t Struct Reference

Data Fields

- uint16_t * pSessionExpiryTimer
- uint16_t * pMinSessionExpiryTimer
- uint8_t * pAmrWbEnable
- uint8_t * pScrAmrEnable
- uint8_t * pScrAmrWbEnable
- uint8_t * pAmrMode
- uint16_t * pAmrWBMode
- uint8_t * pAmrOctetAligned
- uint8_t * pAmrWBOctetAligned
- uint16_t * pRingingTimer
- uint16_t * pRingBackTimer
- uint16_t * pRTPRTCPInactTimer

8.461.1 Detailed Description

This structure contains the SLQSSetIMSVoIPConfig pack parameters.

Parameters

pSessionExpiry-Timer	• Session duration, in seconds
pMinSession-ExpiryTimer	• Minimum allowed value for session expiry timer, in seconds
pAmrWbEnable	• Flag to enable/disable Adaptive Multirate Codec(AMR) WideBand(WB) audio • Values: – True - Enable – False - Disable
pScrAmrEnable	• Flag to enable/disable Source Control Rate(SCR) for AMR NarrowBand (NB) • Values: – True - Enable – False - Disable
pScrAmrWb-Enable	• Flag to enable/disable SCR for AMR WB Audio • Values: – True - Enable – False - Disable

pAmrMode	• BitMask for AMR NB modes allowed • Values: – 0x1 - 4.75 kbps – 0x2 - 5.15 kbps – 0x4 - 5.9 kbps – 0x8 - 6.17 kbps – 0x10 - 7.4 kbps – 0x20 - 7.95 kbps – 0x40 - 10.2 kbps – 0x80 - 12.2 kbps
pAmrWBMode	• BitMask for AMR WB modes allowed • Values: – 0x1 - 6.60 kbps – 0x2 - 8.85 kbps – 0x4 - 12.65 kbps – 0x8 - 14.25 kbps – 0x10 - 15.85 kbps – 0x20 - 18.25 kbps – 0x40 - 19.85 kbps – 0x80 - 23.05 kbps – 0x100 - 23.85 kbps
pAmrOctet-Aligned	• Flag to indicate if the octet is aligned for AMR NB Audio • Values: – True - Aligned – False - Not aligned, Bandwidth Efficient mode
pAmrWBOctet-Aligned	• Flag to indicate if the octet is aligned for AMR WB Audio • Values: – True - Aligned – False - Not aligned, Bandwidth Efficient mode
pRingingTimer	• Duration of ringing timer, in seconds. The ringing timer starts on the ringing event. If the call is not answered within the duration of this timer, the call is disconnected.
pRingBackTimer	• Duration of ringback timer, in seconds. The ringback timer starts on the ringback event. If the call is not answered within the duration of this timer, the call is disconnected.
pRTPRTCP-InactTimer	• Duration of RTP/RTCP inactivity timer, in seconds. If no RTP/RTCP packet is received prior to the expiry of this timer, the call is disconnected.

8.461.2 Field Documentation

8.461.2.1 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrMode

8.461.2.2 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrOctetAligned

8.461.2.3 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrWbEnable

8.461.2.4 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrWBMode

8.461.2.5 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrWBOctetAligned

8.461.2.6 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pMinSessionExpiryTimer

8.461.2.7 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pRingBackTimer

8.461.2.8 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pRingingTimer

8.461.2.9 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pRTPRTCPInactTimer

8.461.2.10 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pScrAmrEnable

8.461.2.11 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pScrAmrWbEnable

8.461.2.12 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pSessionExpiryTimer

8.462 pack_ims_SLQSSetRegMgrConfig_t Struct Reference

Data Fields

- uint16_t * pPriCSCFPort
- uint8_t * pCSCFPortNameLen
- uint8_t * pCSCFPortName
- uint8_t * pIMSTestMode

8.462.1 Detailed Description

This structure contains the SLQSSetRegMgrConfig pack parameters.

Parameters

pPriCSCFPort	• Primary call session control function port
pCSCFPort-NameLen	• Length of the CSCF Port name parameter to follow
pCSCFPort-Name	• Call Session control port, fully qualified domain name • Length of this string must be specified in pCSCFPortNameLen parameter
pIMSTestMode	• IMS Test mode Enabled. – TRUE - Enable, no IMS registration – FALSE - Disable, IMS registration is initiated

8.462.2 Field Documentation

8.462.2.1 uint8_t* pack_ims_SLQSSetRegMgrConfig_t::pCSCFPortName

8.462.2.2 uint8_t* pack_ims_SLQSSetRegMgrConfig_t::pCSCFPortNameLen

8.462.2.3 uint8_t* pack_ims_SLQSSetRegMgrConfig_t::pIMSTestMode

8.462.2.4 uint16_t* pack_ims_SLQSSetRegMgrConfig_t::pPriCSCFPort

8.463 pack_ims_SLQSSetSIPConfig_t Struct Reference

Data Fields

- uint16_t * pSIPLocalPort
- uint32_t * pTimerSIPReg
- uint32_t * pSubscribeTimer
- uint32_t * pTimerT1
- uint32_t * pTimerT2
- uint32_t * pTimerTf
- uint8_t * pSigCompEnabled

8.463.1 Detailed Description

This structure contains the SLQSSetSIPConfig pack parameters.

Parameters

pSIPLocalPort	• Primary call session control function SIP port number
pTimerSIPReg	• Initial SIP registration duration from the User equipment, in seconds
pSubscribeTimer	• Duration of the subscription by the UE for IMS registration notifications, in seconds
pTimerT1	• RTT estimate, in milliseconds
pTimerT2	• The maximum retransmit interval for non-invite requests and invite responses, in milliseconds
pTimerTf	• Non-invite transaction timeout timer, in milliseconds
pSigComp-Enabled	• Sig Comp Status – TRUE - Enable – FALSE - Disable

8.463.2 Field Documentation

8.463.2.1 uint8_t* pack_ims_SLQSSetSIPConfig_t::pSigCompEnabled

8.463.2.2 uint16_t* pack_ims_SLQSSetSIPConfig_t::pSIPLocalPort

8.463.2.3 uint32_t* pack_ims_SLQSSetSIPConfig_t::pSubscribeTimer

8.463.2.4 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerSIPReg

8.463.2.5 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerT1

8.463.2.6 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerT2

8.463.2.7 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerTf

8.464 pack_imsa_SLQSRegisterIMSAIndication_t Struct Reference

Data Fields

- int has_RegStatusConfig
- uint8_t RegStatusConfig
- int has_ServiceStatusConfig
- uint8_t ServiceStatusConfig
- int has_RatHandoverStatusConfig
- uint8_t RatHandoverStatusConfig
- int has_PdpStatusConfig
- uint8_t PdpStatusConfig

8.464.1 Detailed Description

This structure contains parameters of IMSA Config Indication Register

Parameters

RegStatus-Config(optional)	• Register Indication For Registration status. • When this registration is enabled, the device learns of Registration status via the QMI_IMSA_-REGISTRATION_STATUS_IND indication. – 0x00 - Disable – 0x01 - Enable
ServiceStatus-Config(optional)	• Register Indication For Service status Events. • When this registration is enabled, the device learns of Service status via the QMI_IMSA_SER-VICE_STATUS_IND indication. – 0x00 - Disable – 0x01 - Enable
RatHandover-Status-Config(optional)	• Registration Indication For RAT handover status. • When this registration is enabled, the device learns of RAT handover status via the QMI_IMSA-_RAT_HANDOVER_STATUS_IND indication. – 0x00 - Disable – 0x01 - Enable

PdpStatus-Config(optional)	• PDP Status Configuration. – 0x00 - Disable – 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.464.2 Field Documentation

8.464.2.1 int pack_imsa_SLQSRegisterIMSAIndication_t::has_PdpStatusConfig

8.464.2.2 int pack_imsa_SLQSRegisterIMSAIndication_t::has_RatHandoverStatusConfig

8.464.2.3 int pack_imsa_SLQSRegisterIMSAIndication_t::has_RegStatusConfig

8.464.2.4 int pack_imsa_SLQSRegisterIMSAIndication_t::has_ServiceStatusConfig

8.464.2.5 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::PdpStatusConfig

8.464.2.6 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::RatHandoverStatusConfig

8.464.2.7 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::RegStatusConfig

8.464.2.8 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::ServiceStatusConfig

8.465 pack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- loc_SVInfo * pSVInfo
- loc_GnssData * pGnssData
- loc_CellDb * pCellDb
- loc_ClkInfo * pClkInfo
- loc_BdsSVInfo * pBdsSVInfo
- uint16_t Tlvresult

8.465.1 Detailed Description

This structure contains LOC delete assist data pack

Parameters

pSVInfo	• Pointer to struct loc_SVInfo. See loc_SVInfo for more information
pGnssData	• Pointer to struct loc_GnssData. See loc_GnssData for more information
pCellDb	• Pointer to struct loc_CellDb. See loc_CellDb for more information

pClkInfo	• Pointer to struct loc_ClkInfo. See loc_ClkInfo for more information
pBdsSVInfo	• Pointer to struct loc_BdsSVInfo. See loc_BdsSVInfo for more information
Tlvresult	• Pack delete assist data request result.

8.465.2 Field Documentation

8.465.2.1 loc_BdsSVInfo* pack_loc_Delete_Assist_Data_t::pBdsSVInfo

8.465.2.2 loc_CellDb* pack_loc_Delete_Assist_Data_t::pCellDb

8.465.2.3 loc_ClkInfo* pack_loc_Delete_Assist_Data_t::pClkInfo

8.465.2.4 loc_GnssData* pack_loc_Delete_Assist_Data_t::pGnssData

8.465.2.5 loc_SVInfo* pack_loc_Delete_Assist_Data_t::pSVInfo

8.465.2.6 uint16_t pack_loc_Delete_Assist_Data_t::Tlvresult

8.466 pack_loc_EventRegister_t Struct Reference

Data Fields

- uint64_t eventRegister
- uint16_t Tlvresult

8.466.1 Detailed Description

This structure contains the Parameter for RegisterEvents

Parameters

eventRegister	• Specifies the events that the control point is interested in receiving. -Values – 0x00000001 - to receive position report event indications – 0x00000002 - to receive satellite report event indications.These reports are sent at a 1 Hz rate. – 0x00000004 - to receive NMEA reports for position and satellites in view.The report is at a 1 Hz rate. – 0x00000008 - to receive NI Notify/Verify request event indications – 0x00000010 - to receive time injection request event indications. – 0x00000020 - to receive predicted orbits request event indications. – 0x00000040 - to receive position injection request event indications. – 0x00000080 - to receive engine state report event indications. – 0x00000100 - to receive fix session status report event indications. – 0x00000200 - to receive Wi-Fi position request event indications. – 0x00000400 - to receive notifications from the location engine indicating its readiness to accept data from the sensors(accelerometer, gyroscope, etc.). – 0x00000800 - to receive time sync requests from the GPS engine. Time sync enables the GPS engine to synchronize its clock with the sensor processor's clock. – 0x00001000 - to receive Stationary Position Indicator (SPI) streaming report indications. – 0x00002000 - to receive location server requests. These requests are generated when the service wishes to establish a connection with a location server. – 0x00004000 - to receive notifications related to network-initiated Geofences. These events notify the client when a network-initiated Geofence is added, deleted, or edited. – 0x00008000 - to receive Geofence alerts. These alerts are generated to inform the client of the changes that may affect a Geofence, e.g., if GPS is turned off or if the network is unavailable. – 0x00010000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach report is for a single Geofence. – 0x00020000 - to register for pedometer control requests from the location engine. The location engine sends this event to control the injection of pedometer reports. – 0x00040000 - to register for motion data control requests from the location engine. The location engine sends this event to control the injection of motion data. – 0x00080000 - to receive notification when a batch is full. The location engine sends this event to notify of Batch Full for ongoing batching session. – 0x00100000 - to receive position report indications along with an ongoing batching session. The location engine sends this event to notify the batched position report while a batching session is ongoing. – 0x00200000 - to receive Wi-Fi Access Point (AP) data inject request event indications. – 0x00400000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach notification is for multiple Geofences. Breaches from multiple Geofences are all batched and sent in the same notification. – 0x00800000 - to receive notifications from the location engine indicating its readiness to accept vehicle data (vehicle accelerometer, vehicle angular rate, vehicle odometry, etc.). – 0x01000000 - to receive system clock and satellite measurement report events (system clock, SV time, Doppler, etc.). – 0x02000000 - to receive satellite position reports as polynomials.Reports are generated only for the GNSS satellite constellations that are enabled using QMI_LOC_SET_GNSS_-CONSTELL_REPORT_CONFIG.

Note

Multiple events can be registered by OR the individual masks and sending them in this TLV. All unused bits in this mask must be set to 0.

Parameters

TlvResult	• Pack result.

8.466.2 Field Documentation

8.466.2.1 uint64_t pack_loc_EventRegister_t::eventRegister

8.466.2.2 uint16_t pack_loc_EventRegister_t::TlvResult

8.467 pack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint32_t extPowerState
- uint16_t TlvResult

8.467.1 Detailed Description

This structure contains the Parameter External Power Source State pack.

Parameters

extPowerState	• Specifies the Power state; injected by the control point. • Values – 0 - Device is not connected to an external power source – 1 - Device is connected to an external power source – 2 - Unknown external power state
TlvResult	• Pack result.

8.467.2 Field Documentation

8.467.2.1 uint32_t pack_loc_SetExtPowerState_t::extPowerState

8.467.2.2 uint16_t pack_loc_SetExtPowerState_t::TlvResult

8.468 pack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t TlvResult

8.468.1 Detailed Description

This structure contains Set Operation Mode pack

Parameters

mode	• Valid values: – eQMI_LOC_OPER_MODE_DEFAULT (1) - Use the default engine mode – eQMI_LOC_OPER_MODE_MSB (2) - Use the MS-based mode – eQMI_LOC_OPER_MODE_MSA (3) - Use the MS-assisted mode – eQMI_LOC_OPER_MODE_STANDALONE (4) - Use Standalone mode – eQMI_LOC_OPER_MODE_CELL_ID (5) - Use cell ID; this mode is only valid for GSM/UMTS networks – eQMI_LOC_OPER_MODE_WWAN (6) - Use WWAN measurements to calculate the position; if this mode is set, AFLT will be used for 1X networks and OTDOA will be used for LTE networks
Tlvresult	• Pack result.

8.468.2 Field Documentation

8.468.2.1 uint32_t pack_loc_SetOperationMode_t::mode

8.468.2.2 uint16_t pack_loc_SetOperationMode_t::Tlvresult

8.469 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint32_t xid
- uint16_t Tlvresult

8.469.1 Detailed Description

This structure contains Set Operation Mode pack

Parameters

xid	• Identifies the transaction. • The transaction ID is returned in the Get Best Available Position indication.
Tlvresult	• Pack result.

8.469.2 Field Documentation

8.469.2.1 uint16_t pack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.469.2.2 uint32_t pack_loc_SLQSLOCGetBestAvailPos_t::xid

8.470 pack_loc_SLQSLOCGetServer_t Struct Reference

Data Fields

- uint32_t serverType
- int has_serverAddrTypeMask
- uint8_t serverAddrTypeMask

8.470.1 Detailed Description

This structure contains parameters to Gets the location server. Please check has_<Param_Name> field for presence of optional parameters

Parameters

serverType	• Type of server. • Valid values – eQMI_LOC_SERVER_TYPE_CDMA_PDE (1) - Server type is CDMA PDE – eQMI_LOC_SERVER_TYPE_CDMA_MPC (2) - Server type is CDMA MPC – eQMI_LOC_SERVER_TYPE_UMTS_SLP (3) - Server type is UMTS SLP – eQMI_LOC_SERVER_TYPE_CUSTOM_PDE (4)- Server type is custom PDE
serverAddrTypeMask	• Optional parameter • Type of address the client wants. If unspecified, the indication will contain all the types of addresses it has for the specified server type. • Valid bitmasks – 0x01 - IPv4 – 0x02 - IPv6 – 0x04 - URL

8.470.2 Field Documentation

8.470.2.1 int pack_loc_SLQSLOCGetServer_t::has_serverAddrTypeMask

8.470.2.2 uint8_t pack_loc_SLQSLOCGetServer_t::serverAddrTypeMask

8.470.2.3 uint32_t pack_loc_SLQSLOCGetServer_t::serverType

8.471 pack_loc_SLQSLOCInjectPosition_t Struct Reference

Data Fields

- double latitude
- int has_latitude
- double longitude
- int has_longitude
- float horUncCircular

- int has_horUncCircular
- uint8_t horConfidence
- int has_horConfidence
- uint32_t horReliability
- int has_horReliability
- float altitudeWrtEllipsoid
- int has_altitudeWrtEllipsoid
- float altitudeWrtMeanSeaLevel
- int has_altitudeWrtMeanSeaLevel
- float vertUnc
- int has_vertUnc
- uint8_t vertConfidence
- int has_vertConfidence
- uint32_t vertReliability
- int has_vertRelicability
- altSrcInfo_t altitudeSrcInfo
- int has_altitudeSrcInfo
- uint64_t timestampUtc
- int has_timestampUtc
- uint32_t timestampAge
- int has_timestampAge
- uint32_t positionSrc
- int has_positionSrc
- float rawHorUncCircular
- int has_rawHorUncCircular
- uint8_t rawHorConfidence
- int has_rawHorConfidence

8.471.1 Detailed Description

This structure contains LOC Inject Position parameters Please check has_<Param_Name> field for presence of optional parameters

Parameters

latitude	• Optional parameter • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude • Note - This field must be specified together with pLongitude and pHorUncCircular.
longitude	• Optional parameter • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude • Note - This field must be specified together with pLatitude and pHorUncCircular.

horUncCircular	• Optional parameter • Horizontal position uncertainty. • Units - Meters • Note - This field must be specified together with pLatitude and pLongitude.
horConfidence	• Optional parameter • Horizontal confidence. • Units - Percent • Values – Valid Values - 1 to 99 – Invalid Values - 0, 101 to 255 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with horizontal uncertainty. If not specified when pHorUncCircular is set, the default value is 50.
horReliability	• Optional parameter – Values * 0 - Location reliability is not set. * 1 - Location reliability is very low; use it at your own risk * 2 - Location reliability is low; little or no cross-checking is possible. * 3 - Location reliability is medium; limited cross-check passed * 4 - Location reliability is high; strong cross-check passed
altitudeWrt-Ellipsoid	• Optional parameter • Altitude With Respect to Ellipsoid. • Units - Meters • Values – Positive - height – Negative = depth
altitudeWrt-MeanSeaLevel	• Optional parameter • Altitude With Respect to Sea Level. • Units - Meters
vertUnc	• Optional parameter • Vertical uncertainty. • Units - Meters • Note - This is mandatory if either pAltitudeWrtEllipsoid or pAltitudeWrtMeanSeaLevel is specified.

vertConfidence	• Optional parameter • Vertical confidence. • Units - Percentage • Values – Valid Values - 0 to 99 – Invalid Values - 0, 100-256 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with the vertical uncertainty. If not specified, the default value will be 50.
vertReliability	• Optional parameter – Values * 0 - Location reliability is not set. * 1 - Location reliability is very low; use it at your own risk. * 2 - Location reliability is low; little or no cross-checking is possible * 3 - Location reliability is medium; limited cross-check passed * 4 - Location reliability is high; strong cross-check passed
altitudeSrcInfo	• Optional parameter – Pointer to struct altitudeSrcInfo. See altitudeSrcInfo for more information
timestampUtc	• Optional parameter • UTC timestamp • Units - Milliseconds since Jan. 1, 1970
timestampAge	• Optional parameter • Position age, which is an estimate of how long ago this fix was made. • Units - Milliseconds
positionSrc	• Optional parameter • Source from which this position was obtained • Valid values – 0 - Position source is GNSS – 1 - Position source is Cell ID – 2 - Position source is Enhanced Cell ID – 3 - Position source is Wi-Fi – 4 - Position source is Terrestrial – 5 - Position source is GNSS Terrestrial Hybrid – 6 - Other sources • Note - If altitude is specified and the altitude source is not specified, the engine assumes that the altitude was obtained using the specified position source. – If both altitude and altitude source are specified, the engine assumes that only latitude and longitude were obtained using the specified position source.

rawHorUncCircular	• Optional parameter • Horizontal position uncertainty (circular) without any optimization. • Units - Meters
rawHorConfidence	• Optional parameter • Horizontal confidence associated with raw horizontal uncertainty • Units: Percent • Values – Valid values - 1 to 99 – Invalid values - 0, 101 to 255 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with raw horizontal uncertainty. If not specified when rawHorUncCircular is set, the default value is 50.

8.471.2 Field Documentation

8.471.2.1 altSrcInfo_t pack_loc_SLQSLOCInjectPosition_t::altitudeSrcInfo

8.471.2.2 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtEllipsoid

8.471.2.3 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtMeanSeaLevel

8.471.2.4 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeSrcInfo

8.471.2.5 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtEllipsoid

8.471.2.6 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtMeanSeaLevel

8.471.2.7 int pack_loc_SLQSLOCInjectPosition_t::has_horConfidence

8.471.2.8 int pack_loc_SLQSLOCInjectPosition_t::has_horReliability

8.471.2.9 int pack_loc_SLQSLOCInjectPosition_t::has_horUncCircular

8.471.2.10 int pack_loc_SLQSLOCInjectPosition_t::has_latitude

8.471.2.11 int pack_loc_SLQSLOCInjectPosition_t::has_longitude

8.471.2.12 int pack_loc_SLQSLOCInjectPosition_t::has_positionSrc

8.471.2.13 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorConfidence

8.471.2.14 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorUncCircular

8.471.2.15 int pack_loc_SLQSLOCInjectPosition_t::has_timestampAge

8.471.2.16 int pack_loc_SLQSLOCInjectPosition_t::has_timestampUtc

8.471.2.17 int pack_loc_SLQSLOCInjectPosition_t::has_vertConfidence

8.471.2.18 int pack_loc_SLQSLOCInjectPosition_t::has_vertRelicability

8.471.2.19 int pack_loc_SLQSLOCInjectPosition_t::has_vertUnc

8.471.2.20 uint8_t pack_loc_SLQSLOCInjectPosition_t::horConfidence

8.471.2.21 uint32_t pack_loc_SLQSLOCInjectPosition_t::horReliability

8.471.2.22 float pack_loc_SLQSLOCInjectPosition_t::horUncCircular

8.471.2.23 double pack_loc_SLQSLOCInjectPosition_t::latitude

8.471.2.24 double pack_loc_SLQSLOCInjectPosition_t::longitude

8.471.2.25 uint32_t pack_loc_SLQSLOCInjectPosition_t::positionSrc

8.471.2.26 uint8_t pack_loc_SLQSLOCInjectPosition_t::rawHorConfidence

8.471.2.27 float pack_loc_SLQSLOCInjectPosition_t::rawHorUncCircular

8.471.2.28 uint32_t pack_loc_SLQSLOCInjectPosition_t::timestampAge

8.471.2.29 uint64_t pack_loc_SLQSLOCInjectPosition_t::timestampUtc

8.471.2.30 uint8_t pack_loc_SLQSLOCInjectPosition_t::vertConfidence

8.471.2.31 uint32_t pack_loc_SLQSLOCInjectPosition_t::vertReliability

8.471.2.32 float pack_loc_SLQSLOCInjectPosition_t::vertUnc

8.472 pack_loc_SLQSLOCInjectSensorData_t Struct Reference

Data Fields

- int has_opaqueId
- uint32_t opaqueId
- int has_accleroData
- sensorData_t acceleroData
- int has_gyroData
- sensorData_t gyroData
- int has_acceleroTimeSrc
- uint32_t acceleroTimeSrc
- int has_gyroTimeSrc
- uint32_t gyroTimeSrc
- int has_accelTemp
- tempData_t accelTemp
- int has_gyroTemp
- tempData_t gyroTemp

8.472.1 Detailed Description

This structure contains parameters to inject sensor data into the GNSS location engine Please check has_<Param-_Name> field for presence of optional parameters

Parameters

opaqueId	• Opaque Identifier (Optional parameter) • An opaque identifier that is sent in by the client that will be echoed in the indication so the client can relate the indication to the request.
acceleroData	• 3-Axis Accelerometer Data (Optional parameter) • Pointer to struct sensorData. See sensorData_t for more information
gyroData	• 3-Axis Cyroscope Data (Optional parameter) • Pointer to struct sensorData. See sensorData_t for more information
acceleroTimeSrc	• 3-Axis Accelerometer Data Time Source (Optional parameter) • The location service uses this field to identify the time reference used in the accelerometer data time stamps. • If not specified, the location service assumes that the time source for the accelerometer data is unknown. • Valid values – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
gyroTimeSrc	• 3-Axis Gyroscope Data Time Source (Optional) • The location service uses this field to identify the time reference used in the gyroscope data time stamps. • If not specified, the location service assumes that the time source for the gyroscope data is unknown. • Valid values – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
accelTemp	• Accelerometer Temperature Data (Optional parameter) • Pointer to struct tempratureData. See tempData_t for more information
gyroTemp	• Gyroscope Temperature Data (Optional parameter) • Pointer to struct tempratureData. See tempData_t for more information

8.472.2 Field Documentation

8.472.2.1 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::acceleroData

8.472.2.2 uint32_t pack_loc_SLQSLOCInjectSensorData_t::acceleroTimeSrc

8.472.2.3 tempData_t pack_loc_SLQSLOCInjectSensorData_t::accelTemp

8.472.2.4 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::gyroData

8.472.2.5 tempData_t pack_loc_SLQSLOCInjectSensorData_t::gyroTemp

8.472.2.6 uint32_t pack_loc_SLQSLOCInjectSensorData_t::gyroTimeSrc

8.472.2.7 int pack_loc_SLQSLOCInjectSensorData_t::has_acceleroTimeSrc

8.472.2.8 int pack_loc_SLQSLOCInjectSensorData_t::has_accelTemp

8.472.2.9 int pack_loc_SLQSLOCInjectSensorData_t::has_accleroData

8.472.2.10 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroData

8.472.2.11 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTemp

8.472.2.12 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTimeSrc

8.472.2.13 int pack_loc_SLQSLOCInjectSensorData_t::has_opaqueId

8.472.2.14 uint32_t pack_loc_SLQSLOCInjectSensorData_t::opaqueId

8.473 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference

Data Fields

- uint64_t timeMsec
- uint32_t timeUncMsec

8.473.1 Detailed Description

This structure contains inject UTC time parameter.

Parameters

timeMsec	• The UTC time since Jan. 1, 1970
timeUncMsec	• The time Uncertainty

8.473.2 Field Documentation

8.473.2.1 uint64_t pack_loc_SLQSLOCInjectUTCTime_t::timeMsec

8.473.2.2 uint32_t pack_loc_SLQSLOCInjectUTCTime_t::timeUncMsec

8.474 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference

Data Fields

- uint32_t state
- int has_confidence
- uint8_t confidence

8.474.1 Detailed Description

This structure contains parameters to set current cradle mount configuration Please check has_<Param_Name> field for presence of optional parameters

Parameters

state	• Cradle Mount State • Valid values: – 0 - Device is mounted on the cradle – 1 - Device is not mounted on the cradle – 2 - Unknown cradle mount state
confidence	• Cradle Mount Confidence (Optional) • Confidence in the Cradle Mount state expressed as a percentage. • Range - 0 to 100

8.474.2 Field Documentation

8.474.2.1 uint8_t pack_loc_SLQSLOCSetCradleMountConfig_t::confidence

8.474.2.2 int pack_loc_SLQSLOCSetCradleMountConfig_t::has_confidence

8.474.2.3 uint32_t pack_loc_SLQSLOCSetCradleMountConfig_t::state

8.475 pack_loc_SLQSLOCSetServer_t Struct Reference

Data Fields

- uint32_t serverType
- loc_IPv4Config * pIPv4Config
- loc_IPv6Config * pIPv6Config
- loc_URLAddrInfo * pURLAddr

8.475.1 Detailed Description

This structure contains parameters to set A-GPS Server

Parameters

serverType	• Type of server • Valid values: – 1 - Server type is CDMA PDE – 2 - Server type is CDMA MPC – 3 - Server type is UMTS SLP – 4 - Server type is custom PDE

pIPv4Config	• IPv4 address and port (Optional) • See loc_IPv4Config for more information
pIPv6Config	• IPv6 address and port (Optional) • See loc_IPv6Config for more information
pURLAddr	• Uniform Resource Locator (Optional) • See loc_URLAddrInfo for more information

8.475.2 Field Documentation

8.475.2.1 loc_IPv4Config* pack_loc_SLQSLOCSetServer_t::pIPv4Config

8.475.2.2 loc_IPv6Config* pack_loc_SLQSLOCSetServer_t::pIPv6Config

8.475.2.3 loc_URLAddrInfo* pack_loc_SLQSLOCSetServer_t::pURLAddr

8.475.2.4 uint32_t pack_loc_SLQSLOCSetServer_t::serverType

8.476 pack_loc_Start_t Struct Reference

Data Fields

- uint8_t SessionId
- uint32_t * pRecurrenceType
- uint32_t * pHorizontalAccuracyLvl
- uint32_t * pIntermediateReportState
- uint32_t * pMinIntervalTime
- loc_LocApplicationInfo * pApplicationInfo
- uint32_t * pConfigAltitudeAssumed
- uint16_t Tlvresult

8.476.1 Detailed Description

This structure contains the LOC Start pack

Parameters

SessionId	• ID of the session as identified by the control point. • Range: 0 to 255
pRecurrence-Type	• Optional Parameter • Specifies the type of session in which the control point is interested. • Defaults to SINGLE. -Values – 1 - Request periodic position fixes – 2 - Request a single position fix

pHorizontal-AccuracyLvl	• Optional Parameter • Specifies the horizontal accuracy level required by the control point. • Defaults to LOW • Values – 1 - Low accuracy – 2 - Medium accuracy – 3 - High accuracy
pIntermediate-ReportState	• Optional Parameter • Specifies if the control point is interested in receiving intermediate reports. • ON by default. • Values – 1 - Intermediate reports are turned on – 2 - Intermediate reports are turned off
pMinInterval-Time	• Optional Parameter • Minimum time interval, specified by the control point, that must elapse between position reports. • Units - Milliseconds • Default - 1000 ms
LocApplication-Info	• Optional Parameter • LOC Application Parameters • See loc_LocApplicationInfo for more information
pConfigAltitude-Assumed	• Optional Parameter • Configuration for Altitude Assumed Info in GNSS SV Info Event • Defaults to ENABLED. • Values – 1 - Enable Altitude Assumed information in GNSS SV Info Event – 2 - Disable Altitude Assumed information in GNSS SV Info Event
Tlvresult	• Unpack result.

8.476.2 Field Documentation

8.476.2.1 loc_LocApplicationInfo* pack_loc_Start_t::pApplicationInfo

8.476.2.2 uint32_t* pack_loc_Start_t::pConfigAltitudeAssumed

8.476.2.3 uint32_t* pack_loc_Start_t::pHorizontalAccuracyLvl

8.476.2.4 uint32_t* pack_loc_Start_t::pIntermediateReportState

8.476.2.5 uint32_t* pack_loc_Start_t::pMinIntervalTime

8.476.2.6 uint32_t* pack_loc_Start_t::pRecurrenceType

8.476.2.7 uint8_t pack_loc_Start_t::SessionId

8.476.2.8 uint16_t pack_loc_Start_t::Tlvresult

8.477 pack_loc_Stop_t Struct Reference

Data Fields

- uint8_t SessionId
- uint16_t Tlvresult

8.477.1 Detailed Description

This structure contains Stop LOC pack

Parameters

sessionId	• ID of the session as identified by the control point. • Range: 0 to 255
Tlvresult	• Unpack result.

8.477.2 Field Documentation

8.477.2.1 uint8_t pack_loc_Stop_t::SessionId

8.477.2.2 uint16_t pack_loc_Stop_t::Tlvresult

8.478 pack_nas_InitiateDomainAttach_t Struct Reference

Data Fields

- uint32_t action

8.478.1 Detailed Description

Pack structure for Initiates a domain attach/detach of the device.

Parameters

action	• Domain action to attempt 1 - Attach 2 - Detach

8.478.2 Field Documentation

8.478.2.1 uint32_t pack_nas_InitiateDomainAttach_t::action

8.479 pack_nas_PerformNetworkScanPCI_t Struct Reference

Data Fields

- uint8_t * pNetworkType
- uint32_t * pScanType
- uint64_t * pCiotOpModePref
- uint64_t * pLteM1BandPref
- uint64_t * pLteNB1BandPref

8.479.1 Detailed Description

Contain the network scan information.

Parameters

pNetworkType	• network type. • Bit 0 - GSM. • Bit 1 - UMTS. • Bit 2 - LTE. • Bit 3 - TD-SCDMA.
pScanType	• Scan type. • 0x00 - PLMN(default). • 0x01 - Closed subscriber group. • 0x02 - Mode prference. • 0x03 - Physical cell ID.
pCiotOpMode-Pref[IN]	• CIOT LTE Operational Mode Preference • Bitmask representing the Cellular Internet of Things (CIoT) LTE operational mode preference. • Values: – Bit 0 (0x01) - QMI_NAS_CIOT_LTE_OP_MODE_PREF_WB - LTE wideband (legacy LTE) – Bit 1 (0x02) - QMI_NAS_CIOT_LTE_OP_MODE_PREF_M1 - LTE M1 – Bit 2 (0x04) - QMI_NAS_CIOT_LTE_OP_MODE_PREF_NB1 - LTE NB1 – All unlisted bits are reserved for future use and the service point ignores them if used.

pLteM1Band-Pref[IN]	• Bitmask representing the LTE M1 band preferencet. • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – Bit 61 - E-UTRA Operating Band 126 – Bit 62 - E-UTRA Operating Band 127

pLteNb1Band-Pref[IN]	• Bitmask representing the LTE NB1 band preference. • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – Bit 61 - E-UTRA Operating Band 126 – Bit 62 - E-UTRA Operating Band 127

8.479.2 Field Documentation

8.479.2.1 uint64_t* pack_nas_PerformNetworkScanPCI_t::pCiotOpModePref

8.479.2.2 uint64_t* pack_nas_PerformNetworkScanPCI_t::pLteM1BandPref

8.479.2.3 uint64_t* pack_nas_PerformNetworkScanPCI_t::pLteNB1BandPref

8.479.2.4 uint8_t* pack_nas_PerformNetworkScanPCI_t::pNetworkType

8.479.2.5 uint32_t* pack_nas_PerformNetworkScanPCI_t::pScanType

8.480 pack_nas_SetACCOLC_t Struct Reference

Data Fields

- int8_t spc [6]
- uint8_t accolc

8.480.1 Detailed Description

Sets the access overload class (ACCOLC)

Parameters

spc	• service programming code NULL-terminated string of six digit
accolc	• ACCOLC : Valid range is 0 to 15

8.480.2 Field Documentation

8.480.2.1 uint8_t pack_nas_SetACCOLC_t::accolc

8.480.2.2 int8_t pack_nas_SetACCOLC_t::spc[6]

8.481 pack_nas_SetCDMANetworkParameters_t Struct Reference

Data Fields

- char * pSPC
- uint8_t * pForceRev0
- uint8_t * pCustomSCP
- uint32_t * pProtocol
- uint32_t * pBroadcast
- uint32_t * pApplication
- uint32_t * pRoaming

8.481.1 Detailed Description

Pack structure to set CDMA Network parameters.

Parameters

pSPC	• Six digit service programming code (not necessary when only the roaming field is being set)
pForceRev0	• (Optional)Force CDMA 1x-EV-DO Rev. 0 mode – 0 - Disabled – 1 - Enabled Note: Enabled can only be specified if pCustomSCP state is set to Disabled
pCustomSCP	• (Optional)Use a custom config for CDMA 1x-EV-DO SCP – 0 - Disabled – 1 - Enabled Note: Enabled can only be specified if pForceRev0 is set to Disabled
pProtocol	• Protocol mask for custom SCP config – 0x00000001 - Subtype 2 Physical Layer – 0x00000002 - Enhanced CCMAC – 0x00000004 - Enhanced ACMAC – 0x00000008 - Enhanced FTCMAC – 0x00000010 - Subtype 3 RTCMAC – 0x00000020 - Subsystem 1 RTCMAC – 0x00000040 - Enhanced Idle – 0x00000080 - Generic Multimode Capable Disc Port – 0xFFFFFFFF - Unknown
pBroadcast	• Broadcast mask for custom SCP config – 0x00000001 - Generic broadcast enabled – 0xFFFFFFFF - Unknown
pApplication	• Application mask for custom SCP config – 0x00000001 - SN Multiflow Packet Application – 0x00000002 - SN Enhanced Multiflow Packet Application – 0xFFFFFFFF - Unknown
pRoaming	• Roaming preference – 0 - Automatic – 1 - Home Only – 2 - Affiliated Roaming Only – 3 - Home and Affiliated Roaming – 0xFFFFFFFF - Unknown

8.481.2 Field Documentation

8.481.2.1 uint32_t* pack_nas_SetCDMANetworkParameters_t::pApplication

8.481.2.2 uint32_t* pack_nas_SetCDMANetworkParameters_t::pBroadcast

8.481.2.3 uint8_t* pack_nas_SetCDMANetworkParameters_t::pCustomSCP

8.481.2.4 uint8_t* pack_nas_SetCDMANetworkParameters_t::pForceRev0

8.481.2.5 uint32_t* pack_nas_SetCDMANetworkParameters_t::pProtocol

8.481.2.6 uint32_t* pack_nas_SetCDMANetworkParameters_t::pRoaming

8.481.2.7 char* pack_nas_SetCDMANetworkParameters_t::pSPC

8.482 pack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint32_t TechnologyPref
- uint32_t Duration
- uint16_t Tlvresult

8.482.1 Detailed Description

This structure contains pack set network preference parameters.

Parameters

TechnologyPref[-IN]	• Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved
Duration[IN]	• Duration of active preference – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call
Tlvresult	• pack result

8.482.2 Field Documentation

8.482.2.1 uint32_t pack_nas_SetNetworkPreference_t::Duration

8.482.2.2 uint32_t pack_nas_SetNetworkPreference_t::TechnologyPref

8.482.2.3 uint16_t pack_nas_SetNetworkPreference_t::Tlvresult

8.483 pack_nas_SLQSConfigSigInfo_t Struct Reference

Data Fields

- nas_RSSIThresh * pRSSIThresh
- nas_ECIOThresh * pECIOThresh
- nas_HDRSINRThresh * pHDRSINRThresh
- nas_LTESNRThresh * pLTESNRThresh
- nas_IOThresh * pIOThresh
- nas_RSRQThresh * pRSRQThresh
- nas_RSRPThresh * pRSRPThresh
- nas_LTESigRptCfg * pLTESigRptCfg
- nas_TDSCDMASINRCONFThresh * pTDSCDMASINRCONFThresh

8.483.1 Detailed Description

Structure for ConfigSigInfo pack.

Parameters

pRSSIThresh	• RSSI threshold List • See nas_RSSIThresh for more details
pECIOThresh	• ECIO Threshold List • See nas_ECIOThresh for more details
pHDRSINR-Thresh	• HDR SINR Threshold List • See nas_HDRSINRThresh for more details
pLTESNR-Thresh	• LTE SNR Threshold List • See nas_LTESNRThresh for more details
pIOThresh	• IO Threshold List • See nas_IOThresh for more details
pRSRQThresh	• RSRQ Threshold List • See nas_RSRQThresh for more details
pRSRPThresh	• RSRP Threshold List • See nas_RSRPThresh for more details

pLTESigRptCfg	• LTE signal report config • See nas_LTESigRptCfg for more details
pTDSCDMASIN-RCONFThresh	• TD-SCDMA SINR Threshold List • See nas_TDSCDMASINRCONFThresh for more details

8.483.2 Field Documentation

8.483.2.1 nas_ECIOThresh∗ pack_nas_SLQSConfigSigInfo_t::pECIOThresh

8.483.2.2 nas_HDRSINRThresh∗ pack_nas_SLQSConfigSigInfo_t::pHDRSINRThresh

8.483.2.3 nas_IOThresh∗ pack_nas_SLQSConfigSigInfo_t::pIOThresh

8.483.2.4 nas_LTESigRptCfg∗ pack_nas_SLQSConfigSigInfo_t::pLTESigRptCfg

8.483.2.5 nas_LTESNRThresh∗ pack_nas_SLQSConfigSigInfo_t::pLTESNRThresh

8.483.2.6 nas_RSRPThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRPThresh

8.483.2.7 nas_RSRQThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRQThresh

8.483.2.8 nas_RSSIThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSSIThresh

8.483.2.9 nas_TDSCDMASINRCONFThresh∗ pack_nas_SLQSConfigSigInfo_t::pTDSCDMASINRCONFThresh

8.484 pack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t ∗ pMncPcsStatus

8.484.1 Detailed Description

This structure contains pack get operator name for specificed network parameters.

Parameters

mcc	• A 16-bit integer representation of MCC. Range: 0 to 999
mnc	• A 16-bit integer representation of MNC. Range: 0 to 999

pMncPcsStatus	• MNC PCS Digit Include Status • Used to interpret the length of the corresponding MNC reported in the PLMN TLV(0x01). • Values – TRUE - MNC is a three-digit value. e.g. a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a two-digit value. e.g. a reported value of 90 corresponds to an MNC value of 90

Note

If pMncPcsStatus is not present, an MNC smaller than 100 is assumed to be a two-digit value, and an MNC greater than or equal to 100 is assumed to be a three digit value.

8.484.2 Field Documentation

8.484.2.1 uint16_t pack_nas_SLQSGetPLMNName_t::mcc

8.484.2.2 uint16_t pack_nas_SLQSGetPLMNName_t::mnc

8.484.2.3 uint8_t* pack_nas_SLQSGetPLMNName_t::pMncPcsStatus

8.485 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference

Data Fields

- uint32_t regAction
- nas_MNRInfo * pMNRInfo
- uint32_t * pChangeDuration
- uint8_t * pMncPcsDigitStatus

8.485.1 Detailed Description

This structure contains Initiate Network Registration request parameters

Parameters

regAction	• Specifies one of the following register actions : – AUTO_REGISTER - Device registers according to its provisioning and optional parameters supplied with the command are ignored. – MANUAL_REGISTER - Device registers to a specified network and the optional Manual Network Register Information parameter pMNRInfo must also be included for the command to process successfully and supported only for 3GPP.
pMNRInfo	[Optional] • Pointer to structure MNRInfo – See nas_MNRInfo for more information

pChange-Duration	[Optional] • Duration of the change. – 0x00 - Power cycle - Remains active until the next device power cycle – 0x01 - Permanent - Remains active through power cycles until changed by the client
pMncPcsDigit-Status	[Optional] • MNC PCS Digit Include Status – True - MNC is a 3-digit value. – False - MNC is a 2-digit value.

8.485.2 Field Documentation

8.485.2.1 uint32_t* pack_nas_SLQSInitiateNetworkRegistration_t::pChangeDuration

8.485.2.2 uint8_t* pack_nas_SLQSInitiateNetworkRegistration_t::pMncPcsDigitStatus

8.485.2.3 nas_MNRInfo* pack_nas_SLQSInitiateNetworkRegistration_t::pMNRInfo

8.485.2.4 uint32_t pack_nas_SLQSInitiateNetworkRegistration_t::regAction

8.486 pack_nas_SLQSNasConfigSigInfo2_t Struct Reference

Data Fields

- nas_CDMARSSIThresh * pCDMARSSIThresh
- uint16_t * pCDMARSSIDelta
- nas_CDMAECIOThresh * pCDMAECIOThresh
- uint16_t * pCDMAECIODelta
- nas_HDRRSSIThresh * pHDRRSSIThresh
- uint16_t * pHDRRSSIDelta
- nas_HDRECIOThresh * pHDRECIOThresh
- uint16_t * pHDRECIODelta
- nas_HDRSINRThreshold * pHDRSINRThresh
- uint16_t * pHDRSINRDelta
- nas_HDRIOThresh * pHDRIOThresh
- uint16_t * pHDRIODelta
- nas_GSMRSSIThresh * pGSMRSSIThresh
- uint16_t * pGSMRSSIDelta
- nas_WCDMARSSIThresh * pWCDMARSSIThresh
- uint16_t * pWCDMARSSIDelta
- nas_WCDMAECIOThresh * pWCDMAECIOThresh
- uint16_t * pWCDMAECIODelta
- nas_LTERSSIThresh * pLTERSSIThresh
- uint16_t * pLTERSSIDelta
- nas_LTESNRThreshold * pLTESNRThresh
- uint16_t * pLTESNRDelta
- nas_LTERSRQThresh * pLTERSRQThresh
- uint16_t * pLTERSRQDelta
- nas_LTERSRPThresh * pLTERSRPThresh
- uint16_t * pLTERSRPDelta
- nas_LTESigRptConfig * pLTESigRptConfig

- nas_TDSCDMARSCPThresh * pTDSCDMARSCPThresh
- uint16_t * pTDSCDMARSCPDelta
- nas_TDSCDMARSSIThresh * pTDSCDMARSSIThresh
- float * pTDSCDMARSSIDelta
- nas_TDSCDMAECIOThresh * pTDSCDMAECIOThresh
- float * pTDSCDMAECIODelta
- nas_TDSCDMASINRThresh * pTDSCDMASINRThresh
- float * pTDSCDMASINRDelta

8.486.1 Detailed Description

This structure contains pack set the signal strength reporting thresholds parameters.

Parameters

pCDMARSSI-Thresh	• CDMA RSSI threshold List
pCDMARSSI-Delta	• RSSI delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pCDMAECIO-Thresh	• CDMA ECIO Threshold List
pCDMAECIO-Delta	• ECIO delta (in units of 0.1 dB). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRRSSI-Thresh	• HDR RSSI Threshold List
pHDRRSSIDelta	• RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRECIO-Thresh	• HDR ECIO Threshold List
pHDRECIODelta	• ECIO delta (in units of 0.1 dB) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRSINR-Thresh	• HDR SINR Threshold List
pHDRSINRDelta	• SINR delta (in units of 1 SINR level) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRIOThresh	• HDR IO Threshold List

pHDRIODelta	• IO delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pGSMRSSI-Thresh	• GSM RSSI Threshold List • See nas_GSMRSSIThresh for more details
pGSMRSSIDelta	• RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pWCDMARSSI-Thresh	• WCDMA RSSI Threshold List • See nas_WCDMARSSIThresh for more details
pWCDMARSSI-Delta	• RSSI delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pWCDMAECIO-Thresh	• WCDMA ECIO Threshold List
pWCDMAECIO-Delta	• ECIO delta (in units of 0.1 dB) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTERSSI-Thresh	• LTE RSSI Threshold List
pLTERSSIDelta	• RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTESNR-Thresh	• LTE SNR Threshold List
pLTESNRDelta	• SNR delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTERSRQ-Thresh	• LTE RSRQ Threshold List
pLTERSRQ-Delta	• RSRQ delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTERSRP-Thresh	• LTE RSRP Threshold List

pLTERSRPDelta	• RSRP delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTESigRpt-Config	• LTE Signal Report Config
pTDSCDMARS-CPThresh	• TDSCDMA RSCP Threshold List
pTDSCDMARS-CPDelta	• RSCP delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pTDSCDMARS-SIThresh	• TDSCDMA RSSI Threshold List
pTDSCDMARS-SIDelta	• RSSI delta (in dBm) used by TD-SCDMA.
pTDSCDMAECI-OThresh	• TDSCDMA ECIO Threshold List
pTDSCDMAECI-ODelta	• ECIO delta (in dB) used by TD-SCDMA
pTDSCDMASIN-RThresh	• TDSCDMA SINR Threshold List
pTDSCDMASIN-RDelta	• SINR delta (in dB) used by TD-SCDMA.

Note

Mixture of threshold and delta values can be provided in the request. But for each type and RAT,only one of threshold list or delta value is to be provided.

8.486.2 Field Documentation

8.486.2.1 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIODelta

8.486.2.2 nas_CDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIOThresh

8.486.2.3 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIDelta

8.486.2.4 nas_CDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIThresh

8.486.2.5 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIDelta

8.486.2.6 nas_GSMRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIThresh

8.486.2.7 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIODelta

8.486.2.8 **nas_HDRECIOThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIOThresh

8.486.2.9 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pHDRIODelta

8.486.2.10 **nas_HDRIOThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pHDRIOThresh

8.486.2.11 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIDelta

8.486.2.12 **nas_HDRRSSIThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIThresh

8.486.2.13 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRDelta

8.486.2.14 **nas_HDRSINRThreshold*** pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRThresh

8.486.2.15 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPDelta

8.486.2.16 **nas_LTERSRPThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPThresh

8.486.2.17 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQDelta

8.486.2.18 **nas_LTERSRQThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQThresh

8.486.2.19 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIDelta

8.486.2.20 **nas_LTERSSIThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIThresh

8.486.2.21 **nas_LTESigRptConfig*** pack_nas_SLQSNasConfigSigInfo2_t::pLTESigRptConfig

8.486.2.22 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRDelta

8.486.2.23 **nas_LTESNRThreshold*** pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRThresh

8.486.2.24 **float*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIODelta

8.486.2.25 **nas_TDSCDMAECIOThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIOThresh

8.486.2.26 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPDelta

8.486.2.27 **nas_TDSCDMARSCPThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPThresh

8.486.2.28 **float*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIDelta

8.486.2.29 **nas_TDSCDMARSSIThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIThresh

8.486.2.30 **float*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRDelta

8.486.2.31 **nas_TDSCDMASINRThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRThresh

8.486.2.32 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIODelta

8.486.2.33 **nas_WCDMAECIOThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIOThresh

8.486.2.34 **uint16_t*** pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIDelta

8.486.2.35 **nas_WCDMARSSIThresh*** pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIThresh

8.487 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference

Data Fields

- uint8_t namID

8.487.1 Detailed Description

Structure for 3GPP2 Subscription pack.

Parameters

namID	[Mandatory] • NAM ID of the information to be retrieved. The index starts from 0. A nam_id of 0xFF is used to retrieve information of current NAM.

8.487.2 Field Documentation

8.487.2.1 uint8_t pack_nas_SLQSNasGet3GPP2Subscription_t::namID

8.488 pack_nas_SLQSNASGeteDRXParamsExt_t Struct Reference

Data Fields

- uint8_t * pEdrxRAT
- uint32_t * pLteOpMode

8.488.1 Detailed Description

This structure contains the SLQSNASGeteDRXParamsExt request parameters.

Parameters

pEdrxRAT	[Optional] • eDRX Radio Access Technology • Values – NAS_RADIO_IF_NO_SVC (0x00) - None (no service) – NAS_RADIO_IF_CDMA_1X (0x01) - cdma2000 ® 1X – NAS_RADIO_IF_CDMA_1XEVDO (0x02) - cdma2000 ® HRPD (1xEV-DO) – NAS_RADIO_IF_AMPS (0x03) - AMPS – NAS_RADIO_IF_GSM (0x04) - GSM – NAS_RADIO_IF_UMTS (0x05) - UMTS – NAS_RADIO_IF_WLAN (0x06) - WLAN – NAS_RADIO_IF_GPS (0x07) - GPS – NAS_RADIO_IF_LTE (0x08) - LTE – NAS_RADIO_IF_TDSCDMA (0x09) - TD-SCDMA – NAS_RADIO_IF_LTE_M1 (0x0a) - LTE-M1 – NAS_RADIO_IF_LTE_NB1 (0x0b) - LTE-NB1 – NAS_RADIO_IF_NO_CHANGE (-1) - No change

Note

The device will use 0x08 - NAS_RADIO_IF_LTE as the default value if the TLV is omitted.

Parameters

pLteOpMode	[Optional] • LTE Operational Mode • Values – NAS_CIOT_SYS_MODE_NO_ SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_ WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_ M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_ NB1 (0x03) - Camped on LTE NB1 Note: This TLV should be set when edrx_rat_type is LTE.

8.488.2 Field Documentation

8.488.2.1 uint8_t* pack_nas_SLQSNASGeteDRXParamsExt_t::pEdrxRAT

8.488.2.2 uint32_t* pack_nas_SLQSNASGeteDRXParamsExt_t::pLteOpMode

8.489 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- uint8_t radio_if

8.489.1 Detailed Description

Structure for TxRx Info pack

Parameters

radio_if	[Mandatory] • Radio interface technology of the signal being measured • Valid Values – 0x01 - NAS_RADIO_IF_CDMA_1X - CDMA – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - HDR – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE

8.489.2 Field Documentation

8.489.2.1 uint8_t pack_nas_SLQSNasGetTxRxInfo_t::radio_if

8.490 pack_nas_SLQSNasIndicationRegisterExt_t Struct Reference

Data Fields

- uint8_t * pSystemSelectionInd
- uint8_t * pDDTMInd
- uint8_t * pServingSystemInd
- uint8_t * pDualStandByPrefInd
- uint8_t * pSubscriptionInfoInd
- uint8_t * pNetworkTimeInd
- uint8_t * pSysInfoInd
- uint8_t * pSignalStrengthInd
- uint8_t * pErrorRateInd
- uint8_t * pHDRNewUATIAssInd
- uint8_t * pHDRSessionCloseInd
- uint8_t * pManagedRoamingInd
- uint8_t * pNetworkRejectInd
- uint8_t * pSuppressSysInfoInd
- uint8_t * pLTECphyCa
- uint8_t * pEdrxChangeInfoInd

8.490.1 Detailed Description

This structure contains pack indication register parameters.

Parameters

pSystemSelectionInd	[Optional] • System Selection Preference indication registration. The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pDDTMInd	[Optional] • DDTM (Data Dedicated Transmission Mode) indication registration. The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pServingSystemInd	[Optional] • Serving System indication registration. The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pDualStandByPrefInd	[Optional] • Dual Standby Preference indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSubscriptionInfoInd	[Optional] • Subscription Information indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable

pNetworkTimeInd	[Optional] • Network Time indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSysInfoInd	[Optional] • System Information indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSignalStrengthInd	[Optional] • Signal Strength indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pErrorRateInd	[Optional] • Error Rate indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pHDRNewUATIAssInd	[Optional] • HDR New UATI Assigned indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pHDRSessionCloseInd	[Optional] • HDR Session Closed indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pManagedRoamingInd	[Optional] • Managed Roaming indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable
pNetworkRejectInd	[Optional] • Network reject information indication registration.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSuppressSysInfoInd	[Optional] • Controls the reporting of QMI_NAS_SYS_INFO_IND when only the reject_cause field has changed. Values: – 0x00 - Disable – 0x01 - Enable- 0x01 - Enable

pLTECphyCa	[Optional] • LTE Physical Carrier Aggregation Information.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable (default value) – 0x01 - Enable
pEdrxChange-InfoInd	[Optional] • EDRX Change Info indication registration.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable

8.490.2 Field Documentation

8.490.2.1 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDDTMInd

8.490.2.2 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDualStandByPrefInd

8.490.2.3 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pEdrxChangeInfoInd

8.490.2.4 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pErrorRateInd

8.490.2.5 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRNewUATIAssInd

8.490.2.6 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRSessionCloseInd

8.490.2.7 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pLTECphyCa

8.490.2.8 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pManagedRoamingInd

8.490.2.9 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkRejectInd

8.490.2.10 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkTimeInd

8.490.2.11 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pServingSystemInd

8.490.2.12 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSignalStrengthInd

8.490.2.13 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSubscriptionInfoInd

8.490.2.14 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSuppressSysInfoInd

8.490.2.15 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSysInfoInd

8.490.2.16 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSystemSelectionInd

8.491 pack_nas_SLQSNasIndicationRegisterV2_t Struct Reference

Data Fields

- uint8_t * pSystemSelectionInd
- uint8_t * pDDTMInd

- uint8_t * pServingSystemInd
- uint8_t * pDualStandByPrefInd
- uint8_t * pSubscriptionInfoInd
- uint8_t * pNetworkTimeInd
- uint8_t * pSysInfoInd
- uint8_t * pSignalStrengthInd
- uint8_t * pErrorRateInd
- uint8_t * pHDRNewUATIAssInd
- uint8_t * pHDRSessionCloseInd
- uint8_t * pManagedRoamingInd
- uint8_t * pCurrentPLMNNameInd
- uint8_t * pEMBMSStatusInd
- uint8_t * pRFBandInfoInd
- uint8_t * pNetworkRejectInd
- uint8_t * pSuppressSysInfoInd
- uint8_t * pOperatorNameDataInd
- uint8_t * pCSPPLMNModeInd
- uint8_t * pRTREConfigurationInd
- uint8_t * pIMSPrefStatusInd
- uint8_t * pE911StateReadyStatusInd
- uint8_t * pLTESIB16NetworkTimeInd
- uint8_t * pLTECphyCa
- uint8_t * pSubscriptionChangeInd
- uint8_t * pSSACBarringInd
- uint8_t * pT3402TimerValueInd
- uint8_t * pAccessClassBarringInd
- uint8_t * pDataSubscriptionPriorityInd
- uint8_t * pT3346TimerStatusChangeInd
- uint8_t * pCallModeStatusInd
- uint8_t * pSSACBarringExtInd
- uint8_t * pManualNWScanFailureInd
- uint8_t * pTimerExpiryInd
- uint8_t * pEmergencyModeStatusInd
- uint8_t * pGcellInfoInd
- uint8_t * pEdrxChangeInfoInd
- uint8_t * pLTERACHFailInd
- uint8_t * pLTERRCTxInfoInd

8.491.1 Detailed Description

This structure contains the SLQSNasIndicationRegisterExt request parameters.

Parameters

pSystem-SelectionInd	[Optional] • System Selection Preference indication registration. The following callbacks would not be invoked if the indication is disabled. tFNRoamingIndicator tFNDataCapabilities and tFNServingSystem – 0x00 - Disable – 0x01 - Enable

pDDTMInd	[Optional] • DDTM (Data Dedicated Transmission Mode) indication registration. The following callbacks would not be invoked if the indication is disabled. tFNDDTM – 0x00 - Disable – 0x01 - Enable
pServing-SystemInd	[Optional] • Serving System indication registration. The following callbacks would not be invoked if the indication is disabled. tFNBandPreference – 0x00 - Disable – 0x01 - Enable
pDualStandBy-PrefInd	[Optional] • Dual Standby Preference indication registration.The following callbacks would not be invoked if the indication is disabled. tFNDualStandByPref – 0x00 - Disable – 0x01 - Enable
pSubscription-InfoInd	[Optional] • Subscription Information indication registration.The following callbacks would not be invoked if the indication is disabled. tFNSubscriptionInfo – 0x00 - Disable – 0x01 - Enable
pNetworkTime-Ind	[Optional] • Network Time indication registration.The following callbacks would not be invoked if the indication is disabled. tFNNetworkTime – 0x00 - Disable – 0x01 - Enable
pSysInfoInd	[Optional] • System Information indication registration.The following callbacks would not be invoked if the indication is disabled. tFNSysInfo – 0x00 - Disable – 0x01 - Enable
pSignalStrength-Ind	[Optional] • Signal Strength indication registration.The following callbacks would not be invoked if the indication is disabled. tFNSigInfo – 0x00 - Disable – 0x01 - Enable

pErrorRateInd	[Optional] • Error Rate indication registration.The following callbacks would not be invoked if the indication is disabled. tFNErrRate – 0x00 - Disable – 0x01 - Enable
pHDRNewUATI-AssInd	[Optional] • HDR New UATI Assigned indication registration.The following callbacks would not be invoked if the indication is disabled. tFNHDRUATIUpdate – 0x00 - Disable – 0x01 - Enable
pHDRSession-CloseInd	[Optional] • HDR Session Closed indication registration.The following callbacks would not be invoked if the indication is disabled. tFNHDRSessionClose – 0x00 - Disable – 0x01 - Enable
pManaged-RoamingInd	[Optional] • Managed Roaming indication registration.The following callbacks would not be invoked if the indication is disabled. tFNManagedRoaming – 0x00 - Disable – 0x01 - Enable
pCurrentPLMN-NameInd	[Optional] • Current PLMN Name indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_CURRENT_PLMN_NAME_IND – 0x00 - Disable – 0x01 - Enable
pEMBMSStatus-Ind	[Optional] • eMBMS Status indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_EMBMS_STATUS_IND – 0x00 - Disable – 0x01 - Enable
pRFBandInfoInd	[Optional] • RF Band Information indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_RF_BAND_INFO_IND – 0x00 - Disable – 0x01 - Enable

pNetworkRejectInd	[Optional] • Network reject information indication registration.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSuppressSysInfoInd	[Optional] • Controls the reporting of QMI_NAS_SYS_INFO_IND when only the reject_cause field has changed. Values: – 0x00 - Disable – 0x01 - Enable
pOperatorNameDataInd	[Optional] • Operator Name Data indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_OPERATOR_NAME_DATA_IND – 0x00 - Disable – 0x01 - Enable
pCSPPLMNModeInd	[Optional] • CSP PLMN Mode Bit indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_CSP_PLMN_MODE_BIT_IND – 0x00 - Disable – 0x01 - Enable
pRTREConfigurationInd	[Optional] • RTRE Configuration indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_RTRE_CONFIG_IND – 0x00 - Disable – 0x01 - Enable
pIMSPrefStatusInd	[Optional] • IMS Preference Status indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_IMS_PREF_STATUS_IND – 0x00 - Disable – 0x01 - Enable
pE911StateReadyStatusInd	[Optional] • E911 State Ready Status indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_E911_STATE_READY_IND – 0x00 - Disable – 0x01 - Enable

pLTESIB16-NetworkTimeInd	[Optional] • LTE SIB16 Network Time indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_LTE_SIB16_NETWORK_TIME_IND – 0x00 - Disable – 0x01 - Enable
pLTECphyCa	[Optional] • LTE Physical Carrier Aggregation Information indication registration.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSubscription-ChangeInd	[Optional] • Subscription Change indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_SUBSCRIPTION_CHANGE_IND – 0x00 - Disable – 0x01 - Enable
pSSACBarring-Ind	[Optional] • Service-Specific Access Class Barring indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_SSAC_INFO_IND – 0x00 - Disable – 0x01 - Enable
pT3402Timer-ValueInd	[Optional] • T3402 Timer Value indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_EMM_T3402_CHANGED_IND – 0x00 - Disable – 0x01 - Enable
pAccessClass-BarringInd	[Optional] • Access Class Barring indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_ACB_INFO_IND – 0x00 - Disable – 0x01 - Enable
pData-Subscription-PriorityInd	[Optional] • Data Subscription Priority indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_DATA_SUBS_PRIORITY_IND – 0x00 - Disable – 0x01 - Enable

pT3346Timer-StatusChange-Ind	[Optional] • T3346 Timer Status Change indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_DATA_SUBS_PRIORITY_IND – 0x00 - Disable – 0x01 - Enable
pCallMode-StatusInd	[Optional] • Call Mode Status indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_CALL_MODE_IND – 0x00 - Disable – 0x01 - Enable
pSSACBarring-ExtInd	[Optional] • Service-Specific Access Class Barring Ext indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_SSAC_CHANGE_INFO_IND – 0x00 - Disable – 0x01 - Enable
pManualNW-ScanFailureInd	[Optional] • Manual Network Scan Failure indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_MANUAL_SCAN_FAIL_IND – 0x00 - Disable – 0x01 - Enable
pTimerExpiryInd	[Optional] • Timer Expiry indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_TIMER_EXPIRY_IND – 0x00 - Disable – 0x01 - Enable
pEmergency-ModeStatusInd	[Optional] • Emergency Mode Status indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_EMERGENCY_MODE_STATUS_IND – 0x00 - Disable – 0x01 - Enable
pGcellInfoInd	[Optional] • Gcell Info indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_GCELL_INFO_IND – 0x00 - Disable – 0x01 - Enable

pEdrxChange-InfoInd	[Optional] • EDRX Change Info indication registration.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pLTERACHFail-Ind	[Optional] • LTE RACH Failure Indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_LTE_RACH_FAIL_IND – 0x00 - Disable – 0x01 - Enable
pLTERRCTx-InfoInd	[Optional] • LTE RRC Tx Info Indication registration.The following callbacks would not be invoked if the indication is disabled. • QMI_NAS_LTE_RRC_TX_INFO_IND – 0x00 - Disable – 0x01 - Enable

Note

Atleast one parameter must be provided as request. 'NULL' value confirms that the indication value is not sent.

8.491.2 Field Documentation

8.491.2.1 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pAccessClassBarringInd

8.491.2.2 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pCallModeStatusInd

8.491.2.3 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pCSPPLMNModeInd

8.491.2.4 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pCurrentPLMNNameInd

8.491.2.5 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pDataSubscriptionPriorityInd

8.491.2.6 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pDDTMInd

8.491.2.7 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pDualStandByPrefInd

8.491.2.8 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pE911StateReadyStatusInd

8.491.2.9 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pEdrxChangeInfoInd

8.491.2.10 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pEMBMSStatusInd

8.491.2.11 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pEmergencyModeStatusInd

8.491.2.12 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pErrorRateInd

8.491.2.13 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pGcellInfoInd

8.491.2.14 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pHDRNewUATIAssInd

8.491.2.15 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pHDRSessionCloseInd

8.491.2.16 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pIMSPrefStatusInd

8.491.2.17 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pLTECphyCa

8.491.2.18 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pLTERACHFailInd

8.491.2.19 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pLTERRCTxInfoInd

8.491.2.20 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pLTESIB16NetworkTimeInd

8.491.2.21 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pManagedRoamingInd

8.491.2.22 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pManualNWScanFailureInd

8.491.2.23 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pNetworkRejectInd

8.491.2.24 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pNetworkTimeInd

8.491.2.25 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pOperatorNameDataInd

8.491.2.26 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pRFBandInfoInd

8.491.2.27 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pRTREConfigurationInd

8.491.2.28 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pServingSystemInd

8.491.2.29 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSignalStrengthInd

8.491.2.30 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSSACBarringExtInd

8.491.2.31 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSSACBarringInd

8.491.2.32 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSubscriptionChangeInd

8.491.2.33 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSubscriptionInfoInd

8.491.2.34 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSuppressSysInfoInd

8.491.2.35 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSysInfoInd

8.491.2.36 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pSystemSelectionInd

8.491.2.37 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pT3346TimerStatusChangeInd

8.491.2.38 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pT3402TimerValueInd

8.491.2.39 uint8_t* pack_nas_SLQSNasIndicationRegisterV2_t::pTimerExpiryInd

8.492 pack_nas_SLQSNASSeteDRXParams_t Struct Reference

Data Fields

- uint8_t * pEdrxEnable
- uint8_t * pCycleLen
- uint8_t * pPagingTimeWindow
- uint8_t * pEdrxRatType
- uint32_t * pEdrxCiotLteMode

8.492.1 Detailed Description

This structure contains the SLQSNASSeteDRXParams request parameters.

Parameters

pEdrxEnable	[Optional] • eDRX Enable • Values – 0x00 - Disable – 0x01 - Enable
pCycleLen	[Optional] • eDRX cycle length • Value to be set per 3GPP TS 24.008 Rel-13 Section 10.5.5.32
pPagingTime-Window	[Optional] • eDRX paging time window • Value to be set per 3GPP TS 24.008 Rel-13 Section 10.5.5.32
pEdrxRatType	[Optional] • Radio access technology • Values – NAS_RADIO_IF_NO_SVC (0x00) - None (no service) – NAS_RADIO_IF_CDMA_1X (0x01) - cdma2000 ® 1X – NAS_RADIO_IF_CDMA_1XEVDO (0x02) cdma2000 ® HRPD (1xEV-DO) – NAS_RADIO_IF_AMPS (0x03) - AMPS – NAS_RADIO_IF_GSM (0x04) - GSM – NAS_RADIO_IF_UMTS (0x05) - UMTS – NAS_RADIO_IF_WLAN (0x06) - WLAN – NAS_RADIO_IF_GPS (0x07) - GPS – NAS_RADIO_IF_LTE (0x08) - LTE – NAS_RADIO_IF_TDSCDMA (0x09)- TD-SCDMA – NAS_RADIO_IF_LTE_M1 (0x0a) - LTE-M1 – NAS_RADIO_IF_LTE_NB1 (0x0b) - LTE-NB1 – NAS_RADIO_IF_NO_CHANGE (-1)- No change

Note

The device will use 0x08 - NAS_RADIO_IF_LTE as the default value if the TLV is omitted.

Parameters

pEdrxCiotLte-Mode	[Optional] • CIOT LTE mode • Values – NAS_CIOT_SYS_MODE_NO_ SRV(0x00) - No service – NAS_CIOT_SYS_MODE_LTE_ WB(0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_ M1(0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_ NB1(0x03) - Camped on LTE NB1

Note

This TLV should be set when pEdrxRatType is LTE.

8.492.2 Field Documentation

8.492.2.1 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pCycleLen

8.492.2.2 uint32_t* pack_nas_SLQSNASSeteDRXParams_t::pEdrxCiotLteMode

8.492.2.3 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pEdrxEnable

8.492.2.4 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pEdrxRatType

8.492.2.5 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pPagingTimeWindow

8.493 pack_nas_SLQSNasSwiIndicationRegister_t Struct Reference

Data Fields

- uint8_t lteEsmUl
- uint8_t lteEsmDl
- uint8_t lteEmmUl
- uint8_t lteEmmDl
- uint8_t gsmUmtsUl
- uint8_t gsmUmtsDl
- uint8_t * pRankIndicatorInd
- uint8_t * pTimer

8.493.1 Detailed Description

This structure contains the OTA message indication.

Parameters

lteEsmUl	• 0 - do not report • 1 - report LTE ESM uplink messages
lteEsmDl	• 0 - do not report • 1 - report LTE ESM downlink messages

lteEmmUl	• 0 - do not report • 1 - report LTE EMM uplink messages
lteEmmDl	• 0 - do not report • 1 - report GSM/UMTS uplink messages
gsmUmtsUl	• 0 - do not report • 1 - report GSM/UMTS uplink messages
gsmUmtsDl	• 0 - do not report • 1 - report GSM/UMTS downlink messages
pRankIndicator-Ind	• 0 - do not report • 1 - report Rank Indicator messages
pTimer	• 0 - do not report • 1 - report Timer Indicator messages

8.493.2 Field Documentation

8.493.2.1 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsDl

8.493.2.2 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsUl

8.493.2.3 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmDl

8.493.2.4 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmUl

8.493.2.5 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmDl

8.493.2.6 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmUl

8.493.2.7 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pRankIndicatorInd

8.493.2.8 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pTimer

8.494 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN * pWcdmaUARFCN
- nas_lteEARFCN * pLteEARFCN
- nas_ltePCI * pLtePCI

8.494.1 Detailed Description

Structure for ChannelLock pack.

Parameters

pWcdmaUARFCN	[Optional] • See nas_wcdmaUARFCN for more information
pLteEARFCN	[Optional] • See nas_lteEARFCN for more information
pLtePCI	[Optional] • See nas_ltePCI for more information

8.494.2 Field Documentation

8.494.2.1 nas_lteEARFCN∗ pack_nas_SLQSNASSwiSetChannelLock_t::pLteEARFCN

8.494.2.2 nas_ltePCI∗ pack_nas_SLQSNASSwiSetChannelLock_t::pLtePCI

8.494.2.3 nas_wcdmaUARFCN∗ pack_nas_SLQSNASSwiSetChannelLock_t::pWcdmaUARFCN

8.495 pack_nas_SLQSPerformIncNetworkScanReq_t Struct Reference

Data Fields

- uint8_t ∗ pNetworkType
- uint32_t ∗ pScanType
- uint64_t ∗ pBandPref
- uint64_t ∗ pLTEBandPref
- uint64_t ∗ pTDSCDMABandPref
- nas_WCDMAARFCNTlv ∗ pWCDMAArfcn
- nas_LTEARFCNTlv ∗ pLTEArfcn
- uint8_t ∗ pSearchPeriodicity
- uint16_t ∗ pMaxSearchTime
- uint8_t ∗ pIncrementalResults
- uint8_t ∗ pIncResultPeriodicty
- nas_PrefPLMNListTlv ∗ pPrefPLMNList

8.495.1 Detailed Description

This structure contains information about the pack_nas_SLQSPerformIncNetworkScanReq_t request parameters.

Parameters

pNetworkType	• Bitmask representing the network type to scan. • Values – Bit 0 - GSM – Bit 1 - UMTS – Bit 2 - LTE – Bit 3 - TD-SCDMA • Bit to check in ParamPresenceMask - **16**
pScanType	• Network Scan Type. • 0x00 - PLMN (default) • 0x01 - Closed subscriber group • 0x02 - Mode preference • 0x03 - Physical cell ID

- Bit to check in ParamPresenceMask - **17**

Parameters

pBandPref	• Bit mask representing the band preference to be set. • Bit position meanings: – 0 - BC0_A - Band Class 0, A-System – 1 - BC0_B - Band Class 0, B-System, Band Class 0 AB , GSM 850 Band – 2 - BC1 - Band Class 1, all blocks – 3 - BC2 - Band Class 2 place holder – 4 - BC3 - Band Class 3, A-System – 5 - BC4 - Band Class 4, all blocks – 6 - BC5 - Band Class 5, all blocks – 7 - GSM_DCS_1800 - GSM DCS band – 8 - GSM_EGSM_900 - GSM Extended GSM (E-GSM) band – 9 - GSM_PGSM_900 - GSM Primary GSM (P-GSM) band – 10 - BC6 - Band Class 6 – 11 - BC7 - Band Class 7 – 12 - BC8 - Band Class 8 – 13 - BC9 - Band Class 9 – 14 - BC10 - Band Class 10 – 15 - BC11 - Band Class 11 – 16 - GSM_450 - GSM 450 band – 17 - GSM_480 - GSM 480 band – 18 - GSM_750 - GSM 750 band – 19 - GSM_850 - GSM 850 band – 20 - GSM_RGSM_900 - GSM Railways GSM Band – 21 - GSM_PCS_1900 - GSM PCS band – 22 - WCDMA_I_IMT_2000 - WCDMA EUROPE JAPAN and CHINA IMT 2100 band – 23 - WCDMA_II_PCS_1900 - WCDMA US PCS 1900 band – 24 - WCDMA_III_1700 - WCDMA EUROPE and CHINA DCS 1800 band – 25 - WCDMA_IV_1700 - WCDMA US 1700 band – 26 - WCDMA_V_850 - WCDMA US 850 band – 27 - WCDMA_VI_800 - WCDMA JAPAN 800 band – 28 - BC12 - Band Class 12 – 29 - BC14 - Band Class 14 – 30 - RESERVED_2 - Reserved 2 – 31 - BC15 - Band Class 15 – 48 - WCDMA_VII_2600 - WCDMA EUROPE 2600 band – 49 - WCDMA_VIII_900 - WCDMA EUROPE and JAPAN 900 band – 50 - WCDMA_IX_1700 - WCDMA JAPAN 1700 band – 56 - BC16 - Band Class 16 – 57 - BC17 - Band Class 17 – 58 - BC18 - Band Class 18 – 59 - BC19 - Band Class 19 • Bit to check in ParamPresenceMask - **18**

pLTEBandPref	• Bitmask representing the LTE band preference. • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 45 - E-UTRA Operating Band 46 – Bit 46 - E-UTRA Operating Band 47 – Bit 47 - E-UTRA Operating Band 48 – Bit 60 - E-UTRA Operating Band 125 – Bit 61 - E-UTRA Operating Band 126 – Bit 62 - E-UTRA Operating Band 127 • Bit to check in ParamPresenceMask - **19**

pTDSCDMA-BandPref	• Bitmask representing the TD-SCDMA band preference to be set. • Values – 0x01 - NAS_TDSCDMA_BAND_A - TD-SCDMA Band A – 0x02 - NAS_TDSCDMA_BAND_B - TD-SCDMA Band B – 0x04 - NAS_TDSCDMA_BAND_C - TD-SCDMA Band C – 0x08 - NAS_TDSCDMA_BAND_D - TD-SCDMA Band D – 0x10 - NAS_TDSCDMA_BAND_E - TD-SCDMA Band E – 0x20 - NAS_TDSCDMA_BAND_F - TD-SCDMA Band F – All other bits are reserved. • Bit to check in ParamPresenceMask - **20**
pWCDMAArfcn	• See nasWCDMAARFCNTlv for more information. • Bit to check in ParamPresenceMask - **21**
pLTEArfcn	• See nasLTEARFCNTlv for more information. • Bit to check in ParamPresenceMask - **22**
pSearch-Periodicity	• If search periodicity is not provided, or value 0 is provided, it means periodic search is disabled and only one scan will be performed. Only applicable to the following scan types: – NAS_SCAN_TYPE_PLMN – NAS_SCAN_TYPE_MODE_PREF • Bit to check in ParamPresenceMask - **23**
pMaxSearch-Time	• Maximum duration of the periodic search(in seconds) Only applicable to the following scan types: – NAS_SCAN_TYPE_PLMN – NAS_SCAN_TYPE_MODE_PREF • Bit to check in ParamPresenceMask - **24**
pIncremental-Results	• Indicates whether the modern reports incremental results of the network scan to the client. – FALSE - Incremental results are not reported. – TRUE (defalut) - Incremental results are reported. Only applicable to the following scan types: – NAS_SCAN_TYPE_PLMN – NAS_SCAN_TYPE_MODE_PREF • Bit to check in ParamPresenceMask - **25**
pIncResult-Periodicty	• Indicates the periodicity with which the modem should report incremental results to the client(in seconds). Only applicable to the following scan types: – NAS_SCAN_TYPE_PLMN – NAS_SCAN_TYPE_MODE_PREF • Bit to check in ParamPresenceMask - **26**

pPrefPLMNList	• See nasPrefPLMNListTlv for more information. • Bit to check in ParamPresenceMask - **27**

8.495.2 Field Documentation

8.495.2.1 uint64_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pBandPref

8.495.2.2 uint8_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pIncrementalResults

8.495.2.3 uint8_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pIncResultPeriodicty

8.495.2.4 nas_LTEARFCNTlv* pack_nas_SLQSPerformIncNetworkScanReq_t::pLTEArfcn

8.495.2.5 uint64_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pLTEBandPref

8.495.2.6 uint16_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pMaxSearchTime

8.495.2.7 uint8_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pNetworkType

8.495.2.8 nas_PrefPLMNListTlv* pack_nas_SLQSPerformIncNetworkScanReq_t::pPrefPLMNList

8.495.2.9 uint32_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pScanType

8.495.2.10 uint8_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pSearchPeriodicity

8.495.2.11 uint64_t* pack_nas_SLQSPerformIncNetworkScanReq_t::pTDSCDMABandPref

8.495.2.12 nas_WCDMAARFCNTlv* pack_nas_SLQSPerformIncNetworkScanReq_t::pWCDMAArfcn

8.496 pack_nas_SLQSSet3GPP2SubscriptionInfo_t Struct Reference

Data Fields

- uint8_t namID
- nas_Mdn * pMdn

8.496.1 Detailed Description

This structure contains the Set3GPP2SubscriptionInfo request parameters

Parameters

namID	[Mandatory] • NAM ID of the information to be retrieved. The index starts from 0. A nam_id of 0xFF is used to retrieve information of current NAM.
pMdn	[Optional] • See nas_Mdn for more information.

8.496.2 Field Documentation

8.496.2.1 uint8_t pack_nas_SLQSSet3GPP2SubscriptionInfo_t::namID

8.496.2.2 nas_Mdn* pack_nas_SLQSSet3GPP2SubscriptionInfo_t::pMdn

8.497 pack_nas_SLQSSetBandPreference_t Struct Reference

Data Fields

- uint64_t bandPref

8.497.1 Detailed Description

Provides information about the band preference.

Parameters

bandpreference	• Bit mask representing the band preference to be set. • Bit position meanings: – 0 - BC0_A - Band Class 0, A-System – 1 - BC0_B - Band Class 0, B-System, Band Class 0 AB , GSM 850 Band – 2 - BC1 - Band Class 1, all blocks – 3 - BC2 - Band Class 2 place holder – 4 - BC3 - Band Class 3, A-System – 5 - BC4 - Band Class 4, all blocks – 6 - BC5 - Band Class 5, all blocks – 7 - GSM_DCS_1800 - GSM DCS band – 8 - GSM_EGSM_900 - GSM Extended GSM (E-GSM) band – 9 - GSM_PGSM_900 - GSM Primary GSM (P-GSM) band – 10 - BC6 - Band Class 6 – 11 - BC7 - Band Class 7 – 12 - BC8 - Band Class 8 – 13 - BC9 - Band Class 9 – 14 - BC10 - Band Class 10 – 15 - BC11 - Band Class 11 – 16 - GSM_450 - GSM 450 band – 17 - GSM_480 - GSM 480 band – 18 - GSM_750 - GSM 750 band – 19 - GSM_850 - GSM 850 band – 20 - GSM_RGSM_900 - GSM Railways GSM Band – 21 - GSM_PCS_1900 - GSM PCS band – 22 - WCDMA_I_IMT_2000 - WCDMA EUROPE JAPAN and CHINA IMT 2100 band – 23 - WCDMA_II_PCS_1900 - WCDMA US PCS 1900 band – 24 - WCDMA_III_1700 - WCDMA EUROPE and CHINA DCS 1800 band – 25 - WCDMA_IV_1700 - WCDMA US 1700 band – 26 - WCDMA_V_850 - WCDMA US 850 band – 27 - WCDMA_VI_800 - WCDMA JAPAN 800 band – 28 - BC12 - Band Class 12 – 29 - BC14 - Band Class 14 – 30 - RESERVED_2 - Reserved 2 – 31 - BC15 - Band Class 15 – 32 - 47 - Reserved – 48 - WCDMA_VII_2600 - WCDMA EUROPE 2600 band – 49 - WCDMA_VIII_900 - WCDMA EUROPE and JAPAN 900 band – 50 - WCDMA_IX_1700 - WCDMA JAPAN 1700 band – 51 to 55 - Reserved – 56 - BBC16 - Band Class 16 – 57 - BC17 - Band Class 17 – 58 - BC18 - Band Class 18 – 59 - BC19 - Band Class 19 – 60 to 64 - Reserved

8.497.2 Field Documentation

8.497.2.1 uint64_t pack_nas_SLQSSetBandPreference_t::bandPref

8.498 pack_nas_SLQSSetDDTMPref_t Struct Reference

Data Fields

- uint8_t DDTMPrefSetting
- uint16_t DDTMAction
- uint8_t SOListAction
- uint8_t SOLen
- uint16_t ServiceOption [255]

8.498.1 Field Documentation

8.498.1.1 uint16_t pack_nas_SLQSSetDDTMPref_t::DDTMAction

8.498.1.2 uint8_t pack_nas_SLQSSetDDTMPref_t::DDTMPrefSetting

8.498.1.3 uint16_t pack_nas_SLQSSetDDTMPref_t::ServiceOption[255]

8.498.1.4 uint8_t pack_nas_SLQSSetDDTMPref_t::SOLen

8.498.1.5 uint8_t pack_nas_SLQSSetDDTMPref_t::SOListAction

8.499 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference

Data Fields

- uint8_t bEnable
- nas_SLQSSignalStrengthsIndReq * pSigIndReq

8.499.1 Detailed Description

This structure contains paramaters pack set strength thresholds callback.

Parameters

bEnable	0/1 to disable/enable RSSI signal stregth indication
pSigIndReq	parameters to control signal strength indication

8.499.2 Field Documentation

8.499.2.1 uint8_t pack_nas_SLQSSetSignalStrengthsCallback_t::bEnable

8.499.2.2 nas_SLQSSignalStrengthsIndReq* pack_nas_SLQSSetSignalStrengthsCallback_t::pSigIndReq

8.500 pack_nas_SLQSSetSysSelectionPref_t Struct Reference

Data Fields

- uint8_t * pEmerMode
- uint16_t * pModePref

- uint64_t * pBandPref
- uint16_t * pPRLPref
- uint16_t * pRoamPref
- uint64_t * pLTEBandPref
- struct nas_netSelectionPref * pNetSelPref
- uint8_t * pChgDuration
- uint8_t * pMNCIncPCSDigStat
- uint32_t * pSrvDomainPref
- uint32_t * pGWAcqOrderPref
- uint64_t * pTdscdmaBandPref
- struct nas_acqOrderPref * pAcqOrderPref
- uint32_t * pSrvRegRestriction
- struct nas_CSGID * pCSGID
- unsigned char * pRAT

8.500.1 Detailed Description

Contain the system selection preferences.

Parameters

pEmerMode	• Optional parameter specifying the emergency Mode • Values: – 0 - OFF (normal) – 1 - ON (Emergency)
pModePref	• Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE

pBandPref	• Optional parameter • Bit mask representing the band preference • Bit values: – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved

pPRLPref	• Optional parameter indicating the CDMA PRL Preference • Values: – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems
pRoamPref	• Optional parameter indicating the roaming Preference • Values: – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator
pLTEBandPref	• Optional parameter • Bit mask representing the LTE band preference • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – All other bits are reserved

pNetSelPref	- netSelectionPref • Optional parameter for specifying Network Selection Preference • Modem selects networks based on this parameter(if present). • see nas_netSelectionPref for more information
pChgDuration	• Optional parameter specifying the duration of the change • Values: – 0x00 - Power cycle - Remains active until the next device power cycle – 0x01 - Permanent - Remains active through power cycles until changed by client – Device will use “0x01 - permanent” as default if this parameter is omitted
pMNCIncPCS-DigStat	• Optional parameter indicating if MNC includes PCS digit • Values: – TRUE - MNC is a 3 digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a 2-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
pSrvDomainPref	• Optional parameter indicating Service domain preference • Values: – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach
pGWAcqOrder-Pref	• Optional parameter indicating GSM/WCDMA Acquisition order Preference • Values: – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM
pTdscdmaBand-Pref	• Optional parameter indicating bitmask representing the TD-SCDMA band preference to be set. • Values: – 0x01 - TD-SCDMA Band A – 0x02 - TD-SCDMA Band B – 0x04 - TD-SCDMA Band C – 0x08 - TD-SCDMA Band D – 0x10 - TD-SCDMA Band E – 0x20 - TD-SCDMA Band F – All other bits are reserved
pAcqOrderPref	- acqOrderPref • Optional parameter for specifying Acquisition Order Preference • see nas_acqOrderPref for more information

pSrvReg-Restriction	• Optional parameter indicating Network Selection Registration Restriction Preference • Values: – 0x00 - Device follows the normal registration process – 0x01 - Device camps on the network according to its provisioning, but does not register – 0x02 - Device selects the network for limited service – All other values are reserved.
pCSGID	- CSGID • Optional parameter for specifying CSG ID • Either of pNetSelPref or pCSGID can be set. • see nas_CSGID for more information
pRAT	• Optional parameter Radio Access Technology order Preference • Values: – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TDSCDMA

8.500.2 Field Documentation

8.500.2.1 struct nas_acqOrderPref* pack_nas_SLQSSetSysSelectionPref_t::pAcqOrderPref

8.500.2.2 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pBandPref

8.500.2.3 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pChgDuration

8.500.2.4 struct nas_CSGID* pack_nas_SLQSSetSysSelectionPref_t::pCSGID

8.500.2.5 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pEmerMode

8.500.2.6 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pGWAcqOrderPref

8.500.2.7 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pLTEBandPref

8.500.2.8 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pMNCIncPCSDigStat

8.500.2.9 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pModePref

8.500.2.10 struct nas_netSelectionPref* pack_nas_SLQSSetSysSelectionPref_t::pNetSelPref

8.500.2.11 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pPRLPref

8.500.2.12 unsigned char* pack_nas_SLQSSetSysSelectionPref_t::pRAT

8.500.2.13 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pRoamPref

8.500.2.14 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvDomainPref

8.500.2.15 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvRegRestriction

8.500.2.16 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pTdscdmaBandPref

8.501 pack_nas_SLQSSetSysSelectionPrefExt_t Struct Reference

Data Fields

- uint8_t * pEmerMode
- uint16_t * pModePref
- uint64_t * pBandPref
- uint16_t * pPRLPref
- uint16_t * pRoamPref
- uint64_t * pLTEBandPref
- struct nas_netSelectionPref * pNetSelPref
- uint8_t * pChgDuration
- uint8_t * pMNCIncPCSDigStat
- uint32_t * pSrvDomainPref
- uint32_t * pGWAcqOrderPref
- uint64_t * pTdscdmaBandPref
- struct nas_acqOrderPref * pAcqOrderPref
- uint32_t * pSrvRegRestriction
- struct nas_CSGID * pCSGID
- unsigned char * pRAT
- struct nas_lteBandPrefExt * pLTEBandPrefExt
- uint64_t * pCiotLteOpModePref
- uint64_t * pLteM1BandPref
- uint64_t * pLteNb1BandPref
- uint32_t * pCiotLteOpMode
- struct nas_ciotAcqOrderPref * pCiotAcqOrderPref
- struct nas_nr5gBandPref * pNr5gBandPref

8.501.1 Detailed Description

Contain the system selection preferences.

Parameters

pEmerMode	• Optional parameter specifying the emergency Mode • Values: – 0 - OFF (normal) – 1 - ON (Emergency)

pModePref	• Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE – Bit 5 - TD-SCDMA – Bit 6 - NR5G

pBandPref	• Optional parameter • Bit mask representing the band preference • Bit values: – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved

pPRLPref	• Optional parameter indicating the CDMA PRL Preference • Values: – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems
pRoamPref	• Optional parameter indicating the roaming Preference • Values: – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator

pLTEBandPref	• Optional parameter • Bit mask representing the LTE band preference • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – All other bits are reserved

pNetSelPref	- netSelectionPref • Optional parameter for specifying Network Selection Preference • Modem selects networks based on this parameter(if present). • Either of pNetSelPref or pCSGID can be set. • see nas_netSelectionPref for more information
pChgDuration	• Optional parameter specifying the duration of the change • At least one system selection setting to be set if pChgDuration is populated. • Values: – 0x00 - Power cycle - Remains active until the next device power cycle – 0x01 - Permanent - Remains active through power cycles until changed by client – Device will use “0x01 - permanent” as default if this parameter is omitted
pMNCIncPCS-DigStat	• Optional parameter indicating if MNC includes PCS digit • pNetSelPref is expected if MNC includes PCS digit is set to 1. • Values: – TRUE - MNC is a 3 digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a 2-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
pSrvDomainPref	• Optional parameter indicating Service domain preference • Values: – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach
pGWAcqOrder-Pref	• Optional parameter indicating GSM/WCDMA Acquisition order Preference • Values: – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM
pTdscdmaBand-Pref	• Optional parameter indicating bitmask representing the TD-SCDMA band preference to be set. • Values: – 0x01 - TD-SCDMA Band A – 0x02 - TD-SCDMA Band B – 0x04 - TD-SCDMA Band C – 0x08 - TD-SCDMA Band D – 0x10 - TD-SCDMA Band E – 0x20 - TD-SCDMA Band F – All other bits are reserved

pAcqOrderPref	- acqOrderPref • Optional parameter for specifying Acquisition Order Preference • see nas_acqOrderPref for more information
pSrvReg-Restriction	• Optional parameter indicating Network Selection Registration Restriction Preference • Values: – 0x00 - Device follows the normal registration process – 0x01 - Device camps on the network according to its provisioning, but does not register – 0x02 - Device selects the network for limited service – All other values are reserved.
pCSGID	- CSGID • Optional parameter for specifying CSG ID • Either of pNetSelPref or pCSGID can be set. • see nas_CSGID for more information
pRAT	• Optional parameter Radio Access Technology order Preference • Values: – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TDSCDMA – 0x0A - LTE-M1 – 0x0B - LTE-NB1 – 0x0C - NR5G
pLTEBandPref-Ext[IN]	• LTE Band Preference Extended • see nas_lteBandPrefExt for more information
pCiotLteOp-ModePref[IN]	• CIOT LTE Operational Mode Preference • Bitmask representing the CIoT LTE operational mode preference to be set. • Values : – Bit 0 - LTE wideband – Bit 1 - LTE M1 – Bit 2 - LTE NB1 All unlisted bits are reserved for future use and the service point ignores them if used. If this TLV is not present, the modem will use the value read from the NV during bootup.

pLteM1Band-Pref[IN]	• Bitmask representing the LTE M1 band preference to be set. • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – Bit 61 - E-UTRA Operating Band 126 – Bit 62 - E-UTRA Operating Band 127

- If this field is not present, the modem will use the M1 band preference value read from the NV.

Parameters

pLteNb1Band-Pref[IN]	• Bitmask representing the LTE NB1 band preference to be set. • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 22 - E-UTRA Operating Band 23 – Bit 23 - E-UTRA Operating Band 24 – Bit 24 - E-UTRA Operating Band 25 – Bit 25 - E-UTRA Operating Band 26 – Bit 27 - E-UTRA Operating Band 28 – Bit 28 - E-UTRA Operating Band 29 – Bit 29 - E-UTRA Operating Band 32 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – Bit 40 - E-UTRA Operating Band 41 – Bit 41 - E-UTRA Operating Band 42 – Bit 42 - E-UTRA Operating Band 43 – Bit 60 - E-UTRA Operating Band 125 – Bit 61 - E-UTRA Operating Band 126 – Bit 62 - E-UTRA Operating Band 127 • If this field is not present, the modem will use the NB1 band preference value read from the NV.

pCiotLteOpMode	• This TLV is used when the net_sel_pref is MANUAL and RAT TLV is LTE. • This TLV indicates the the operational mode to be used during the manual LTE PLMN selection. • Values : – 0x00 - No service – 0x01 - Camped on LTE wideband – 0x02 - Camped on LTE M1 – 0x03 - Camped on LTE NB1
pCiotAcqOrder-Pref[IN]	• CIOT Acquisition Order Preference • see nas_ciotAcqOrderPref for more information
pNr5gBandPref[-IN]	• NR5G Band Preference • see nas_nr5gBandPref for more information

8.501.2 Field Documentation

8.501.2.1 struct nas_acqOrderPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pAcqOrderPref

8.501.2.2 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pBandPref

8.501.2.3 uint8_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pChgDuration

8.501.2.4 struct nas_ciotAcqOrderPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pCiotAcqOrderPref

8.501.2.5 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pCiotLteOpMode

8.501.2.6 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pCiotLteOpModePref

8.501.2.7 struct nas_CSGID* pack_nas_SLQSSetSysSelectionPrefExt_t::pCSGID

8.501.2.8 uint8_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pEmerMode

8.501.2.9 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pGWAcqOrderPref

8.501.2.10 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pLTEBandPref

8.501.2.11 struct nas_lteBandPrefExt* pack_nas_SLQSSetSysSelectionPrefExt_t::pLTEBandPrefExt

8.501.2.12 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pLteM1BandPref

8.501.2.13 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pLteNb1BandPref

8.501.2.14 uint8_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pMNCIncPCSDigStat

8.501.2.15 uint16_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pModePref

8.501.2.16 struct nas_netSelectionPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pNetSelPref

8.501.2.17 struct nas_nr5gBandPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pNr5gBandPref

8.501.2.18 uint16_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pPRLPref

8.501.2.19 unsigned char* pack_nas_SLQSSetSysSelectionPrefExt_t::pRAT

8.501.2.20 uint16_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pRoamPref

8.501.2.21 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pSrvDomainPref

8.501.2.22 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pSrvRegRestriction

8.501.2.23 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pTdscdmaBandPref

8.502 pack_nas_SLQSSwiPSDetach_t Struct Reference

Data Fields

- uint8_t * pDetachAction

8.502.1 Detailed Description

Structure for PS connection detach pack.

Parameters

pDetachAction[I-N]	• Values – 2- Initiates an immediate packet domain detach.

8.502.2 Field Documentation

8.502.2.1 uint8_t* pack_nas_SLQSSwiPSDetach_t::pDetachAction

8.503 pack_pds_PDSInjectTimeReference_t Struct Reference

Data Fields

- uint64_t systemTime
- uint16_t systemDiscontinuities

8.503.1 Detailed Description

Structure contain the parameter for PDS Inject time.

Parameters

systemTime	• System time(milliseconds)
system-Discontinuities	• Number of system time discontinuities

8.503.2 Field Documentation

8.503.2.1 uint16_t pack_pds_PDSInjectTimeReference_t::systemDiscontinuities

8.503.2.2 uint64_t pack_pds_PDSInjectTimeReference_t::systemTime

8.504 pack_pds_ResetPDSData_t Struct Reference

Data Fields

- uint32_t * pGPSDataMask
- uint32_t * pCellDataMask

8.504.1 Detailed Description

Structure contain the parameter to Resets the specified PDS data.

Parameters

pGPSDataMask	• Bitmask of GPS data to clear (optional) – 0x00000001 - EPH – 0x00000002 - ALM – 0x00000004 - POS – 0x00000008 - TIME – 0x00000010 - IONO – 0x00000020 - UTC – 0x00000040 - HEALTH – 0x00000080 - SVDIR – 0x00000100 - SVSTEER – 0x00000200 - SADATA – 0x00000400 - RTI – 0x00000800 - ALM_CORR – 0x00001000 - FREQ_BIAS_EST
pCellDataMask	• Bitmask of cell data to clear (optional) – 0x00000001 - POS – 0x00000002 - LATEST_GPS_POS – 0x00000004 - OTA_POS – 0x00000008 - EXT_REF_POS – 0x00000010 - TIMETAG – 0x00000020 - CELLID – 0x00000040 - CACHED_CELLID – 0x00000080 - LAST_SRV_CELL – 0x00000100 - CUR_SRV_CELL – 0x00000200 - NEIGHBOR_INFO

8.504.2 Field Documentation

8.504.2.1 uint32_t* pack_pds_ResetPDSData_t::pCellDataMask

8.504.2.2 uint32_t* pack_pds_ResetPDSData_t::pGPSDataMask

8.505 pack_pds_SetEventReportCallback_t Struct Reference

Data Fields

- uint8_t posDataNmea
- uint8_t rptPosData

8.505.1 Detailed Description

Structure contain the parameter for the automatic tracking configuration for the NMEA COM port.

Parameters

posDataNmea	• Report new position data in NMEA format. • Values – 0x00 - Disable – 0x01 - Enable
rptPosData	• Report new position data in raw format. • Values – 0x00 - Disable – 0x01 - Enable

8.505.2 Field Documentation

8.505.2.1 uint8_t pack_pds_SetEventReportCallback_t::posDataNmea

8.505.2.2 uint8_t pack_pds_SetEventReportCallback_t::rptPosData

8.506 pack_pds_SetPDSDefaults_t Struct Reference

Data Fields

- uint32_t operation
- uint8_t timeout
- uint32_t interval
- uint32_t accuracy

8.506.1 Detailed Description

Structure containing session configuration parameters.

Parameters

operation	• Current session operating mode – 0 - Standalone – 1 - MS based – 2 - MS assisted
timeout	• Maximum amount of time (seconds) to work on each fix, maximum is 255
interval	• Interval (seconds) between fix requests
accuracy	• Preferred accuracy threshold (meters)

8.506.2 Field Documentation

8.506.2.1 uint32_t pack_pds_SetPDSDefaults_t::accuracy

8.506.2.2 uint32_t pack_pds_SetPDSDefaults_t::interval

8.506.2.3 uint32_t pack_pds_SetPDSDefaults_t::operation

8.506.2.4 uint8_t pack_pds_SetPDSDefaults_t::timeout

8.507 pack_pds_SetPDSState_t Struct Reference

Data Fields

- uint32_t enable

8.507.1 Detailed Description

Structure contain PDS state parameters.

Parameters

enable	• Desired PDS state – Zero - disable – Non-Zero - enable

8.507.2 Field Documentation

8.507.2.1 uint32_t pack_pds_SetPDSState_t::enable

8.508 pack_pds_SetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t bAuto

8.508.1 Detailed Description

Structure contain the parameter for the automatic tracking configuration for the NMEA COM port.

Parameters

bAuto	• Enable automatic tracking for NMEA COM port – 0x00 - Disabled – 0x01 - Enabled

8.508.2 Field Documentation

8.508.2.1 uint32_t pack_pds_SetPortAutomaticTracking_t::bAuto

8.509 pack_pds_SetServiceAutomaticTracking_t Struct Reference

Data Fields

- uint32_t bAuto

8.509.1 Detailed Description

Structure contain the parameter to Sets the automatic tracking state.

Parameters

bAuto	• Automatic tracking session started for service – 0x00 - Disabled – 0x01 - Enabled

8.509.2 Field Documentation

8.509.2.1 uint32_t pack_pds_SetServiceAutomaticTracking_t::bAuto

8.510 pack_pds_SetXTRAAutomaticDownload_t Struct Reference

Data Fields

- uint32_t bEnabled
- uint16_t interval

8.510.1 Detailed Description

Structure contain the parameter to set the XTRA automatic database download configuration.

Parameters

bEnabled	• Automatic XTRA download status – 0 - Disabled – 1 - Enabled
interval	• Interval (hours) between XTRA downloads

8.510.2 Field Documentation

8.510.2.1 uint32_t pack_pds_SetXTRAAutomaticDownload_t::bEnabled

8.510.2.2 uint16_t pack_pds_SetXTRAAutomaticDownload_t::interval

8.511 pack_pds_SetXTRANetwork_t Struct Reference

Data Fields

- uint32_t preference

8.511.1 Detailed Description

Structure contain the parameter to Sets the XTRA WWAN network preference

Parameters

preference	• XTRA WWAN network preference – 0x00 - None (any available network) – 0x01 - Home-only, only when on home systems – 0x02 - Roam-only, only when on non-home systems

8.511.2 Field Documentation

8.511.2.1 uint32_t pack_pds_SetXTRANetwork_t::preference

8.512 pack_pds_SLQSGetAGPSConfig_t Struct Reference

Data Fields

- uint8_t * pNetworkMode

8.512.1 Detailed Description

Structure contain the parameter to PDS AGPS (MS-based) configuration pack.

Parameters

pNetworkMode	• Network Mode of AGPS Server [optional - should be present in Multimode Systems] – 0x00 - UMTS – 0x01 - CDMA

8.512.2 Field Documentation

8.512.2.1 uint8_t* pack_pds_SLQSGetAGPSConfig_t::pNetworkMode

8.513 pack_pds_SLQSPDSInjectAbsoluteTimeReference_t Struct Reference

Data Fields

- uint64_t timeMsec
- uint32_t timeUncMsec
- uint8_t timeBase
- uint8_t forceFlag

8.513.1 Detailed Description

Structure contain the parameter to Injects a absolute time reference.

Parameters

timeMsec	• Represents the number of milliseconds elapsed since either a GPS or UTC time base. If the time base is UTC, this value should NOT include leap seconds
timeUncMsec	• Time uncertainty in milliseconds
timeBase	• Time base – 0x00 - GPS (midnight, Jan 6, 1980) – 0x01 - UTC (midnight, Jan 1, 1970)
forceFlag	• Force acceptance of data

8.513.2 Field Documentation

8.513.2.1 uint8_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::forceFlag

8.513.2.2 uint8_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeBase

8.513.2.3 uint64_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeMsec

8.513.2.4 uint32_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeUncMsec

8.514 pack_pds_SLQSPDSInjectPositionData_t Struct Reference

Data Fields

- uint64_t * pTimeStamp
- uint64_t * pLatitude
- uint64_t * pLongitude
- uint32_t * pAltitudeWrtEllipsoid
- uint32_t * pAltitudeWrtSealevel
- uint32_t * pHorizontalUncCircular
- uint32_t * pVerticalUnc
- uint8_t * pHorizontalConfidence
- uint8_t * pVerticalConfidence
- uint8_t * pPositionSource
- uint8_t * pTimeType

8.514.1 Detailed Description

Position Data Parameters from the external source to be injected to PDS engine.

Parameters

pTimeStamp	• Timestamp of the injected position in msec. The time can be of type UTC, GPS, or Age and is defined in the pTimeType parameter. If the pTimeType is not present, the timestamp shall be assumed to be UTC time
pLatitude	• Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. Value (in decimal degrees) in the range from -90 degrees to +90 degrees.Value in double float format (refer toIEEE Std 754-1985)
pLongitude	• Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of Greenwich meridian. Value (in decimal degrees) in the range from -180 degrees to +180 degrees.
pAltitudeWrt-Ellipsoid	• Height above the WGS-84 reference ellipsoid. Value conveys height (in meters). When injecting altitude information, the control point should include either this parameter or the pAltitudeWrt-Sealevel parameter. Value in single float format (refer to IEEE Std 754-1985)
pAltitudeWrt-Sealevel	• Height of MS above the mean sea level in units (in meters). When injecting altitude information, the control point should include either this parameter or the pAltitudeWrtEllipsoid parameter. Value in single float format (refer to IEEE Std 754-1985)
pHorizontalUnc-Circular	• Circular horizontal uncertainty (in meters). This parameter must be included if the latitude and longitude parameters are specified. Value in single float format (refer to IEEE Std 754-1985)

pVerticalUnc	• Vertical uncertainty (in meters). This parameter must be included if one of the altitude parameter are specified.Value in single float format (refer to IEEE Std 754-1985)
pHorizontal-Confidence	• Confidence value of the location horizontal uncertainty, specified as percentage, 1 to 100. This parameter must be included if the latitude and longitude parameters are specified.
pVertical-Confidence	• Confidence value of the location vertical uncertainty, specified as percentage, 1 to 100. This parameter must be included if one of the altitude paramters are specified.
pPositionSource	• Source of injected position: – 0x00 - Unknown – 0x01 - GPS – 0x02 - Cell ID – 0x03 - Enhanced cell ID – 0x04 - WiFi – 0x05 - Terrestrial – 0x06 - Terrestrial hybrid – 0x07 - Other
pTimeType	• Defines the time value set in the pTimeStamp parameter. – 0x00 - UTC Time: starting Jan 1, 1970 – 0x01 - GPS Time: starting Jan 6, 1980 – 0x02 - Age: Age of position information

8.514.2 Field Documentation

8.514.2.1 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pAltitudeWrtEllipsoid

8.514.2.2 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pAltitudeWrtSealevel

8.514.2.3 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pHorizontalConfidence

8.514.2.4 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pHorizontalUncCircular

8.514.2.5 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pLatitude

8.514.2.6 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pLongitude

8.514.2.7 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pPositionSource

8.514.2.8 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pTimeStamp

8.514.2.9 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pTimeType

8.514.2.10 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pVerticalConfidence

8.514.2.11 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pVerticalUnc

8.515 pack_pds_SLQSSetAGPSConfig_t Struct Reference

Data Fields

- uint32_t * pServerAddress
- uint32_t * pServerPort
- uint8_t * pServerURL
- uint8_t * pServerURLLength
- uint8_t * pNetworkMode

8.515.1 Detailed Description

Structure contain the parameter to Sets the PDS AGPS (MS-based) configuration.

Parameters

pServerAddress	• IPv4 address of AGPS server [optional]
pServerPort	• Port number of AGPS server [optional - should be present when pServerAddress is present]
pServerURL	• URL of the AGPS server [optional]
pServerURL-Length	• URL length of AGPS server [optional - should be present when pServerURL is present]
pNetworkMode	• Network Mode of AGPS Server [optional - should be present in Multimode Systems] – 0x00 - UMTS – 0x01 - CDMA

8.515.2 Field Documentation

8.515.2.1 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pNetworkMode

8.515.2.2 uint32_t* pack_pds_SLQSSetAGPSConfig_t::pServerAddress

8.515.2.3 uint32_t* pack_pds_SLQSSetAGPSConfig_t::pServerPort

8.515.2.4 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pServerURL

8.515.2.5 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pServerURLLength

8.516 pack_pds_SLQSSetPositionMethodState_t Struct Reference

Data Fields

- uint8_t * pXtraTimeState

- uint8_t * pXtraDataState
- uint8_t * pWifiState

8.516.1 Detailed Description

Parameters to Set state of positioning method for a device.

Parameters

pXtraTimeState	• XTRA Time Position Method State. • Values: – 0x00 - Disable – 0x01 - Enable
pXtraDataState	• XTRA Data Position Method State. • Values: – 0x00 - Disable – 0x01 - Enable
Latitude	• WiFi Position Method State • Values: – 0x00 - Disable – 0x01 - Enable

8.516.2 Field Documentation

8.516.2.1 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pWifiState

8.516.2.2 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pXtraDataState

8.516.2.3 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pXtraTimeState

8.517 pack_pds_StartPDSTrackingSessionExt_t Struct Reference

Data Fields

- uint8_t sessionControl
- uint8_t sessionType
- uint8_t sessionOperation
- uint8_t sessionServerOption
- uint8_t fixTimeout
- uint32_t fixInterval
- uint32_t fixCount
- uint32_t fixAccuracy

8.517.1 Detailed Description

Structure contain the parameter for PDS tracking session.

Parameters

sessionControl	• Control method: – 0x0 - Manual
sessionType	• Type: – 0x0 - New
session-Operation	• Operating mode: – 0x00 - Standalone – 0x01 - MS-based
sessionServer-Option	• Location server option: – 0x0 - Default
fixTimeout	• Maximum time to work on each fix (in seconds, max 255)
fixCount	• Count of position fix requests for this session (must be at least 1)
fixInterval	• interval between position fix requests (in seconds)
fixAccuracy	• Preferred accuracy threshold(in meters)

8.517.2 Field Documentation

8.517.2.1 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixAccuracy

8.517.2.2 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixCount

8.517.2.3 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixInterval

8.517.2.4 uint8_t pack_pds_StartPDSTrackingSessionExt_t::fixTimeout

8.517.2.5 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionControl

8.517.2.6 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionOperation

8.517.2.7 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionServerOption

8.517.2.8 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionType

8.518 pack_qmi_t Struct Reference

Data Fields

- uint16_t xid
- int timeout
- uint16_t msgid
- uint8_t svc

8.518.1 Detailed Description

qmi request context

Parameters

in	*xid*	transaction id
out	*timeout*	timeout recommended in seconds
out	*msgid*	message id
out	*svc*	qmi service

8.518.2 Field Documentation

8.518.2.1 uint16_t pack_qmi_t::msgid

8.518.2.2 uint8_t pack_qmi_t::svc

8.518.2.3 int pack_qmi_t::timeout

8.518.2.4 uint16_t pack_qmi_t::xid

8.519 pack_qos_BindDataPort_t Struct Reference

Data Fields

- qos_BindDataPortPeripheralEndPointID_t * pPeripheralEndPointID
- qos_BindDataPortMuxID_t * pMuxID
- qos_BindDataPortSIODataPort_t * pSIODataPort

8.519.1 Detailed Description

Structure that contains the request Binds a control point to a data port.

Parameters

pPeripheralEnd-PointID	• See qos_BindDataPortPeripheralEndPointID_t for more information.
pMuxID	• See qos_BindDataPortMuxID_t for more information.
pSIODataPort	• See qos_BindDataPortSIODataPort_t for more information.

8.519.2 Field Documentation

8.519.2.1 qos_BindDataPortMuxID_t* pack_qos_BindDataPort_t::pMuxID

8.519.2.2 qos_BindDataPortPeripheralEndPointID_t* pack_qos_BindDataPort_t::pPeripheralEndPointID

8.519.2.3 qos_BindDataPortSIODataPort_t* pack_qos_BindDataPort_t::pSIODataPort

8.520 pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference

Data Fields

- uint32_t apnId

8.520.1 Detailed Description

Structure that contains the APN ID to obtain extra APN parameters

Parameters

apnId[IN]	• APN id

8.520.2 Field Documentation

8.520.2.1 uint32_t pack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.521 pack_qos_SLQSQosSwiReadDataStats_t Struct Reference

Data Fields

- uint32_t apnId

8.521.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

apnId[IN]	• APN id

8.521.2 Field Documentation

8.521.2.1 uint32_t pack_qos_SLQSQosSwiReadDataStats_t::apnId

8.522 pack_qos_SLQSSetQosEventCallback_t Struct Reference

Data Fields

- uint8_t enable

8.522.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

in	*enable*	• 1 - Enable QoS event reporting • 0 - Disable QoS event reporting

8.522.2 Field Documentation

8.522.2.1 uint8_t pack_qos_SLQSSetQosEventCallback_t::enable

8.523 pack_rms_SetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask

8.523.1 Detailed Description

Pack set SMS wake parameters

Parameters

enabled	• SMS wake functionality enabled – 0 - Disabled – 1 - Enabled
wake_mask	• SMS wake mask to search for incoming messages (only relevant when enabled)

8.523.2 Field Documentation

8.523.2.1 uint32_t pack_rms_SetSMSWake_t::enabled

8.523.2.2 uint32_t pack_rms_SetSMSWake_t::wake_mask

8.524 pack_sar_SLQSSetRfSarState_t Struct Reference

Data Fields

- uint32_t RfSarState

8.524.1 Detailed Description

This sturcture contain sets the specified RF SAR state parameter.

Parameters

RfSarState	• SAR RF State – QMI_SAR_RF_STATE_DEFAULT – QMI_SAR_RF_STATE_1 – QMI_SAR_RF_STATE_2 – QMI_SAR_RF_STATE_3 – QMI_SAR_RF_STATE_4 – QMI_SAR_RF_STATE_5 – QMI_SAR_RF_STATE_6 – QMI_SAR_RF_STATE_7 – QMI_SAR_RF_STATE_8

8.524.2 Field Documentation

8.524.2.1 uint32_t pack_sar_SLQSSetRfSarState_t::RfSarState

8.525 pack_sms_SaveSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage

8.525.1 Detailed Description

Sutructure containing parameters to Save an SMS message to device memory

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
messageFormat	• Message format – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
messageSize	• The length of the message contents in bytes

pMessage	• The message contents

8.525.2 Field Documentation

8.525.2.1 uint32_t pack_sms_SaveSMS_t::messageFormat

8.525.2.2 uint32_t pack_sms_SaveSMS_t::messageSize

8.525.2.3 uint8_t* pack_sms_SaveSMS_t::pMessage

8.525.2.4 uint32_t pack_sms_SaveSMS_t::storageType

8.526 pack_sms_SendSMS_t Struct Reference

Data Fields

- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage
- uint8_t * pLinktimer

8.526.1 Detailed Description

This structure contains pack send SMS parameters.

Parameters

messageFormat	• Message format – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
messageSize	• The length of the message contents in bytes
pLinktimer	• GW SMS link open for the specified number of second
pMessage	• The message contents in PDU format contains SMS header and payload message

8.526.2 Field Documentation

8.526.2.1 uint32_t pack_sms_SendSMS_t::messageFormat

8.526.2.2 uint32_t pack_sms_SendSMS_t::messageSize

8.526.2.3 uint8_t* pack_sms_SendSMS_t::pLinktimer

8.526.2.4 uint8_t* pack_sms_SendSMS_t::pMessage

8.527 pack_sms_SetNewSMSCallback_t Struct Reference

Data Fields

- enum eqmiCbkSetStatus status

8.527.1 Detailed Description

This strucure contains pack set new SMS callback parameters.

Parameters

status	callback parameter

8.527.2 Field Documentation

8.527.2.1 enum eqmiCbkSetStatus pack_sms_SetNewSMSCallback_t::status

8.528 pack_sms_SetSMSCAddress_t Struct Reference

Data Fields

- uint8_t * pSMSCAddress
- uint8_t * pSMSCType

8.528.1 Detailed Description

Structure to Set the SMS center address.

Parameters

SMSCAddress	• The SMS center address represented as a NULL terminated string
SMSCType	• The SMS center address type represented as a NULL terminated string (optional).

8.528.2 Field Documentation

8.528.2.1 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCAddress

8.528.2.2 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCType

8.529 pack_sms_SLQSDeleteSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t * pMessageIndex
- uint32_t * pMessageTag
- uint8_t * pMessageMode

8.529.1 Detailed Description

This structure contains pack delete SMS parameters.

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
pMessageIndex	• (Optional) message index
pMessageTag	• (Optional) message tag – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
pMessageMode	• (Optional) message mode • this must be included if the device is capable of supporting more than one protocol • e.g. CDMA and GW – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)

8.529.2 Field Documentation

8.529.2.1 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageIndex

8.529.2.2 uint8_t* pack_sms_SLQSDeleteSMS_t::pMessageMode

8.529.2.3 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageTag

8.529.2.4 uint32_t pack_sms_SLQSDeleteSMS_t::storageType

8.530 pack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex
- uint8_t * pMessageMode

8.530.1 Detailed Description

This sturcture contains pack get SMS parameters.

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
messageIndex	• Message index
pMessageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.530.2 Field Documentation

8.530.2.1 uint32_t pack_sms_SLQSGetSMS_t::messageIndex

8.530.2.2 uint8_t* pack_sms_SLQSGetSMS_t::pMessageMode

8.530.2.3 uint32_t pack_sms_SLQSGetSMS_t::storageType

8.531 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode

8.531.1 Detailed Description

This structure contains get SMS boardcast configure parameter.

Parameters

mode	• Mode – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)

8.531.2 Field Documentation

8.531.2.1 uint8_t pack_sms_SLQSGetSmsBroadcastConfig_t::mode

8.532 pack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t * pRequestedTag
- uint8_t * pMessageMode

8.532.1 Detailed Description

This structure contains pack get SMS list parameters.

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
requestedTag	• (Optional) Message tag – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
messageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.532.2 Field Documentation

8.532.2.1 uint8_t* pack_sms_SLQSGetSMSList_t::pMessageMode

8.532.2.2 uint32_t* pack_sms_SLQSGetSMSList_t::pRequestedTag

8.532.2.3 uint32_t pack_sms_SLQSGetSMSList_t::storageType

8.533 pack_sms_SLQSModifySMSStatus_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex
- uint32_t messageTag
- uint8_t * pMessageMode

8.533.1 Detailed Description

This structure contains pack modify SMS status parameters.

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
messageIndex	• Message index
messageTag	• Message tag – 0 - Read – 1 - Not read
pMessageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.533.2 Field Documentation

8.533.2.1 uint32_t pack_sms_SLQSModifySMSStatus_t::messageIndex

8.533.2.2 uint32_t pack_sms_SLQSModifySMSStatus_t::messageTag

8.533.2.3 uint8_t* pack_sms_SLQSModifySMSStatus_t::pMessageMode

8.533.2.4 uint32_t pack_sms_SLQSModifySMSStatus_t::storageType

8.534 pack_sms_SLQSSendAsyncSMS_t Struct Reference

Data Fields

- sms_sendAsyncsmsParams * pSendSmsParams

8.534.1 Detailed Description

Structure contain Parameter to Send an SMS message for immediate over-the-air transmission

Parameters

pSendSms-Params	• structure containing the SMS parameters. Refer sms_sendasyncsmsparams

8.534.2 Field Documentation

8.534.2.1 sms_sendAsyncsmsParams* pack_sms_SLQSSendAsyncSMS_t::pSendSmsParams

8.535 pack_sms_SLQSSetIndicationRegister_t Struct Reference

Data Fields

- sms_setIndicationReg * pSetIndicationRegReq

8.535.1 Detailed Description

Structure contain Parameter that sets the registration state of different WMS indications.

Parameters

pSetIndication-Reg	• Pointer to structure of indicationRegReqParams – See sms_setIndicationReg for more information

8.535.2 Field Documentation

8.535.2.1 sms_setIndicationReg* pack_sms_SLQSSetIndicationRegister_t::pSetIndicationRegReq

8.536 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference

Data Fields

- uint8_t mode
- uint8_t broadcastActivate

8.536.1 Detailed Description

Structure contain parameters that Enables or disables the reception of broadcast SMS messages.

Parameters

Mode	• Mode • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)
broadcast-Activate	• 0x00 - Disable broadcast • 0x01 - Activate broadcast

8.536.2 Field Documentation

8.536.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::broadcastActivate

8.536.2.2 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::mode

8.537 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode
- sms_qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig
- sms_qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig

8.537.1 Detailed Description

Structure containing parameters that provides the information about the SMS BroadcastConfiguration

Parameters

mode	• Mode – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)
pBroadcast-Config	• The data for 3GPP Broadcast Information(Optional).
pCDMA-BroadcastConfig	• The data for 3GPP2 Broadcast Information(Optional).

8.537.2 Field Documentation

8.537.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastConfig_t::mode

8.537.2.2 sms_qaQmi3GPPBroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pBroadcastConfig

8.537.2.3 sms_qaQmi3GPP2BroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pCDMABroadcastConfig

8.538 pack_sms_SLQSSetSmsStorage_t Struct Reference

Data Fields

- uint8_t smsStorage

8.538.1 Detailed Description

Structure contain Parameter to set the SMS Storage on the device

Parameters

smsStorage	• SMS Storage – 0x01 - device's permanent memory – 0x02 - UICC

8.538.2 Field Documentation

8.538.2.1 uint8_t pack_sms_SLQSSetSmsStorage_t::smsStorage

8.539 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeReq * pMaxStorageSizeReq

8.539.1 Detailed Description

This structure contains get maximum storage size.

Parameters

pMaxStorage-SizeReq	• Request parameters for SmsSLQSGetMaxStorageSize – See sms_maxStorageSizeReq for more information

8.539.2 Field Documentation

8.539.2.1 sms_maxStorageSizeReq* pack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeReq

8.540 pack_sms_SLQSSmsSetRoutes_t Struct Reference

Data Fields

- sms_setRoutesReq * pSetRoutesReq

8.540.1 Detailed Description

Structure contain parameter that Sets the action performed on SMS message receipt for specified message routes. It also specifies the action performed on SMS receipt of status reports.

Parameters

pSetRoutesReq	• Pointer to structure of smsSetRoutesReq – See sms_setRoutesReq for more information

8.540.2 Field Documentation

8.540.2.1 sms_setRoutesReq* pack_sms_SLQSSmsSetRoutes_t::pSetRoutesReq

8.541 pack_swiaudio_SLQSGetM2MAudioProfile_t Struct Reference

Data Fields

- uint8_t * pGenerator

8.541.1 Detailed Description

This structure contains the SLQSGetM2MAudioProfile pack parameters.

Parameters

p-Generator[optional]	• Generator – 0 - Voice

8.541.2 Field Documentation

8.541.2.1 uint8_t* pack_swiaudio_SLQSGetM2MAudioProfile_t::pGenerator

8.542 pack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator

8.542.1 Detailed Description

This structure contains the SLQSGetM2MAudioVolume pack parameters.

Parameters

Profile	• Audio Profile – 0-5
Generator	• Generator – 0 - Voice

8.542.2 Field Documentation

8.542.2.1 uint8_t pack_swiaudio_SLQSGetM2MAudioVolume_t::Generator

8.542.2.2 uint8_t pack_swiaudio_SLQSGetM2MAudioVolume_t::Profile

8.543 pack_swiaudio_SLQSGetM2MAVMute_t Struct Reference

Data Fields

- uint8_t Profile

8.543.1 Detailed Description

This structure contains the SLQSGetM2MAVMute pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5

8.543.2 Field Documentation

8.543.2.1 uint8_t pack_swiaudio_SLQSGetM2MAVMute_t::Profile

8.544 pack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference

Data Fields

- uint8_t Profile

8.544.1 Detailed Description

This structure contains the SLQSGetM2MSpkrGain pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5

8.544.2 Field Documentation

8.544.2.1 uint8_t pack_swiaudio_SLQSGetM2MSpkrGain_t::Profile

8.545 pack_swiaudio_SLQSSetM2MAudioAVCFG_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Device
- uint8_t PIFACEId
- swiaudio_PCMparams * pPCMParams

8.545.1 Detailed Description

This structure contains the SLQSSetM2MAudioAVCFG pack parameters.

Parameters

Profile	• Audio Profile – 0-5

Device	• ACDB Device • See qaGobiApiTableSwiAudio.h for more information on ACDB Device
PIFACEId	• Physical Interface • See qaGobiApiTableSwiAudio.h for more information on physical interface
pPCMParams	• PCM parameters • See swiaudio_PCMparams for more information

8.545.2 Field Documentation

8.545.2.1 uint8_t pack_swiaudio_SLQSSetM2MAudioAVCFG_t::Device

8.545.2.2 uint8_t pack_swiaudio_SLQSSetM2MAudioAVCFG_t::PIFACEId

8.545.2.3 swiaudio_PCMparams* pack_swiaudio_SLQSSetM2MAudioAVCFG_t::pPCMParams

8.545.2.4 uint8_t pack_swiaudio_SLQSSetM2MAudioAVCFG_t::Profile

8.546 pack_swiaudio_SLQSSetM2MAudioLPBK_t Struct Reference

Data Fields

- uint8_t Enable

8.546.1 Detailed Description

This structure contains the SLQSSetM2MAudioLPBK pack parameters.

Parameters

Enable	• Operation to be performed – 0 - stop – 1 - VOCODER loop – 2 - internal codec loop

8.546.2 Field Documentation

8.546.2.1 uint8_t pack_swiaudio_SLQSSetM2MAudioLPBK_t::Enable

8.547 pack_swiaudio_SLQSSetM2MAudioProfile_t Struct Reference

Data Fields

- uint8_t Profile

- uint8_t * pEarMute
- uint8_t * pMicMute
- uint8_t * pGenerator
- uint8_t * pVolume
- uint8_t * pCwtMute

8.547.1 Detailed Description

This structure contains the SLQSSetM2MAudioProfile pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5
pEarMute	• Ear Mute – 0 - mute – 1 - unmute
pMicMute	• Mic Mute – 0 - mute – 1 - unmute
pGenerator	• Generator – 0 - voice
pVolume	• Set RX Volume level – 0-5
pCwtMute	• Call Waiting Tone Mute – 0 - Mute – 1 - UnMute

8.547.2 Field Documentation

8.547.2.1 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pCwtMute

8.547.2.2 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pEarMute

8.547.2.3 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pGenerator

8.547.2.4 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pMicMute

8.547.2.5 uint8_t pack_swiaudio_SLQSSetM2MAudioProfile_t::Profile

8.547.2.6 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pVolume

8.548 pack_swiaudio_SLQSSetM2MAudioVolume_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator
- uint8_t Level

8.548.1 Detailed Description

This structure contains the SLQSSetM2MAudioProfile pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5
Generator	• Generator – 0 - voice
Level	• Audio volume level – 0-5

8.548.2 Field Documentation

8.548.2.1 uint8_t pack_swiaudio_SLQSSetM2MAudioVolume_t::Generator

8.548.2.2 uint8_t pack_swiaudio_SLQSSetM2MAudioVolume_t::Level

8.548.2.3 uint8_t pack_swiaudio_SLQSSetM2MAudioVolume_t::Profile

8.549 pack_swiaudio_SLQSSetM2MAVMute_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t EarMute
- uint8_t MicMute
- uint8_t * pCwtMute

8.549.1 Detailed Description

This structure contains the SLQSSetM2MAVMute pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5

EarMute	• Ear Mute – 0-1
MicMute	• Mic Mute – 0-1
pCwtMute	[Optional] • Call Waiting Tone Mute – 0-1

8.549.2 Field Documentation

8.549.2.1 uint8_t pack_swiaudio_SLQSSetM2MAVMute_t::EarMute

8.549.2.2 uint8_t pack_swiaudio_SLQSSetM2MAVMute_t::MicMute

8.549.2.3 uint8_t* pack_swiaudio_SLQSSetM2MAVMute_t::pCwtMute

8.549.2.4 uint8_t pack_swiaudio_SLQSSetM2MAVMute_t::Profile

8.550 pack_swiaudio_SLQSSetM2MSpkrGain_t Struct Reference

Data Fields

- uint8_t Profile
- uint16_t Value

8.550.1 Detailed Description

This structure contains the SLQSSetM2MSpkrGain pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5
Value	• RX speakerphone gain – 0x0 - 0x7fff

8.550.2 Field Documentation

8.550.2.1 uint8_t pack_swiaudio_SLQSSetM2MSpkrGain_t::Profile

8.550.2.2 uint16_t pack_swiaudio_SLQSSetM2MSpkrGain_t::Value

8.551 pack_swiavms_SLQSAVMSSendSelection_t Struct Reference

Data Fields

- uint8_t selection
- uint32_t * pDeferTime
- uint8_t * pClientPerformOperationFlag
- uint8_t * pPackageID
- uint16_t * pRejectReason

8.551.1 Detailed Description

Structure containing the AVMS selection

Parameters

selection[IN]	• User Selection – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer
pDeferTime[IN/-OPTIONAL]	• Defer time in minutes. A value of 0 will cause the prompt to be resent immediately.
pClientPerform-OperationFlag[I-N/OPTIONAL]	• Client operation flag after accept. – 0: if modem performs the operation (download or update) – 1: if client performs the operation (download or update)
pPackageID[IN/-OPTIONAL]	• Package ID.
pRejectReason[-IN/OPTIONAL]	• Reject Reason.

8.551.2 Field Documentation

8.551.2.1 uint8_t* pack_swiavms_SLQSAVMSSendSelection_t::pClientPerformOperationFlag

8.551.2.2 uint32_t* pack_swiavms_SLQSAVMSSendSelection_t::pDeferTime

8.551.2.3 uint8_t* pack_swiavms_SLQSAVMSSendSelection_t::pPackageID

8.551.2.4 uint16_t* pack_swiavms_SLQSAVMSSendSelection_t::pRejectReason

8.551.2.5 uint8_t pack_swiavms_SLQSAVMSSendSelection_t::selection

8.552 pack_swiavms_SLQSAVMSSetSettings_t Struct Reference

Data Fields

- uint8_t AutoConnect
- uint8_t AutoReboot
- uint8_t PromptFwDownload
- uint8_t PromptFwUpdate
- uint8_t * pFwAutoSDM
- uint32_t * pPollingTimer
- PackSwiAvmsSetSettingsConnectionRetryTimers * pConnectionRetryTimers
- PackSwiAvmsSetSettingsAPNInfo * pAPNInfo
- uint8_t * pNotifStore
- PackSwiAvmsSetSettingsPeriodInfo * pPeriodInfo

8.552.1 Detailed Description

Structure containing the AVMS settings to be set on the device This maps to structure SLQSAVMSSetSettings

Parameters

AutoConnect[IN]	• Auto Connect to AirVantage server – 0x00 - FALSE – 0x01 - TRUE
AutoReboot[IN]	• Automatic device reboot when the request is received from AirVantage server – 0x00 - FALSE – 0x01 - TRUE
PromptFwDownload[IN]	• Firmware Auto Download – 0x00 - FALSE – 0x01 - TRUE
PromptFwUpdate[IN]	• 1 byte parameter indicating FOTA Automatic update – 0x00 - Firmware autoupdate FALSE – 0x01 - Firmware autoupdate TRUE
pFwAutoSDM[IN/OPTIONAL]	• OMA Automatic UI Alert Response – 0x00 - DISABLED – 0x01 - ENABLED ACCEPT – 0x02 - ENABLED REJECT
pPollingTimer[IN/OPTIONAL]	• Polling timer to connect to AVMS server – 0-525600 (min) – 0:disabled
pConnectionRetryTimers[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsConnectionRetryTimers for more information

pAPNInfo[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information
pNotifStore[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information
pPeriodInfo[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information

8.552.2 Field Documentation

8.552.2.1 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::AutoConnect

8.552.2.2 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::AutoReboot

8.552.2.3 PackSwiAvmsSetSettingsAPNInfo* pack_swiavms_SLQSAVMSSetSettings_t::pAPNInfo

8.552.2.4 PackSwiAvmsSetSettingsConnectionRetryTimers* pack_swiavms_SLQSAVMSSetSettings_t::pConnectionRetryTimers

8.552.2.5 uint8_t* pack_swiavms_SLQSAVMSSetSettings_t::pFwAutoSDM

8.552.2.6 uint8_t* pack_swiavms_SLQSAVMSSetSettings_t::pNotifStore

8.552.2.7 PackSwiAvmsSetSettingsPeriodInfo* pack_swiavms_SLQSAVMSSetSettings_t::pPeriodInfo

8.552.2.8 uint32_t* pack_swiavms_SLQSAVMSSetSettings_t::pPollingTimer

8.552.2.9 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::PromptFwDownload

8.552.2.10 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::PromptFwUpdate

8.553 pack_swiavms_SLQSAVMSSetSettings_v2_t Struct Reference

Data Fields

- uint8_t AutoConnect
- uint8_t PromptFwDownload
- uint8_t PromptFwUpdate
- uint8_t * pFwAutoSDM
- uint32_t * pPollingTimer
- PackSwiAvmsSetSettingsConnectionRetryTimers * pConnectionRetryTimers
- PackSwiAvmsSetSettingsAPNInfo * pAPNInfo
- uint8_t * pNotifStore
- PackSwiAvmsSetSettingsPeriodInfo * pPeriodInfo
- uint8_t * pAutoReboot

8.553.1 Detailed Description

Structure containing the AVMS settings to be set on the device This maps to structure SLQSAVMSSetSettings_v2 (For AVC2 service)

Parameters

AutoConnect[IN]	• Auto Connect to AirVantage server – 0x00 - FALSE – 0x01 - TRUE
PromptFwDownload[IN]	• Firmware Auto Download – 0x00 - FALSE – 0x01 - TRUE
PromptFwUpdate[IN]	• 1 byte parameter indicating FOTA Automatic update – 0x00 - Firmware autoupdate FALSE – 0x01 - Firmware autoupdate TRUE
pFwAutoSDM[IN/OPTIONAL]	• OMA Automatic UI Alert Response – 0x00 - DISABLED – 0x01 - ENABLED ACCEPT – 0x02 - ENABLED REJECT
pPollingTimer[IN/OPTIONAL]	• Polling timer to connect to AVMS server – 0-525600 (min) – 0:disabled
pConnectionRetryTimers[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsConnectionRetryTimers for more information
pAPNInfo[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information
pNotifStore[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information
pPeriodInfo[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information
pAutoReboot[IN/OPTIONAL]	• Automatic device reboot when the request is received from AirVantage server – 0x00 - FALSE – 0x01 - TRUE

Note

Setting PromptFwDownload/PromptFwUpdate as TRUE implies AutoConnect to be also TRUE even if AutoConnect was set FALSE initially. Automatic firmware download/update selection trumps the connection setting flag.

8.553.2 Field Documentation

8.553.2.1 uint8_t pack_swiavms_SLQSAVMSSetSettings_v2_t::AutoConnect

8.553.2.2 PackSwiAvmsSetSettingsAPNInfo* pack_swiavms_SLQSAVMSSetSettings_v2_t::pAPNInfo

8.553.2.3 uint8_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pAutoReboot

8.553.2.4 PackSwiAvmsSetSettingsConnectionRetryTimers* pack_swiavms_SLQSAVMSSetSettings_v2_t::pConnectionRetryTimers

8.553.2.5 uint8_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pFwAutoSDM

8.553.2.6 uint8_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pNotifStore

8.553.2.7 PackSwiAvmsSetSettingsPeriodInfo* pack_swiavms_SLQSAVMSSetSettings_v2_t::pPeriodInfo

8.553.2.8 uint32_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pPollingTimer

8.553.2.9 uint8_t pack_swiavms_SLQSAVMSSetSettings_v2_t::PromptFwDownload

8.553.2.10 uint8_t pack_swiavms_SLQSAVMSSetSettings_v2_t::PromptFwUpdate

8.554 pack_swiavms_SLQSAVMSStartSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.554.1 Detailed Description

Structure that contains the session type request for AVMS start session command

Parameters

sessionType[IN]	• Session type – 0x01 - FOTA, to check availability of FW Update. This field is mandatory in OMA-DM case but not necessary in LWM2M one. To keep compatibility, this shall be used in LWM2M without any specific treatment.

8.554.2 Field Documentation

8.554.2.1 uint8_t pack_swiavms_SLQSAVMSStartSession_t::sessionType

8.555 pack_swiavms_SLQSAVMSStopSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.555.1 Detailed Description

Structure that contains the session type for AVMS Stop session command

Parameters

sessionType	• Session type – 0x01 - FOTA, to suspend FOTA session. – 0xFF - Suspend ongoing FOTA session or stop any other active AVMS session.
resultcode	• Tlv Result Code

8.555.2 Field Documentation

8.555.2.1 uint8_t pack_swiavms_SLQSAVMSStopSession_t::sessionType

8.556 pack_swidms_SLQSSwiDmsSetHWWatchdog_t Struct Reference

Data Fields

- uint32_t timeout
- uint32_t resetDelay
- uint8_t enable

8.556.1 Detailed Description

This structure contains the set hardware watchdong settings request parameter.

Parameters

timeout	• timeout value for HW watchdog (unit in second)
resetDelay	• delay before reset after watchdog timeout (unit in second)
enable	• 0 to disable watchdog; 1 to enable watchdog

8.556.2 Field Documentation

8.556.2.1 uint8_t pack_swidms_SLQSSwiDmsSetHWWatchdog_t::enable

8.556.2.2 uint32_t pack_swidms_SLQSSwiDmsSetHWWatchdog_t::resetDelay

8.556.2.3 uint32_t pack_swidms_SLQSSwiDmsSetHWWatchdog_t::timeout

8.557 pack_swidms_SLQSSwiDmsSetMTU_t Struct Reference

Data Fields

- uint16_t MTUSize

8.557.1 Detailed Description

This structure contains the Set MTU request parameter.

Parameters

MTUSize	• Set MTU size • MTU Values – 0 - use default value – 576 to 2000 - other values required by carrier. – Set the same MTU for all RAT/interfaces. – New MTU size effective after modem reboot.

8.557.2 Field Documentation

8.557.2.1 uint16_t pack_swidms_SLQSSwiDmsSetMTU_t::MTUSize

8.558 pack_swidms_SLQSSwiDmsSetUimSlot_t Struct Reference

Data Fields

- uint8_t slotSelectaction
- uint8_t * pSlotSelect
- swidms_SetMultiplexedModeTlv * pMultiplexedSimMode
- uint8_t * pAutoSwitchMode

8.558.1 Detailed Description

This structure contains the set Uim slot

Parameters

slotSelectaction	• UIM Slot selection action • Values – 1 - select UIM slot – 2 - select multiplexed SIM slot – 3 - set UIM auto-switch mode
pSlotSelect	• Select UIM slot • Values – 0 - UIM1 – 1 - UIM2 – 2 - Reserved (Remote SIM)
pMultiplexed-SimMode	• Multiplexed SIM mode • See swidms_SetMultiplexedModeTlv for more information

pAutoSwitch-Mode	• Values – 0 - Auto switch off – 1 - Auto switch UIM1 prefer – 2 - Auto switch UIM2 prefer

8.558.2 Field Documentation

8.558.2.1 uint8_t* pack_swidms_SLQSSwiDmsSetUimSlot_t::pAutoSwitchMode

8.558.2.2 swidms_SetMultiplexedModeTlv* pack_swidms_SLQSSwiDmsSetUimSlot_t::pMultiplexedSimMode

8.558.2.3 uint8_t* pack_swidms_SLQSSwiDmsSetUimSlot_t::pSlotSelect

8.558.2.4 uint8_t pack_swidms_SLQSSwiDmsSetUimSlot_t::slotSelectaction

8.559 pack_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference

Data Fields

- uint32_t CfgValue

8.559.1 Detailed Description

This structure contains the Set Usb Interface Config request parameter.

Parameters

CfgValue	• Host composition bit mask, see supported USB interface bitmasks, not supported by QMI object setting for 9x50 modules – 0x00000001 - DIAG interface – 0x00000002 - ADB interface – 0x00000004 - NMEA interface – 0x00000008 - MODEM interface – 0x00000010 - RESERVED5 – 0x00000020 - RESERVED6 – 0x00000040 - RESERVED7 – 0x00000080 - RESERVED8 – 0x00000100 - RMENT0 interface – 0x00000200 - RESERVED10 – 0x00000400 - RMENT1 interface – 0x00000800 - RESERVED12 – 0x00001000 - MBIM interface – 0x00002000 - RESERVED14 – 0x00004000 - RNDIS interface – 0x00008000 - RESERVED16 – 0x00010000 - AUDIO interface – 0x00020000 - RESERVED18 – 0x00080000 - ECM interface – 0x00100000 - RESERVED21 – 0x00200000 - RESERVED22 – 0xFFC00000 - RESERVED

8.559.2 Field Documentation

8.559.2.1 uint32_t pack_swidms_SLQSSwiDmsSetUsbComp_t::CfgValue

8.560 pack_swidms_SLQSSwiDmsSetUsbNetNum_t Struct Reference

Data Fields

- uint8_t nUsbNetNum

8.560.1 Detailed Description

This structure contains usb net number to set on remote endpoint for QMAP configuration

Parameters

nUsbNetNum	• value of usb net numbers to set for the device • range is 0-8

8.560.2 Field Documentation

8.560.2.1 uint8_t pack_swidms_SLQSSwiDmsSetUsbNetNum_t::nUsbNetNum

8.561 pack_swiloc_SwiLocSetAutoStart_t Struct Reference

Data Fields

- uint8_t function
- int set_function
- uint8_t fix_type
- int set_fix_type
- uint8_t max_time
- int set_max_time
- uint32_t max_dist
- int set_max_dist
- uint32_t fix_rate
- int set_fix_rate

8.561.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

function	• Setting to indicate when modem should start an automatic GNSS fix – 0 - disabled – 1 - At bootup – 2 - When NMEA port is opened
set_function	• 0 - do not set to modem • l - set to modem
fix_type	• Type of GNSS fix: – 1 - Default Engine mode – 2 - MS-Based – 3 - MS-Assisted – 4 - Standalone
set_fix_type	• 0 - do not set to modem • l - set to modem
max_time	• Maximum time allowed for the receiver to get a fix in seconds • Valid range: 1-255
set_max_time	• 0 - do not set to modem • l - set to modem

max_dist	• Maximum uncertainty of a fix measured by distance in meters • Valid range: 1 - 4294967280
set_max_dist	• 0 - do not set to modem • l - set to modem
fix_rate	• Time between fixes in seconds • Valid range: 1–65535
set_fix_rate	• 0 - do not set to modem • l - set to modem

8.561.2 Field Documentation

8.561.2.1 uint32_t pack_swiloc_SwiLocSetAutoStart_t::fix_rate

8.561.2.2 uint8_t pack_swiloc_SwiLocSetAutoStart_t::fix_type

8.561.2.3 uint8_t pack_swiloc_SwiLocSetAutoStart_t::function

8.561.2.4 uint32_t pack_swiloc_SwiLocSetAutoStart_t::max_dist

8.561.2.5 uint8_t pack_swiloc_SwiLocSetAutoStart_t::max_time

8.561.2.6 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_rate

8.561.2.7 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_type

8.561.2.8 int pack_swiloc_SwiLocSetAutoStart_t::set_function

8.561.2.9 int pack_swiloc_SwiLocSetAutoStart_t::set_max_dist

8.561.2.10 int pack_swiloc_SwiLocSetAutoStart_t::set_max_time

8.562 pack_swioma_SLQSOMADMCancelSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.562.1 Detailed Description

Structure that contains the session type for OMA cancel session command

Parameters

sessionType	[in] • Session type – 0x01 - FOTA, to check availability of FW Update – 0xFF - Cancel any active OMADM session

8.562.2 Field Documentation

8.562.2.1 uint32_t pack_swioma_SLQSOMADMCancelSession_t::sessionType

8.563 pack_swioma_SLQSOMADMCancelSessionExt_t Struct Reference

Data Fields

- uint32_t sessionType

8.563.1 Detailed Description

Structure that contains the session type for OMA cancel session command

Parameters

sessionType	[IN] • Session type – 0x01 - FOTA, to check availability of FW Update – 0xFF - Cancel any active OMADM session

8.563.2 Field Documentation

8.563.2.1 uint32_t pack_swioma_SLQSOMADMCancelSessionExt_t::sessionType

8.564 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference

Data Fields

- uint32_t SessionType

8.564.1 Detailed Description

Structure that contains the session type for OMA get session info command

Parameters

SessionType	[in] • Session type – 0x01 - FOTA – 0xFF - Any active OMADM session. If no active sessions are available, then previous OMADM session info is returned

8.564.2 Field Documentation

8.564.2.1 uint32_t pack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.565 pack_swioma_SLQSOMADMSendSelection_t Struct Reference

Data Fields

- uint32_t selection
- uint32_t * pDeferTime
- uint32_t * pRejectReason

8.565.1 Detailed Description

Structure containing the OMA DM selection

Parameters

selection	[in] • OMA-DM NIA Selection – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer
pDeferTime	[in] • Defer time in minutes. A value of 0 will cause the prompt to be resent immediately. • This TLV is mandatory if selection is set to 0x03.
pRejectReason	[in] • Reject Reason • This TLV is processed if selection is set to 0x02. If it is not present, the reject reason 0 is used as default.

8.565.2 Field Documentation

8.565.2.1 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pDeferTime

8.565.2.2 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pRejectReason

8.565.2.3 uint32_t pack_swioma_SLQSOMADMSendSelection_t::selection

8.566 pack_swioma_SLQSOMADMSendSelectionExt_t Struct Reference

Data Fields

- uint32_t selection

8.566.1 Detailed Description

Structure containing the OMA DM selection

Parameters

selection	[IN] • OMA-DM NIA Selection – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer

8.566.2 Field Documentation

8.566.2.1 uint32_t pack_swioma_SLQSOMADMSendSelectionExt_t::selection

8.567 pack_swioma_SLQSOMADMSetSettings_t Struct Reference

Data Fields

- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t * pAutosdm
- uint8_t * pFwAutoCheck

8.567.1 Detailed Description

Structure containing the OMA DM settings to be set on the device This maps to structure SLQSOMADMSettings-ReqParams3

Parameters

FOTAdownload	• 1 Byte parameter indicating support for FOTA Automatic download – 0x00 - Firmware autodownload FALSE – 0x01 - Firmware autodownload TRUE
FOTAUpdate	• 1 byte parameter indicating FOTA Automatic update – 0x00 - Firmware autoupdate FALSE – 0x01 - Firmware autoupdate TRUE
pAutosdm	• Optional 1 byte parameter indicating OMA Automatic UI Alert Response – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject
pFwAutoCheck	• Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response – 0x00 - Disabled – 0x01 - Enabled

8.567.2 Field Documentation

8.567.2.1 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAdownload

8.567.2.2 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAUpdate

8.567.2.3 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pAutosdm

8.567.2.4 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pFwAutoCheck

8.568 pack_swioma_SLQSOMADMSetSettingsExt_t Struct Reference

Data Fields

- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t OMADMEnable
- uint8_t OMADMLogEnable
- uint8_t FUMOEnable
- uint8_t PRLEnable

8.568.1 Detailed Description

Structure containing the OMA DM settings to be set on the device This maps to structure SLQSOMADMSetSettingsExt

Parameters

FOTAdownload	• 1 Byte parameter indicating support for FOTA Automatic download – 0x00 - Firmware autodownload FALSE – 0x01 - Firmware autodownload TRUE
FOTAUpdate	• 1 byte parameter indicating FOTA Automatic update – 0x00 - Firmware autoupdate FALSE – 0x01 - Firmware autoupdate TRUE
OMADMEnable	[IN] • 1 byte parameter indicating OMA DM – 0x00 - OMA DM Disabled – 0x01 - OMA DM Enabled
OMADMLogEnable	[IN] • 1 byte parameter indicating OMA DM Logs – 0x00 - OMA DM Logs Disabled – 0x01 - OMA DM Logs Enabled
FUMOEnable	[IN] • 1 byte parameter indicating FUMO enabled – 0x00 - FUMO Disabled – 0x01 - FUMO enabled

PRLEnable	[IN] • 1 byte parameter indicating PRL Enabled – 0x00 - PRL Disabled – 0x01 - PRL Enabled – 0x02 - Launch a CI PRL – 0x03 - PRL Enabled and PRL automatic update every 45 days – 0x04 - PRL Enabled and PRL update ever 90 days

8.568.2 Field Documentation

8.568.2.1 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::FOTAdownload

8.568.2.2 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::FOTAUpdate

8.568.2.3 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::FUMOEnable

8.568.2.4 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::OMADMEnable

8.568.2.5 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::OMADMLogEnable

8.568.2.6 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::PRLEnable

8.569 pack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.569.1 Detailed Description

Structure that contains the session type for OMA start session command

Parameters

sessionType[IN]	• Session type – 0x01 - FOTA, to check availability of FW Update – 0x02 - DM, to check availability of DM Update – 0x03 - PRL, to check availability of PRL Update

8.569.2 Field Documentation

8.569.2.1 uint32_t pack_swioma_SLQSOMADMStartSession_t::sessionType

8.570 pack_swioma_SLQSOMADMStartSessionExt_t Struct Reference

Data Fields

- uint32_t sessionType

8.570.1 Detailed Description

Structure that contains the session type for OMA start session command

Parameters

sessionType	[IN] • Session type – 0x01 - FOTA, to check availability of FW Update – 0x02 - DM, to check availability of DM Update – 0x03 - PRL, to check availability of PRL Update

8.570.2 Field Documentation

8.570.2.1 uint32_t pack_swioma_SLQSOMADMStartSessionExt_t::sessionType

8.571 pack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference

Data Fields

- uint8_t mitigationDevIDLen
- char mitigationDevID [255]

8.571.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

mitigationDevID-Len	• Number of sets of the following elements – mitigation_dev_id
mitigationDevID	• Mitigation device ID

8.571.2 Field Documentation

8.571.2.1 char pack_tmd_SLQSTmdDeRegNotMitigationLvl_t::mitigationDevID[255]

8.571.2.2 uint8_t pack_tmd_SLQSTmdDeRegNotMitigationLvl_t::mitigationDevIDLen

8.572 pack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference

Data Fields

- uint8_t mitigationDevIDLen
- char mitigationDevID [255]

8.572.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

mitigationDevID-Len	• Number of sets of the following elements – mitigation_dev_id
mitigationDevID	• Mitigation device ID

8.572.2 Field Documentation

8.572.2.1 char pack_tmd_SLQSTmdGetMitigationLvl_t::mitigationDevID[255]

8.572.2.2 uint8_t pack_tmd_SLQSTmdGetMitigationLvl_t::mitigationDevIDLen

8.573 pack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference

Data Fields

- uint8_t mitigationDevIDLen
- char mitigationDevID [255]

8.573.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

mitigationDevID-Len	• Number of sets of the following elements – mitigation_dev_id
mitigationDevID	• Mitigation device ID

8.573.2 Field Documentation

8.573.2.1 char pack_tmd_SLQSTmdRegNotMitigationLvl_t::mitigationDevID[255]

8.573.2.2 uint8_t pack_tmd_SLQSTmdRegNotMitigationLvl_t::mitigationDevIDLen

8.574 pack_uim_ChangePin_t Struct Reference

Data Fields

- uim_encryptedPIN1 EncryptedPIN1
- uint32_t * pIndicationToken

- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_changeUIMPIN changePIN
- uint16_t TlvResult

8.574.1 Detailed Description

This structure contains information of the request parameters associated with a Change PIN API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
changePIN	• See uim_changeUIMPIN for more information.
pKeyReferenceID(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.574.2 Field Documentation

8.574.2.1 uim_changeUIMPIN pack_uim_ChangePin_t::changePIN

8.574.2.2 uim_encryptedPIN1 pack_uim_ChangePin_t::EncryptedPIN1

8.574.2.3 uint32_t* pack_uim_ChangePin_t::pIndicationToken

8.574.2.4 uint8_t* pack_uim_ChangePin_t::pKeyReferenceID

8.574.2.5 uim_sessionInformation pack_uim_ChangePin_t::sessionInfo

8.574.2.6 uint16_t pack_uim_ChangePin_t::TlvResult

8.575 pack_uim_ReadTransparent_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_fileInfo fileIndex
- uim_readTransparentInfo readTransparent
- uint32_t * pIndicationToken
- uint8_t * pEncryptData
- uint16_t TlvResult

8.575.1 Detailed Description

This structure contains information of the request parameters associated with a Read Transparent API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
fileIndex	• See uim_fileInfo for more information.
readTransparent	• See uim_readTransparentInfo for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function
pEncrypt-Data(optional)	• Encrypt Data. • Indicates whether the data read from the card is to be encrypted.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.575.2 Field Documentation

8.575.2.1 uim_fileInfo pack_uim_ReadTransparent_t::fileIndex

8.575.2.2 uint8_t* pack_uim_ReadTransparent_t::pEncryptData

8.575.2.3 uint32_t* pack_uim_ReadTransparent_t::pIndicationToken

8.575.2.4 uim_readTransparentInfo pack_uim_ReadTransparent_t::readTransparent

8.575.2.5 uim_sessionInformation pack_uim_ReadTransparent_t::sessionInfo

8.575.2.6 uint16_t pack_uim_ReadTransparent_t::TlvResult

8.576 pack_uim_SetPinProtection_t Struct Reference

Data Fields

- uim_encryptedPIN1 EncryptedPIN1
- uint32_t * pIndicationToken
- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_setPINProtection pinProtection
- uint16_t TlvResult

8.576.1 Detailed Description

This structure contains information of the request parameters associated with a set pin protection API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
pinProtection	• See uim_setPINProtection for more information.
pKeyReferenceID(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.576.2 Field Documentation

8.576.2.1 uim_encryptedPIN1 pack_uim_SetPinProtection_t::EncryptedPIN1

8.576.2.2 uint32_t* pack_uim_SetPinProtection_t::pIndicationToken

8.576.2.3 uim_setPINProtection pack_uim_SetPinProtection_t::pinProtection

8.576.2.4 uint8_t* pack_uim_SetPinProtection_t::pKeyReferenceID

8.576.2.5 uim_sessionInformation pack_uim_SetPinProtection_t::sessionInfo

8.576.2.6 uint16_t pack_uim_SetPinProtection_t::TlvresultTlvresult

8.577 pack_uim_SLQSUIMAddProfile_t Struct Reference

Data Fields

- uint8_t slot
- uim_UIMProfileCode profileActInfo
- uim_UIMProfileCode * pProfileConfInfo
- uint8_t * pUserConsent

8.577.1 Detailed Description

This structure contains information of the request parameters associated with a Add Profile API.

Parameters

slot	• any slot number 1 to 5
profileActInfo	• See uim_UIMProfileCode for more information.
pProfileConfInfo	(Optional) • See uim_UIMProfileCode for more information.
pUserConsent	(Optional) • 0: User consent not supported • 1: User consent supported

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.577.2 Field Documentation

8.577.2.1 uim_UIMProfileCode* pack_uim_SLQSUIMAddProfile_t::pProfileConfInfo

8.577.2.2 uim_UIMProfileCode pack_uim_SLQSUIMAddProfile_t::profileActInfo

8.577.2.3 uint8_t* pack_uim_SLQSUIMAddProfile_t::pUserConsent

8.577.2.4 uint8_t pack_uim_SLQSUIMAddProfile_t::slot

8.578 pack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_authenticationData authData
- uint32_t * pIndicationToken

8.578.1 Detailed Description

This structure contains information of the request parameters associated with a SLQSUIMAuthenticate.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
authData	• See uim_authenticationData for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.578.2 Field Documentation

8.578.2.1 uim_authenticationData pack_uim_SLQSUIMAuthenticate_t::authData

8.578.2.2 uint32_t* pack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.578.2.3 uim_sessionInformation pack_uim_SLQSUIMAuthenticate_t::sessionInfo

8.579 pack_uim_SLQSUIMDeleteProfile_t Struct Reference

Data Fields

- uint8_t slot
- uint32_t profileId

8.579.1 Detailed Description

This structure contains information of the request parameters associated with a Delete Profile API.

Parameters

slot	• any slot number 1 to 5
profileId	• any profile number from 1 to 8

8.579.2 Field Documentation

8.579.2.1 uint32_t pack_uim_SLQSUIMDeleteProfile_t::profileId

8.579.2.2 uint8_t pack_uim_SLQSUIMDeleteProfile_t::slot

8.580 pack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_depersonalizationInformation depersonilisationInfo

8.580.1 Detailed Description

This structure contains information of the request parameters associated with a Depersonalization.

Parameters

depersonilisation-Info	• See uim_depersonalizationInformation for more information.

8.580.2 Field Documentation

8.580.2.1 uim_depersonalizationInformation pack_uim_SLQSUIMDepersonalization_t::depersonilisationInfo

8.581 pack_uim_SLQSUIMeUICCMemReset_t Struct Reference

Data Fields

- uint8_t uimSlot
- uint64_t * pResetOption

8.581.1 Detailed Description

This structure contains information of the request parameters associated with Resets the memory of the eUICC card API.

Parameters

uimSlot	• Indicates the recovered slot. • Valid values: – 0x01 - Slot 1 – 0x02 - Slot 2 – 0x03 - Slot 3 – 0x04 - Slot 4 – 0x05 - Slot 5
pResetOption	• Indicates the reset option for eUICC. • Valid values: – 0x01 - Delete all the test profiles – 0x02 - Delete all operational profiles – 0x04 - Reset the default SM-DP+ address All other bits are reserved for future use. if the TLV is missing, complete eUICC card will be reset.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.581.2 Field Documentation

8.581.2.1 uint64_t* pack_uim_SLQSUIMeUICCMemReset_t::pResetOption

8.581.2.2 uint8_t pack_uim_SLQSUIMeUICCMemReset_t::uimSlot

8.582 pack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask

8.582.1 Detailed Description

This structure contains pack event register parameter.

Parameters

eventMask	- bit 1 - card status • bit 4 - physical slot status

8.582.2 Field Documentation

8.582.2.1 uint32_t pack_uim_SLQSUIMEventRegister_t::eventMask

8.583 pack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint32_t * pConfigurationMask

8.583.1 Detailed Description

This structure contains information of the request parameters associated with Get Configuration to get the modem configuration for the UIM module

Parameters

pConfiguration-Mask(optional)	• Requested configurations – Bit 0 - Automatic selection – Bit 1 - Personalization status – Bit 2 - Halt subscription – All other bits are reserved for future use

Note

- if the TLV is missing, the service returns all configuration items in the response.

8.583.2 Field Documentation

8.583.2.1 uint32_t* pack_uim_SLQSUIMGetConfiguration_t::pConfigurationMask

8.584 pack_uim_SLQSUIMGetEID_t Struct Reference

Data Fields

- uint8_t * pUimSlotID

8.584.1 Detailed Description

This structure contains information of the request parameters associated with Get EID API.

Parameters

pUimSlotID	• Indicates the slot for which the EID is requested. Valid values: – UIM_SLOT_1 (0x01) - Slot 1 – UIM_SLOT_2 (0x02) - Slot 2 – UIM_SLOT_3 (0x03) - Slot 3 – UIM_SLOT_4 (0x04) - Slot 4 – UIM_SLOT_5 (0x05) - Slot 5

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.584.2 Field Documentation

8.584.2.1 uint8_t* pack_uim_SLQSUIMGetEID_t::pUimSlotID

8.585 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_fileInfo fileIndex
- uint32_t * pIndicationToken

8.585.1 Detailed Description

This structure contains information of the request parameters associated with a Get File Attributes

Parameters

sessionInfo	• See uim_sessionInformation for more information.
fileIndex	• See uim_fileInfo for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.585.2 Field Documentation

8.585.2.1 uim_fileInfo pack_uim_SLQSUIMGetFileAttributes_t::fileIndex

8.585.2.2 uint32_t* pack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.585.2.3 uim_sessionInformation pack_uim_SLQSUIMGetFileAttributes_t::sessionInfo

8.586 pack_uim_SLQSUIMGetProfileInfo_t Struct Reference

Data Fields

- uint8_t slotNo
- uint32_t profileId

8.586.1 Detailed Description

This structure contains information of the request parameters associated with profile information API.

Parameters

slotNo	• Indicates the recovered slot. • Valid values: – 0x01 - Slot 1 – 0x02 - Slot 2 – 0x03 - Slot 3 – 0x04 - Slot 4 – 0x05 - Slot 5

profileId	• Indicates the profile ID to switch. • Valid values – 0x01 - Profile 1 – 0x02 - Profile 2 – 0x03 - Profile 3 – 0x04 - Profile 4 – 0x05 - Profile 5 – 0x06 - Profile 6 – 0x07 - Profile 7 – 0x08 - Profile 8

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.586.2 Field Documentation

8.586.2.1 uint32_t pack_uim_SLQSUIMGetProfileInfo_t::profileId

8.586.2.2 uint8_t pack_uim_SLQSUIMGetProfileInfo_t::slotNo

8.587 pack_uim_SLQSUIMGetServiceStatus_t Struct Reference

Data Fields

- uim_UIMSessionInformation sessionInfo
- uint32_t capMask

8.587.1 Detailed Description

This structure contains information of the request parameters associated with a Get Service Status API.

Parameters

sessionInfo	• See uim_UIMSessionInformation for more information.
capMask	Bitmask of the capabilities the client retrieves from the card • Bit 0 - FDN status • Bit 1 - Hidden key PIN status (only for USIM cards) • Bit 2 - Index in the EF-DIR (only for UICC cards)

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.587.2 Field Documentation

8.587.2.1 uint32_t pack_uim_SLQSUIMGetServiceStatus_t::capMask

8.587.2.2 uim_UIMSessionInformation pack_uim_SLQSUIMGetServiceStatus_t::sessionInfo

8.588 pack_uim_SLQSUIMGetSIMProfile_t Struct Reference

Data Fields

- uint8_t slotNo

8.588.1 Detailed Description

This structure contains information of the request parameters associated with a Get SIM Profile.

Parameters

slotNo	• Indicates the recovered slot. • Valid values: – 0x01 - Slot 1 – 0x02 - Slot 2 – 0x03 - Slot 3 – 0x04 - Slot 4 – 0x05 - Slot 5

8.588.2 Field Documentation

8.588.2.1 uint8_t pack_uim_SLQSUIMGetSIMProfile_t::slotNo

8.589 pack_uim_SLQSUIMPowerDown_t Struct Reference

Data Fields

- uint8_t slot

8.589.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

slot	• Indicates the slot to be used. – 1 - Slot 1 – 2 - Slot 2

8.589.2 Field Documentation

8.589.2.1 uint8_t pack_uim_SLQSUIMPowerDown_t::slot

8.590 pack_uim_SLQSUIMPowerUp_t Struct Reference

Data Fields

- uint8_t slot
- uint8_t * pIgnoreHotSwapSwitch

8.590.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

slot	• Indicates the slot to be used. – 1 - Slot 1 – 2 - Slot 2
pIgnoreHotSwapSwitch(optional)	• Hot-swap switch status. – 0 - Checks the hot-swap switch status – 1 - Ignores the hot-swap switch status

8.590.2 Field Documentation

8.590.2.1 uint8_t* pack_uim_SLQSUIMPowerUp_t::pIgnoreHotSwapSwitch

8.590.2.2 uint8_t pack_uim_SLQSUIMPowerUp_t::slot

8.591 pack_uim_SLQSUIMReadRecord_t Struct Reference

Data Fields

- uim_UIMSessionInformation sessionInfo
- uim_fileInfo fileIndex
- uim_readRecordInfo readRecord
- uint16_t * pLastRecord
- uint32_t * pIndicationToken

8.591.1 Detailed Description

This structure contains information of the request parameters associated with a Read Record API.

Parameters

sessionInfo	• See uim_UIMSessionInformation for more information.
fileIndex	• See uim_fileInfo for more information.

readRecord	• See uim_readRecordInfo for more information.
pLastRecord	• Last record. This value is used to read multiple records at the same time.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.591.2 Field Documentation

8.591.2.1 uim_FileInfo pack_uim_SLQSUIMReadRecord_t::fileIndex

8.591.2.2 uint32_t* pack_uim_SLQSUIMReadRecord_t::pIndicationToken

8.591.2.3 uint16_t* pack_uim_SLQSUIMReadRecord_t::pLastRecord

8.591.2.4 uim_readRecordInfo pack_uim_SLQSUIMReadRecord_t::readRecord

8.591.2.5 uim_UIMSessionInformation pack_uim_SLQSUIMReadRecord_t::sessionInfo

8.592 pack_uim_SLQSUIMRefreshComplete_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t refreshComplete

8.592.1 Detailed Description

This structure contains information of the request parameters associated with a Refresh Complete event.

Parameters

sessionInfo(-Mandatory)	• See uim_sessionInformation for more information.
refresh-Complete(-Mandatory)	• Indicates whether the refresh was successful. Valid values: – 0 - Refresh was not completed successfully – 1 - Refresh was completed successfully

8.592.2 Field Documentation

8.592.2.1 uint8_t pack_uim_SLQSUIMRefreshComplete_t::refreshComplete

8.592.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshComplete_t::sessionInfo

8.593 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo

8.593.1 Detailed Description

This structure contains information of the request parameters associated with a Refresh Get Last Event.

Parameters

sessionInfo(-Mandatory)	• See uim_sessionInformation for more information.

8.593.2 Field Documentation

8.593.2.1 uim_sessionInformation pack_uim_SLQSUIMRefreshGetLastEvent_t::sessionInfo

8.594 pack_uim_SLQSUIMRefreshOK_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t OKtoRefresh

8.594.1 Detailed Description

This structure contains Parameters of the Session Information

Parameters

sessionInfo	• Session Information • See uim_sessionInformation for more information
OKtoRefresh	• Indicates whether a refresh is OK. Valid values: – 0 - Not OK to refresh – 1 - OK to refresh

8.594.2 Field Documentation

8.594.2.1 uint8_t pack_uim_SLQSUIMRefreshOK_t::OKtoRefresh

8.594.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshOK_t::sessionInfo

8.595 pack_uim_SLQSUIMRefreshRegister_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_registerRefresh regRefresh

8.595.1 Detailed Description

This structure contains information of the request parameters associated with a Refresh Register.

Parameters

sessionInfo	• Session Information params • See uim_sessionInformation for more information
regRefresh	• Register Refresh parameters • See uim_registerRefresh for more information

8.595.2 Field Documentation

8.595.2.1 uim_registerRefresh pack_uim_SLQSUIMRefreshRegister_t::regRefresh

8.595.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshRegister_t::sessionInfo

8.596 pack_uim_SLQSUIMSetServiceStatus_t Struct Reference

Data Fields

- uim_UIMSessionInformation sessionInfo
- uint8_t * pFDNStatus

8.596.1 Detailed Description

This structure contains information of the request parameters associated with a Set Service Status API.

Parameters

sessionInfo	• See uim_UIMSessionInformation for more information.
pFDNStatus	(Optional) • 0: Disables the FDN • 1: Enables the FDN

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.596.2 Field Documentation

8.596.2.1 uint8_t* pack_uim_SLQSUIMSetServiceStatus_t::pFDNStatus

8.596.2.2 uim_UIMSessionInformation pack_uim_SLQSUIMSetServiceStatus_t::sessionInfo

8.597 pack_uim_SLQSUIMSetSIMProfile_t Struct Reference

Data Fields

- uint8_t setSlot
- uint32_t profileId
- uint8_t * pProfileStatus

8.597.1 Detailed Description

This structure contains information of the request parameters associated with a Set SIM profile API.

Parameters

setSlot	• Indicates the recovered slot. • Valid values: – 0x01 - Slot 1 – 0x02 - Slot 2 – 0x03 - Slot 3 – 0x04 - Slot 4 – 0x05 - Slot 5
profileId	• Indicates the profile ID to switch. • Valid values: – 0x01 - Profile 1 – 0x02 - Profile 2 – 0x03 - Profile 3 – 0x04 - Profile 4 – 0x05 - Profile 5 – 0x06 - Profile 6 – 0x07 - Profile 7 – 0x08 - Profile 8
pProfileStatus	(Optional) • Indicates whether a profile must be enabled or disabled. If the TLV is missing, the request is used to enable the profile. • Valid values: – 0 - Disable the profile – 1 - Enable the profile

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.597.2 Field Documentation

8.597.2.1 uint8_t* pack_uim_SLQSUIMSetSIMProfile_t::pProfileStatus

8.597.2.2 uint32_t pack_uim_SLQSUIMSetSIMProfile_t::profileId

8.597.2.3 uint8_t pack_uim_SLQSUIMSetSIMProfile_t::setSlot

8.598 pack_uim_SLQSUIMSwitchSlot_t Struct Reference

Data Fields

- uint8_t bLogicalSlot
- uint32_t ulPhysicalSlot

8.598.1 Detailed Description

This structure contains information of the request parameters associated with a Switch Slot.

Parameters

bLogicalSlot	• Indicates the slot to be used. – 1 - Slot 1 – 2 - Slot 2 – 3 - Slot 3 – 4 - Slot 4 – 5 - Slot 5
ulPhysicalSlot	• 1 - Slot 1 • 2 - Slot 2 • 3 - Slot 3 • 4 - Slot 4 • 5 - Slot 5

8.598.2 Field Documentation

8.598.2.1 uint8_t pack_uim_SLQSUIMSwitchSlot_t::bLogicalSlot

8.598.2.2 uint32_t pack_uim_SLQSUIMSwitchSlot_t::ulPhysicalSlot

8.599 pack_uim_SLQSUIMUpdateProfile_t Struct Reference

Data Fields

- uint8_t slot

- uint32_t profileId
- uim_UIMNickName * pNickName

8.599.1 Detailed Description

This structure contains information of the request parameters associated with a Update Profile API.

Parameters

slot	• any slot number from 1 to 5
profileId	• any profile number from 1 to 8
pNickName	(Optional) • See uim_UIMNickName for more information.

8.599.2 Field Documentation

8.599.2.1 uim_UIMNickName* pack_uim_SLQSUIMUpdateProfile_t::pNickName

8.599.2.2 uint32_t pack_uim_SLQSUIMUpdateProfile_t::profileId

8.599.2.3 uint8_t pack_uim_SLQSUIMUpdateProfile_t::slot

8.600 pack_uim_SLQSUIMWriteRecord_t Struct Reference

Data Fields

- uim_UIMSessionInformation sessionInfo
- uim_fileInfo fileIndex
- uim_writeRecordInfo writeRecord
- uint32_t * pIndicationToken

8.600.1 Detailed Description

This structure contains information of the request parameters associated with a Write Record API.

Parameters

sessionInfo	• See uim_UIMSessionInformation for more information.
fileIndex	• See uim_fileInfo for more information.
writeRecord	• See uim_writeRecordInfo for more information.

pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.600.2 Field Documentation

8.600.2.1 uim_FileInfo pack_uim_SLQSUIMWriteRecord_t::fileIndex

8.600.2.2 uint32_t* pack_uim_SLQSUIMWriteRecord_t::pIndicationToken

8.600.2.3 uim_UIMSessionInformation pack_uim_SLQSUIMWriteRecord_t::sessionInfo

8.600.2.4 uim_writeRecordInfo pack_uim_SLQSUIMWriteRecord_t::writeRecord

8.601 pack_uim_SLQSUIMWriteTransparent_t Struct Reference

Data Fields

- uim_UIMSessionInformation sessionInfo
- uim_fileInfo fileIndex
- uim_writeTransparentInfo writeTransparent
- uint32_t * pIndicationToken

8.601.1 Detailed Description

This structure contains information of the request parameters associated with a Write Transparent API.

Parameters

sessionInfo	• See uim_UIMSessionInformation for more information.
fileId	• See uim_fileInfo for more information.
writeTransparent	• See uim_writeTransparentInfo for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.601.2 Field Documentation

8.601.2.1 uim_fileInfo pack_uim_SLQSUIMWriteTransparent_t::fileIndex

8.601.2.2 uint32_t* pack_uim_SLQSUIMWriteTransparent_t::pIndicationToken

8.601.2.3 uim_UIMSessionInformation pack_uim_SLQSUIMWriteTransparent_t::sessionInfo

8.601.2.4 uim_writeTransparentInfo pack_uim_SLQSUIMWriteTransparent_t::writeTransparent

8.602 pack_uim_UnblockPin_t Struct Reference

Data Fields

- uim_encryptedPIN1 EncryptedPIN1
- uint32_t * pIndicationToken
- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_unblockUIMPIN pinProtection
- uint16_t TlvResult

8.602.1 Detailed Description

This structure contains information of the request parameters associated with a Unblock PIN API.

Parameters

EncryptedPIN1	• See uim_encryptedPIN1 for more information.
sessionInfo	• See uim_sessionInformation for more information.
pinProtection	• See uim_unblockUIMPIN for more information.
pKeyReferenceID(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

8.602.2 Field Documentation

8.602.2.1 uim_encryptedPIN1 pack_uim_UnblockPin_t::EncryptedPIN1

8.602.2.2 uint32_t* pack_uim_UnblockPin_t::pIndicationToken

8.602.2.3 uim_unblockUIMPIN pack_uim_UnblockPin_t::pinProtection

8.602.2.4 uint8_t* pack_uim_UnblockPin_t::pKeyReferenceID

8.602.2.5 uim_sessionInformation pack_uim_UnblockPin_t::sessionInfo

8.602.2.6 uint16_t pack_uim_UnblockPin_t::Tlvresult

8.603 pack_uim_VerifyPin_t Struct Reference

Data Fields

- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_verifyUIMPIN verifyPIN
- uint16_t Tlvresult

8.603.1 Detailed Description

This structure contains information of the request parameters associated with a verify PIN API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
verifyPIN	• See uim_verifyUIMPIN for more information.
pEncryptedPIN1(optional)	• See uim_encryptedPIN1 for more information.
pKeyReferenceID(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.603.2 Field Documentation

8.603.2.1 uim_encryptedPIN1* pack_uim_VerifyPin_t::pEncryptedPIN1

8.603.2.2 uint32_t* pack_uim_VerifyPin_t::pIndicationToken

8.603.2.3 uint8_t* pack_uim_VerifyPin_t::pKeyReferenceID

8.603.2.4 uim_sessionInformation pack_uim_VerifyPin_t::sessionInfo

8.603.2.5 uint16_t pack_uim_VerifyPin_t::Tlvresult

8.603.2.6 uim_verifyUIMPIN pack_uim_VerifyPin_t::verifyPIN

8.604 pack_voice_AnswerUSSD_t Struct Reference

Data Fields

- uint8_t * pInfo

8.604.1 Detailed Description

This structure contains pack anser USSD parameter.

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.604.2 Field Documentation

8.604.2.1 uint8_t* pack_voice_AnswerUSSD_t::pInfo

8.605 pack_voice_OriginateUSSD_t Struct Reference

Data Fields

- uint8_t * pInfo

8.605.1 Detailed Description

This structure contains pack orginate USSD parameter.

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.605.2 Field Documentation

8.605.2.1 uint8_t* pack_voice_OriginateUSSD_t::pInfo

8.606 pack_voice_SLQSOriginateUSSD_t Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.606.1 Detailed Description

This structure contains USS Information

Parameters

ussDCS	• 1 - ASCII coding scheme • 2 - 8-BIT coding scheme • 3 - UCS2
ussLen	• Range 1 to 182
ussData	• Data encoded as per the DCS

8.606.2 Field Documentation

8.606.2.1 uint8_t pack_voice_SLQSOriginateUSSD_t::ussData[182]

8.606.2.2 uint8_t pack_voice_SLQSOriginateUSSD_t::ussDCS

8.606.2.3 uint8_t pack_voice_SLQSOriginateUSSD_t::ussLen

8.607 pack_voice_SLQSVoiceALSSelectLine_t Struct Reference

Data Fields

- uint8_t lineValue

8.607.1 Detailed Description

This structure contains ALS Select Line Information Parameters.

Parameters

lineValue	• ALS Line Value. – 0x00 - ALS_LINE1 - Line 1 (default) – 0x01 - ALS_LINE2 - Line 2

8.607.2 Field Documentation

8.607.2.1 uint8_t pack_voice_SLQSVoiceALSSelectLine_t::lineValue

8.608 pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference

Data Fields

- uint8_t switchOption

8.608.1 Detailed Description

This structure contains ALS Set Line Switching Information Parameters.

Parameters

switchOption	• Switch Option. – 0x00 - VOICE_LINE_SWITCHING_NOT_ALLOWED - Line switching is not allowed – 0x01 - VOICE_LINE_SWITCHING_ALLOWED - Line switching is allowed

8.608.2 Field Documentation

8.608.2.1 uint8_t pack_voice_SLQSVoiceALSSetLineSwitching_t::switchOption

8.609 pack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.609.1 Detailed Description

Contains the parameters passed for pack voice Answer Call.

Parameters

pCallId	• Unique call identifier for the call that must be answered.

8.609.2 Field Documentation

8.609.2.1 uint8_t* pack_voice_SLQSVoiceAnswerCall_t::pCallId

8.610 pack_voice_SLQSVoiceBindSubscription_t Struct Reference

Data Fields

- uint8_t subsType

8.610.1 Detailed Description

This structure contains Bind Subscription Information Parameters.

Parameters

subsType	• Subscription Type. – 0x00 - VOICE_SUBS_TYPE_PRIMARY - Primary – 0x01 - VOICE_SUBS_TYPE_SECONDARY - Secondary

8.610.2 Field Documentation

8.610.2.1 uint8_t pack_voice_SLQSVoiceBindSubscription_t::subsType

8.611 pack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- voice_burstDTMFInfo BurstDTMFInfo
- voice_DTMFLengths * pBurstDTMFLengths

8.611.1 Detailed Description

This structure contains parameters of burst Dual-Tone Multifrequency (DTMF)

Parameters

BurstDTMFInfo	• Burst DTMF Information – See voice_burstDTMFInfo for more information
pBurstDTMF-Lengths	[optional] • DTMF Lengths – See voice_DTMFLengths for more information • NULL pointer - Invalid data.

8.611.2 Field Documentation

8.611.2.1 voice_burstDTMFInfo pack_voice_SLQSVoiceBurstDTMF_t::BurstDTMFInfo

8.611.2.2 voice_DTMFLengths* pack_voice_SLQSVoiceBurstDTMF_t::pBurstDTMFLengths

8.612 pack_voice_SLQSVoiceDialCall_t Struct Reference

Data Fields

- uint8_t callNumber [81]
- uint8_t * pCallType
- uint8_t * pCLIRType
- voice_UUSInfo * pUUSInFo
- voice_CUGInfo * pCUGInfo
- uint8_t * pEmergencyCategory
- voice_calledPartySubAdd * pCallPartySubAdd
- uint8_t * pSvcType

8.612.1 Detailed Description

This structure contains Voice Call Request Parameters

Parameters

callNumber[81]	• Number to be dialed in ASCII string, NULL terminated. • Length Range [1 to 81]
pCall-Type(optional)	• the type of call to be dialed. CALL_TYPE_VOICE is automatically selected if this parameter is not provided. When CALL_TYPE_NON_STD_OTASP is selected, the call is sent as a nonstandard OTASP call regardless of the digit string Call type values are: – 0x00 - CALL_TYPE_VOICE - Voice (automatic selection) – 0x01 - CALL_TYPE_VOICE_FORCED - Avoid modem call classification – 0x08 - CALL_TYPE_NON_STD_OTASP - Nonstandard OTASP* – 0x09 - CALL_TYPE_EMERGENCY - Emergency
pCLIR-Type(optional)	• CLIR type values are: – 0x01 - CLIR_SUPPRESSION - Suppression – 0x02 - CLIR_INVOCATION - Invocation
pUUSIn-Fo(optional)	• Pointer to structure of UUSInfo – See voice_UUSInfo for more information
pCUG-Info(optional)	• Pointer to structure of CUGInfo – See voice_CUGInfo for more information

pEmergency-Category(optional)	• Bit mask of emergency number categories. This is only applicable when the call type is set to Emergency. – Bit 0 - VOICE_EMER_CAT_POLICE_BIT - Police – Bit 1 - VOICE_EMER_CAT_AMBULANCE_BIT - Ambulance – Bit 2 - VOICE_EMER_CAT_FIRE_BRIGADE_BIT- Fire brigade – Bit 3 - VOICE_EMER_CAT_MARINE_GUARD_BIT - Marine guard – Bit 4 - VOICE_EMER_CAT_MOUNTAIN_RESCUE_BIT - Mountain rescue – Bit 5 - VOICE_EMER_CAT_MANUAL_ECALL_BIT - Manual emergency call – Bit 6 - VOICE_EMER_CAT_AUTO_ECALL_BIT - Automatic emergency call – Bit 7 - VOICE_EMER_CAT_SPARE_BIT - Spare bit
pCallPartySub-Add(optional)	• Pointer to structure of calledPartySubAdd – See voice_calledPartySubAdd for more information
pSvc-Type(optional)	• Service Type. – 0x01 - VOICE_DIAL_CALL_SRV_TYPE_AUTOMATIC - Automatic – 0x02 - VOICE_DIAL_CALL_SRV_TYPE_GSM - GSM – 0x03 - VOICE_DIAL_CALL_SRV_TYPE_WCDMA - WCDMA – 0x04 - VOICE_DIAL_CALL_SRV_TYPE_CDMA_AUTOMATIC - CDMA automatic – 0x05 - VOICE_DIAL_CALL_SRV_TYPE_GSM_WCDMA - GSM or WCDMA – 0x06 - VOICE_DIAL_CALL_SRV_TYPE_LTE -LTE

8.612.2 Field Documentation

8.612.2.1 uint8_t pack_voice_SLQSVoiceDialCall_t::callNumber[81]

8.612.2.2 voice_calledPartySubAdd* pack_voice_SLQSVoiceDialCall_t::pCallPartySubAdd

8.612.2.3 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCallType

8.612.2.4 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCLIRType

8.612.2.5 voice_CUGInfo* pack_voice_SLQSVoiceDialCall_t::pCUGInfo

8.612.2.6 uint8_t* pack_voice_SLQSVoiceDialCall_t::pEmergencyCategory

8.612.2.7 uint8_t* pack_voice_SLQSVoiceDialCall_t::pSvcType

8.612.2.8 voice_UUSInfo* pack_voice_SLQSVoiceDialCall_t::pUUSInFo

8.613 pack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.613.1 Detailed Description

This structure contain pack voice end call parameter.

Parameters

pCallId	• Unique call identifier for the call that must be ended

8.613.2 Field Documentation

8.613.2.1 uint8_t* pack_voice_SLQSVoiceEndCall_t::pCallId

8.614 pack_voice_SLQSVoiceGetCallBarring_t Struct Reference

Data Fields

- uint8_t reason
- uint8_t * pSvcClass

8.614.1 Detailed Description

This structure contains Voice Get Call Barring Pack Parameters

Parameters

reason	• Call Barring Reason • Values: – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME - Outgoing external to home – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available

8.614.2 Field Documentation

8.614.2.1 uint8_t* pack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.614.2.2 uint8_t pack_voice_SLQSVoiceGetCallBarring_t::reason

8.615 pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t * pSvcClass

8.615.1 Detailed Description

This structure contains Voice Get Call Forwarding Status Pack Parameters

Parameters

Reason	• Call Forwarding Reason • Values: – 0x01 - QMI_VOICE_REASON_FWDREASON_UNCONDITIONAL - Unconditional call forwarding – 0x02 - QMI_VOICE_REASON_FWDREASON_MOBILEBUSY - Forward when the mobile is busy – 0x03 - QMI_VOICE_REASON_FWDREASON_NOREPLY - Forward when there is no reply – 0x04 - QMI_VOICE_REASON_FWDREASON_UNREACHABLE - Forward when the call is unreachable – 0x05 - QMI_VOICE_REASON_FWDREASON_ALLFORWARDING - All forwarding – 0x06 - QMI_VOICE_REASON_FWDREASON_ALLCONDITIONAL - All conditional forwarding
pSvcClass(optional)	• Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.

8.615.2 Field Documentation

8.615.2.1 uint8_t* pack_voice_SLQSVoiceGetCallForwardingStatus_t::pSvcClass

8.615.2.2 uint8_t pack_voice_SLQSVoiceGetCallForwardingStatus_t::Reason

8.616 pack_voice_SLQSVoiceGetCallInfo_t Struct Reference

Data Fields

- uint8_t callID

8.616.1 Detailed Description

This structure contains information of the request parameters associated with a call.

Parameters

callID	• Call identifier for the call queried for information.

8.616.2 Field Documentation

8.616.2.1 uint8_t pack_voice_SLQSVoiceGetCallInfo_t::callID

8.617 pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference

Data Fields

- uint8_t * pSvcClass

8.617.1 Detailed Description

This structure contains Voice Get Call Waiting Response Parameters

Parameters

pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available

8.617.2 Field Documentation

8.617.2.1 uint8_t* pack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.618 pack_voice_SLQSVoiceGetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnswer
- uint8_t * pAirTimer
- uint8_t * pRoamTimer
- uint8_t * pTTYMode
- uint8_t * pPrefVoiceSO
- uint8_t * pAMRStatus
- uint8_t * pPrefVoicePrivacy
- uint8_t * pNamID
- uint8_t * pVoiceDomainPref

8.618.1 Detailed Description

This structure contains Voice Get Configuration Request Parameters

Parameters

pAuto-Answer(optional)	• Indicator to retrieve the Auto Answer Information. – 0x01 - Include auto answer information
pAir-Timer(optional)	• Indicator to retrieve the Air Timer Information. – 0x01 - Include air calls timer count information • Currently Not Supported.
pRoam-Timer(optional)	• Indicator to retrieve the Roam Timer Information. – 0x01 - Include roam calls timer information • Currently Not Supported.
pTTY-Mode(optional)	• Indicator to retrieve the TTY Mode Information. – 0x01 - Include TTY configuration status information
pPrefVoiceS-O(optional)	• Indicator to retrieve the Preferred Voice SO Information. – 0x01 - Include preferred voice configuration status information • Currently Not Supported.
pAMR-Status(optional)	• Indicator to retrieve the AMR Status Information. – 0x01 - Include AMR status information
pPrefVoice-Privacy(optional)	• Indicator to retrieve the Preferred Voice Privacy Information. – 0x01 - Include preferred voice privacy status information
pNamI-D(optional)	• Index of the Number Assignment Module Index (CDMA subscription) to be configured • Range: 0 to 3. • Some modems support only 1 or 2 NAMs. • The NAM Index is valid only when the request contains at least one of Air Timer, Roam Timer, and Preferred Voice SO. • If no nam_id value is specified in the request,the default value is 0.
pVoiceDomain-Pref(optional)	• Indicator to retrieve the Preferred Voice Domain Information. – 0x01 - Include voice domain preference information

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.618.2 Field Documentation

8.618.2.1 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAirTimer

8.618.2.2 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAMRStatus

8.618.2.3 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAutoAnswer

8.618.2.4 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pNamID

8.618.2.5 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoicePrivacy

8.618.2.6 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoiceSO

8.618.2.7 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pRoamTimer

8.618.2.8 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pTTYMode

8.618.2.9 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pVoiceDomainPref

8.619 pack_voice_SLQSVoiceIndicationRegister_t Struct Reference

Data Fields

- uint8_t * pRegDTMFEvents
- uint8_t * pRegVoicePrivacyEvents
- uint8_t * pSuppsNotifEvents

8.619.1 Detailed Description

This structure contains parameters of Indication Register Information

Parameters

pRegDTMF-Events(optional)	• Registration Indication For DTMF Events. • When this registration is enabled, the device learns of DTMF events via the QMI_VOICE_DTMF_IND indication. – 0x00 - Disable – 0x01 - Enable
pRegVoice-Privacy-Events(optional)	• Registration Indication For Voice Privacy Events. • When this registration is enabled, the device learns of DTMF events via the QMI_VOICE_PRIVACY_IND indication. – 0x00 - Disable – 0x01 - Enable
pSuppsNotif-Events(optional)	• Registration Indication For Supplementary Service Notification Events. • When this registration is enabled, the device learns of DTMF events via the QMI_VOICE_SUPS_NOTIFICATION_IND indication. – 0x00 - Disable – 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.619.2 Field Documentation

8.619.2.1 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegDTMFEvents

8.619.2.2 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegVoicePrivacyEvents

8.619.2.3 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pSuppsNotifEvents

8.620 pack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint8_t SUPSType
- uint8_t * pCallID

8.620.1 Detailed Description

This structure contains Manage Calls Information.

Parameters

SUPSType	• Supplementary service type during the call. – 0x01 - SUPS_TYPE_RELEASE_HELD_OR_WAITING * Release is held or waiting – 0x02 - SUPS_TYPE_RELEASE_ACTIVE_ACCEPT_HELD_OR_WAITING * Release is active and accepting held or waiting – 0x03 - SUPS_TYPE_HOLD_ACTIVE_ACCEPT_WAITING_OR_HELD * Hold is active and accepting waiting or held – 0x04 - SUPS_TYPE_HOLD_ALL_EXCEPT_SPECIFIED_CALL * Hold all calls except a specified one – 0x05 - SUPS_TYPE_MAKE_CONFERENCE_CALL * Make a conference call – 0x06 - SUPS_TYPE_EXPLICIT_CALL_TRANSFER * Explicit call transfer – 0x07 - SUPS_TYPE_CCBS_ACTIVATION * Activate completion of calls to busy subscriber – 0x08 - SUPS_TYPE_END_ALL_CALLS * End all calls – 0x09 - SUPS_TYPE_RELEASE_SPECIFIED_CALL * Release a specified call
pCallID[Optional]	• Applicable only for SUPSType 0x04, 0x07, and 0x09 • NULL pointer - Invalid data.

8.620.2 Field Documentation

8.620.2.1 uint8_t* pack_voice_SLQSVoiceManageCalls_t::pCallID

8.620.2.2 uint8_t pack_voice_SLQSVoiceManageCalls_t::SUPSType

8.621 pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference

Data Fields

- struct voice_USSInfo USSInformation

8.621.1 Detailed Description

This structure contains Orig USSD No Wait Information Parameters.

Parameters

USSInformation	• See voice_USSInfo for more information.

8.621.2 Field Documentation

8.621.2.1 struct voice_USSInfo pack_voice_SLQSVoiceOrigUSSDNoWait_t::USSInformation

8.622 pack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t * pFlashPayLd
- uint8_t * pFlashType

8.622.1 Detailed Description

This structure contains the flash information associated with a call.

Parameters

pCallID	• Unique call identifier associated with the current call.
pFlashPayLd(optional)	• Payload in ASCII to be sent in Flash. • Variable Length, NULL terminated.
pFlashType(optional)	• Flash type. – 0 - Simple Flash (default) – 1 - Activate answer hold – 2 - Deactivate answer hold

8.622.2 Field Documentation

8.622.2.1 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pCallID

8.622.2.2 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashPayLd

8.622.2.3 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashType

8.623 pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t oldPasswd [4]
- uint8_t newPasswd [4]
- uint8_t newPasswdAgain [4]

8.623.1 Detailed Description

This structure contains Voice Set Call Barring Password Pack Parameters

Parameters

Reason	• Call Barring Reason • Values: – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME - Outgoing external to home – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
oldPasswd[BARRING_PASSWORD_LENGTH]	• Old password. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.
newPasswd[BARRING_PASSWORD_LENGTH]	• New password. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.
newPasswdAgain[BARRING_PASSWORD_LENGTH]	• New password Again. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.

8.623.2 Field Documentation

8.623.2.1 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswd[4]

8.623.2.2 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswdAgain[4]

8.623.2.3 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::oldPasswd[4]

8.623.2.4 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::Reason

8.624 pack_voice_SLQSVoiceSetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnswer
- voice_airTimer * pAirTimerConfig
- voice_roamTimer * pRoamTimerConfig
- uint8_t * pTTYMode
- voice_prefVoiceSO * pPrefVoiceSO
- uint8_t * pPrefVoiceDomain

8.624.1 Detailed Description

This structure contains information about the Set Configuration Request Parameters.

Parameters

pAutoAnswer	• Value specified is written to NV_AUTO_ANSWER_I. (optional) • Values: – 0x00 - Disable – 0x01 - Enable
pAirTimerConfig	• Value specified is written to NV_AIR_CNT_I. (optional) • See voice_airTimer for more information
pRoamTimer-Config	• Value specified is written to NV_ROAM_CNT_I. (optional) • See voice_roamTimer for more information
pTTYMode	• Value specified is written to NV_TTY_I. (optional) • Values: – 0x00 - TTY_MODE_FULL - Full – 0x01 - TTY_MODE_VCO - Voice carry over – 0x02 - TTY_MODE_HCO - Hearing carry over – 0x03 - TTY_MODE_OFF - Off
pPrefVoiceSO	• Value specified is written to NV_PREF_VOICE_SO_I. (optional) • See voice_prefVoiceSO for more information

pPrefVoice-Domain	• Preferred Voice-Domain. (optional) • Values: – 0x00 - VOICE_DOMAIN_PREF_CS_ONLY - Circuit-switched (CS) only – 0x01 - VOICE_DOMAIN_PREF_PS_ONLY - Packet-switched (PS) only – 0x02 - VOICE_DOMAIN_PREF_CS_PREF - CS is preferred, PS is secondary – 0x03 - VOICE_DOMAIN_PREF_PS_PREF - PS is preferred, CS is secondary

Note

One of the optional parameters must be present in the request.

8.624.2 Field Documentation

8.624.2.1 voice_airTimer* pack_voice_SLQSVoiceSetConfig_t::pAirTimerConfig

8.624.2.2 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pAutoAnswer

8.624.2.3 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceDomain

8.624.2.4 voice_prefVoiceSO* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSO

8.624.2.5 voice_roamTimer* pack_voice_SLQSVoiceSetConfig_t::pRoamTimerConfig

8.624.2.6 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pTTYMode

8.625 pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference

Data Fields

- uint8_t privacyPref

8.625.1 Detailed Description

This structure contains the preferred voice privacy values.

Parameters

privacyPref	• Voice Privacy Preference – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy

8.625.2 Field Documentation

8.625.2.1 uint8_t pack_voice_SLQSVoiceSetPreferredPrivacy_t::privacyPref

8.626 pack_voice_SLQSVoiceSetSUPSService_t Struct Reference

Data Fields

- uint8_t voiceSvc
- uint8_t reason
- uint8_t * pServiceClass
- uint8_t * pCallBarringPasswd
- uint8_t * pCallForwardingNumber
- uint8_t * pTimerVal
- voice_callFwdTypeAndPlan * pCallFwdTypeAndPlan

8.626.1 Detailed Description

This structure contains Supplementary Service request parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

voiceSvc	• Manages all call-independent supplementary services, such as activation, deactivation, registration, and erasure (mandatory) – 0x01 - VOICE_SERVICE_ACTIVATE – 0x02 - VOICE_SERVICE_DEACTIVATE – 0x03 - VOICE_SERVICE_REGISTER – 0x04 - VOICE_SERVICE_ERASE

reason	• supplementary service reason values (mandatory) – 0x01 - QMI_VOICE_REASON_FWD_UNCONDITIONAL Unconditional call forwarding – 0x02 - QMI_VOICE_REASON_FWD_MOBILEBUSY Forward when the mobile is busy – 0x03 - QMI_VOICE_REASON_FWD_NOREPLY Forward when there is no reply – 0x04 - QMI_VOICE_REASON_FWD_UNREACHABLE Forward when the call is unreachable – 0x05 - QMI_VOICE_REASON_FWD_ALLFORWARDING All forwarding – 0x06 - QMI_VOICE_REASON_FWD_ALLCONDITIONAL All conditional forwarding – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING All outgoing calls are barred – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT Outgoing internal calls are barred – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME Outgoing calls external to home are barred – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING All incoming calls are barred – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING Roaming incoming calls are barred – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING All incoming calls are barred – 0x0F - QMI_VOICE_REASON_CALLWAITING Call waiting
pServiceClass	• Service class is a combination (sum) of information class constants (optional) – See liteServiceClassInformation for more information
pCallBarring-Passwd	• Password is required if call barring is provisioned using a password. Password consists of 4 ASCII digits. Range: 0000 to 9999 (optional)
pCallForwarding-Number	• Call forwarding number to be registered with the network. This has to be included in the request only when the service is set to VOICE_SERVICE_REGISTER. NULL terminated ASCII string. (optional)
pTimerVal	• Call forwarding no reply timer value in seconds. This has to be included in the request only when the service is set to VOICE_SERVICE_REGISTER and the reason is QMI_VOICE_REASON_FWD_NOREPLY. (optional) – Range: 5 to 30 in steps of 5

pCallFwdType-AndPlan	• Information about call forwarding type and plan. This parameter is ignored when the Call Forwarding Number is not included (optional) – See voice_callFwdTypeAndPlan for more information

8.626.2 Field Documentation

8.626.2.1 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallBarringPasswd

8.626.2.2 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallForwardingNumber

8.626.2.3 voice_callFwdTypeAndPlan* pack_voice_SLQSVoiceSetSUPSService_t::pCallFwdTypeAndPlan

8.626.2.4 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pServiceClass

8.626.2.5 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pTimerVal

8.626.2.6 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::reason

8.626.2.7 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::voiceSvc

8.627 pack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t DTMFdigit

8.627.1 Detailed Description

This structure contains parameters of continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Start continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device
DTMFdigit	• DTMF digit in ASCII.

8.627.2 Field Documentation

8.627.2.1 uint8_t pack_voice_SLQSVoiceStartContDTMF_t::DTMFdigit

8.627.2.2 uint8_t* pack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.628 pack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID

8.628.1 Detailed Description

This structure contains parameters of stop continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Stop continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device

8.628.2 Field Documentation

8.628.2.1 uint8_t pack_voice_SLQSVoiceStopContDTMF_t::callID

8.629 pack_wda_GetDataFormat_t Struct Reference

Data Fields

- uint32_t ep_type
- uint32_t iface_id

8.629.1 Detailed Description

Structure to get configured data format

Parameters

ep_type	• Peripheral endpoint type. – Values: * DATA_EP_TYPE_RESERVED (0x00) - Reserved * DATA_EP_TYPE_HSIC (0x01) - High-speed inter-chip interface * DATA_EP_TYPE_HSUSB (0x02) - High-speed universal serial bus * DATA_EP_TYPE_PCIE (0x03) - Peripheral component interconnect express * DATA_EP_TYPE_EMBEDDED (0x04) - Embedded * DATA_EP_TYPE_BAM_DMUX (0x05) - BAM demux * All other values are reserved and are ignored.

iface_id	• Peripheral interface number.

8.629.2 Field Documentation

8.629.2.1 uint32_t pack_wda_GetDataFormat_t::ep_type

8.629.2.2 uint32_t pack_wda_GetDataFormat_t::iface_id

8.630 pack_wda_GetQmapSettings_t Struct Reference

Data Fields

- uint32_t * pEp_type
- uint32_t * pIface_id

8.630.1 Detailed Description

Structure to get qmap settings format

Parameters

pEp_type	• Peripheral endpoint type. – Values: * DATA_EP_TYPE_RESERVED (0x00) - Reserved * DATA_EP_TYPE_HSIC (0x01) - High-speed inter-chip interface * DATA_EP_TYPE_HSUSB (0x02) - High-speed universal serial bus * DATA_EP_TYPE_PCIE (0x03) - Peripheral component interconnect express * DATA_EP_TYPE_EMBEDDED (0x04) - Embedded * DATA_EP_TYPE_BAM_DMUX (0x05) - BAM demux * All other values are reserved and are ignored.
pIface_id	• Peripheral interface number.

8.630.2 Field Documentation

8.630.2.1 uint32_t* pack_wda_GetQmapSettings_t::pEp_type

8.630.2.2 uint32_t* pack_wda_GetQmapSettings_t::pIface_id

8.631 pack_wda_SetDataFormat_t Struct Reference

Data Fields

- uint8_t * pQos_format
- uint32_t * pQlink_prot
- uint32_t * pUl_data_aggregation_protocol

- uint32_t * pDl_data_aggregation_protocol
- uint32_t * pNdp_signature
- uint32_t * pDl_data_aggregation_max_datagrams
- uint32_t * pDl_data_aggregation_max_size
- uint32_t * pEp_type
- uint32_t * pIface_id
- uint32_t * pQos_header_format
- uint32_t * pDl_minimum_padding
- uint8_t * pFlow_control
- uint32_t * pUl_data_aggregation_max_datagrams
- uint32_t * pUl_data_aggregation_max_size
- uint8_t * pTcp_coalescing
- uint8_t * pUdp_coalescing

8.631.1 Detailed Description

Structure to set data format for RMNET QMAP

Parameters

pQos_format	• Configured QOS data format. – 0 - QOS flow header is not present (default) – 1 - QOS flow header is present
pQlink_prot	• 0x01 - 802.3 Ethernet mode (Not supported) – 0x02 - IP mode (Default)
pUl_data_aggregation_protocol	• 0x00 - UL data aggregation is disabled (default) • 0x01 - UL TLP is enabled • 0x02 - UL QC_NCM is enabled • 0x03 - UL MBIM is enabled • 0x04 - UL RNDIS is enabled • 0x05 - UL QMAP is enabled • 0x06 - UL QMAP V2 is enabled • 0x07 - UL QMAP V3 is enabled • 0x08 - UL QMAP V4 is enabled • 0x09 - UL QMAP V5 is enabled

pDL_data_aggregation_protocol	• 0x00 - DL data aggregation is disabled(Default) – 0x05 - DL QMAP is enabled – 0x00 - DL data aggregation is disabled (default) – 0x01 - DL TLP is enabled – 0x02 - DL QC_NCM is enabled – 0x03 - DL MBIM is enabled – 0x04 - DL RNDIS is enabled – 0x05 - DL QMAP is enabled – 0x06 - DL QMAP V2 is enabled – 0x07 - DL QMAP V3 is enabled – 0x08 - DL QMAP V4 is enabled – 0x09 - DL QMAP V5 is enabled
pNdp_signature	• Uses the default value based on the data aggregation protocol.
pDl_data_aggregation_max_datagrams	• Maximum number of datagrams in a single aggregated packet on downlink. The value applies to all downlink data aggregation protocols when downlink data aggregation is enabled. Zero means no limit.
pDl_data_aggregation_max_size	• Maximum size in bytes of a single aggregated packet allowed on downlink. The value applies to all downlink data aggregation protocols when downlink data aggregation is enabled.
pEp_type	• Peripheral endpoint type. – 0x00 - Reserved – 0x01 - High-speed inter-chip interface – 0x02 - High-speed universal serial bus – 0x03 - Peripheral component interconnect express – 0x04 - Embedded – 0x05 - BAM demux All other values are reserved and are ignored.
pIface_id	• Peripheral interface number (0x08 for Sierra modems)
pQos_header_format	• QOS header format to use on the uplink, on all the protocols, if QOS is enabled. – 0x00 - Reserved – 0x01 - QOS 6 byte default header – 0x02 - QOS 8 byte header
pDl_minimum_padding	• Indicates the minimum padding bytes added in between aggregated downlink QMAP packets.
Flow_Control	• Indicates whether flow control will be done by the TE. – 0 - Flow control will not be done by the TE (Default) – 1 - Flow control will be done by the TE (Default)

pUl_data_aggregation_max_size	• Maximum size in bytes of a single aggregated packet allowed on uplink. The value applies only to QMAP uplink data aggregation protocol when it is enabled.
pTcp_coalescing	• Specifies whether TCP coalescing is enabled or disabled on the modem. – FALSE - Disable TCP coalecsing (default). – TRUE - Enable TCP coalescing
pUdp_coalescing	• Specifies whether UDP coalescing is enabled or disabled on the modem. – FALSE - Disable UDP coalecsing (default). – TRUE - Enable UDP coalescing

8.631.2 Field Documentation

8.631.2.1 uint32_t* pack_wda_SetDataFormat_t::pDl_data_aggregation_max_datagrams

8.631.2.2 uint32_t* pack_wda_SetDataFormat_t::pDl_data_aggregation_max_size

8.631.2.3 uint32_t* pack_wda_SetDataFormat_t::pDl_data_aggregation_protocol

8.631.2.4 uint32_t* pack_wda_SetDataFormat_t::pDl_minimum_padding

8.631.2.5 uint32_t* pack_wda_SetDataFormat_t::pEp_type

8.631.2.6 uint8_t* pack_wda_SetDataFormat_t::pFlow_control

8.631.2.7 uint32_t* pack_wda_SetDataFormat_t::pIface_id

8.631.2.8 uint32_t* pack_wda_SetDataFormat_t::pNdp_signature

8.631.2.9 uint32_t* pack_wda_SetDataFormat_t::pQlink_prot

8.631.2.10 uint8_t* pack_wda_SetDataFormat_t::pQos_format

8.631.2.11 uint32_t* pack_wda_SetDataFormat_t::pQos_header_format

8.631.2.12 uint8_t* pack_wda_SetDataFormat_t::pTcp_coalescing

8.631.2.13 uint8_t* pack_wda_SetDataFormat_t::pUdp_coalescing

8.631.2.14 uint32_t* pack_wda_SetDataFormat_t::pUl_data_aggregation_max_datagrams

8.631.2.15 uint32_t* pack_wda_SetDataFormat_t::pUl_data_aggregation_max_size

8.631.2.16 uint32_t* pack_wda_SetDataFormat_t::pUl_data_aggregation_protocol

8.632 pack_wda_SetQmapSettings_t Struct Reference

Data Fields

- uint32_t * pEp_type
- uint32_t * pIface_id
- uint8_t * pQmap_dfc

8.632.1 Detailed Description

Structure to set WDA Set QMAP Settings

Parameters

pEp_type	• Peripheral endpoint type. – Values: * DATA_EP_TYPE_RESERVED (0x00) - Reserved * DATA_EP_TYPE_HSIC (0x01) - High-speed inter-chip interface * DATA_EP_TYPE_HSUSB (0x02) - High-speed universal serial bus * DATA_EP_TYPE_PCIE (0x03) - Peripheral component interconnect express * DATA_EP_TYPE_EMBEDDED (0x04) - Embedded * DATA_EP_TYPE_BAM_DMUX (0x05) - BAM demux All other values are reserved and are ignored.
pIface_id	• Peripheral interface number.
pQmap_dfc	• Configures the data flow control via QMAP. – Values: * 0 - Disables QMAP Data flow control * 1 - Enables QMAP Data flow control

8.632.2 Field Documentation

8.632.2.1 uint32_t* pack_wda_SetQmapSettings_t::pEp_type

8.632.2.2 uint32_t* pack_wda_SetQmapSettings_t::pIface_id

8.632.2.3 uint8_t* pack_wda_SetQmapSettings_t::pQmap_dfc

8.633 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference

Data Fields

- uint8_t * pEnableNotification

8.633.1 Detailed Description

WDS SWI DHCPv4 Client Lease Change Structure

Parameters

pEnable-Notification	• Enable Notification or not

8.633.2 Field Documentation

8.633.2.1 uint8_t* pack_wds_DHCPv4ClientLeaseChange_t::pEnableNotification

8.634 pack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t profiletype

8.634.1 Detailed Description

This structure contains pack get default profile information.

Parameters

profiletype	profile type • 0 - WDS_PROFILE_TYPE_3GPP (0x00) - 3GPP • 1 - WDS_PROFILE_TYPE_3GPP (0x01) - 3GPP2 • 2 - WDS_PROFILE_TYPE_EPC (0x02) - EPC

8.634.2 Field Documentation

8.634.2.1 uint32_t pack_wds_GetDefaultProfile_t::profiletype

8.635 pack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family

8.635.1 Detailed Description

This structure contains pack Get Default Profile Number information.

Parameters

type	profile type • 0 - 3GPP • 1 - 3GPP2
family	profile family • 0 - Embedded • 1 - Tethered

8.635.2 Field Documentation

8.635.2.1 uint8_t pack_wds_GetDefaultProfileNum_t::family

8.635.2.2 uint8_t pack_wds_GetDefaultProfileNum_t::type

8.636 pack_wds_GetDefaultProfileV2_t Struct Reference

Data Fields

- uint32_t profiletype

8.636.1 Detailed Description

This structure contains pack get default profile information.

Parameters

profiletype	profile type • 0 - WDS_PROFILE_TYPE_3GPP (0x00) - 3GPP • 1 - WDS_PROFILE_TYPE_3GPP (0x01) - 3GPP2 • 2 - WDS_PROFILE_TYPE_EPC (0x02) - EPC

8.636.2 Field Documentation

8.636.2.1 uint32_t pack_wds_GetDefaultProfileV2_t::profiletype

8.637 pack_wds_GetDormancyState_t Struct Reference

8.637.1 Detailed Description

This structure contains pack get dormancy state information.

Parameters

NULL	• this is a dummy structure

8.638 pack_wds_GetLastMobileIPError_t Struct Reference

8.638.1 Detailed Description

This structure contains pack get last mobile IP error information.

Parameters

NULL	• this is a dummy structure

8.639 pack_wds_GetMobileIP_t Struct Reference

8.639.1 Detailed Description

This structure contains pack get mobile IP information.

Parameters

NULL	• this is a dummy structure

8.640 pack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t index

8.640.1 Detailed Description

This structure contains pack get mobile IP profile information.

Parameters

index	• Mobile IP profile ID

8.640.2 Field Documentation

8.640.2.1 uint8_t pack_wds_GetMobileIPProfile_t::index

8.641 pack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * pStatMask

8.641.1 Detailed Description

This structure contains pack get packet statistics information.

Parameters

pStatMask	• Packet Statistics Mask – 0x00000001 - Tx packets OK – 0x00000002 - Rx packets OK – 0x00000004 - Tx packet errors – 0x00000008 - Rx packet errors – 0x00000010 - Tx overflows – 0x00000020 - Rx overflows – 0x00000040 - Tx bytes OK – 0x00000080 - Rx bytes OK

8.641.2 Field Documentation

8.641.2.1 uint32_t* pack_wds_GetPacketStatistics_t::pStatMask

8.642 pack_wds_GetPacketStatus_t Struct Reference

Data Fields

- uint32_t statmask

8.642.1 Detailed Description

This structure contains pack get packet status information.

Parameters

statmask	packet statistics mask • 0x00000001 - Tx packets OK • 0x00000002 - Rx packets OK • 0x00000004 - Tx packet errors • 0x00000008 - Rx packet errors • 0x00000010 - Tx overflows • 0x00000020 - Rx overflows • 0x00000040 - Tx bytes OK • 0x00000080 - Rx bytes OK • 0x00000100 - Tx packets dropped • 0x00000200 - Rx packets dropped

8.642.2 Field Documentation

8.642.2.1 uint32_t pack_wds_GetPacketStatus_t::statmask

8.643 pack_wds_GetSessionDuration_t Struct Reference

8.643.1 Detailed Description

This structure contains pack get session duration information.

Parameters

NULL	• this is a dummy structure

8.644 pack_wds_RMSetTransferStatistics_t Struct Reference

Data Fields

- rmTrasnferStaticsReq RmTrasnferStaticsReq

8.644.1 Detailed Description

This structure contains pack fetch current data system transfer statistics information.

Parameters

RmTrasnfer-StaticsReq[IN]	• See rmTrasnferStaticsReq for more information

8.644.2 Field Documentation

8.644.2.1 rmTrasnferStaticsReq pack_wds_RMSetTransferStatistics_t::RmTrasnferStaticsReq

8.645 pack_wds_SetAutoconnect_t Struct Reference

Data Fields

- uint8_t acsetting
- uint8_t acroamsetting

8.645.1 Detailed Description

auto connect data session parameters.

Parameters

acsetting	• Current autoconnect setting: – 0x00 - Autoconnect disabled – 0x01 - Autoconnect enabled – 0x02 - Autoconnect paused (resume on powercycle)

acroamsetting	• Current autoconnect roaming status – 0x00 - Autoconnect always allowed – 0x01 - Autoconnect while in home service area only

8.645.2 Field Documentation

8.645.2.1 uint8_t pack_wds_SetAutoconnect_t::acroamsetting

8.645.2.2 uint8_t pack_wds_SetAutoconnect_t::acsetting

8.646 pack_wds_SetDefaultProfile_t Struct Reference

Data Fields

- uint32_t profileType
- uint32_t pdpType
- uint32_t ipAddress
- uint32_t primaryDNS
- uint32_t secondaryDNS
- uint32_t authentication
- uint8_t * pName
- uint8_t * pUsername
- uint8_t * pApnname
- uint8_t * pPassword

8.646.1 Detailed Description

Writes the default profile settings to the device. The default profile is used to establish an autoconnect data session.

Parameters

profileType	• Type of profile – 0 - 3GPP – 1 - 3GPP2
pdpType	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile (optional) – 0 - PDP-IP (IPv4) – 1 - PDP-PPP – 2 - PDP-IPv6 – 3 - PDP-IPv4v6
ipAddress	• Preferred IPv4 addr to be assigned to device (optional)
primaryDNS	• Primary DNS Ipv4 address preference (optional)

secondaryDNS	• Secondary DNS Ipv4 address preference (optional)
authentication	• Bitmap that indicates authentication algorithm preference (optional) – 0x00000001 - PAP preference * 0 - Never performed * 1 - May be performed – 0x00000002 - CHAP preference * 0 - Never performed * 1 - May be performed – All other bits are reserved and must be set to 0 – If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.
pName	• profile Name (optional)
pUsername	• Username used during network authentication (optional)
pAPNName	• Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network (optional) • If value is NULL or omitted, then subscription default value will be requested.
pPassword	• Password used during network authentication (optional)

8.646.2 Field Documentation

8.646.2.1 uint32_t pack_wds_SetDefaultProfile_t::authentication

8.646.2.2 uint32_t pack_wds_SetDefaultProfile_t::ipAddress

8.646.2.3 uint8_t* pack_wds_SetDefaultProfile_t::pApnname

8.646.2.4 uint32_t pack_wds_SetDefaultProfile_t::pdpType

8.646.2.5 uint8_t* pack_wds_SetDefaultProfile_t::pName

8.646.2.6 uint8_t* pack_wds_SetDefaultProfile_t::pPassword

8.646.2.7 uint32_t pack_wds_SetDefaultProfile_t::primaryDNS

8.646.2.8 uint32_t pack_wds_SetDefaultProfile_t::profileType

8.646.2.9 uint8_t* pack_wds_SetDefaultProfile_t::pUsername

8.646.2.10 uint32_t pack_wds_SetDefaultProfile_t::secondaryDNS

8.647 pack_wds_SetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family
- uint8_t index

8.647.1 Detailed Description

This structure to hold Set default profile number

Parameters

type	Identifies the technology type of the profile • 0 - 3GPP • 1 - 3GPP2
family	Identifies the family of profile • 0 - Embedded • 1 - Tethered
index	Profile number to be set as default profile.

8.647.2 Field Documentation

8.647.2.1 uint8_t pack_wds_SetDefaultProfileNum_t::family

8.647.2.2 uint8_t pack_wds_SetDefaultProfileNum_t::index

8.647.2.3 uint8_t pack_wds_SetDefaultProfileNum_t::type

8.648 pack_wds_SetMobileIP_t Struct Reference

Data Fields

- uint32_t mode

8.648.1 Detailed Description

This structure contains set mobile IP pack information.

Parameters

mode	• Mobile IP setting – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only

8.648.2 Field Documentation

8.648.2.1 uint32_t pack_wds_SetMobileIP_t::mode

8.649 pack_wds_SetMobileIPParameters_t Struct Reference

Data Fields

- char * pSPC
- uint32_t * pMode
- uint8_t * pRetryLimit
- uint8_t * pRetryInterval
- uint8_t * pReRegPeriod
- uint8_t * pReRegTraffic
- uint8_t * pHAAuthenticator
- uint8_t * pHA2002bis

8.649.1 Detailed Description

Mobile IP parameters information.

Parameters

pSPC	• NULL-terminated string representing six digit service programming code.
pMode	• Mode to be set (optional) – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only
pRetryLimit	• Registration retry attempt limit (optional)
pRetryInterval	• Registration retry attempt interval used to determine the time between registration attempts (optional)
pReRegPeriod	• Period (in minutes) to attempt re-registration before current registration expires (optional)
pReRegTraffic	• Re-registration only if traffic since last attempt (optional) – Zero - Disabled – NonZero - Enabled
pHA-Authenticator	• MH-HA authenticator calculator (optional) – Zero - Disabled – NonZero - Enabled

pHA2002bis	• MH-HA RFC 2002bis authentication instead of RFC2002 (optional) – Zero - Disabled – NonZero - Enabled

8.649.2 Field Documentation

8.649.2.1 uint8_t* pack_wds_SetMobileIPParameters_t::pHA2002bis

8.649.2.2 uint8_t* pack_wds_SetMobileIPParameters_t::pHAAuthenticator

8.649.2.3 uint32_t* pack_wds_SetMobileIPParameters_t::pMode

8.649.2.4 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegPeriod

8.649.2.5 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegTraffic

8.649.2.6 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryInterval

8.649.2.7 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryLimit

8.649.2.8 char* pack_wds_SetMobileIPParameters_t::pSPC

8.650 pack_wds_SetMobileIPProfile_t Struct Reference

Data Fields

- int8_t spc [10]
- uint8_t index
- uint8_t * pEnabled
- uint32_t * pAddress
- uint32_t * pPrimaryHA
- uint32_t * pSecondaryHA
- uint8_t * pRevTunneling
- int8_t * pNAI
- uint32_t * pHASPI
- uint32_t * pAAASPI
- int8_t * pMNHA
- int8_t * pMNAAA

8.650.1 Detailed Description

Sets the mobile IP parameters.

Parameters

spc	• Six digit service programming code string
index	• Index of the profile to modify

pEnabled	• (Optional) Enable profile? 0 - Disabled Nonzero - Enabled
pAddress	• (Optional) Home IPv4 address
pPrimaryHA	• (Optional) Primary home agent IPv4 address
pSecondaryHA	• (Optional) Secondary home agent IPv4 address
pRevTunneling	• (Optional) Enable reverse tunneling? 0 - Disabled Nonzero - Enabled
pNAI	• (Optional) Network access identifier string
pHASPI	• (Optional) Home agent security parameter index
pAAASPI	• (Optional) AAA server security parameter index
pMNHA	• (Optional) MN-HA key string
pMNAAA	• (Optional) MN-AAA key string

8.650.2 Field Documentation

8.650.2.1 uint8_t pack_wds_SetMobileIPProfile_t::index

8.650.2.2 uint32_t* pack_wds_SetMobileIPProfile_t::pAAASPI

8.650.2.3 uint32_t* pack_wds_SetMobileIPProfile_t::pAddress

8.650.2.4 uint8_t* pack_wds_SetMobileIPProfile_t::pEnabled

8.650.2.5 uint32_t* pack_wds_SetMobileIPProfile_t::pHASPI

8.650.2.6 int8_t* pack_wds_SetMobileIPProfile_t::pMNAAA

8.650.2.7 int8_t* pack_wds_SetMobileIPProfile_t::pMNHA

8.650.2.8 int8_t* pack_wds_SetMobileIPProfile_t::pNAI

8.650.2.9 uint32_t* pack_wds_SetMobileIPProfile_t::pPrimaryHA

8.650.2.10 uint8_t* pack_wds_SetMobileIPProfile_t::pRevTunneling

8.650.2.11 uint32_t* pack_wds_SetMobileIPProfile_t::pSecondaryHA

8.650.2.12 int8_t pack_wds_SetMobileIPProfile_t::spc[10]

8.651 pack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo * pCurProfile

8.651.1 Detailed Description

This structure contains pack create profile.

Parameters

pProfileId	• 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM7430/EM7455/MC7430/MC7455 onwards) • 101 to 106 for 3GPP2 profile
pProfileType	• Identifies the technology type of the profile – 0x00 - 3GPP – 0x01 - 3GPP2 – NULL is not allowed
pCurProfile	• union of 3GPP and 3GPP2 profile • See wds_profileInfo

Note

- If profileID is NULL, 3GPP profile will be created and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be created

8.651.2 Field Documentation

8.651.2.1 wds_profileInfo* pack_wds_SLQSCreateProfile_t::pCurProfile

8.651.2.2 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileId

8.651.2.3 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileType

8.652 pack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.652.1 Detailed Description

This structure contains the information about the profile to be deleted.

Parameters

profileType	• Identifies the type of profile – 0x00 - 3GPP • Note: Deletion of 3GPP2 profiles is not supported.
profileIndex	• Index of the configured profile to be deleted – Value between 1 - 16 (EM/MC73xx or earlier) – Value between 1 - 24 (EM7430/EM7455/MC7430/MC7455 onwards)

8.652.2 Field Documentation

8.652.2.1 uint8_t pack_wds_SLQSDeleteProfile_t::profileIndex

8.652.2.2 uint8_t pack_wds_SLQSDeleteProfile_t::profileType

8.653 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

8.653.1 Detailed Description

This structure contains pack get current data system state information.

Parameters

NULL	• this is a dummy structure

8.654 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference

8.654.1 Detailed Description

This structure contains pack get data bearer technology information.

Parameters

NULL	• this is a dummy structure

8.655 pack_wds_SLQSGetDUNCallInfo_t Struct Reference

Data Fields

- uint32_t Mask

- uint8_t * pReportConnStatus
- wds_transferStatInd * pTransferStatInd
- uint8_t * pReportDormStatus
- uint8_t * pReportDataBearerTech
- uint8_t * pReportChannelRate

8.655.1 Detailed Description

This structure contains the DUN Call Info Request parameters.

Parameters

Mask	• Mandatory parameter • Set the bits corresponding to the information requested to 1 • All other bits must be set to 0. • If any values are not available or applicable, the corresponding TLVs are not returned in the response. – Bit 0 - Connection Status – Bit 1 - Last call end reason – Bit 2 - Tx/Rx bytes OK – Bit 3 - Dormancy status – Bit 4 - Data bearer – Bit 5 - Channel rate – Bit 6 - Call active duration
pReportConnStatus	• Connect Status Indicator – 0 - Do not report – 1 - Report connection status and call end reason
pTransferStatInd	• See wds_transferStatInd for more information
pReportDormStatus	• Dormancy Status Indicator – 0 - Do not report – 1 - Report traffic channel state of interface used for data connection
pReportDataBearerTech	• Current Data Bearer Technology Indicator – 0 - Do not report – 1 - Report radio interface used for data transfer when it changes
pReportChannelRate	• Channel Rate Indicator – 0 - Do not report – 1 - Report channel rate

8.655.2 Field Documentation

8.655.2.1 uint32_t pack_wds_SLQSGetDUNCallInfo_t::Mask

8.655.2.2 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportChannelRate

8.655.2.3 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportConnStatus

8.655.2.4 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDataBearerTech

8.655.2.5 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDormStatus

8.655.2.6 wds_transferStatInd* pack_wds_SLQSGetDUNCallInfo_t::pTransferStatInd

8.656 pack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- uint8_t ProfileId
- uint8_t ProfileType

8.656.1 Detailed Description

This structure contains pack get profile settings information.

Parameters

ProfileID	• 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM7430/EM7455/MC7430/MC7455 onwards) • 101 to 106 for 3GPP2 profile
ProfileType	• Identifies the technology type of the profile – 0x00 - 3GPP – 0x01 - 3GPP2

Note

- If profileID is NULL, 3GPP profile will be fetched and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be fetched

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

8.656.2 Field Documentation

8.656.2.1 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileId

8.656.2.2 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileType

8.657 pack_wds_SLQSGetRuntimeSettings_t Struct Reference

Data Fields

- uint32_t * pReqSettings

8.657.1 Detailed Description

This structure contains pack get runtime settings information

Parameters

pReqSettings	Requested Settings (Optional Parameter) • Set bits to 1, corresponding to requested information. All other bits must be set to 0. • If the values are not available, the corresponding TLVs are not returned in the response. • Absence of this mask TLV results in the device returning all of the available information corresponding to bits 0 through 12. • In cases where the information from bit 13 or greater is required, this TLV with all the necessary bits set must be present in the request. • Values – Bit 0 - Profile identifier – Bit 1 - Profile name – Bit 2 - PDP type – Bit 3 - APN name – Bit 4 - DNS address – Bit 5 - UMTS/GPRS granted QoS – Bit 6 - Username – Bit 7 - Authentication Protocol – Bit 8 - IP address – Bit 9 - Gateway info (address and subnet mask) – Bit 10 - PCSCF address using PCO flag – Bit 11 - PCSCF server address list – Bit 12 - PCSCF domain name list – Bit 13 - MTU – Bit 14 - domain name list – Bit 15 - IP family – Bit 16 - IM_CM flag – Bit 17 - Technology name – Bit 18 - Operator reserved PCO (Not Supported on MC/EM73xx)

8.657.2 Field Documentation

8.657.2.1 uint32_t* pack_wds_SLQSGetRuntimeSettings_t::pReqSettings

8.658 pack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo curProfile

8.658.1 Detailed Description

This structure contains pack modify profile information.

Parameters

pProfileId	• 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM7430/EM7455/MC7430/MC7455 onwards) • 101 to 106 for 3GPP2 profile
pProfileType	• Identifies the technology type of the profile – 0x00 - 3GPP – 0x01 - 3GPP2 – NULL is not allowed
curProfile	• union of 3GPP and 3GPP2 profile • See wds_profileInfo

Note

- If profileID is NULL, 3GPP profile will be created and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be created

8.658.2 Field Documentation

8.658.2.1 wds_profileInfo pack_wds_SLQSModifyProfile_t::curProfile

8.658.2.2 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileId

8.658.2.3 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileType

8.659 pack_wds_SLQSSet3GPPConfigItem_t Struct Reference

Data Fields

- uint16_t * pLTEAttachProfile
- uint16_t * pProfileList
- uint8_t * pDefaultPDNEnabled
- uint8_t * p3gppRelease
- uint16_t * pLTEAttachProfileList
- uint16_t LTEAttachProfileListLen

8.659.1 Detailed Description

This structure contains pack set 3GPP configuration item information.

Parameters

pLTEAttach-Profile	• Optional parameter • LTE Attach Profile – points to a single WORD Value indicating the attached LTE Profile – Optional parameter with possible values 1-16 (EM/MC73xx or earlier) • This setting is deprecated on MC7430/MC7455/EM7430/EM7455
ProfileList	Profile List • an array of 4 profile configurations • Each element points to a single WORD value indicating profile • Optional parameter with possible values – 1 - 16 (MC/EM73xx and before) – 1 - 24 (MC7430/MC7455/EM7430/EM7455 and onwards) • function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
pDefaultPDN-Enabled	• Optional parameter – 0 - disabled – 1 - enabled
p3gppRelease	3GPP release • Optional parameter – 0 - Release_99 – 1 - Release_5 – 2 - Release_6 – 3 - Release_7 – 4 - Release_8 • In 9x30 and onwards – 5 - Release 9 – 6 - Release 10 – 7 - Release 11
pLTEAttach-ProfileList	• pointer to WORD array indicating LTE Attach Profile List – Optional parameter – possible values: 1-24 – This setting is only supported for MC7430/MC7455/EM7430/EM7455 onwards – Please provide attach profiles in order of decreasing priority in this list.
LTEAttach-ProfileListLen	• Number of element in pLTEAttachProfileList – valid range: 1-24 – This setting is only supported for MC7430/MC7455/EM7430/EM7455 onwards

8.659.2 Field Documentation

8.659.2.1 uint16_t pack_wds_SLQSSet3GPPConfigItem_t::LTEAttachProfileListLen

8.659.2.2 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::p3gppRelease

8.659.2.3 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::pDefaultPDNEnabled

8.659.2.4 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfile

8.659.2.5 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfileList

8.659.2.6 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pProfileList

8.660 pack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint8_t IPFamilyPreference

8.660.1 Detailed Description

This structure contains pack set IP family preference information

Parameters

IPFamily-Preference	IP Family preference • PACK_WDS_IPV4 IP Version 4 • PACK_WDS_IPV6 IP Version 6

8.660.2 Field Documentation

8.660.2.1 uint8_t pack_wds_SLQSSetIPFamilyPreference_t::IPFamilyPreference

8.661 pack_wds_SLQSSetWdsEventCallback_t Struct Reference

Data Fields

- uint8_t dataBearer
- uint8_t dormancyStatus
- uint8_t mobileIP
- uint8_t transferStats
- uint8_t currentDataBearer
- uint8_t dataSystemStatus
- uint8_t dataBearerTechExt
- uint8_t interval

8.661.1 Detailed Description

This structure contains set WDS event callback information.

Parameters

dataBearer	data bearer
dormancyStatus	dormancy status
mobileIP	mobile IP
currentData-Bearer	current data bearer
dataSystem-Status	data system status
dataBearerTech-Ext	data Bearer Technology Extended
interval	interval

8.661.2 Field Documentation

8.661.2.1 uint8_t pack_wds_SLQSSetWdsEventCallback_t::currentDataBearer

8.661.2.2 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearer

8.661.2.3 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearerTechExt

8.661.2.4 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataSystemStatus

8.661.2.5 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dormancyStatus

8.661.2.6 uint8_t pack_wds_SLQSSetWdsEventCallback_t::interval

8.661.2.7 uint8_t pack_wds_SLQSSetWdsEventCallback_t::mobileIP

8.661.2.8 uint8_t pack_wds_SLQSSetWdsEventCallback_t::transferStats

8.662 pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4ProfileId * pProfileId

8.662.1 Detailed Description

This structure contain get DHCPv4 client configure.

Parameters

pProfileId	pointer to Profile Id structure • See wdsDhcpv4ProfileId

8.662.2 Field Documentation

8.662.2.1 wdsDhcpv4ProfileId* pack_wds_SLQSSGetDHCPv4ClientConfig_t::pProfileId

8.663 pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wds_DHCPv4ProfileId * pProfileId
- wds_DHCPv4HWConfig * pHwConfig
- wds_DHCPv4OptionList * pRequestOptionList

8.663.1 Detailed Description

WDS SWI DHCPv4 Config Structure

Parameters

pProfileId	• pointer to Profile Id structure
pHWConfig	• pointer to HW Config structure
pRequestOption-List	• pointer to Option List structure to be sent in DHCP request

8.663.2 Field Documentation

8.663.2.1 wds_DHCPv4HWConfig* pack_wds_SLQSSSetDHCPv4ClientConfig_t::pHwConfig

8.663.2.2 wds_DHCPv4ProfileId* pack_wds_SLQSSSetDHCPv4ClientConfig_t::pProfileId

8.663.2.3 wds_DHCPv4OptionList* pack_wds_SLQSSSetDHCPv4ClientConfig_t::pRequestOptionList

8.664 pack_wds_SLQSSSetLoopback_t Struct Reference

Data Fields

- uint8_t loopbackMode
- uint8_t loopbackMultiplier

8.664.1 Detailed Description

This structure contains pack set loopback information.

Parameters

loopbackMode	• Loopback Mode. – 0 - Disable – 1 - Enable
loopback-Multiplier	• Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.664.2 Field Documentation

8.664.2.1 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMode

8.664.2.2 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMultiplier

8.665 pack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint8_t * pTech
- uint32_t * pprofileid3gpp
- uint32_t * pprofileid3gpp2
- uint32_t * pAuth
- char * pUser
- char * pPass

8.665.1 Detailed Description

This structure contains pack Start Data Session Information.

Parameters

pTech	• Indicates the technology preference – 1 - UMTS – 2 - CDMA – 3 - eMBMS – 4 - Modem Link Label. Modem Link is an interface for transferring data between entities on AP and modem. • optional
pprofileid3gpp	• pointer to 3GPP profile id • optional
pprofileid3gpp2	• pointer to 3GPP2 profile id • optional
pAuth	• Authentication type, it can be PAP or CHAP • optional
pUser	• username for authentication process • optional
pPass	• password for authentication process • optional

8.665.2 Field Documentation

8.665.2.1 uint32_t* pack_wds_SLQSStartDataSession_t::pAuth

8.665.2.2 char* pack_wds_SLQSStartDataSession_t::pPass

8.665.2.3 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp

8.665.2.4 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp2

8.665.2.5 uint8_t* pack_wds_SLQSStartDataSession_t::pTech

8.665.2.6 char* pack_wds_SLQSStartDataSession_t::pUser

8.666 pack_wds_SLQSStopDataSession_t Struct Reference

Data Fields

- uint32_t * psid

8.666.1 Detailed Description

This structure contains pack stop data session information.

Parameters

psid	session id

8.666.2 Field Documentation

8.666.2.1 uint32_t* pack_wds_SLQSStopDataSession_t::psid

8.667 pack_wds_SLQSSwiProfileChangeCallback_t Struct Reference

Data Fields

- uint8_t * pProfileChangeInd

8.667.1 Detailed Description

This structure contains Profile Change Enable/Disable Notification Parameter

Parameters

pProfileChange-Ind	• Enable/Disable Notification

8.667.2 Field Documentation

8.667.2.1 uint8_t* pack_wds_SLQSSwiProfileChangeCallback_t::pProfileChangeInd

8.668 pack_wds_SLQSWdsSetEventReport_t Struct Reference

Data Fields

- uint8_t * pCurrChannelRateInd
- wds_TrStatInd * pTransferStatInd
- uint8_t * pDataBearerTechInd
- uint8_t * pDormancyStatusInd
- uint8_t * pMIPStatusInd
- uint8_t * pCurrDataBearerTechInd
- uint8_t * pDataCallStatusChangeInd
- uint8_t * pCurrPrefDataSysInd
- uint8_t * pEVDOPageMonPerChangeInd
- uint8_t * pDataSystemStatusChangeInd

8.668.1 Detailed Description

This structure contains the information about the Set Event Report Request parameters.

Parameters

pCurrChannelRateInd	(optional) • Current Channel Rate Indicator. – 0 - Do not report – 1 - Report channel rate when it changes
pTransferStatInd	(optional) • See wds_TrStatInd for more information.
pDataBearerTechInd	(optional) • Data Bearer Technology Indicator. – 0 - Do not report – 1 - Report radio interface used for data transfer when it changes
pDormancyStatusInd	(optional) • Dormancy Status indicator. – 0 - Do not report – 1 - Report traffic channel state of interface used for data connection
pMIPStatusInd	(optional) • MIP Status Indicator. – 0 - Do not report – 1 - Report MIP status
pCurrDataBearerTechInd	(optional) • Current Data Bearer Technology Indicator. – 0 - Do not report – 1 - Report current data bearer technology when it changes
pDataCallStatusChangeInd	(optional) • Data Call Status Change Indicator. – 0 - Do not report – 1 - Report data call status change when it changes

pCurrPrefData-SysInd	(optional) • Current Preferred Data System Indicator. – 0 - Do not report – 1 - Report preferred data system when it changes
pEVDOPage-MonPerChange-Ind	(optional) • EV-DO Page Monitor Period Change Indicator. – 0 - Do not report – 1 - Report EV-DO page monitor period change event
pDataSystem-StatusChange-Ind	(optional) • Data System Status Change Indicator. – 0 - Do not report – 1 - Report data system status change event

Note

At least one parameter should be present.

8.668.2 Field Documentation

8.668.2.1 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrChannelRateInd

8.668.2.2 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrDataBearerTechInd

8.668.2.3 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrPrefDataSysInd

8.668.2.4 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataBearerTechInd

8.668.2.5 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataCallStatusChangeInd

8.668.2.6 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataSystemStatusChangeInd

8.668.2.7 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDormancyStatusInd

8.668.2.8 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pEVDOPageMonPerChangeInd

8.668.2.9 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pMIPStatusInd

8.668.2.10 wds_TrStatInd* pack_wds_SLQSWdsSetEventReport_t::pTransferStatInd

8.669 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference

Data Fields

- uint8_t contextId
- uint8_t contextType

8.669.1 Detailed Description

This structure contains the PDP Runtime Settings Request parameters.

Parameters

contextId	• Context Identifier
contextType	• Identifies technology type – 0 - 3GPP – 1 - 3GPP2

8.669.2 Field Documentation

8.669.2.1 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.669.2.2 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextType

8.670 PackCreateProfileOut Struct Reference

Data Fields

- uint8_t ProfileType
- uint8_t ProfileIndex
- uint16_t ExtErrorCode

8.670.1 Detailed Description

This structure contains out parameter Information

Parameters

ProfileType	• Identifies the type of profile 0x00 = 3GPP 0x01 = 3GPP2 • Bit to check in ParamPresenceMask - **1**
ProfileIndex	• Index identifying the profile that was created • Bit to check in ParamPresenceMask - **1**
ExtErrorCode	• The extended error code received from DS Profile subsystem • Bit to check in ParamPresenceMask - **224**

8.670.2 Field Documentation

8.670.2.1 uint16_t PackCreateProfileOut::ExtErrorCode

8.670.2.2 uint8_t PackCreateProfileOut::ProfileIndex

8.670.2.3 uint8_t PackCreateProfileOut::ProfileType

8.671 packgetDyingGaspCfg Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.671.1 Detailed Description

This struture contains the TLV required to get the Dysing GASP Config.

Parameters

pDestSMSNum[IN]	• SMS Destination Number as string of 8 bit ASCII Characters Max 20 chars. • Optional parameter.
pDestSMSContent[IN]	• SMS COntent as a string of 8 bit ASCII text characters Max 160 chars. • Optional parameter.

8.671.2 Field Documentation

8.671.2.1 uint8_t* packgetDyingGaspCfg::pDestSMSContent

8.671.2.2 uint8_t* packgetDyingGaspCfg::pDestSMSNum

8.672 packgetDyingGaspStatistics Struct Reference

Data Fields

- uint32_t * pTimeStamp
- uint8_t * pSMSAttemptedFlag

8.672.1 Detailed Description

This structure contains Get Dysing GASP Statistics

Parameters

pTimeStamp[OUT]	• Timestamp of the last time power loss was detected and Dying Gasp feature was triggered. • UTC time in seconds since Jan 06, 1980 (GPS Epoch).
pSMSAttemptedFlag[OUT]	• Indicates whether device attempted to send SMS in the last power loss event. • 0 - SMS not attempted • 1 - SMS attempted • This only indicates device sent the SMS but does not guarantee network delivery.

8.672.2 Field Documentation

8.672.2.1 uint8_t* packgetDyingGaspStatistics::pSMSAttemptedFlag

8.672.2.2 uint32_t* packgetDyingGaspStatistics::pTimeStamp

8.673 PackSwiAvmsSetSettingsAPNInfo Struct Reference

Data Fields

- uint8_t bAPNLength
- uint8_t szAPN [49]
- uint8_t bUnameLength
- uint8_t szUname [29]
- uint8_t bPWDLength
- uint8_t szPWD [29]

8.673.1 Detailed Description

This structure contains pack set settings APN information.

Parameters

bAPNLength	APN Length.
szAPN	APN String.
bUnameLength	Username Length.
szUname	User Name String.
bPWDLength	Password Length.
szPWD	Password String.

8.673.2 Field Documentation

8.673.2.1 uint8_t PackSwiAvmsSetSettingsAPNInfo::bAPNLength

8.673.2.2 uint8_t PackSwiAvmsSetSettingsAPNInfo::bPWDLength

8.673.2.3 uint8_t PackSwiAvmsSetSettingsAPNInfo::bUnameLength

8.673.2.4 uint8_t PackSwiAvmsSetSettingsAPNInfo::szAPN[49]

8.673.2.5 uint8_t PackSwiAvmsSetSettingsAPNInfo::szPWD[29]

8.673.2.6 uint8_t PackSwiAvmsSetSettingsAPNInfo::szUname[29]

8.674 PackSwiAvmsSetSettingsConnectionRetryTimers Struct Reference

Data Fields

- uint16_t Timers [8]

8.674.1 Detailed Description

This structure contains the SLQSAVMSSetSettings Polling timer to connect to AVMS server.

Parameters

Timers[0]	- Timer 1. • 0-20160 (min) • 0:retry disabled
Timers[1]	- Timer 2. • 1-20160 (min)
Timers[2]	- Timer 3. • 1-20160 (min)
Timers[3]	- Timer 4. • 1-20160 (min)
Timers[4]	- Timer 5. • 1-20160 (min)
Timers[5]	- Timer 6. • 1-20160 (min)
Timers[6]	- Timer 7. • 1-20160 (min)
Timers[7]	- Timer 8. • 1-20160 (min)

8.674.2 Field Documentation

8.674.2.1 uint16_t PackSwiAvmsSetSettingsConnectionRetryTimers::Timers[8]

8.675 PackSwiAvmsSetSettingsPeriodInfo Struct Reference

Data Fields

- uint32_t ulMin
- uint32_t ulMax

8.675.1 Detailed Description

This structure contain pack set settings period information.

Parameters

ulMin	Min period in seconds.
ulMax	Max period in seconds.

8.675.2 Field Documentation

8.675.2.1 uint32_t PackSwiAvmsSetSettingsPeriodInfo::ulMax

8.675.2.2 uint32_t PackSwiAvmsSetSettingsPeriodInfo::ulMin

8.676 PackSwiAVMSSettingsAPNInfo Struct Reference

Data Fields

- uint8_t bAPNLength
- uint8_t * pAPN
- uint8_t bUnameLength
- uint8_t * pUname
- uint8_t bPWDLength
- uint8_t * pPWD

8.676.1 Detailed Description

This structure contains the SLQSAVMSGetSettings APN Info to connect to AVMS server.

Parameters

bAPNLength	- APN Length(Max 49).
szAPN	- APN.
bUnameLength	- User Name Length (Max 29).
szUname	- User Name.
bPWDLength	- Password Length (Max 29).
szPWD	- Password.

8.676.2 Field Documentation

8.676.2.1 uint8_t PackSwiAVMSSettingsAPNInfo::bAPNLength

8.676.2.2 uint8_t PackSwiAVMSSettingsAPNInfo::bPWDLength

8.676.2.3 uint8_t PackSwiAVMSSettingsAPNInfo::bUnameLength

8.676.2.4 uint8_t* PackSwiAVMSSettingsAPNInfo::pAPN

8.676.2.5 uint8_t* PackSwiAVMSSettingsAPNInfo::pPWD

8.676.2.6 uint8_t* PackSwiAVMSSettingsAPNInfo::pUname

8.677 PackSwiAVMSSettingsConnectionRetryTimers Struct Reference

Data Fields

- uint16_t Timers [8]

8.677.1 Detailed Description

This structure contains the SLQSAVMSGetSettings Polling timer to connect to AVMS server.

Parameters

Timers[0]	- Timer 1. • 0-20160 (min) • 0:retry disabled

Timers[1]	- Timer 2. • 1-20160 (min)
Timers[2]	- Timer 3. • 1-20160 (min)
Timers[3]	- Timer 4. • 1-20160 (min)
Timers[4]	- Timer 5. • 1-20160 (min)
Timers[5]	- Timer 6. • 1-20160 (min)
Timers[6]	- Timer 7. • 1-20160 (min)
Timers[7]	- Timer 8. • 1-20160 (min)

8.677.2 Field Documentation

8.677.2.1 uint16_t PackSwiAVMSSettingsConnectionRetryTimers::Timers[8]

8.678 PackSwiAVMSSettingsPeriodsInfo Struct Reference

Data Fields

- uint32_t min
- uint32_t max

8.678.1 Detailed Description

This structure contains the SLQSAVMSSettings Min/Max Period of an Observation.

Parameters

min	- Min period in seconds.
max	- Max period in seconds.

8.678.2 Field Documentation

8.678.2.1 uint32_t PackSwiAVMSSettingsPeriodsInfo::max

8.678.2.2 uint32_t PackSwiAVMSSettingsPeriodsInfo::min

8.679 QcqmiService Struct Reference

Data Fields

- bool useQcqmi
- char * devicePath
- bool readyToRead

8.679.1 Detailed Description

Parameters

useQcqmi	• Using QCQMI interface.
devicePath	• Pointer to device path.
readyToRead	• QCQMI device ready to read.

8.679.2 Field Documentation

8.679.2.1 char* QcqmiService::devicePath

8.679.2.2 bool QcqmiService::readyToRead

8.679.2.3 bool QcqmiService::useQcqmi

8.680 QmiHdr Struct Reference

Data Fields

- uint16_t messageId
- uint16_t length

8.680.1 Field Documentation

8.680.1.1 uint16_t QmiHdr::length

8.680.1.2 uint16_t QmiHdr::messageId

8.681 QmiService Struct Reference

Data Fields

- struct QmiService * pNext
- uint8_t serviceType
- uint8_t clientId
- struct QmuxTransport * pTransport
- uint16_t transactionId
- pthread_mutex_t transactionIdLock
- QmiTransaction transactionPool [10]
- pthread_mutex_t transactionPoolLock

- QMI_INDICATION_CALLBACK pIndicationCallback
- void * pIndicationCallbackContext
- int deviceFd
- struct QmuxTransport * pQcqmitransport

8.681.1 Detailed Description

This structure abstracts a QMI service. It is responsible issuing transaction IDs. It tracks QMI requests and responses using transaction objects from a pool. It invokes a user callback when QMI indications which match its client ID (or a broadcast ID) are received from the transport layer.

Parameters

pNext	• A link so we can store this object in a linked list by the transport layer.
serviceType	• QMI service type, CTL, NAS, etc.
clientId	• QMI client ID
pTransport	• Pointer to underlying QMUX transport object.
transactionId	• QMI transaction ID. Upper limit 0xffff for regular QMI services, 0xff for CTL service.
transactionId-Lock	• Mutex to make transactionId access thread-safe.
transactionPool	• Pool of transaction objects which may be assigned to QMI requests made by clients of this service.
transactionPool-Lock	• Mutex to make transactionPool access thread-safe
pIndication-Callback	• User callback for QMI indications of this service.
pIndication-CallbackContext	• User context for user callback.
deviceFd	• Store service fd.
pQcqmitransport	• Pointer to QMI service transport object.

8.681.2 Field Documentation

8.681.2.1 uint8_t QmiService::clientId

8.681.2.2 int QmiService::deviceFd

8.681.2.3 QMI_INDICATION_CALLBACK QmiService::pIndicationCallback

8.681.2.4 void* QmiService::pIndicationCallbackContext

8.681.2.5 struct QmiService* QmiService::pNext

8.681.2.6 struct QmuxTransport* QmiService::pQcqmitransport

8.681.2.7 struct QmuxTransport* QmiService::pTransport

8.681.2.8 uint8_t QmiService::serviceType

8.681.2.9 uint16_t QmiService::transactionId

8.681.2.10 pthread_mutex_t QmiService::transactionIdLock

8.681.2.11 QmiTransaction QmiService::transactionPool[10]

8.681.2.12 pthread_mutex_t QmiService::transactionPoolLock

8.682 qmiSmsMessageList Struct Reference

Data Fields

- uint32_t messageIndex
- uint32_t messageTag

8.682.1 Detailed Description

This structure contains SMS message list.

Parameters

messageIndex	• Message index of each matched message
messageTag	• Messagetag

8.682.2 Field Documentation

8.682.2.1 uint32_t qmiSmsMessageList::messageIndex

8.682.2.2 uint32_t qmiSmsMessageList::messageTag

8.683 QmiSyncObject Struct Reference

Data Fields

- pthread_mutex_t mutex
- pthread_cond_t condition
- uint8_t buffer [(2048-6)]
- uint16_t bufferSize

8.683.1 Detailed Description

This structure is a wrapper around a mutex and condition object and a buffer. It is a convenience object which may be used to create synchronous QMI APIs.

Parameters

mutex	• For thread safety.
condition	• Triggered on completion.
buffer	• QMI request packet is written to this buffer. On request completion, contains the QMI response packet.
bufferSize	• Set to size in bytes of QMI request packet. On request completion, set to size of QMI response packet.

Note

The buffer is intended for QMI packet only i.e. QMI header and QMI payload. This buffer becomes the SDU of a QMUX packet.

8.683.2 Field Documentation

8.683.2.1 uint8_t QmiSyncObject::buffer[(2048-6)]

8.683.2.2 uint16_t QmiSyncObject::bufferSize

8.683.2.3 pthread_cond_t QmiSyncObject::condition

8.683.2.4 pthread_mutex_t QmiSyncObject::mutex

8.684 QmiTransaction Struct Reference

Data Fields

- bool inUse
- uint16_t transactionId
- uint16_t messageId
- QMI_RESPONSE_CALLBACK pResponseCallback
- void * pResponseCallbackContext

8.684.1 Detailed Description

This structure abstracts a QMI transaction. It is responsible for tracking a QMI request to its response.

Parameters

inUse	• true indicates that this object has been allocated by the QMI service and is un use.
transactionId	• QMI transaction ID. Upper limit 0xffff for regular QMI services, 0xff for CTL service.
messageId	• QMI message ID.
pResponse-Callback	• User callback for QMI response.
pResponse-CallbackContext	• User context for user callback.

8.684.2 Field Documentation

8.684.2.1 bool QmiTransaction::inUse

8.684.2.2 uint16_t QmiTransaction::messageId

8.684.2.3 QMI_RESPONSE_CALLBACK QmiTransaction::pResponseCallback

8.684.2.4 void* QmiTransaction::pResponseCallbackContext

8.684.2.5 uint16_t QmiTransaction::transactionId

8.685 QmiTransactionHdr Struct Reference

Data Fields

- uint8_t flags
- uint16_t transactionId

8.685.1 Field Documentation

8.685.1.1 uint8_t QmiTransactionHdr::flags

8.685.1.2 uint16_t QmiTransactionHdr::transactionId

8.686 qmiWDSDataBearerTechnology Struct Reference

Data Fields

- uint8_t currentNetwork
- uint32_t ratMask
- uint32_t soMask

8.686.1 Detailed Description

Structure to hold the current data bearer technology values

Parameters

currentNetwork[-OUT]	• current selected network – 0 - UNKNOWN – 1 - 3GPP2 – 2 - 3GPP – 0xff - Invalid data.
ratMask[OUT]	RAT Mask
soMask[OUT]	SO Mask

8.686.2 Field Documentation

8.686.2.1 uint8_t qmiWDSDataBearerTechnology::currentNetwork

8.686.2.2 uint32_t qmiWDSDataBearerTechnology::ratMask

8.686.2.3 uint32_t qmiWDSDataBearerTechnology::soMask

8.687 qmTlvResult Struct Reference

Data Fields

- swi_uint256_t TlvPresenceMask
- qmulong TlvResultCode
- qmuint16 DeviceResult
- qmuint16 DeviceError

8.687.1 Field Documentation

8.687.1.1 qmuint16 qmTlvResult::DeviceError

8.687.1.2 qmuint16 qmTlvResult::DeviceResult

8.687.1.3 swi_uint256_t qmTlvResult::TlvPresenceMask

8.687.1.4 qmulong qmTlvResult::TlvResultCode

8.688 QMUX_HDR Struct Reference

Data Fields

- uint8_t IfType
- uint16_t Length
- uint8_t CtlFlags
- uint8_t QMIType
- uint8_t ClientId

8.688.1 Field Documentation

8.688.1.1 uint8_t QMUX_HDR::ClientId

8.688.1.2 uint8_t QMUX_HDR::CtlFlags

8.688.1.3 uint8_t QMUX_HDR::IfType

8.688.1.4 uint16_t QMUX_HDR::Length

8.688.1.5 uint8_t QMUX_HDR::QMIType

8.689 QmuxTransport Struct Reference

Data Fields

- bool useMbim
- void * pDataLayer
- int deviceFd
- int shutdownFd
- struct QmiService * qmiServiceList
- pthread_mutex_t writeLock
- pthread_mutex_t qmiServiceListLock
- pthread_t readThread
- QcqmiService qcqmi

8.689.1 Detailed Description

This structure abstracts a QMUX transport layer. It is responsible for applying QMUX headers to QMI packets from QMI services and forwarding them to the device. It is responsible for receiving QMUX packets from the device. On the basis of QMUX headers, it distributes the SDUs (QMI packets) to the appropriate QMI service. It maintains a list of connected QMI service objects.

Parameters

useMbim	• Using MBIM interface as opposed to direct QMUX interface for data layer.
pDataLayer	• Pointer to instantiated data layer.
deviceFd	• File descriptor of device.
shutdownFd	• Event descriptor. Used to signal read thread to terminate.
qmiServiceList	• List of connected QMI service objects.
writeLock	• Provides mutual exclusion for write operations.

qmiServiceList-Lock	• Provides thread safety for the service list.
readThread	• Thread responsible for reading from device.
qcqmi	• QCQMI data.

8.689.2 Field Documentation

8.689.2.1 int QmuxTransport::deviceFd

8.689.2.2 void* QmuxTransport::pDataLayer

8.689.2.3 QcqmiService QmuxTransport::qcqmi

8.689.2.4 struct QmiService* QmuxTransport::qmiServiceList

8.689.2.5 pthread_mutex_t QmuxTransport::qmiServiceListLock

8.689.2.6 pthread_t QmuxTransport::readThread

8.689.2.7 int QmuxTransport::shutdownFd

8.689.2.8 bool QmuxTransport::useMbim

8.689.2.9 pthread_mutex_t QmuxTransport::writeLock

8.690 qos_BindDataPortMuxID_t Struct Reference

Data Fields

- uint8_t MuxID

8.690.1 Detailed Description

Structure that contains the request Binds a control point to a data port Mux ID data.

Parameters

MuxID	Mux ID: • 0x80-0x8F : valid value

8.690.2 Field Documentation

8.690.2.1 uint8_t qos_BindDataPortMuxID_t::MuxID

8.691 qos_BindDataPortPeripheralEndPointID_t Struct Reference

Data Fields

- uint32_t EndPointType
- uint32_t IfaceID

8.691.1 Detailed Description

Structure that contains the request Binds a control point to a data port Periapheral End Point Type data.

Parameters

EndPointType	Peripheral end point type. Values: • 0 - Reserved • 1 - HSIC • 2 - HSUSB • 3 - PCIE • 4 - Embedded
IfaceID	Peripheral interface number.

8.691.2 Field Documentation

8.691.2.1 uint32_t qos_BindDataPortPeripheralEndPointID_t::EndPointType

8.691.2.2 uint32_t qos_BindDataPortPeripheralEndPointID_t::IfaceID

8.692 qos_BindDataPortSIODataPort_t Struct Reference

Data Fields

- uint16_t SIODataPort

8.692.1 Detailed Description

Structure that contains the request Binds a control point to a data port SIO Data Port data.

Parameters

u16SIODataPort	SIO Data Port to which the client binds

8.692.2 Field Documentation

8.692.2.1 uint16_t qos_BindDataPortSIODataPort_t::SIODataPort

8.693 ReleaseClientIdReqMsg Struct Reference

Data Fields

- uint16_t messageId
- uint16_t length
- uint8_t tlvType
- uint16_t tlvLength

- uint8_t serviceType
- uint8_t clientId

8.693.1 Field Documentation

8.693.1.1 uint8_t ReleaseClientIdReqMsg::clientId

8.693.1.2 uint16_t ReleaseClientIdReqMsg::length

8.693.1.3 uint16_t ReleaseClientIdReqMsg::messageId

8.693.1.4 uint8_t ReleaseClientIdReqMsg::serviceType

8.693.1.5 uint16_t ReleaseClientIdReqMsg::tlvLength

8.693.1.6 uint8_t ReleaseClientIdReqMsg::tlvType

8.694 ReleaseClientIdRspMsg Struct Reference

Data Fields

- uint16_t messageId
- uint16_t length
- uint8_t tlvType
- uint16_t tlvLength
- uint16_t result
- uint16_t error
- uint8_t tlv2Type
- uint16_t tlv2Length
- uint8_t serviceType
- uint8_t clientId

8.694.1 Field Documentation

8.694.1.1 uint8_t ReleaseClientIdRspMsg::clientId

8.694.1.2 uint16_t ReleaseClientIdRspMsg::error

8.694.1.3 uint16_t ReleaseClientIdRspMsg::length

8.694.1.4 uint16_t ReleaseClientIdRspMsg::messageId

8.694.1.5 uint16_t ReleaseClientIdRspMsg::result

8.694.1.6 uint8_t ReleaseClientIdRspMsg::serviceType

8.694.1.7 uint16_t ReleaseClientIdRspMsg::tlv2Length

8.694.1.8 uint8_t ReleaseClientIdRspMsg::tlv2Type

8.694.1.9 uint16_t ReleaseClientIdRspMsg::tlvLength

8.694.1.10 uint8_t ReleaseClientIdRspMsg::tlvType

8.695 ResultTlv Struct Reference

Data Fields

- uint8_t tlvType
- uint16_t tlvLength
- uint16_t result
- uint16_t error

8.695.1 Field Documentation

8.695.1.1 uint16_t ResultTlv::error

8.695.1.2 uint16_t ResultTlv::result

8.695.1.3 uint16_t ResultTlv::tlvLength

8.695.1.4 uint8_t ResultTlv::tlvType

8.696 RFBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBandClass
- uint16_t activeChannel

8.696.1 Detailed Description

This structure contains the RFBandInfo response parameters.

Parameters

radioInterface	• Radio interface technology – See Tables for Radio Interface
activeBandClass	• Active Band Class – See Tables for Band Classes
activeChannel	• Active channel (0 if channel is not relevant to the reported technology)

8.696.2 Field Documentation

8.696.2.1 uint16_t RFBandInfoElements::activeBandClass

8.696.2.2 uint16_t RFBandInfoElements::activeChannel

8.696.2.3 uint8_t RFBandInfoElements::radioInterface

8.697 rmTrasnferStaticsReq Struct Reference

Data Fields

- uint8_t bResetStatistics
- uint32_t ulMask

8.697.1 Detailed Description

RM Transfer Satistics Structure

Parameters

bResetStatistics	• Reset Statistics • Values: • 0 - Not Reset • Other - Reset
ulMask	• Enable/Disable RM Transfer Satatistics Indiscation Mask • Bit 0: Tx Packet Ok • Bit 1: Rx Packet Ok • Bit 2: Tx Bytes Ok • Bit 3: Rx Bytes Ok • Bit 4: Tx Packets Dropped • Bit 5: Rx Packets Dropped • Value: -0 - Disable -1 - Enable

8.697.2 Field Documentation

8.697.2.1 uint8_t rmTrasnferStaticsReq::bResetStatistics

8.697.2.2 uint32_t rmTrasnferStaticsReq::ulMask

8.698 sensorData_t Struct Reference

Data Fields

- uint32_t timeOfFirstSample
- uint8_t flags
- uint8_t sensorDataLen
- uint16_t timeOffset [64]
- uint32_t xAxis [64]
- uint32_t yAxis [64]
- uint32_t zAxis [64]

8.698.1 Detailed Description

This structure specifies information regarding the 3-Axis Sensor Data. Please check has_<Param_Name> field for presence of optional parameters

Parameters

timeOfFirst-Sample	• Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds
flags	• Flags to indicate any deviation from the default measurement assumptions. • All unused bits in this field must be set to 0. • Valid bitmasks – 0x01 - Bitmask to specify that a sign reversal is required while interpreting the sensor data; only applies to the accelerometer samples – 0x02 - Bitmask to specify that the sensor time stamp is the same as the modem time stamp
sensorDataLen	• Number of sets of the following elements – timeOffset – xAxis – yAxis – zAxis
timeOffset	• Sample time offset • Units - Milliseconds
xAxis	• Sensor x-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
yAxis	• Sensor Y-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
xAxis	• Sensor Z-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second

8.698.2 Field Documentation

8.698.2.1 uint8_t sensorData_t::flags

8.698.2.2 uint8_t sensorData_t::sensorDataLen

8.698.2.3 uint32_t sensorData_t::timeOfFirstSample

8.698.2.4 uint16_t sensorData_t::timeOffset[64]

8.698.2.5 uint32_t sensorData_t::xAxis[64]

8.698.2.6 uint32_t sensorData_t::yAxis[64]

8.698.2.7 uint32_t sensorData_t::zAxis[64]

8.699 slot_t Struct Reference

Data Fields

- uint32_t uPhyCardStatus
- uint32_t uPhySlotStatus
- uint8_t bLogicalSlot
- uint8_t bICCIDLength
- uint8_t bICCID [255]

8.699.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

uPhyCardStatus	• State of the card in the Pyhsical Slot Status. – 0x00 - Unknown. – 0x01 - Absent. – 0x02 - Present.
uPhySlotStatus	• State of the Physical Slot status. – 0x00 Inactive. – 0x01 Activate.
bLogicalSlot	• Logical Slot associated with this physical slot. THis is valid if the physical slot is active. – 1 - Slot 1. – 2 - Slot 2. – 3 - Slot 3. – 4 - Slot 4. – 5 - Slot 5.
bLogicalSlot	• Number of sets the sets of ICCCID
bICCID[MAX_ICCID_LENGTH]	• Contains the ICCID of the card in the physical slot.

8.699.2 Field Documentation

8.699.2.1 uint8_t slot_t::bICCID[255]

8.699.2.2 uint8_t slot_t::bICCIDLength

8.699.2.3 uint8_t slot_t::bLogicalSlot

8.699.2.4 uint32_t slot_t::uPhyCardStatus

8.699.2.5 uint32_t slot_t::uPhySlotStatus

8.700 slotInf Struct Reference

Data Fields

- uint8_t cardState
- uint8_t upinState
- uint8_t upinRetries
- uint8_t upukRetries
- uint8_t errorState
- uint8_t numApp
- appStats AppStatus [10]

8.700.1 Detailed Description

This structure contains information about the SLOTS present.

Parameters

cardState	• Indicates the state of the card for each slot. – 0 - Absent – 1 - Present – 2 - Error
upinState	• Indicates the state of UPIN. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked – 0xFF - Not Available
upinRetries	• Indicates the number of retries remaining to verify the UPIN. • If 0xFF, information not available.
upukRetries	• Indicates the number of retries remaining to unblock the UPIN. • If 0xFF, information not available.

errorState	• Indicates the reason for the card error, and is valid only when the card state is Error – 0 - Unknown – 1 - Power down – 2 - Poll error – 3 - No ATR received – 4 - Volt mismatch – 5 - Parity error – 6 - Unknown; possibly removed – 7 - Card returned technical problems – 0xFF - Not Available • Other values are possible and reserved for future use. • When an unknown value is received, it is to be handled as “Unknown”.
numApp	• Indicates the number of applications available on the card. • The following block is repeated for each application. i.e. AppStatus. • If zero(0) then no AppStatus information exists.
AppStatus[MAX_NO_OF_APPLICATIONS]	• See appStats for more information.

8.700.2 Field Documentation

8.700.2.1 appStats slotInf::AppStatus[10]

8.700.2.2 uint8_t slotInf::cardState

8.700.2.3 uint8_t slotInf::errorState

8.700.2.4 uint8_t slotInf::numApp

8.700.2.5 uint8_t slotInf::upinRetries

8.700.2.6 uint8_t slotInf::upinState

8.700.2.7 uint8_t slotInf::upukRetries

8.701 slots_t Struct Reference

Data Fields

- slot_t uimSlotStatus [255]

8.701.1 Detailed Description

This structure contains slots status.

Parameters

uimSlotStatus	• See slot_t

8.701.2 Field Documentation

8.701.2.1 slot_t slots_t::uimSlotStatus[255]

8.702 sms_BroadcastConfig Struct Reference

Data Fields

- uint16_t fromServiceId
- uint16_t toServiceId
- uint8_t selected

8.702.1 Detailed Description

This structure contains BroadcastConfig parameters

Parameters

fromServiceId	• Starting point of range of CBM message identifiers
toServiceId	• Ending point of range of CBM message identifiers
selected	• Range of CBM message identifiers indicated by from_service_id and to_service_id – 0x00 - Not selected – 0x01 - Selected

8.702.2 Field Documentation

8.702.2.1 uint16_t sms_BroadcastConfig::fromServiceId

8.702.2.2 uint8_t sms_BroadcastConfig::selected

8.702.2.3 uint16_t sms_BroadcastConfig::toServiceId

8.703 sms_CDMABroadcastConfig Struct Reference

Data Fields

- uint16_t serviceCategory
- uint16_t language
- uint8_t selected

8.703.1 Detailed Description

This structure contains CDMABroadcastConfig parameters

Parameters

serviceCategory	• Service category
language	• Language
selected	• Specified service_category and language – 0x00 - Not selected – 0x01 - Selected

8.703.2 Field Documentation

8.703.2.1 uint16_t sms_CDMABroadcastConfig::language

8.703.2.2 uint8_t sms_CDMABroadcastConfig::selected

8.703.2.3 uint16_t sms_CDMABroadcastConfig::serviceCategory

8.704 sms_getIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.704.1 Detailed Description

This structure contains Get Indication Register Response parameters

Parameters

pRegTrans-LayerInfoEvt	- • Optional parameter indicating registration status of transport layer information events • Values: – 0x00 - Disabled – 0x01 - Enabled
pRegTransNW-RegInfoEvt	- • Optional parameter indicating registration status of transport network registration information events • Values: – 0x00 - Disabled – 0x01 - Enabled

pRegCallStat-InfoEvt	- Optional parameter indicating registration status of call status information events
	- Values:
	  – 0x00 - Disabled
	  – 0x01 - Enabled

8.704.2 Field Documentation

8.704.2.1 uint8_t* sms_getIndicationReg::pRegCallStatInfoEvt

8.704.2.2 uint8_t* sms_getIndicationReg::pRegTransLayerInfoEvt

8.704.2.3 uint8_t* sms_getIndicationReg::pRegTransNWRegInfoEvt

8.705 sms_getMsgWaitingInfo Struct Reference

Data Fields

- uint8_t numInstances
- sms_messageWaitingInfoContent msgWaitInfo [0xFF]

8.705.1 Detailed Description

This structure contains Get Message Waiting Info Response parameters

Parameters

numInstances	- Number of sets of the elements in structure sms_messageWaitingInfoContent
pMsgWaitInfo	- Pointer to structure of sms_messageWaitingInfoContent. – See sms_messageWaitingInfoContent for more information.

8.705.2 Field Documentation

8.705.2.1 sms_messageWaitingInfoContent sms_getMsgWaitingInfo::msgWaitInfo[0xFF]

8.705.2.2 uint8_t sms_getMsgWaitingInfo::numInstances

8.706 sms_getTransLayerInfo Struct Reference

Data Fields

- uint8_t * pRegInd
- sms_transLayerInfo * pTransLayerInfo

8.706.1 Detailed Description

This structure contains Get Transport Layer Info Response parameters

Parameters

pRegInd	- • Optional parameter indicating if transport layer is registered • Values: – 0x00 - Transport layer is not registered – 0x01 - Transport layer is registered • returns a default value 0xFF if no response is received from the device.
pTransLayerInfo	• Pointer to structure of sms_transLayerInfo. – Optional parameter – See sms_transLayerInfo for more information • returns a default value 0xFF for parameter values if no response is received from the device.

8.706.2 Field Documentation

8.706.2.1 uint8_t* sms_getTransLayerInfo::pRegInd

8.706.2.2 sms_transLayerInfo* sms_getTransLayerInfo::pTransLayerInfo

8.707 sms_getTransNWRegInfo Struct Reference

Data Fields

- uint8_t * pRegStatus

8.707.1 Detailed Description

This structure contains transport network registration info parameter

Parameters

pRegStatus	- • Optional parameter indicating transport layer network registration status • Values: – 0x00 - No service – 0x01 - In progress – 0x02 - Failed – 0x03 - Limited Service – 0x04 - Full Service • returns a default value 0xFF if no response is received from the device.

8.707.2 Field Documentation

8.707.2.1 uint8_t* sms_getTransNWRegInfo::pRegStatus

8.708 sms_maxStorageSizeReq Struct Reference

Data Fields

- uint8_t storageType
- uint8_t * pMessageMode

8.708.1 Detailed Description

This structure contains get store max size request parameters

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
pMessage-Mode(optional	parameter) • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

Note

The Message Mode TLV must be included if the device is capable of supporting more than one protocol

8.708.2 Field Documentation

8.708.2.1 uint8_t* sms_maxStorageSizeReq::pMessageMode

8.708.2.2 uint8_t sms_maxStorageSizeReq::storageType

8.709 sms_maxStorageSizeResp Struct Reference

Data Fields

- uint32_t maxStorageSize
- uint32_t freeSlots

8.709.1 Detailed Description

This structure contains get store max size response parameters

Parameters

maxStorageSize	- • Memory Store Size

freeSlots	- Optional parameter indicating how much Memory is available - returns a default value 0xFFFFFFFF for parameter values if no response is received from the device.

8.709.2 Field Documentation

8.709.2.1 uint32_t sms_maxStorageSizeResp::freeSlots

8.709.2.2 uint32_t sms_maxStorageSizeResp::maxStorageSize

8.710 sms_messageWaitingInfoContent Struct Reference

Data Fields

- uint8_t msgType
- uint8_t activeInd
- uint8_t msgCount

8.710.1 Detailed Description

This structure contains message waiting information per instance

Parameters

msgType	• Message type – 0x00 - MWI_MESSAGE_TYPE_VOICMAIL - Voicemail – 0x01 - MWI_MESSAGE_TYPE_FAX - Fax – 0X02 - MWI_MESSAGE_TYPE_EMAIL - Email – 0x03 - MWI_MESSAGE_TYPE_OTHER - Other – 0x04 - MWI_MESSAGE_TYPE_VIDEOMAIL - Videomail
activeInd	• Indicates whether the indication is active – 0x00 - Inactive – 0x01 - Active
msgCount	• Number of messages

8.710.2 Field Documentation

8.710.2.1 uint8_t sms_messageWaitingInfoContent::activeInd

8.710.2.2 uint8_t sms_messageWaitingInfoContent::msgCount

8.710.2.3 uint8_t sms_messageWaitingInfoContent::msgType

8.711 sms_msgProtocolResp Struct Reference

Data Fields

- uint8_t msgProtocol

8.711.1 Detailed Description

This structure contains get message protocol response parameters

Parameters

msgProtocol	- • Message Protocol • Values: – 0x00 - MESSAGE_PROTOCOL_CDMA – 0x01 - MESSAGE_PROTOCOL_WCDMA

8.711.2 Field Documentation

8.711.2.1 uint8_t sms_msgProtocolResp::msgProtocol

8.712 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_CDMABroadcastConfig CDMABroadcastConfig [0x05]

8.712.1 Detailed Description

This structure contains the 3GPP2 Broadcast Configuration Information parameters

Parameters

activated_ind	• Broadcast SMS – 0x00 - Deactivated – 0x01 - Activated
num_instances	• Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. – serviceCategory – language – selected
broadcastConfig	• A CDMABroadcastConfig structure array. • Further defined by the structure sms_CDMABroadcastConfig

8.712.2 Field Documentation

8.712.2.1 uint8_t sms_qaQmi3GPP2BroadcastCfgInfo::activated_ind

8.712.2.2 sms_CDMABroadcastConfig sms_qaQmi3GPP2BroadcastCfgInfo::CDMABroadcastConfig[0x05]

8.712.2.3 uint16_t sms_qaQmi3GPP2BroadcastCfgInfo::num_instances

8.713 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_BroadcastConfig broadcastConfig [0x05]

8.713.1 Detailed Description

This structure contains the 3GPP Broadcast Configuration Information parameters

Parameters

activated_ind	• Broadcast SMS – 0x00 - Deactivated – 0x01 - Activated
num_instances	• Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. – fromServiceId – toServiceId – selected
broadcastConfig	• A BroadcastConfig structure array. • Further defined by the structure sms_BroadcastConfig

8.713.2 Field Documentation

8.713.2.1 uint8_t sms_qaQmi3GPPBroadcastCfgInfo::activated_ind

8.713.2.2 sms_BroadcastConfig sms_qaQmi3GPPBroadcastCfgInfo::broadcastConfig[0x05]

8.713.2.3 uint16_t sms_qaQmi3GPPBroadcastCfgInfo::num_instances

8.714 sms_routeEntry Struct Reference

Data Fields

- uint8_t messageType
- uint8_t messageClass

- uint8_t routeStorage
- uint8_t receiptAction

8.714.1 Detailed Description

This structure contains SMS route entry details

Parameters

messageType	- • Message type matching this route • Values: – 0x00 - MESSAGE_TYPE_POINT_TO_POINT
messageClass	- • Message Class • Values: – 0x00 - MESSAGE_CLASS_0 – 0x01 - MESSAGE_CLASS_1 – 0x02 - MESSAGE_CLASS_2 – 0x03 - MESSAGE_CLASS_3 – 0x04 - MESSAGE_CLASS_NONE – 0x05 - MESSAGE_CLASS_CDMA
routeStorage	- • If the receiptAction is store where to store the message • Values: – 0x00 - STORAGE_TYPE_UIM – 0x01 - STORAGE_TYPE_NV – 0xFF - STORAGE_TYPE_NONE
receiptAction	- • Action to be taken on receipt of a message matching the specified type and class for this route • Values: – 0x00 - DISCARD (discarded without notification) – 0x01 - STORE AND NOTIFY (stored and notified to the registered clients) – 0x02 - TRANSFER ONLY (transferred to the client, client expected to send the ACK) – 0x03 - TRANSFER AND ACK (transferred to the client, device expected to send the ACK)

8.714.2 Field Documentation

8.714.2.1 uint8_t sms_routeEntry::messageClass

8.714.2.2 uint8_t sms_routeEntry::messageType

8.714.2.3 uint8_t sms_routeEntry::receiptAction

8.714.2.4 uint8_t sms_routeEntry::routeStorage

8.715 sms_sendAsyncsmsParams Struct Reference

Data Fields

- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage
- uint8_t * pForceOnDC
- uint8_t * pServiceOption
- uint8_t * pFollowOnDC
- uint8_t * pLinktimer
- uint8_t * pSmsOnIms
- uint8_t * pRetryMessage
- uint32_t * pRetryMessageId
- uint32_t * pUserData

8.715.1 Detailed Description

This structure contains SMS parameters

Parameters

messageFormat	• Message format • Values: – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
messageSize	• The length of the message contents in bytes
pMessage	• The message contents
pForceOnDC	• Force the message to be sent on the CDMA dedicated channel. • Values: – 0x00 - Do not care about the channel on which the message is sent – 0x01 - Request to send the message over the dedicated channel
pServiceOption	• Service option: • Values: – 0x00 - SO_AUTO - AUTO (choose the best service option) – 0x06 - SO_6 - Service option 6 – 0x0E - SO_14 - Service option 14
pFollowOnDC	• Flag to request not to disconnect the CDMA dedicated channel after the send operation is complete. • This TLV can be included if more messages are expected to follow. • Values: – 0x01 - FOLLOW_ON_DC_ON - On (don't disconnect after send operation) Any value other than 0x01 is treated as an absence of this TLV.

pLinktimer	• Keeps the GW SMS link open for the specified number of seconds; can be enabled if more messages are expected to follow
pSmsOnIms	• Indicates whether the message is to be sent on IMS. • Values: – 0x00 - Message is not to be sent on IMS – 0x01 - Message is to be sent on IMS – 0x02 to 0xFF - Reserved
pRetryMessage	• Indicates this message is a retry message. • Values: – 0x01 - WMS_MESSAGE_IS_A_RETRY - Message is a retry message Note: Any value other than 0x01 in this field is treated as an absence of this TLV.
pRetryMessage-Id	• Message ID to be used in the retry message. • The message ID specified here is used instead of the messsage ID encoded in the raw message.
pUserData	• Enables the control point to associate the request with the corresponding indication. • The control point might send numerous requests. • This TLV will help the control point to identify the request for which the received indication belongs.

8.715.2 Field Documentation

8.715.2.1 uint32_t sms_sendAsyncsmsParams::messageFormat

8.715.2.2 uint32_t sms_sendAsyncsmsParams::messageSize

8.715.2.3 uint8_t* sms_sendAsyncsmsParams::pFollowOnDC

8.715.2.4 uint8_t* sms_sendAsyncsmsParams::pForceOnDC

8.715.2.5 uint8_t* sms_sendAsyncsmsParams::pLinktimer

8.715.2.6 uint8_t* sms_sendAsyncsmsParams::pMessage

8.715.2.7 uint8_t* sms_sendAsyncsmsParams::pRetryMessage

8.715.2.8 uint32_t* sms_sendAsyncsmsParams::pRetryMessageId

8.715.2.9 uint8_t* sms_sendAsyncsmsParams::pServiceOption

8.715.2.10 uint8_t* sms_sendAsyncsmsParams::pSmsOnIms

8.715.2.11 uint32_t* sms_sendAsyncsmsParams::pUserData

8.716 sms_setIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.716.1 Detailed Description

This structure contains Indication Register request parameters

Parameters

pRegTransLayerInfoEvt	- • Optional parameter indicating registration status of transport layer information events • Values: – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
pRegTransNWRegInfoEvt	- • Optional parameter indicating registration status of transport network registration information events • Values: – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
pRegCallStatInfoEvt	- • Optional parameter indicating registration status of call status information events • Values: – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device

8.716.2 Field Documentation

8.716.2.1 uint8_t* sms_setIndicationReg::pRegCallStatInfoEvt

8.716.2.2 uint8_t* sms_setIndicationReg::pRegTransLayerInfoEvt

8.716.2.3 uint8_t* sms_setIndicationReg::pRegTransNWRegInfoEvt

8.717 sms_setRoutesReq Struct Reference

Data Fields

- uint16_t numOfRoutes

- sms_routeEntry routeList [0x0A]
- uint8_t * pTransferStatusReport

8.717.1 Detailed Description

This structure contains SMS route request parameters

Parameters

numOfRoutes	- • Number of sets of the following element
routeList	- • Array containing the set of sms_routeEntry
pTransferStatus-Report	- • 0x01 - Status report are transferred to the client (optional)

8.717.2 Field Documentation

8.717.2.1 uint16_t sms_setRoutesReq::numOfRoutes

8.717.2.2 uint8_t* sms_setRoutesReq::pTransferStatusReport

8.717.2.3 sms_routeEntry sms_setRoutesReq::routeList[0x0A]

8.718 sms_transLayerInfo Struct Reference

Data Fields

- uint8_t TransType
- uint8_t TransCap

8.718.1 Detailed Description

This structure contains Transport Layer Information

Parameters

TransType	• Transport Type – 0x00 - IMS
TransCap	• Transport Capability • Values: – 0x00 - CDMA – 0x01 - GW

8.718.2 Field Documentation

8.718.2.1 uint8_t sms_transLayerInfo::TransCap

8.718.2.2 uint8_t sms_transLayerInfo::TransType

8.719 sMSCAddressInfo Struct Reference

Data Fields

- uint8_t length
- uint8_t data [256]

8.719.1 Detailed Description

This structure contains SMSC information

Parameters

length	• Number of sets of following element
data	• SMSC address

8.719.2 Field Documentation

8.719.2.1 uint8_t sMSCAddressInfo::data[256]

8.719.2.2 uint8_t sMSCAddressInfo::length

8.720 sMSCAddressTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSCAddressInfo SMSCInfo

8.720.1 Detailed Description

This structure contains SMSC TLV information

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
SMSCInfo	• SMSC Address • See sMSCAddressInfo for more information

8.720.2 Field Documentation

8.720.2.1 sMSCAddressInfo sMSCAddressTlv::SMSCInfo

8.720.2.2 uint8_t sMSCAddressTlv::TlvPresent

8.721 sMSEtwsMessageInfo Struct Reference

Data Fields

- uint8_t notificationType
- uint16_t length
- uint8_t data [1254]

8.721.1 Detailed Description

This structure contains ETWS information

Parameters

notificationType	• Message mode 0x00 - Primary 0x01 - Secondary GSM 0x02 - Secondary UMTS
length	• Number of sets of following elements
data	• Raw message data

8.721.2 Field Documentation

8.721.2.1 uint8_t sMSEtwsMessageInfo::data[1254]

8.721.2.2 uint16_t sMSEtwsMessageInfo::length

8.721.2.3 uint8_t sMSEtwsMessageInfo::notificationType

8.722 sMSEtwsMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsMessageInfo EtwsMessageInfo

8.722.1 Detailed Description

This structure contains ETWS TLV information

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response

EtwsMessage-Info	• ETWS Message • See sMSEtwsMessageInfo for more information

8.722.2 Field Documentation

8.722.2.1 sMSEtwsMessageInfo sMSEtwsMessageTlv::EtwsMessageInfo

8.722.2.2 uint8_t sMSEtwsMessageTlv::TlvPresent

8.723 sMSEtwsPlmnInfo Struct Reference

Data Fields

- uint16_t mobileCountryCode
- uint16_t mobileNetworkCode

8.723.1 Detailed Description

This structure contains ETWS PLMN information

Parameters

mobileCountry-Code	• 16 bit representation of MCC value range : 0 -999
mobileNetwork-Code	• 16 bit representation of MNC value range : 0 -999

8.723.2 Field Documentation

8.723.2.1 uint16_t sMSEtwsPlmnInfo::mobileCountryCode

8.723.2.2 uint16_t sMSEtwsPlmnInfo::mobileNetworkCode

8.724 sMSMessageModeInfo Struct Reference

Data Fields

- uint8_t messageMode

8.724.1 Detailed Description

This structure contains message mode information.

Parameters

messageMode	• Message Mode – 0x00 - CDMA – 0x01 - GW

8.724.2 Field Documentation

8.724.2.1 uint8_t sMSMessageModeInfo::messageMode

8.725 sMSMTMessageInfo Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex

8.725.1 Detailed Description

This structure contains MT message information.

Parameters

storageType	memory storage 0x00-UIM 0x01-NV
messageIndex	MT Message index

8.725.2 Field Documentation

8.725.2.1 uint32_t sMSMTMessageInfo::messageIndex

8.725.2.2 uint32_t sMSMTMessageInfo::storageType

8.726 sMSOnIMSInfo Struct Reference

Data Fields

- uint8_t smsOnIMS

8.726.1 Detailed Description

This structure contains SMS on IMS information

Parameters

smsOnIMS	SMS on IMS

8.726.2 Field Documentation

8.726.2.1 uint8_t sMSOnIMSInfo::smsOnIMS

8.727 sMSOnIMSTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSOnIMSInfo IMSInfo

8.727.1 Detailed Description

This structure contains SMS on IMS TLV information.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
IMSInfo	• SMS on IMS • See sMSOnIMSInfo for more information

8.727.2 Field Documentation

8.727.2.1 sMSOnIMSInfo sMSOnIMSTlv::IMSInfo

8.727.2.2 uint8_t sMSOnIMSTlv::TlvPresent

8.728 sMSTransferRouteMTMessageInfo Struct Reference

Data Fields

- uint8_t ackIndicator
- uint32_t transactionID
- uint8_t format
- uint16_t length
- uint8_t data [256]

8.728.1 Detailed Description

This structure contains Transfer route MT message information.

Parameters

ackIndicator	• Parameter to indicate if ACK must be sent by the control point 0x00 - Send ACK 0x01 - Do not send ACK
transactionID	• Transaction ID of the message
format	• Message format 0x00 - CDMA 0x02 - 0x05 - Reserved 0x06 - GW_PP 0x07 - GW_BC

length	• Length of the raw message. This length should not exceed the maximum WMS payload length of 256 bytes
data	• Raw message data

8.728.2 Field Documentation

8.728.2.1 uint8_t sMSTransferRouteMTMessageInfo::ackIndicator

8.728.2.2 uint8_t sMSTransferRouteMTMessageInfo::data[256]

8.728.2.3 uint8_t sMSTransferRouteMTMessageInfo::format

8.728.2.4 uint16_t sMSTransferRouteMTMessageInfo::length

8.728.2.5 uint32_t sMSTransferRouteMTMessageInfo::transactionID

8.729 swi_uint256_t Struct Reference

Data Fields

- uint16_t word [16]

8.729.1 Detailed Description

SWI 256 bit data type

8.729.2 Field Documentation

8.729.2.1 uint16_t swi_uint256_t::word[16]

8.730 swiaudio_PCMparams Struct Reference

Data Fields

- uint8_t iFaceTabLen
- uint8_t iFaceTab [255]

8.730.1 Detailed Description

This structure contains the PCM parameters.

Parameters

iFaceTabLen	• Number of sets of iface table

iFaceTab	• Physical Interface Parameters • See qaGobiApiTableSwiAudio.h for more information on physical interface parameters

8.730.2 Field Documentation

8.730.2.1 uint8_t swiaudio_PCMparams::iFaceTab[255]

8.730.2.2 uint8_t swiaudio_PCMparams::iFaceTabLen

8.731 swidms_AutoUimSwitchModeTlv Struct Reference

Data Fields

- uint8_t uimSlot
- uint8_t TlvPresent

8.731.1 Detailed Description

This structure contains swidms_UimSlotSelected Tlv the Get UIM Slot Response.

Parameters

autoUimMode	• Values – 0 - Auto switch off – 1 - Auto switch UIM1 prefer – 2 - Auto switch UIM2 prefer
TlvPresent	• Tlv Present.

8.731.2 Field Documentation

8.731.2.1 uint8_t swidms_AutoUimSwitchModeTlv::TlvPresent

8.731.2.2 uint8_t swidms_AutoUimSwitchModeTlv::uimSlot

8.732 swidms_ehrpdMTUSizeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ehrpdMTUSize

8.732.1 Detailed Description

This structure contains the parameters for EHRPD MTU Size.

Parameters

TlvPresent	• Tlv Present.
ehrpdMTUSize	EHRPD size • Size of EHRPD MTU

8.732.2 Field Documentation

8.732.2.1 uint16_t swidms_ehrpdMTUSizeTlv::ehrpdMTUSize

8.732.2.2 uint8_t swidms_ehrpdMTUSizeTlv::TlvPresent

8.733 swidms_hrpdMTUSizeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t hrpdMTUSize

8.733.1 Detailed Description

This structure contains the parameters for HRPD MTU Size.

Parameters

TlvPresent	• Tlv Present.
hrpdMTUSize	HRPD size • Size of HRPD MTU

8.733.2 Field Documentation

8.733.2.1 uint16_t swidms_hrpdMTUSizeTlv::hrpdMTUSize

8.733.2.2 uint8_t swidms_hrpdMTUSizeTlv::TlvPresent

8.734 swidms_intfaceCfgTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t CurrentCfgType
- uint32_t CfgValue

8.734.1 Detailed Description

This structure contains the parameters for host usb interface composition

Parameters

TlvPresent	• Tlv Present.
CurrentCfgType	• Current configure type, see values below: – 1 - MBIM – 2 - USBIF + PCIE – 3 - RMNET – 4 - RNDIS – 5 - PCIE only
CfgValue	• Host composition bit mask, see supported USB interface bitmasks, not supported by QMI object setting for 9x50 modules – 0x00000001 - DIAG interface – 0x00000002 - ADB interface – 0x00000004 - NMEA interface – 0x00000008 - MODEM interface – 0x00000010 - RESERVED5 – 0x00000020 - RESERVED6 – 0x00000040 - RESERVED7 – 0x00000080 - RESERVED8 – 0x00000100 - RMENT0 interface – 0x00000200 - RESERVED10 – 0x00000400 - RMENT1 interface – 0x00000800 - RESERVED12 – 0x00001000 - MBIM interface – 0x00002000 - RESERVED14 – 0x00004000 - RNDIS interface – 0x00008000 - RESERVED16 – 0x00010000 - AUDIO interface – 0x00020000 - RESERVED18 – 0x00080000 - ECM interface – 0x00100000 - RESERVED21 – 0x00200000 - RESERVED22 – 0xFFC00000 - RESERVED

8.734.2 Field Documentation

8.734.2.1 uint32_t swidms_intfaceCfgTlv::CfgValue

8.734.2.2 uint32_t swidms_intfaceCfgTlv::CurrentCfgType

8.734.2.3 uint8_t swidms_intfaceCfgTlv::TlvPresent

8.735 swidms_mtuSize3gppTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t MTUSize3gpp

8.735.1 Detailed Description

This structure contains the parameters for 3GPP MTU Size.

Parameters

TlvPresent	• Tlv Present.
MTUSize3gpp	3GPP MTU size • Size of 3GPP MTU

8.735.2 Field Documentation

8.735.2.1 uint16_t swidms_mtuSize3gppTlv::MTUSize3gpp

8.735.2.2 uint8_t swidms_mtuSize3gppTlv::TlvPresent

8.736 swidms_MultiplexedSimModeTlv Struct Reference

Data Fields

- uint8_t uimSlot
- uint8_t mode
- uint8_t TlvPresent

8.736.1 Detailed Description

This structure contains swidms_MultiplexedSimModeTlv Tlv the Get UIM Slot Response.

Parameters

uimSlot	• Values – 0 - Multiplexed SIM via UIM1 – 1 - Reserved via UIM2
mode	• Values – 0 - No Multiplexed SIM – 1 - Multiplexed SIM1 selected – 2 - Multiplexed SIM2 selected
TlvPresent	• Tlv Present.

8.736.2 Field Documentation

8.736.2.1 uint8_t swidms_MultiplexedSimModeTlv::mode

8.736.2.2 uint8_t swidms_MultiplexedSimModeTlv::TlvPresent

8.736.2.3 uint8_t swidms_MultiplexedSimModeTlv::uimSlot

8.737 swidms_SetMultiplexedModeTlv Struct Reference

Data Fields

- uint8_t uimSlot
- uint8_t mode

8.737.1 Detailed Description

This structure contains swidms_SetMultiplexedModeTlv Tlv the Set UIM Slot Request.

Parameters

uimSlot	• Values – 0 - Multiplexed SIM via UIM1 – 1 - Reserved via UIM2
mode	• Values – 0 - No Multiplexed SIM – 1 - Multiplexed SIM1 selected – 2 - Multiplexed SIM2 selected

8.737.2 Field Documentation

8.737.2.1 uint8_t swidms_SetMultiplexedModeTlv::mode

8.737.2.2 uint8_t swidms_SetMultiplexedModeTlv::uimSlot

8.738 swidms_supportedIntBitmaskTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ValidBitmasks

8.738.1 Detailed Description

This structure contains the parameters for supported usb bitmasks

Parameters

TlvPresent	• Tlv Present.
ValidBitmasks	• for current configuration type, this is the supported bitmasks that the host can change.

8.738.2 Field Documentation

8.738.2.1 uint8_t swidms_supportedIntBitmaskTlv::TlvPresent

8.738.2.2 uint32_t swidms_supportedIntBitmaskTlv::ValidBitmasks

8.739 swidms_SwiDmsGetHWWatchdog Struct Reference

Data Fields

- uint32_t timeout
- uint32_t resetDelay
- uint8_t enable
- uint32_t count

8.739.1 Detailed Description

This structure contains the parameters for hardware watchdog settings

Parameters

timeout	• timeout value for HW watchdog (unit in second)
resetDelay	• delay before reset after watchdog timeout (unit in second)
enable	• 0 to disable watchdog; 1 to enable watchdog
count	• once <timeout> has occurred, the <count> will be increased by 1 and the timer will be restarted automatically. This <count> indicates the number of renewals.

8.739.2 Field Documentation

8.739.2.1 uint32_t swidms_SwiDmsGetHWWatchdog::count

8.739.2.2 uint8_t swidms_SwiDmsGetHWWatchdog::enable

8.739.2.3 uint32_t swidms_SwiDmsGetHWWatchdog::resetDelay

8.739.2.4 uint32_t swidms_SwiDmsGetHWWatchdog::timeout

8.740 swidms_UimSlotSelectedTlv Struct Reference

Data Fields

- uint8_t uimSlot
- uint8_t TlvPresent

8.740.1 Detailed Description

This structure contains swidms_UimSlotSelected Tlv the Get UIM Slot Response.

Parameters

uimSlot	• UIM slot selected • Values – 0 - UIM1 – 1 - UIM2 – 2 - Reserved (Remote SIM)
TlvPresent	• Tlv Present.

8.740.2 Field Documentation

8.740.2.1 uint8_t swidms_UimSlotSelectedTlv::TlvPresent

8.740.2.2 uint8_t swidms_UimSlotSelectedTlv::uimSlot

8.741 swidms_usbMTUSizeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t UsbMTUSize

8.741.1 Detailed Description

This structure contains the parameters for USB MTU Size

Parameters

TlvPresent	• Tlv Present.
UsbMTUSize	USB MTU size • Size of USB MTU

8.741.2 Field Documentation

8.741.2.1 uint8_t swidms_usbMTUSizeTlv::TlvPresent

8.741.2.2 uint16_t swidms_usbMTUSizeTlv::UsbMTUSize

8.742 tdscdmaSigInfoExt Struct Reference

Data Fields

- float rssi
- float rscp
- float ecio
- float sinr

8.742.1 Detailed Description

This structure contains the TDSCDMA Signal Strength Info Extended

Parameters

rssi	• Measured RSSI in dB
rscp[Optional]	• Measured RSCP in dBm
ecio[Optional]	• Measured ECIO in dBm.
sinr[Optional]	• Measured SINR in dB. -15 dB is sent to clients if the actual SINR is less than -15 dB

8.742.2 Field Documentation

8.742.2.1 float tdscdmaSigInfoExt::ecio

8.742.2.2 float tdscdmaSigInfoExt::rscp

8.742.2.3 float tdscdmaSigInfoExt::rssi

8.742.2.4 float tdscdmaSigInfoExt::sinr

8.743 tempData_t Struct Reference

Data Fields

- uint32_t timeSource
- uint32_t timeOfFirstSample
- uint8_t temperatureDataLen
- uint16_t timeOffset [64]
- uint32_t temperature [64]

8.743.1 Detailed Description

This structure specifies information regarding the Temperature Data. Please check has_<Param_Name> field for presence of optional parameters

Parameters

timeSource	• Time source of the sensor data • Valid values – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
timeOfFirst-Sample	• Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds
temperature-DataLen	• Number of sets of the following elements – timeOffset – temperature
timeOffset	• Sample time offset • Units - Milliseconds
temperature	• Sensor temperature. • Type - Floating point • Units - Degrees Celsius • Range -50 to +100.00

8.743.2 Field Documentation

8.743.2.1 uint32_t tempData_t::temperature[64]

8.743.2.2 uint8_t tempData_t::temperatureDataLen

8.743.2.3 uint32_t tempData_t::timeOfFirstSample

8.743.2.4 uint16_t tempData_t::timeOffset[64]

8.743.2.5 uint32_t tempData_t::timeSource

8.744 tmd_mitigationDevList Struct Reference

Data Fields

- uint8_t mitigationDevIdLen
- char mitigationDevId [255]
- uint8_t maxMitigationLevel

8.744.1 Detailed Description

This structure contains mitigation devices list

Parameters

mitigationDevId-Len	• Number of sets of the following elements – mitigation_dev_id
mitigationDevId	• Mitigation device ID.
maxMitigation-Level	• Maximum valid mitigation level. • Valid range - 0 to max_mitigation_level.

8.744.2 Field Documentation

8.744.2.1 uint8_t tmd_mitigationDevList::maxMitigationLevel

8.744.2.2 char tmd_mitigationDevList::mitigationDevId[255]

8.744.2.3 uint8_t tmd_mitigationDevList::mitigationDevIdLen

8.745 transferRouteMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSTransferRouteMTMessageInfo TransferRouteMTMessageInfo

8.745.1 Detailed Description

This structure contains Transfer route MT message TLV information.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
TransferRouteM-TMessageInfo	• Transfer Route MT Message • See sMSTransferRouteMTMessageInfo for more information

8.745.2 Field Documentation

8.745.2.1 uint8_t transferRouteMessageTlv::TlvPresent

8.745.2.2 sMSTransferRouteMTMessageInfo transferRouteMessageTlv::TransferRouteMTMessageInfo

8.746 uim_activeProfileInfo Struct Reference

Data Fields

- uint32_t profileId
- uint32_t profileType
- uint8_t TlvPresent

8.746.1 Detailed Description

This structure contains the Active Profile Information

Parameters

profileId	• Indicates the profile ID of the card. • Valid values: – 0x01 - Profile 1 – 0x02 - Profile 2 – 0x03 - Profile 3 – 0x04 - Profile 4 – 0x05 - Profile 5 – 0x06 - Profile 6 – 0x07 - Profile 7 – 0x08 - Profile 8
profileId	• Indicates the profile type of the card. • Valid values: – 0x00 - Regular – 0x01 - Emergency
TlvPresent	• Tlv Present.

8.746.2 Field Documentation

8.746.2.1 uint32_t uim_activeProfileInfo::profileId

8.746.2.2 uint32_t uim_activeProfileInfo::profileType

8.746.2.3 uint8_t uim_activeProfileInfo::TlvPresent

8.747 uim_additionalReadResult Struct Reference

Data Fields

- uint16_t additionalRecordLen
- uint8_t additionalRecord [255]
- uint8_t TlvPresent

8.747.1 Detailed Description

This structure contains the information for additional result.

Parameters

additionalRecordLen	• Number of sets of additionalRecord.
additionalRecord[MAX_DESCRIPTION_LENGTH]	• Read content of all the additional records. Each record has the same size as the first record.
TlvPresent	• Tlv Present.

8.747.2 Field Documentation

8.747.2.1 uint8_t uim_additionalReadResult::additionalRecord[255]

8.747.2.2 uint16_t uim_additionalReadResult::additionalRecordLen

8.747.2.3 uint8_t uim_additionalReadResult::TlvPresent

8.748 uim_appStatus Struct Reference

Data Fields

- uint8_t appType
- uint8_t appState
- uint8_t persoState
- uint8_t persoFeature
- uint8_t persoRetries
- uint8_t persoUnblockRetries
- uint8_t aidLength
- uint8_t aidVal [255]
- uint8_t univPin
- uint8_t pin1State
- uint8_t pin1Retries
- uint8_t puk1Retries
- uint8_t pin2State
- uint8_t pin2Retries
- uint8_t puk2Retries

8.748.1 Detailed Description

This structure contains Application Status Information loaded on the card.

Parameters

appType	• Indicates the type of the application. – 0 - Unknown – 1 - SIM card – 2 - USIM application – 3 - RUIM card – 4 - CSIM application – 5 - ISIM application • Other values are reserved for the future and are to be handled as “Unknown”.
appState	• Indicates the state of the application. – 0 - Unknown – 1 - Detected – 2 - PIN1 or UPIN is required – 3 - PUK1 or PUK for UPIN is required – 4 - Personalization state must be checked – 5 - PIN1 is blocked – 6 - Illegal – 7 - Ready
persoState	• Indicates the state of the personalization for the application. – 0 - Unknown – 1 - Personalization operation is in progress – 2 - Ready – 3 - Personalization code is required – 4 - PUK for personalization code is required – 5 - Permanently blocked
persoFeature	• Indicates the personalization feature. • This applies only when a personalization code is required to deactivate or unblock personalization. – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization – 11 - Unknown
persoRetries	• Indicates the number of retries remaining to disable the personalization.

persoUnblock-Retries	• Indicates the number of retries remaining to unblock the personalization.
aidLength	• Number of sets of the following elements. i.e. aidVal • If zero(0) then no aidVal information exists.
aidVal[MAX_DE-SCRIPTION_LE-NGTH]	• Application identifier value.
univPin	• Indicates whether UPIN replaces PIN1. – 0 - PIN1 is used – 1 - UPIN replaces PIN1
pin1State	• Indicates the state of PIN1. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin1Retries	• Indicates the number of retries remaining to verify PIN1.
puk1Retries	• Indicates the number of retries remaining to unblock PIN1.
pin2State	• Indicates the state of PIN2. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin2Retries	• Indicates the number of retries remaining to verify PIN2.
puk2Retries	• Indicates the number of retries remaining to unblock PIN2.

8.748.2 Field Documentation

8.748.2.1 uint8_t uim_appStatus::aidLength

8.748.2.2 uint8_t uim_appStatus::aidVal[255]

8.748.2.3 uint8_t uim_appStatus::appState

8.748.2.4 uint8_t uim_appStatus::appType

8.748.2.5 uint8_t uim_appStatus::persoFeature

8.748.2.6 uint8_t uim_appStatus::persoRetries

8.748.2.7 uint8_t uim_appStatus::persoState

8.748.2.8 uint8_t uim_appStatus::persoUnblockRetries

8.748.2.9 uint8_t uim_appStatus::pin1Retries

8.748.2.10 uint8_t uim_appStatus::pin1State

8.748.2.11 uint8_t uim_appStatus::pin2Retries

8.748.2.12 uint8_t uim_appStatus::pin2State

8.748.2.13 uint8_t uim_appStatus::puk1Retries

8.748.2.14 uint8_t uim_appStatus::puk2Retries

8.748.2.15 uint8_t uim_appStatus::univPin

8.749 uim_authenticateResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [1024]

8.749.1 Detailed Description

This structure contains the information about the authenticate result.

Parameters

contentLen	• Length of the following elements i.e. content.
content[UIM_MAX_CONTENT_LENGTH]	• Authenticate data. • This value is a sequence of bytes returned from the card.

8.749.2 Field Documentation

8.749.2.1 uint8_t uim_authenticateResult::content[1024]

8.749.2.2 uint16_t uim_authenticateResult::contentLen

8.750 uim_authenticationData Struct Reference

Data Fields

- uint8_t context
- uint16_t dataLen
- uint8_t data [1024]

8.750.1 Detailed Description

This structure contains the Session Information.

Parameters

context	• Authenticate context. – 0 - Runs the GSM algorithm (valid only on a 2GSIM card) – 1 - Runs the CAVE algorithm (valid only on a RUIM card) – 2 - GSM security context (valid only on a USIM application) – 3 - 3G security context (valid only on a USIM application) – 4 - VGCS/VBS security context (valid only on aUSIM application) – 5 - GBA security context, Bootstrapping mode (valid only on a USIM or ISIM application) – 6 - GBA security context, NAF Derivation mode (valid only on a USIM or ISIM application) – 7 - MBMS security context, MSK Update mode (valid only on a USIM application) – 8 - MBMS security context, MTK Generation mode (valid only on a USIM application) – 9 - MBMS security context, MSK Deletion mode (valid only on a USIM application) – 10 - MBMS security context, MUK Deletion mode (valid only on a USIM application) – 11 - IMS AKA security context (valid only on aISIM application) – 12 - HTTP-digest security context (valid only onan ISIM application) – 13 - Compute IP authentication, CHAP (valid onlyon RUIM or CSIM) – 14 - Compute IP authentication, MN-HA authenticator (valid only on RUIM or CSIM) – 15 - Compute IP authentication, MIP-RRQ hash (valid only on RUIM or CSIM) – 16 - Compute IP authentication, MN-AAA authenticator (valid only on RUIM or CSIM) – 17 - Compute IP authentication, HRPD access authenticator (valid only on RUIM or CSIM) • Other values are possible and reserved for future use.
dataLen	• Length of the following elements i.e. data.
data[UIM_MAX-_DESCRIPTIO-N_LENGTH]	• Authenticate Data.

8.750.2 Field Documentation

8.750.2.1 uint8_t uim_authenticationData::context

8.750.2.2 uint8_t uim_authenticationData::data[1024]

8.750.2.3 uint16_t uim_authenticationData::dataLen

8.751 uim_cardResult Struct Reference

Data Fields

- uint8_t sw1
- uint8_t sw2

8.751.1 Detailed Description

This structure contains the information about the card result.

Parameters

sw1	• SW1 received from the card.
sw2	• SW2 received from the card.

8.751.2 Field Documentation

8.751.2.1 uint8_t uim_cardResult::sw1

8.751.2.2 uint8_t uim_cardResult::sw2

8.752 uim_cardResultInfo Struct Reference

Data Fields

- uint8_t sw1
- uint8_t sw2
- uint8_t TlvPresent

8.752.1 Detailed Description

This structure contains the information about the card result.

Parameters

sw1	• SW1 received from the card.
sw2	• SW2 received from the card.
TlvPresent	• Tlv Present.

8.752.2 Field Documentation

8.752.2.1 uint8_t uim_cardResultInfo::sw1

8.752.2.2 uint8_t uim_cardResultInfo::sw2

8.752.2.3 uint8_t uim_cardResultInfo::TlvPresent

8.753 uim_cardStatus Struct Reference

Data Fields

- uint16_t indexGwPri
- uint16_t index1xPri
- uint16_t indexGwSec
- uint16_t index1xSec
- uint8_t numSlot
- uim_slotInfo SlotInfo [5]

8.753.1 Detailed Description

This structure contains Card Status Information.

Parameters

indexGwPri	• Index of the primary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
index1xPri	• Index of the primary 1X provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
indexGwSec	• Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
index1xSec	• Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
numSlot	• Indicates the number of slots available on the device. • The following block is repeated for each slot. i.e. cardState • If zero(0) then no cardState information exists.
SlotInfo	• See uim_slotInfo for more information.

8.753.2 Field Documentation

8.753.2.1 uint16_t uim_cardStatus::index1xPri

8.753.2.2 uint16_t uim_cardStatus::index1xSec

8.753.2.3 uint16_t uim_cardStatus::indexGwPri

8.753.2.4 uint16_t uim_cardStatus::indexGwSec

8.753.2.5 uint8_t uim_cardStatus::numSlot

8.753.2.6 uim_slotInfo uim_cardStatus::SlotInfo[5]

8.754 uim_changeUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t oldPINLen
- uint8_t oldPINVal [255]
- uint8_t pinLen
- uint8_t pinVal [255]

8.754.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

pinID	• Indicates the PIN ID to be changed. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN – 4 - Hidden key
oldPINLen	• Length of the following elements i.e. old pin value.
oldPINVal[MAX_DESCRIPTION_LENGTH]	• Old PIN value. • This value is a sequence of ASCII characters.
pinLen	• Length of the following elements i.e. new pin value.
pinVal[MAX_DESCRIPTION_LENGTH]	• New PIN value. • This value is a sequence of ASCII characters.

8.754.2 Field Documentation

8.754.2.1 uint8_t uim_changeUIMPIN::oldPINLen

8.754.2.2 uint8_t uim_changeUIMPIN::oldPINVal[255]

8.754.2.3 uint8_t uim_changeUIMPIN::pinID

8.754.2.4 uint8_t uim_changeUIMPIN::pinLen

8.754.2.5 uint8_t uim_changeUIMPIN::pinVal[255]

8.755 uim_depersonalizationInformation Struct Reference

Data Fields

- uint8_t feature
- uint8_t operation
- uint8_t ckLen
- uint8_t ckVal [255]

8.755.1 Detailed Description

This structure contains the Depersonalization Information.

Parameters

feature	• Indicates the personalization feature to de-activate or unblock. – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization
operation	• Indicates the operation to perform. – 0 - Deactivate personalization. – 1 - Unblock personalization.
ckLen	• Length of the following elements i.e. control key value.
ckVal[MAX_DESCRIPTION_LENGTH]	• Control key value. • This value is a sequence of ASCII characters.

8.755.2 Field Documentation

8.755.2.1 uint8_t uim_depersonalizationInformation::ckLen

8.755.2.2 uint8_t uim_depersonalizationInformation::ckVal[255]

8.755.2.3 uint8_t uim_depersonalizationInformation::feature

8.755.2.4 uint8_t uim_depersonalizationInformation::operation

8.756 uim_eidInfo Struct Reference

Data Fields

- uint8_t eidValLen
- uint8_t eidVal [255]
- uint8_t TlvPresent

8.756.1 Detailed Description

This structure contains the EID informaton

Parameters

eidValLen	• Number of sets of the following elements • eid_value
eidVal	• eUICC ID value
TlvPresent	• TLV presents

8.756.2 Field Documentation

8.756.2.1 uint8_t uim_eidInfo::eidVal[255]

8.756.2.2 uint8_t uim_eidInfo::eidValLen

8.756.2.3 uint8_t uim_eidInfo::TlvPresent

8.757 uim_encryptedPIN1 Struct Reference

Data Fields

- uint8_t pin1Len
- uint8_t pin1Val [255]

8.757.1 Detailed Description

This structure contains the encrypted PIN1 Information.

Parameters

pin1Len	• Number of sets of the following elements ie encrypted PIN1 value. • If zero(0), no information follows.
pin1Val	• Encrypted PIN1 value.

Note

This value is returned only when PIN1 is enabled successfully and the feature is supported.

8.757.2 Field Documentation

8.757.2.1 uint8_t uim_encryptedPIN1::pin1Len

8.757.2.2 uint8_t uim_encryptedPIN1::pin1Val[255]

8.758 uim_fileAttributes Struct Reference

Data Fields

- uint16_t fileSize
- uint16_t fileID
- uint8_t fileType
- uint16_t recordSize
- uint16_t recordCount
- uint8_t secRead
- uint16_t secReadMask
- uint8_t secWrite
- uint16_t secWriteMask
- uint8_t secIncrease
- uint16_t secIncreaseMask
- uint8_t secDeactivate
- uint16_t secDeactivateMask
- uint8_t secActivate
- uint16_t secActivateMask
- uint16_t rawLen
- uint8_t rawValue [255]

8.758.1 Detailed Description

This structure contains the information about the File Attributes.

Parameters

fileSize	• Indicates the size of the file.
fileID	• Indicates the ID of the file.

fileType	• Indicates the type of the file. – 0 - Transparent – 1 - Cyclic – 2 - Linear fixed – 3 - Dedicated file – 4 - Master file
recordSize	• Indicates the size of the records. • Only for cyclic and linear fixed files
recordCount	• Indicates the total no. of the records. • Only for linear fixed files
secRead	• Read security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secReadMask	• Mask with read security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
secWrite	• Write security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secWriteMask	• Mask with write security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM

secIncrease	• Increase security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secIncrease-Mask	• Mask with increase security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
secDeactivate	• Deactivate security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secDeactivate-Mask	• Mask with deactivate security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
secActivate	• Activate security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secActivateMask	• Mask with activate security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
rawLen	• Length of the following elements i.e. raw value.

rawValue[MAX_-DESCRIPTION-_LENGTH]	• Raw value of file attributes.

8.758.2 Field Documentation

8.758.2.1 uint16_t uim_fileAttributes::fileID

8.758.2.2 uint16_t uim_fileAttributes::fileSize

8.758.2.3 uint8_t uim_fileAttributes::fileType

8.758.2.4 uint16_t uim_fileAttributes::rawLen

8.758.2.5 uint8_t uim_fileAttributes::rawValue[255]

8.758.2.6 uint16_t uim_fileAttributes::recordCount

8.758.2.7 uint16_t uim_fileAttributes::recordSize

8.758.2.8 uint8_t uim_fileAttributes::secActivate

8.758.2.9 uint16_t uim_fileAttributes::secActivateMask

8.758.2.10 uint8_t uim_fileAttributes::secDeactivate

8.758.2.11 uint16_t uim_fileAttributes::secDeactivateMask

8.758.2.12 uint8_t uim_fileAttributes::secIncrease

8.758.2.13 uint16_t uim_fileAttributes::secIncreaseMask

8.758.2.14 uint8_t uim_fileAttributes::secRead

8.758.2.15 uint16_t uim_fileAttributes::secReadMask

8.758.2.16 uint8_t uim_fileAttributes::secWrite

8.758.2.17 uint16_t uim_fileAttributes::secWriteMask

8.759 uim_fileInfo Struct Reference

Data Fields

- uint16_t fileID
- uint8_t pathLen
- uint16_t path [255]

8.759.1 Detailed Description

This structure contains paramaters for file Information

Parameters

fileID	• This is Identifier to SIM files; e.g. in UIM “6F07” is Identifier of IMSI File
pathLen	• Length of file Path
path	• Path value. This value must be the complete path of the file, which is a sequence block of 2 bytes (e.g., 0x3F00 0x7FFF) is for LTE (0x3F00 ,0x7F20) is for GSM.

8.759.2 Field Documentation

8.759.2.1 uint16_t uim_fileInfo::fileID

8.759.2.2 uint16_t uim_fileInfo::path[255]

8.759.2.3 uint8_t uim_fileInfo::pathLen

8.760 uim_getSimProfileInfo Struct Reference

Data Fields

- uint8_t profileInfoLen
- uint32_t profileId [255]
- uint32_t profileType [255]
- uint8_t TlvPresent

8.760.1 Detailed Description

This structure contains the SIM Profile Information.

Parameters

profileInfoLen	• Profile Information Length • Number of sets of the following elements: – profileId – profileType
profileId[MAX_DESCRIPTION_LENGTH]	• Indicates the profile ID of the card. • Valid values: – 0x01 - Profile 1 – 0x02 - Profile 2 – 0x03 - Profile 3 – 0x04 - Profile 4 – 0x05 - Profile 5 – 0x06 - Profile 6 – 0x07 - Profile 7 – 0x08 - Profile 8

profileType[MAX_DESCRIPTION_LENGTH]	• Indicates the profile type of the card. • Valid values: – 0x00 - Regular – 0x01 - Emergency
TlvPresent	• Tlv Present.

8.760.2 Field Documentation

8.760.2.1 uint32_t uim_getSimProfileInfo::profileId[255]

8.760.2.2 uint8_t uim_getSimProfileInfo::profileInfoLen

8.760.2.3 uint32_t uim_getSimProfileInfo::profileType[255]

8.760.2.4 uint8_t uim_getSimProfileInfo::TlvPresent

8.761 uim_GetSlotsInfoTlv Struct Reference

Data Fields

- uint8_t NumberOfPhySlotInfo
- uim_physlotInfo uimSlotInfo [255]
- uint8_t TlvPresent

8.761.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Information TLV.

Parameters

NumberOfPhySlotInfo	• Number of sets of the Slot Information.
uimSlotInfo	• Slots information See uim_physlotInfo for more information.
TlvPresent	• Physical slot information TLV present.

8.761.2 Field Documentation

8.761.2.1 uint8_t uim_GetSlotsInfoTlv::NumberOfPhySlotInfo

8.761.2.2 uint8_t uim_GetSlotsInfoTlv::TlvPresent

8.761.2.3 uim_physlotInfo uim_GetSlotsInfoTlv::uimSlotInfo[255]

8.762 uim_GetSlotsStatusTlv Struct Reference

Data Fields

- uint8_t NumberOfPhySlot
- slot_t uimSlotStatus [255]
- uint8_t TlvPresent

8.762.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status TLV.

Parameters

NumberOfPhySlot	• Number of sets of the Physical Slot Status.
uimSlotStatus	• Slots Status See slot_t for more information.
TlvPresent	• Physical slot status TLV present.

8.762.2 Field Documentation

8.762.2.1 uint8_t uim_GetSlotsStatusTlv::NumberOfPhySlot

8.762.2.2 uint8_t uim_GetSlotsStatusTlv::TlvPresent

8.762.2.3 slot_t uim_GetSlotsStatusTlv::uimSlotStatus[255]

8.763 uim_hotSwapStatus Struct Reference

Data Fields

- uint8_t hotSwapLength
- uint8_t hotSwap [255]

8.763.1 Detailed Description

This structure contains Hot Swap Status Information.

Parameters

hotSwapLength	• Number of sets of the following elements. i.e. hot_swap
hotSwap	• Indicates the status of the hot-swap switch. – 0 - Hot-swap is not supported – 1 - Hot-swap is supported, but the status of the switch is not supported – 2 - Switch indicates that the card is present – 3 - Switch indicates that the card is not present

8.763.2 Field Documentation

8.763.2.1 uint8_t uim_hotSwapStatus::hotSwap[255]

8.763.2.2 uint8_t uim_hotSwapStatus::hotSwapLength

8.764 uim_indToken Struct Reference

Data Fields

- uint32_t token
- uint8_t TlvPresent

8.764.1 Detailed Description

This structure contains the information for indication.

Parameters

pIndicationToken	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.
TlvPresent	• Tlv Present.

8.764.2 Field Documentation

8.764.2.1 uint8_t uim_indToken::TlvPresent

8.764.2.2 uint32_t uim_indToken::token

8.765 uim_personalizationStatus Struct Reference

Data Fields

- uint8_t numFeatures
- uint8_t feature [12]
- uint8_t verifyLeft [12]

- uint8_t unblockLeft [12]

8.765.1 Detailed Description

This structure contains the information about the card result.

Parameters

numFeatures	• Number of active personalization features. The following block is repeated for each feature.
feature	• Indicates the personalization feature to deactivate or unblock. Valid values: – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization
verifyLeft	• Number of the remaining attempts to verify the personalization feature.
unblockLeft	• Number of the remaining attempts to unblock the personalization feature.

8.765.2 Field Documentation

8.765.2.1 uint8_t uim_personalizationStatus::feature[12]

8.765.2.2 uint8_t uim_personalizationStatus::numFeatures

8.765.2.3 uint8_t uim_personalizationStatus::unblockLeft[12]

8.765.2.4 uint8_t uim_personalizationStatus::verifyLeft[12]

8.766 uim_physlotInfo Struct Reference

Data Fields

- uint32_t cardProtocol
- uint8_t numApp
- uint8_t atrValueLen
- uint8_t atrValue [255]
- uint8_t iseUICC

8.766.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

cardProtocol	• Protocol of the card. • Valid values: – 0x00 - Unknown – 0x01 - ICC Protocol – 0x02 - UICC Protocol
numApp	• Number of valid applications present in the EF-DIR of card.
atrValueLen	• Number of sets of the atrValue
atrValue	• Raw value of the ATR sent by the card during the initialization.
iseUICC	• Indicates whether the card is an eUICC card based on the ATR. • Values – 0 - Not an eUICC card – 1 - eUICC card

8.766.2 Field Documentation

8.766.2.1 uint8_t uim_physlotInfo::atrValue[255]

8.766.2.2 uint8_t uim_physlotInfo::atrValueLen

8.766.2.3 uint32_t uim_physlotInfo::cardProtocol

8.766.2.4 uint8_t uim_physlotInfo::iseUICC

8.766.2.5 uint8_t uim_physlotInfo::numApp

8.767 uim_physlotsInfo Struct Reference

Data Fields

- uim_physlotInfo uimSlotInfo [255]

8.767.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

uimSlotInfo[MAX_PHY_SLOTS_INFO]	• Contain all slots Information.

8.767.2 Field Documentation

8.767.2.1 uim_physlotInfo uim_physlotsInfo::uimSlotInfo[255]

8.768 uim_profileClassTlv Struct Reference

Data Fields

- uint32_t profileClass
- uint8_t TlvPresent

8.768.1 Detailed Description

This structure contains the information for profile Class.

Parameters

profileIconType	• Indicates the profile class. • Valid values: – 0x00 - Test profile – 0x01 - Provisioning profile – 0x02 - Operational profile Other values are reserved for future use and should be handled as operational. If the TLV is missing, the profile should be handled as operational.
TlvPresent	• TLV presents

8.768.2 Field Documentation

8.768.2.1 uint32_t uim_profileClassTlv::profileClass

8.768.2.2 uint8_t uim_profileClassTlv::TlvPresent

8.769 uim_profileICCIDTlv Struct Reference

Data Fields

- uint8_t profileICCIDLen
- uint8_t profileICCID [255]
- uint8_t TlvPresent

8.769.1 Detailed Description

This structure contains the information for profile ICCID.

Parameters

profileICCIDLen	• Number of sets of the profileICCID
profileICCID[MAX_DESCRIPTION_LENGTH]	• Indicates the profile ICCID coded as in EF-ICCID
TlvPresent	• TLV presents

8.769.2 Field Documentation

8.769.2.1 uint8_t uim_profileICCIDTlv::profileICCID[255]

8.769.2.2 uint8_t uim_profileICCIDTlv::profileICCIDLen

8.769.2.3 uint8_t uim_profileICCIDTlv::TlvPresent

8.770 uim_profileIconTlv Struct Reference

Data Fields

- uint16_t profileIconDataLen
- uint8_t profileIconData [255]
- uint8_t TlvPresent

8.770.1 Detailed Description

This structure contains the information for profile Name.

Parameters

profileIconDataLen	• Number of sets of the profileIconData
profileIconData[MAX_DESCRIPTION_LENGTH]	• Indicates the profile icon.
TlvPresent	• TLV presents

8.770.2 Field Documentation

8.770.2.1 uint8_t uim_profileIconTlv::profileIconData[255]

8.770.2.2 uint16_t uim_profileIconTlv::profileIconDataLen

8.770.2.3 uint8_t uim_profileIconTlv::TlvPresent

8.771 uim_profileIconTypeTlv Struct Reference

Data Fields

- uint32_t profileIconType
- uint8_t TlvPresent

8.771.1 Detailed Description

This structure contains the information for profile Icon Type.

Parameters

profileIconType	• Indicates the profile icon type. • Valid values: – 0x00 - No icon information – 0x01 - JPEG icon – 0x02 - PNG icon Other values are reserved for future use and should be handled as NONE.
TlvPresent	• TLV presents

8.771.2 Field Documentation

8.771.2.1 uint32_t uim_profileIconTypeTlv::profileIconType

8.771.2.2 uint8_t uim_profileIconTypeTlv::TlvPresent

8.772 uim_profileNameTlv Struct Reference

Data Fields

- uint8_t profileNameLen
- uint8_t profileName [255]
- uint8_t TlvPresent

8.772.1 Detailed Description

This structure contains the information for profile Name.

Parameters

profileNameLen	• Number of sets of the profileName
profileName[MAX_DESCRIPTION_LENGTH]	• Indicates the profile name.
TlvPresent	• TLV presents

8.772.2 Field Documentation

8.772.2.1 uint8_t uim_profileNameTlv::profileName[255]

8.772.2.2 uint8_t uim_profileNameTlv::profileNameLen

8.772.2.3 uint8_t uim_profileNameTlv::TlvPresent

8.773 uim_profileNicknameTlv Struct Reference

Data Fields

- uint8_t profileNicknameLen
- uint8_t profileNickname [255]
- uint8_t TlvPresent

8.773.1 Detailed Description

This structure contains the information for profile Nickname.

Parameters

profileNickname-Len	• Number of sets of the profileNickname
profile-Nickname[MAX-_DESCRIPTIO-N_LENGTH]	• Indicates the profile nickname.
TlvPresent	• TLV presents

8.773.2 Field Documentation

8.773.2.1 uint8_t uim_profileNicknameTlv::profileNickname[255]

8.773.2.2 uint8_t uim_profileNicknameTlv::profileNicknameLen

8.773.2.3 uint8_t uim_profileNicknameTlv::TlvPresent

8.774 uim_profilePolicyRulesTlv Struct Reference

Data Fields

- uint64_t profilePolicyRules
- uint8_t TlvPresent

8.774.1 Detailed Description

This structure contains the information for profile Policy Rules.

Parameters

profilePolicy-Rules	• Indicates the profile policy rules applicable for this profile. If the TLV is missing, it indicates there is no profile policy rule associated with this profile. Other bits are reserved for future use and shall be ignored by the client.
TlvPresent	• TLV presents

8.774.2 Field Documentation

8.774.2.1 uint64_t uim_profilePolicyRulesTlv::profilePolicyRules

8.774.2.2 uint8_t uim_profilePolicyRulesTlv::TlvPresent

8.775 uim_profileSPNTlv Struct Reference

Data Fields

- uint8_t profileSPNLen
- uint8_t profileSPN [255]
- uint8_t TlvPresent

8.775.1 Detailed Description

This structure contains the information for profile Service Provider Name.

Parameters

profileSPNLen	• Number of sets of the profileSPN
profileSPN[MAX-_DESCRIPTIO-N_LENGTH]	• Indicates the profile service provider name.
TlvPresent	• TLV presents

8.775.2 Field Documentation

8.775.2.1 uint8_t uim_profileSPNTlv::profileSPN[255]

8.775.2.2 uint8_t uim_profileSPNTlv::profileSPNLen

8.775.2.3 uint8_t uim_profileSPNTlv::TlvPresent

8.776 uim_profileStateTlv Struct Reference

Data Fields

- uint32_t profileState
- uint8_t TlvPresent

8.776.1 Detailed Description

This structure contains the information for profile State.

Parameters

profileState	• Indicates the profile state. • Valid values: – 0x00 - Inactive profile – 0x01 - Active profile Other values are reserved for future and should be handled as if the profile is inactive.
TlvPresent	• TLV presents

8.776.2 Field Documentation

8.776.2.1 uint32_t uim_profileStateTlv::profileState

8.776.2.2 uint8_t uim_profileStateTlv::TlvPresent

8.777 uim_readRecordInfo Struct Reference

Data Fields

- uint16_t record
- uint16_t length

8.777.1 Detailed Description

This structure contains the information for read record information.

Parameters

record	• Record number (starting from 1).
length	• Length of the content to be read. • The value 0 is used to read the complete record.

8.777.2 Field Documentation

8.777.2.1 uint16_t uim_readRecordInfo::length

8.777.2.2 uint16_t uim_readRecordInfo::record

8.778 uim_readResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [255]

8.778.1 Detailed Description

This structure contains the information for write operation.

Parameters

contentLen	• Number of sets of content.
content[255]	• Read content. • The content is the sequence of bytes as read from the card.

8.778.2 Field Documentation

8.778.2.1 uint8_t uim_readResult::content[255]

8.778.2.2 uint16_t uim_readResult::contentLen

8.779 uim_readResultInfo Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [255]
- uint8_t TlvPresent

8.779.1 Detailed Description

This structure contains the information for read operation.

Parameters

contentLen	• Number of sets of content.
content[MAX_DESCRIPTION_LENGTH]	• Read content. • The content is the sequence of bytes as read from the card.
TlvPresent	• Tlv Present.

8.779.2 Field Documentation

8.779.2.1 uint8_t uim_readResultInfo::content[255]

8.779.2.2 uint16_t uim_readResultInfo::contentLen

8.779.2.3 uint8_t uim_readResultInfo::TlvPresent

8.780 uim_readTransparentInfo Struct Reference

Data Fields

- uint16_t offset
- uint16_t length

8.780.1 Detailed Description

This structure contains the information for read operation.

Parameters

offset	• Offset for the read operation.
length	• Length of the content to be read. • The value 0 is used to read the complete file.

8.780.2 Field Documentation

8.780.2.1 uint16_t uim_readTransparentInfo::length

8.780.2.2 uint16_t uim_readTransparentInfo::offset

8.781 uim_refreshevent Struct Reference

Data Fields

- uint8_t stage
- uint8_t mode
- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]
- uint16_t numOfFiles
- uim_fileInfo arrfileInfo [255]

8.781.1 Detailed Description

This structure contains information of parameters associated with the Refresh Event.

Parameters

stage	• Indicates the stage of the Refresh procedure. – 0 - Waiting for OK to refresh – 1 - Refresh started – 2 - Refresh ended successfully – 3 - Refresh failed
mode	• Indicates the Refresh mode. – 0 - Reset – 1 - Init – 2 - Init and FCN – 3 - FCN – 4 - Init and Full FCN – 5 - Application reset – 6 - 3G session reset
sessionType	• Indicates the session type. – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Nonprovisioning on slot 1 – 5 - Nonprovisioning on slot 2 – 6 - Card on slot 1 – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2
aidLength	• Number of sets of the following elements – Application Identifier
aid	• Application identifier value or channel ID. This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases
numFiles	• Number of sets of the following elements: – file_id – path_len – path
arrfileInfo	• Array of file Information struct

8.781.2 Field Documentation

8.781.2.1 uint8_t uim_refreshevent::aid[255]

8.781.2.2 uint8_t uim_refreshevent::aidLength

8.781.2.3 uim_fileInfo uim_refreshevent::arrfileInfo[255]

8.781.2.4 uint8_t uim_refreshevent::mode

8.781.2.5 uint16_t uim_refreshevent::numOfFiles

8.781.2.6 uint8_t uim_refreshevent::sessionType

8.781.2.7 uint8_t uim_refreshevent::stage

8.782 uim_registerRefresh Struct Reference

Data Fields

- uint8_t registerFlag
- uint8_t voteForInit
- uint16_t numFiles
- uim_fileInfo arrfileInfo [255]

8.782.1 Detailed Description

This structure contains parameters of refresh Information

Parameters

registerFlag	• Flag that indicates whether to register or deregister for refresh indications. Valid values: – 0 - Deregister – 1 - Register
voteForInit	• Flag that indicates whether to vote for the init when there is a refresh. Valid values: – 0 - Client does not vote for initialization – 1 - Client votes for initialization
numFiles	• Number of sets of the following elements: – file_id – path_len – path
arrfileInfo	• Array of file Information structure. • See uim_fileInfo for more information

8.782.2 Field Documentation

8.782.2.1 uim_fileInfo uim_registerRefresh::arrfileInfo[255]

8.782.2.2 uint16_t uim_registerRefresh::numFiles

8.782.2.3 uint8_t uim_registerRefresh::registerFlag

8.782.2.4 uint8_t uim_registerRefresh::voteForInit

8.783 uim_remainingRetries Struct Reference

Data Fields

- uint8_t verifyLeft
- uint8_t unblockLeft

8.783.1 Detailed Description

This structure contains the information about the retries remaining.

Parameters

verifyLeft	• Number of remaining attempts to verify the PIN. • 0xFF, if unavailable.
unblockLeft	• Number of remaining attempts to unblock the PIN. • 0xFF, if unavailable.

Note

This value is returned only when the enable/disable operation has failed. This information is not sent for a hidden key PIN type.

8.783.2 Field Documentation

8.783.2.1 uint8_t uim_remainingRetries::unblockLeft

8.783.2.2 uint8_t uim_remainingRetries::verifyLeft

8.784 uim_sessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.784.1 Detailed Description

This structure contains the Session Information.

Parameters

sessionType	• Indicates the session type. – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Non-provisioning on slot 1 – 5 - Non-provisioning on slot 2 – 6 - Card on slot 1 – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2
aidLength	• Length of the following elements i.e. Application Identifier.
aid	• Application identifier value or channel ID. • This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases.

8.784.2 Field Documentation

8.784.2.1 uint8_t uim_sessionInformation::aid[255]

8.784.2.2 uint8_t uim_sessionInformation::aidLength

8.784.2.3 uint8_t uim_sessionInformation::sessionType

8.785 uim_setPINProtection Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinOperation
- uint8_t pinLength
- uint8_t pinValue [255]

8.785.1 Detailed Description

This structure contains the information about the pin protection parameters that need to be set.

Parameters

pinID	• Indicates the PIN ID to be enabled or disabled. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN – 4 - Hidden key

pinOperation	• Indicates whether the PIN is enabled or disabled. – 0 - Disable the PIN – 1 - Enable the PIN
pinLength	• Length of the following elements i.e. pin value.
pinValue[MAX_-DESCRIPTION-_LENGTH]	• PIN value. • This value is a sequence of ASCII characters.

8.785.2 Field Documentation

8.785.2.1 uint8_t uim_setPINProtection::pinID

8.785.2.2 uint8_t uim_setPINProtection::pinLength

8.785.2.3 uint8_t uim_setPINProtection::pinOperation

8.785.2.4 uint8_t uim_setPINProtection::pinValue[255]

8.786 uim_simBusyStatus Struct Reference

Data Fields

- uint8_t simBusyLength
- uint8_t simBusy [255]

8.786.1 Detailed Description

This structure contains Sim Busy Status Information.

Parameters

simBusyLength	• Number of sets of the following elements. i.e. simBusy
simBusy[MAX_-DESCRIPTION-_LENGTH]	• Indicates the status of the valid card. – 0 - Sim card is not busy – 1 - Sim card is busy

8.786.2 Field Documentation

8.786.2.1 uint8_t uim_simBusyStatus::simBusy[255]

8.786.2.2 uint8_t uim_simBusyStatus::simBusyLength

8.787 uim_slotInfo Struct Reference

Data Fields

- uint8_t cardState
- uint8_t upinState
- uint8_t upinRetries
- uint8_t upukRetries
- uint8_t errorState
- uint8_t numApp
- uim_appStatus AppStatus [10]

8.787.1 Detailed Description

This structure contains information about the SLOTS present.

Parameters

cardState	• Indicates the state of the card for each slot. – 0 - Absent – 1 - Present – 2 - Error
upinState	• Indicates the state of UPIN. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked – 0xFF - Not Available
upinRetries	• Indicates the number of retries remaining to verify the UPIN. • If 0xFF, information not available.
upukRetries	• Indicates the number of retries remaining to unblock the UPIN. • If 0xFF, information not available.

errorState	• Indicates the reason for the card error, and is valid only when the card state is Error – 0 - Unknown – 1 - Power down – 2 - Poll error – 3 - No ATR received – 4 - Volt mismatch – 5 - Parity error – 6 - Unknown; possibly removed – 7 - Card returned technical problems – 0xFF - Not Available • Other values are possible and reserved for future use. • When an unknown value is received, it is to be handled as “Unknown”.
numApp	• Indicates the number of applications available on the card. • The following block is repeated for each application. i.e. AppStatus. • If zero(0) then no AppStatus information exists.
AppStatus	• See uim_appStatus for more information.

8.787.2 Field Documentation

8.787.2.1 uim_appStatus uim_slotInfo::AppStatus[10]

8.787.2.2 uint8_t uim_slotInfo::cardState

8.787.2.3 uint8_t uim_slotInfo::errorState

8.787.2.4 uint8_t uim_slotInfo::numApp

8.787.2.5 uint8_t uim_slotInfo::upinRetries

8.787.2.6 uint8_t uim_slotInfo::upinState

8.787.2.7 uint8_t uim_slotInfo::upukRetries

8.788 uim_UIMGetFDNStatus Struct Reference

Data Fields

- uint8_t FDNStatus
- uint8_t TlvPresent

8.788.1 Detailed Description

This structure contains the information for FDN Status.

Parameters

FDNStatus	0 - FDN is not available 1 - FDN is available and disabled 2 - FDN is available and enabled
TlvPresent	• Tlv Present.

8.788.2 Field Documentation

8.788.2.1 uint8_t uim_UIMGetFDNStatus::FDNStatus

8.788.2.2 uint8_t uim_UIMGetFDNStatus::TlvPresent

8.789 uim_UIMGetHiddenKeyStatus Struct Reference

Data Fields

- uint8_t hiddenKey
- uint8_t TlvPresent

8.789.1 Detailed Description

This structure contains the information for Hidden Key Status.

Parameters

hiddenKey	• Status of hidden key PIN 0 - Not supported 1 - Enabled and not verified 2 - Enabled and verified 3 - Disabled
TlvPresent	• Tlv Present.

8.789.2 Field Documentation

8.789.2.1 uint8_t uim_UIMGetHiddenKeyStatus::hiddenKey

8.789.2.2 uint8_t uim_UIMGetHiddenKeyStatus::TlvPresent

8.790 uim_UIMGetIndex Struct Reference

Data Fields

- uint8_t index
- uint8_t TlvPresent

8.790.1 Detailed Description

This structure contains the information for getting Index.

Parameters

index	• Index of the application in EF_DIR file, starting from 1.
TlvPresent	• Tlv Present.

8.790.2 Field Documentation

8.790.2.1 uint8_t uim_UIMGetIndex::index

8.790.2.2 uint8_t uim_UIMGetIndex::TlvPresent

8.791 uim_UIMNickName Struct Reference

Data Fields

- uint8_t dataLen
- uint8_t data [255]

8.791.1 Detailed Description

This structure contains the information for the nick name of a profile.

Parameters

dataLen	• Length of the following elements i.e. data.
data[UIM_MAX_DESCRIPTION_LENGTH]	• Nick name for the profile.

8.791.2 Field Documentation

8.791.2.1 uint8_t uim_UIMNickName::data[255]

8.791.2.2 uint8_t uim_UIMNickName::dataLen

8.792 uim_UIMProfileCode Struct Reference

Data Fields

- uint16_t dataLen
- uint8_t data [255]

8.792.1 Detailed Description

This structure contains the information for activation/confirmation code for a profile.

Parameters

dataLen	• Length of the following elements i.e. data.
data[UIM_MAX-_DESCRIPTIO-N_LENGTH]	• Activation or confirmation code.

8.792.2 Field Documentation

8.792.2.1 uint8_t uim_UIMProfileCode::data[255]

8.792.2.2 uint16_t uim_UIMProfileCode::dataLen

8.793 uim_UIMSessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.793.1 Detailed Description

This structure contains the Session Information.

Parameters

sessionType	• Indicates the session type. – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Non-provisioning on slot 1 – 5 - Non-provisioning on slot 2 – 6 - Card on slot 1 – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2
aidLength	• Length of the following elements i.e. Application Identifier.
aid	• Application identifier value or channel ID. • This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases.

8.793.2 Field Documentation

8.793.2.1 uint8_t uim_UIMSessionInformation::aid[255]

8.793.2.2 uint8_t uim_UIMSessionInformation::aidLength

8.793.2.3 uint8_t uim_UIMSessionInformation::sessionType

8.794 uim_unblockUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pukLen
- uint8_t pukVal [255]
- uint8_t newPINLen
- uint8_t newPINVal [255]

8.794.1 Detailed Description

This structure contains the information about the unblock pin parameters.

Parameters

pinID	• Indicates the PIN ID to be changed. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN
pukLen	• Length of the following elements i.e. puk value.
pukVal[UIM_MAX_DESCRIPTION_LENGTH]	• PIN Unlock Key value. • This value is a sequence of ASCII characters.
newPINLen	• Length of the following elements i.e. new pin value.
newPINVal[UIM_MAX_DESCRIPTION_LENGTH]	• New PIN value. • This value is a sequence of ASCII characters.

8.794.2 Field Documentation

8.794.2.1 uint8_t uim_unblockUIMPIN::newPINLen

8.794.2.2 uint8_t uim_unblockUIMPIN::newPINVal[255]

8.794.2.3 uint8_t uim_unblockUIMPIN::pinID

8.794.2.4 uint8_t uim_unblockUIMPIN::pukLen

8.794.2.5 uint8_t uim_unblockUIMPIN::pukVal[255]

8.795 uim_validCardStatus Struct Reference

Data Fields

- uint8_t validCardLength
- uint8_t validCard [255]

8.795.1 Detailed Description

This structure contains Valid Card Status Information.

Parameters

validCardLength	• Number of sets of the following elements. i.e. validCard
validCard[MAX_-DESCRIPTION-_LENGTH]	• Indicates the status of the valid card. – 0 - Status of the card is unknown – 1 - Status of the card is valid

8.795.2 Field Documentation

8.795.2.1 uint8_t uim_validCardStatus::validCard[255]

8.795.2.2 uint8_t uim_validCardStatus::validCardLength

8.796 uim_verifyUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinLen
- uint8_t pinVal [255]

8.796.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

pinID	• Indicates the PIN ID to be verified. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN – 4 - Hidden key

pinLen	• Length of the following elements i.e. pin value.
pinVal[MAX_DESCRIPTION_LENGTH]	• PIN value. • This value is a sequence of ASCII characters.

8.796.2 Field Documentation

8.796.2.1 uint8_t uim_verifyUIMPIN::pinID

8.796.2.2 uint8_t uim_verifyUIMPIN::pinLen

8.796.2.3 uint8_t uim_verifyUIMPIN::pinVal[255]

8.797 uim_writeRecordInfo Struct Reference

Data Fields

- uint16_t record
- uint16_t dataLen
- uint8_t data [255]

8.797.1 Detailed Description

This structure contains the information for write operation.

Parameters

record	• Record number (starting from 1). • This field is ignored for cyclic files.
length	• Number of sets of content.
content[MAX_DESCRIPTION_LENGTH]	• Content to write.

8.797.2 Field Documentation

8.797.2.1 uint8_t uim_writeRecordInfo::data[255]

8.797.2.2 uint16_t uim_writeRecordInfo::dataLen

8.797.2.3 uint16_t uim_writeRecordInfo::record

8.798 uim_writeTransparentInfo Struct Reference

Data Fields

- uint16_t offset
- uint16_t dataLen
- uint8_t data [1024]

8.798.1 Detailed Description

This structure contains the information for write operation.

Parameters

offset	• Offset for the write operation.
dataLen	• Length of the following elements i.e. data.
data[MAX_CONTENT_LENGTH]	• Content to write.

8.798.2 Field Documentation

8.798.2.1 uint8_t uim_writeTransparentInfo::data[1024]

8.798.2.2 uint16_t uim_writeTransparentInfo::dataLen

8.798.2.3 uint16_t uim_writeTransparentInfo::offset

8.799 unpack_audio_SLQSGetAudioPathConfig_t Struct Reference

Data Fields

- uint8_t * pECMode
- uint8_t * pNSEnable
- uint16_t * pTXGain
- uint16_t * pDTMFTXGain
- uint16_t * pCodecSTGain
- audio_TXPCMIIRFltr * pTXPCMIIRFltr
- audio_RXPCMIIRFltr * pRXPCMIIRFltr
- uint8_t * pMICGainSelect
- uint8_t * pRXAVCAGCSwitch
- uint8_t * pTXAVCSwitch
- audio_RXAGCList * pRXAGCList
- audio_RXAVCList * pRXAVCList
- audio_TXAGCList * pTXAGCList
- swi_uint256_t ParamPresenceMask

8.799.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig response parameters.

Parameters

pECMode	[Optional] • AV_EC – 0 - Echo cancellation off – 1 - Handset mode – 2 - Headset mode – 3 - Car kit mode – 4 - Speaker Mode • Bit to check in ParamPresenceMask - **33**
pNSEnable	[Optional] • AV_NS – 0 - Noise suppression off – 1 - Noise suppression on • Bit to check in ParamPresenceMask - **34**
pTXGain	[Optional] • AV_TXVOL – 0x0000 - 0xffff • Bit to check in ParamPresenceMask - **35**
pDTMFTXGain	[Optional] • AV_DTMFTXG – 0x0000 - 0xffff • Bit to check in ParamPresenceMask - **36**
pCodecSTGain	[Optional] • AV_CODECSTG – 0x0000 - 0xffff • Bit to check in ParamPresenceMask - **37**
pTXPCMIIRFltr	[Optional] • See audio_TXPCMIIRFltr for more information • Bit to check in ParamPresenceMask - **38**
pRXPCMIIRFltr	[Optional] • See audio_RXPCMIIRFltr for more information • Bit to check in ParamPresenceMask - **39**
pMICGainSelect	[Optional] • AV_MICGAIN • Bit to check in ParamPresenceMask - **40**
pRXAVCAGC-Switch	[Optional] • RX AVC/AGC Switch • Bit to check in ParamPresenceMask - **41**
pTXAVCSwitch	[Optional] • TX AVC Switch • Bit to check in ParamPresenceMask - **42**

pRXAGCList	[Optional] • See audio_RXAGCList for more information • Bit to check in ParamPresenceMask - **43**
pRXAVCList	[Optional] • See audio_RXAVCList for more information • Bit to check in ParamPresenceMask - **44**
pTXAGCList	[Optional] • See audio_TXAGCList for more information • Bit to check in ParamPresenceMask - **45**

8.799.2 Field Documentation

8.799.2.1 swi_uint256_t unpack_audio_SLQSGetAudioPathConfig_t::ParamPresenceMask

8.799.2.2 uint16_t* unpack_audio_SLQSGetAudioPathConfig_t::pCodecSTGain

8.799.2.3 uint16_t* unpack_audio_SLQSGetAudioPathConfig_t::pDTMFTXGain

8.799.2.4 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pECMode

8.799.2.5 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pMICGainSelect

8.799.2.6 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pNSEnable

8.799.2.7 audio_RXAGCList* unpack_audio_SLQSGetAudioPathConfig_t::pRXAGCList

8.799.2.8 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pRXAVCAGCSwitch

8.799.2.9 audio_RXAVCList* unpack_audio_SLQSGetAudioPathConfig_t::pRXAVCList

8.799.2.10 audio_RXPCMIIRFltr* unpack_audio_SLQSGetAudioPathConfig_t::pRXPCMIIRFltr

8.799.2.11 audio_TXAGCList* unpack_audio_SLQSGetAudioPathConfig_t::pTXAGCList

8.799.2.12 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pTXAVCSwitch

8.799.2.13 uint16_t* unpack_audio_SLQSGetAudioPathConfig_t::pTXGain

8.799.2.14 audio_TXPCMIIRFltr* unpack_audio_SLQSGetAudioPathConfig_t::pTXPCMIIRFltr

8.800 unpack_audio_SLQSGetAudioProfile_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t EarMute
- uint8_t MicMute
- uint8_t Volume
- swi_uint256_t ParamPresenceMask

8.800.1 Detailed Description

This structure contains the unpack parameters to Get Audio Profile.

Parameters

Profile	• Audio Profile – 0 - Handset – 1 - Headset – 2 - Car Kit – 3 - Speaker phone – 4 - Auxiliary – 5 - TTY – 6 - Auxiliary external PCM – 7 - Primary external PCM – 8 - External slave PCM – 9 - I2S • Bit to check in ParamPresenceMask - **16**
EarMute	• Ear Mute Setting – 0 - unmuted – 1 - muted • Bit to check in ParamPresenceMask - **17**
MicMute	• MIC Mute Setting – 0 - unmuted – 1 - muted • Bit to check in ParamPresenceMask - **18**
Volume	• Audio Volume Level – 0 to 7 • Bit to check in ParamPresenceMask - **20**

8.800.2 Field Documentation

8.800.2.1 uint8_t unpack_audio_SLQSGetAudioProfile_t::EarMute

8.800.2.2 uint8_t unpack_audio_SLQSGetAudioProfile_t::MicMute

8.800.2.3 swi_uint256_t unpack_audio_SLQSGetAudioProfile_t::ParamPresenceMask

8.800.2.4 uint8_t unpack_audio_SLQSGetAudioProfile_t::Profile

8.800.2.5 uint8_t unpack_audio_SLQSGetAudioProfile_t::Volume

8.801 unpack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint16_t ResCode
- swi_uint256_t ParamPresenceMask

8.801.1 Detailed Description

This structure contains the unpack parameters for SLQSGetAudioVolTLBConfig.

Parameters

ResCode	• Result of requested item • Bit to check in ParamPresenceMask - **33**

8.801.2 Field Documentation

8.801.2.1 swi_uint256_t unpack_audio_SLQSGetAudioVolTLBConfig_t::ParamPresenceMask

8.801.2.2 uint16_t unpack_audio_SLQSGetAudioVolTLBConfig_t::ResCode

8.802 unpack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint16_t ResCode
- swi_uint256_t ParamPresenceMask

8.802.1 Detailed Description

This structure contains the unpack parameters for SLQSSetAudioVolTLBConfig.

Parameters

ResCode	• Result of requested item • Bit to check in ParamPresenceMask - **33**

8.802.2 Field Documentation

8.802.2.1 swi_uint256_t unpack_audio_SLQSSetAudioVolTLBConfig_t::ParamPresenceMask

8.802.2.2 uint16_t unpack_audio_SLQSSetAudioVolTLBConfig_t::ResCode

8.803 unpack_cat_SetCatEventCallback_ind_t Struct Reference

Data Fields

- uint8_t event_Index
- struct cat_commonEventTlv CCETlv [11]

- swi_uint256_t ParamPresenceMask

8.803.1 Detailed Description

Structure used to store all CAT Notification Parameters.

Parameters

event_Index	- Event Index
CCETlv	- CAT common event Tlv cat_commonEventTlv for more info. • [0] - Bit to check in ParamPresenceMask for Display Text - **16** • [1] - Bit to check in ParamPresenceMask for Get In-Key - **17** • [2] - Bit to check in ParamPresenceMask for Get Input - **18** • [3] - Bit to check in ParamPresenceMask for Setup Menu - **19** • [4] - Bit to check in ParamPresenceMask for Select Item - **20** • [5] - Bit to check in ParamPresenceMask for Alpha Identifier - **21** • [6] - Bit to check in ParamPresenceMask for Setup Event List - **22** • [7] - Bit to check in ParamPresenceMask for Setup Idle Mode Text - **23** • [8] - Bit to check in ParamPresenceMask for Language Notification - **24** • [9] - Bit to check in ParamPresenceMask for Refresh - **25** • [10] - Bit to check in ParamPresenceMask for End Proactive Session - **26**

8.803.2 Field Documentation

8.803.2.1 struct cat_commonEventTlv unpack_cat_SetCatEventCallback_ind_t::CCETlv[11]

8.803.2.2 uint8_t unpack_cat_SetCatEventCallback_ind_t::event_Index

8.803.2.3 swi_uint256_t unpack_cat_SetCatEventCallback_ind_t::ParamPresenceMask

8.804 unpack_cat_SetCATEventCallback_t Struct Reference

Data Fields

- uint32_t errorMask
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.804.1 Detailed Description

This structure contains sEnables/disables the CAT event callback unpack variable.

Parameters

errorMask	• error bitmask. Each bit set indicates the proactive command that caused the error – 0x00000001 - Display Text – 0x00000002 - Get In-Key – 0x00000004 - Get Input – 0x00000008 - Setup Menu – 0x00000010 - Select Item – 0x00000020 - Send SMS - Alpha Identifier – 0x00000040 - Setup Event: User Activity – 0x00000080 - Setup Event: Idle Screen Notify – 0x00000100 - Setup Event: Language Sel Notify – 0x00000200 - Setup Idle Mode Text – 0x00000400 - Language Notification – 0x00000800 - Refresh – 0x00001000 - End Proactive Session • Bit to check in ParamPresenceMask - **16**
Tlvresult	• Unpack Result

8.804.2 Field Documentation

8.804.2.1 uint32_t unpack_cat_SetCATEventCallback_t::errorMask

8.804.2.2 swi_uint256_t unpack_cat_SetCATEventCallback_t::ParamPresenceMask

8.804.2.3 uint16_t unpack_cat_SetCATEventCallback_t::Tlvresult

8.805 unpack_ctl_GetClientId_t Struct Reference

Data Fields

- uint8_t serviceType
- uint8_t clientId
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.805.1 Detailed Description

Structure to unpack the QMI client ID for service.

Parameters

serviceType	• QMI service for which client ID has been obtained.
clientId	• client ID that has been obtained.

TlvResult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.805.2 Field Documentation

8.805.2.1 uint8_t unpack_ctl_GetClientId_t::clientId

8.805.2.2 swi_uint256_t unpack_ctl_GetClientId_t::ParamPresenceMask

8.805.2.3 uint8_t unpack_ctl_GetClientId_t::serviceType

8.805.2.4 uint16_t unpack_ctl_GetClientId_t::TlvResult

8.806 unpack_ctl_ReleaseClientId_t Struct Reference

Data Fields

- uint8_t serviceType
- uint8_t clientId
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.806.1 Detailed Description

Structure to unpack the QMI release client ID for service.

Parameters

serviceType	• QMI service for which client ID is to be released.
clientId	• client ID that has been obtained.
TlvResult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.806.2 Field Documentation

8.806.2.1 uint8_t unpack_ctl_ReleaseClientId_t::clientId

8.806.2.2 swi_uint256_t unpack_ctl_ReleaseClientId_t::ParamPresenceMask

8.806.2.3 uint8_t unpack_ctl_ReleaseClientId_t::serviceType

8.806.2.4 uint16_t unpack_ctl_ReleaseClientId_t::Tlvresult

8.807 unpack_dms_GetActivationState_t Struct Reference

Data Fields

- uint8_t state
- swi_uint256_t ParamPresenceMask

8.807.1 Detailed Description

This structure contains unpack_dms_GetActivationState parameters.

Parameters

state[OUT]	• Service Activation Code 0 - Service not activated 1 - Service activated 2 - Activation connecting 3 - Activation connected 4 - OTASP security authenticated 5 - OTASP NAM downloaded 6 - OTASP MDN downloaded 7 - OTASP IMSI downloaded 8 - OTASP PRL downloaded 9 - OTASP SPC downloaded 10 - OTASP settings committed • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.807.2 Field Documentation

8.807.2.1 swi_uint256_t unpack_dms_GetActivationState_t::ParamPresenceMask

8.807.2.2 uint8_t unpack_dms_GetActivationState_t::state

8.808 unpack_dms_GetBandCapability_t Struct Reference

Data Fields

- uint64_t BandCapability
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.808.1 Detailed Description

Structure to store the band capability of the device.

Parameters

BandCapability	Bitmask of bands supported by the device • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19 • Bit 60 - WCDMA Japan 850 band • Bit 61 - WCDMA 1500 band • Bits 62 and 63 - Reserved

- Bit to check in ParamPresenceMask - **1**

Parameters

Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.808.2 Field Documentation

8.808.2.1 uint64_t unpack_dms_GetBandCapability_t::BandCapability

8.808.2.2 swi_uint256_t unpack_dms_GetBandCapability_t::ParamPresenceMask

8.808.2.3 uint16_t unpack_dms_GetBandCapability_t::Tlvresult

8.809 unpack_dms_GetCrashAction_t Struct Reference

Data Fields

- uint8_t DevCrashState
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.809.1 Detailed Description

This structure is used to store the Crash State from the device.

Parameters

DevCrashState	• Device Crash State • Values: – 0 - USB Memory Download Modem will reset after a crash and will stay in USB download mode with only ttyUSB0 enumerated. ramdump tool is to be used to recover the crash dump. Modem needs to be reset again to come back in ONLINE mode. – 1 - Reset Modem will reset and come back in ONLINE mode. Minimal crash data will be available and can be extracted with at!gcdump? AT command or SLQSSwiGetCrashInfo() SDK API – 2 - No action • Bit to check in ParamPresenceMask - **1**
Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.809.2 Field Documentation

8.809.2.1 uint8_t unpack_dms_GetCrashAction_t::DevCrashState

8.809.2.2 swi_uint256_t unpack_dms_GetCrashAction_t::ParamPresenceMask

8.809.2.3 uint16_t unpack_dms_GetCrashAction_t::Tlvresult

8.810 unpack_dms_GetCustFeature_t Struct Reference

Data Fields

- uint32_t GpsEnable
- uint8_t DisableIMSI
- uint16_t IPFamSupport
- uint8_t RMAutoConnect
- uint8_t GPSSel
- uint8_t SMSSupport
- uint8_t IsVoiceEnabled
- uint8_t DHCPRelayEnabled
- uint8_t GPSLPM
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.810.1 Detailed Description

This structure contains current settings of custom features

Parameters

GpsEnable	• describes if GPS is enabled or disabled • values: – 0x00 - GPS is disabled – 0x01 - GPS is enabled • Bit to check in ParamPresenceMask - **16**
DisableIMSI	• optional 1 byte parameter • describes if IMSI display is enabled or disabled • values: – 0x00 - Allow display of IMSI – 0x01 - Do not display IMSI • Bit to check in ParamPresenceMask - **17**
IPFamSupport	• optional 2 byte BitMask • bitmask representing the IP families supported • values: – 0x01 - IPv4 – 0x02 - IPv6 – 0x04 - IPv4v6 • Bit to check in ParamPresenceMask - **18**

RMAutoConnect	• optional 1 byte parameter • QMI Mode RM Net Auto Connect Support • values: – 0x00 - Not Supported – 0x01 - Supported • Bit to check in ParamPresenceMask - **19**
GPSSel	• optional 1 byte parameter • GPS Antenna Select • values: – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied • Bit to check in ParamPresenceMask - **20**
SMSSupport	• optional 1 byte parameter • SMS support • values: – 0x00 - Not supported – 0x01 - supported • Bit to check in ParamPresenceMask - **21** • Used to determine whether or not to hide SMS from user
IsVoiceEnabled	• optional 1 byte parameter • Voice support • values: – 0x00 - Enable voice on both AT and QMI interface (default) – 0x01 - Reserved – 0x02 - Disable voice on both AT and QMI interface • Bit to check in ParamPresenceMask - **22**
DHCPRelay-Enabled	• optional 1 byte parameter • DHCP Relay support • values: – 0x00 - Disable DHCP relay – 0x01 - Enable DHCP relay • Bit to check in ParamPresenceMask - **23**
GPSLPM	• optional 1 byte parameter • GPSLPM support • values: – 0x00 - Enable GPS in Low Power Mode – 0x01 - Disable GPS in Low Power Mode • Bit to check in ParamPresenceMask - **24**

Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.810.2 Field Documentation

8.810.2.1 uint8_t unpack_dms_GetCustFeature_t::DHCPRelayEnabled

8.810.2.2 uint8_t unpack_dms_GetCustFeature_t::DisableIMSI

8.810.2.3 uint32_t unpack_dms_GetCustFeature_t::GpsEnable

8.810.2.4 uint8_t unpack_dms_GetCustFeature_t::GPSLPM

8.810.2.5 uint8_t unpack_dms_GetCustFeature_t::GPSSel

8.810.2.6 uint16_t unpack_dms_GetCustFeature_t::IPFamSupport

8.810.2.7 uint8_t unpack_dms_GetCustFeature_t::IsVoiceEnabled

8.810.2.8 swi_uint256_t unpack_dms_GetCustFeature_t::ParamPresenceMask

8.810.2.9 uint8_t unpack_dms_GetCustFeature_t::RMAutoConnect

8.810.2.10 uint8_t unpack_dms_GetCustFeature_t::SMSSupport

8.810.2.11 uint16_t unpack_dms_GetCustFeature_t::Tlvresult

8.811 unpack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- DMSgetCustomFeatureV2 GetCustomFeatureV2
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.811.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

GetCustom-FeatureV2	• See DMSgetCustomFeatureV2 for more information • pCustSettingInfo – Bit to check in ParamPresenceMask - **32** • pCustSettingList – Bit to check in ParamPresenceMask - **33**

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.811.2 Field Documentation

8.811.2.1 DMSgetCustomFeatureV2 unpack_dms_GetCustFeaturesV2_t::GetCustomFeatureV2

8.811.2.2 swi_uint256_t unpack_dms_GetCustFeaturesV2_t::ParamPresenceMask

8.811.2.3 uint16_t unpack_dms_GetCustFeaturesV2_t::Tlvresult

8.812 unpack_dms_GetDeviceCap_t Struct Reference

Data Fields

- uint32_t MaxTXChannelRate
- uint32_t MaxRXChannelRate
- uint32_t DataServiceCapability
- uint32_t SimCapability
- uint32_t RadioIfacesSize
- uint8_t RadioIfaces [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.812.1 Detailed Description

Structure to store the device capabilities

Parameters

MaxTXChannel-Rate	• Maximum transmission rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies • Bit to check in ParamPresenceMask - **1**
MaxRXChannel-Rate	• Maximum reception rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies • Bit to check in ParamPresenceMask - **1**
DataService-Capability	• CS/PS data service capability – 0 - No data services supported – 1 - Only Circuit Switched (CS) services supported – 2 - Only Packet Switched (PS) services supported – 3 - Simultaneous CS and PS – 4 - Non-simultaneous CS and PS • Bit to check in ParamPresenceMask - **1**

SimCapability	Device SIM capability • 0 - SIM not supported • 1 - SIM supported

- Bit to check in ParamPresenceMask - **1**

Parameters

RadioIfacesSize	• Upon input, the maximum number of elements that the radio interface array can contain • Upon successful output, actual number of elements in the radio interface array • Bit to check in ParamPresenceMask - **1**
RadioIfaces	• Radio interface array. This is a structure of array containing the elements below. uint8_t RadioIfaces – See qaGobiApiTableRadioInterfaces.h for Radio Interfaces • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.812.2 Field Documentation

8.812.2.1 uint32_t unpack_dms_GetDeviceCap_t::DataServiceCapability

8.812.2.2 uint32_t unpack_dms_GetDeviceCap_t::MaxRXChannelRate

8.812.2.3 uint32_t unpack_dms_GetDeviceCap_t::MaxTXChannelRate

8.812.2.4 swi_uint256_t unpack_dms_GetDeviceCap_t::ParamPresenceMask

8.812.2.5 uint8_t unpack_dms_GetDeviceCap_t::RadioIfaces[255]

8.812.2.6 uint32_t unpack_dms_GetDeviceCap_t::RadioIfacesSize

8.812.2.7 uint32_t unpack_dms_GetDeviceCap_t::SimCapability

8.812.2.8 uint16_t unpack_dms_GetDeviceCap_t::Tlvresult

8.813 unpack_dms_GetDeviceCapabilities_t Struct Reference

Data Fields

- uint32_t maxTxChannelRate
- uint32_t maxRxChannelRate
- uint32_t dataServiceCaCapability

- uint32_t simCapability
- uint32_t radioIfacesSize
- uint8_t RadioIfaces [255]
- swi_uint256_t ParamPresenceMask

8.813.1 Detailed Description

Gets the device capabilities structure

Parameters

maxTxChannelRate	• Maximum transmission rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies • Bit to check in ParamPresenceMask - **1**
maxRxChannelRate	• Maximum reception rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies • Bit to check in ParamPresenceMask - **1**
dataServiceCaCapability	• CS/PS data service capability – 0 - No data services supported – 1 - Only Circuit Switched (CS) services supported – 2 - Only Packet Switched (PS) services supported – 3 - Simultaneous CS and PS – 4 - Non-simultaneous CS and PS • Bit to check in ParamPresenceMask - **1**
simCapability	• Device SIM capability – 0 - SIM not supported – 1 - SIM supported • Bit to check in ParamPresenceMask - **1**
radioIfacesSize	• Upon input, the maximum number of elements that the radio interface array can contain • Upon successful output, actual number of elements in the radio interface array • Bit to check in ParamPresenceMask - **1**
RadioIfaces[OUT]	• Radio interface array. This is a structure of array containing the elements below. – See qaGobiApiTableRadioInterfaces.h for Radio Interfaces • Bit to check in ParamPresenceMask - **1**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.813.2 Field Documentation

8.813.2.1 uint32_t unpack_dms_GetDeviceCapabilities_t::dataServiceCaCapability

8.813.2.2 uint32_t unpack_dms_GetDeviceCapabilities_t::maxRxChannelRate

8.813.2.3 uint32_t unpack_dms_GetDeviceCapabilities_t::maxTxChannelRate

8.813.2.4 swi_uint256_t unpack_dms_GetDeviceCapabilities_t::ParamPresenceMask

8.813.2.5 uint8_t unpack_dms_GetDeviceCapabilities_t::RadioIfaces[255]

8.813.2.6 uint32_t unpack_dms_GetDeviceCapabilities_t::radioIfacesSize

8.813.2.7 uint32_t unpack_dms_GetDeviceCapabilities_t::simCapability

8.814 unpack_dms_GetDeviceCapabilitiesV2_t Struct Reference

Data Fields

- dms_devCaps DevCaps
- uint32_t * pDevSrvCaps
- uint64_t * pDevVoiceCaps
- uint64_t * pDevVoiceDataCaps
- dms_devMultiSimCaps * pDevMultiSimCaps
- dms_devMultiSimVoiceDataCaps * pDevMultiSimVoiceDataCaps
- dms_devCurSubsCaps * pDevCurSubsCaps
- dms_devSubsVoiceDataCaps * pDevSubsVoiceDataCaps
- dms_devSubsFeatureModeCaps * pDevSubsFeatureModeCaps
- uint8_t * pDevMaxActDataSubsCaps
- dms_devMaxSubsCaps * pDevMaxSubsCaps
- dms_devMaxCfgListCaps * pDevMaxCfgListCaps
- int16_t * pDevExplicitCfgIndex
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.814.1 Detailed Description

This structure contains the unpack_dms_GetDeviceCapabilitiesV2 response parameters.

Parameters

DevCaps	• See dms_devCaps for more information • Bit to check in ParamPresenceMask - **1**
pDevSrvCaps	• Device service capabilities – DMS_DEVICE_CAP_DATA_ONLY -0x01 – DMS_DEVICE_CAP_VOICE_ONLY -0x02 – DMS_DEVICE_CAP_SIMUL_VOICE_AND_DATA -0x03 – DMS_DEVICE_CAP_NONSIMUL_VOICE_AND_DATA -0x04 • Bit to check in ParamPresenceMask - **16**

pDevVoiceCaps	• Device voice capabilities – Bit 0 - GW CSFB * 0 - Not capable * 1 - Capable – Bit 1 - 1x CSFB * 0 - Not capable * 1 - Capable – Bit 2 - VoLTE * 0 - Not capable * 1 - Capable

- Bit to check in ParamPresenceMask - **17**

Parameters

pDevVoiceDataCaps	• DevVoiceDataCaps – Bit 0 - SVLTE capability – Bit 1 - SVDO capability – Bit 2 - SGLTE capability • Bit to check in ParamPresenceMask - **18**
pDevMultiSimCaps	• See dms_devMultiSimCaps for more information • Bit to check in ParamPresenceMask - **19**
pDevMultiSimVoiceDataCaps	• See dms_devMultiSimVoiceDataCaps for more information • Bit to check in ParamPresenceMask - **20**
pDevCurSubsCaps	• See dms_devCurSubsCaps for more information • Bit to check in ParamPresenceMask - **21**
pDevSubsVoiceDataCaps	• See dms_devSubsVoiceDataCaps for more information • Bit to check in ParamPresenceMask - **22**
pDevSubsFeatureModeCaps	• See dms_devSubsFeatureModeCaps for more information • Bit to check in ParamPresenceMask - **23**
pDevMaxActDataSubsCaps	• Max number of subscriptions for data activity • Bit to check in ParamPresenceMask - **24**
pDevMaxSubsCaps	• See dms_devMaxSubsCaps for more information • Bit to check in ParamPresenceMask - **25**

pDevMaxAct-DataSubsCaps	• See dms_devMaxCfgListCaps for more information • Bit to check in ParamPresenceMask - **26**
pDevExplicitCfg-Index	• Explicit Cfg Index – (-1) - Modem controlled cfg – any other valid value • Bit to check in ParamPresenceMask - **27**
Tlvresult	• unpack Tlv Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.814.2 Field Documentation

8.814.2.1 dms_devCaps unpack_dms_GetDeviceCapabilitiesV2_t::DevCaps

8.814.2.2 swi_uint256_t unpack_dms_GetDeviceCapabilitiesV2_t::ParamPresenceMask

8.814.2.3 dms_devCurSubsCaps* unpack_dms_GetDeviceCapabilitiesV2_t::pDevCurSubsCaps

8.814.2.4 int16_t* unpack_dms_GetDeviceCapabilitiesV2_t::pDevExplicitCfgIndex

8.814.2.5 uint8_t* unpack_dms_GetDeviceCapabilitiesV2_t::pDevMaxActDataSubsCaps

8.814.2.6 dms_devMaxCfgListCaps* unpack_dms_GetDeviceCapabilitiesV2_t::pDevMaxCfgListCaps

8.814.2.7 dms_devMaxSubsCaps* unpack_dms_GetDeviceCapabilitiesV2_t::pDevMaxSubsCaps

8.814.2.8 dms_devMultiSimCaps* unpack_dms_GetDeviceCapabilitiesV2_t::pDevMultiSimCaps

8.814.2.9 dms_devMultiSimVoiceDataCaps* unpack_dms_GetDeviceCapabilitiesV2_t::pDevMultiSimVoiceDataCaps

8.814.2.10 uint32_t* unpack_dms_GetDeviceCapabilitiesV2_t::pDevSrvCaps

8.814.2.11 dms_devSubsFeatureModeCaps* unpack_dms_GetDeviceCapabilitiesV2_t::pDevSubsFeatureModeCaps

8.814.2.12 dms_devSubsVoiceDataCaps* unpack_dms_GetDeviceCapabilitiesV2_t::pDevSubsVoiceDataCaps

8.814.2.13 uint64_t* unpack_dms_GetDeviceCapabilitiesV2_t::pDevVoiceCaps

8.814.2.14 uint64_t* unpack_dms_GetDeviceCapabilitiesV2_t::pDevVoiceDataCaps

8.814.2.15 uint16_t unpack_dms_GetDeviceCapabilitiesV2_t::Tlvresult

8.815 unpack_dms_GetDeviceHardwareRev_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.815.1 Detailed Description

Structure to store the hardware revision of the device

Parameters

stringSize	• The maximum number of characters (including NULL terminator) that the string array can contain • Bit to check in ParamPresenceMask - **1**
String	• NULL terminated Hardware Revision string • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
TlvResult	• Unpack Result
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.815.2 Field Documentation

8.815.2.1 swi_uint256_t unpack_dms_GetDeviceHardwareRev_t::ParamPresenceMask

8.815.2.2 char unpack_dms_GetDeviceHardwareRev_t::String[255]

8.815.2.3 uint8_t unpack_dms_GetDeviceHardwareRev_t::stringSize

8.815.2.4 uint16_t unpack_dms_GetDeviceHardwareRev_t::TlvResult

8.816 unpack_dms_GetDeviceMfr_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.816.1 Detailed Description

This structure is used to store unpack_dms_GetDeviceMfr parameters

Parameters

stringSize	• The maximum number of characters (including NULL terminator) that the string array can contain

- Bit to check in ParamPresenceMask - **1**

Parameters

String	• NULL terminated Device Manufacture string • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.816.2 Field Documentation

8.816.2.1 swi_uint256_t unpack_dms_GetDeviceMfr_t::ParamPresenceMask

8.816.2.2 char unpack_dms_GetDeviceMfr_t::String[255]

8.816.2.3 uint8_t unpack_dms_GetDeviceMfr_t::stringSize

8.816.2.4 uint16_t unpack_dms_GetDeviceMfr_t::Tlvresult

8.817 unpack_dms_GetDeviceSerialNumbers_t Struct Reference

Data Fields

- uint8_t esnSize
- char ESNString [255]
- uint8_t imeiSize
- char IMEIString [255]
- uint8_t meidSize
- char MEIDString [255]
- uint8_t imeiSvnSize
- char ImeiSvnString [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.817.1 Detailed Description

This structure used to store unpack_dms_GetDeviceSerialNumbers parameters

Parameters

esnSize	• The maximum number of characters (including NULL terminator) that the ESN string array can contain • Bit to check in ParamPresenceMask - **16**
ESNString	• NULL-terminated ESN string. Empty string is returned when ESN is not supported/programmed • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **16**
imeiSize	• The maximum number of characters (including NULL terminator) that the IMEI string array can contain • Bit to check in ParamPresenceMask - **17**
IMEIString	• NULL terminated IMEI string. Empty string is returned when IMEI is not supported/programmed • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **17**
meidSize	• The maximum number of characters (including NULL terminator) that the MEID string array can contain • Bit to check in ParamPresenceMask - **18**
MEIDString	• NULL-terminated MEID string. Empty string is returned when MEID is not supported/programmed • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **18**
imeiSvnSize	• The maximum number of characters (including NULL terminator) that the IMEI SVN string array can contain. • Bit to check in ParamPresenceMask - **19**
ImeiSvnString	• NULL-terminated IMEI SVN string. Empty string is returned when IMEI SVN is not supported/programmed. • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **19**
Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.817.2 Field Documentation

8.817.2.1 uint8_t unpack_dms_GetDeviceSerialNumbers_t::esnSize

8.817.2.2 char unpack_dms_GetDeviceSerialNumbers_t::ESNString[255]

8.817.2.3 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSize

8.817.2.4 char unpack_dms_GetDeviceSerialNumbers_t::IMEIString[255]

8.817.2.5 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSvnSize

8.817.2.6 char unpack_dms_GetDeviceSerialNumbers_t::ImeiSvnString[255]

8.817.2.7 uint8_t unpack_dms_GetDeviceSerialNumbers_t::meidSize

8.817.2.8 char unpack_dms_GetDeviceSerialNumbers_t::MEIDString[255]

8.817.2.9 swi_uint256_t unpack_dms_GetDeviceSerialNumbers_t::ParamPresenceMask

8.817.2.10 uint16_t unpack_dms_GetDeviceSerialNumbers_t::Tlvresult

8.818 unpack_dms_GetFirmwareInfo_t Struct Reference

Data Fields

- char modelid_str [20]
- char bootversion_str [85]
- char appversion_str [85]
- char sku_str [15]
- char packageid_str [85]
- char carrier_str [20]
- char priversion_str [16]
- char cur_carr_name [17]
- char cur_carr_rev [13]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.818.1 Detailed Description

This structure used to store unpack_dms_GetFirmwareInfo parameters

Parameters

modelid_str	- Mode ID String. • NULL-terminated Mode ID String. • Bit to check in ParamPresenceMask - **16**
bootversion_str	- Boot Version. • NULL-terminated Boot Version String. • Bit to check in ParamPresenceMask - **17**
appversion_str	- Application Version String. • NULL-terminated Application Version String. • Bit to check in ParamPresenceMask - **18**

sku_str	- SKU String. • NULL-terminated SKU String. • Bit to check in ParamPresenceMask - **19**
packageid_str	- Package ID String. • NULL-terminated Package ID String. • deprecated on EM7430/EM7455/MC7430/MC7455(9x30) devices • Bit to check in ParamPresenceMask - **20**
carrier_str	- Carrier String. • NULL-terminated Carrier String. • Bit to check in ParamPresenceMask - **21**
priversion_str	- PRI Version String. • NULL-terminated PRI Version String. • Bit to check in ParamPresenceMask - **22**
cur_carr_name	- Current Carrier Name String. • NULL-terminated Current Carrier Name String. • Bit to check in ParamPresenceMask - **23**
cur_carr_rev	- Current Carrier Revision String. • NULL-terminated Current Carrier Revision String. • Bit to check in ParamPresenceMask - **24**
Tlvresult	- unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.818.2 Field Documentation

8.818.2.1 char unpack_dms_GetFirmwareInfo_t::appversion_str[85]

8.818.2.2 char unpack_dms_GetFirmwareInfo_t::bootversion_str[85]

8.818.2.3 char unpack_dms_GetFirmwareInfo_t::carrier_str[20]

8.818.2.4 char unpack_dms_GetFirmwareInfo_t::cur_carr_name[17]

8.818.2.5 char unpack_dms_GetFirmwareInfo_t::cur_carr_rev[13]

8.818.2.6 char unpack_dms_GetFirmwareInfo_t::modelid_str[20]

8.818.2.7 char unpack_dms_GetFirmwareInfo_t::packageid_str[85]

8.818.2.8 swi_uint256_t unpack_dms_GetFirmwareInfo_t::ParamPresenceMask

8.818.2.9 char unpack_dms_GetFirmwareInfo_t::priversion_str[16]

8.818.2.10 char unpack_dms_GetFirmwareInfo_t::sku_str[15]

8.818.2.11 uint16_t unpack_dms_GetFirmwareInfo_t::Tlvresult

8.819 unpack_dms_GetFirmwareRevision_t Struct Reference

Data Fields

- uint8_t amssSize
- char AMSSString [255]
- char PRIString [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.819.1 Detailed Description

This structure used to store unpack_dms_GetFirmwareRevision parameters

Parameters

amssSize	• Maximum number of characters (including NULL terminator) that the AMSS string array can contain. • Bit to check in ParamPresenceMask - **1**
AMSSString	• NULL-terminated AMSS revision string. • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
PRIString[OUT]	• NULL-terminated PRI revision string. • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **17**
Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.819.2 Field Documentation

8.819.2.1 uint8_t unpack_dms_GetFirmwareRevision_t::amssSize

8.819.2.2 char unpack_dms_GetFirmwareRevision_t::AMSSString[255]

8.819.2.3 swi_uint256_t unpack_dms_GetFirmwareRevision_t::ParamPresenceMask

8.819.2.4 char unpack_dms_GetFirmwareRevision_t::PRIString[255]

8.819.2.5 uint16_t unpack_dms_GetFirmwareRevision_t::Tlvresult

8.820 unpack_dms_GetFirmwareRevisions_t Struct Reference

Data Fields

- uint8_t amssSize
- char AMSSString [255]
- uint8_t bootSize
- char BootString [255]
- uint8_t priSize
- char PRIString [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.820.1 Detailed Description

This structure used to store unpack_dms_GetFirmwareRevisions parameters

Parameters

amssSize	• Maximum number of characters (including NULL terminator) that the AMSS string array can contain • Bit to check in ParamPresenceMask - **1**
AMSSString	• NULL-terminated AMSS revision string • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
bootSize	• Maximum number of characters (including NULL terminator) that the boot string array can contain • Bit to check in ParamPresenceMask - **16**
BootString	• NULL-terminated boot code revision string • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **16**
priSize	• Maximum number of characters (including NULL terminator) that the PRI string array can contain • Bit to check in ParamPresenceMask - **17**
PRIString	• NULL-terminated PRI revision string • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **17**
Tlvresult	• unpack Tlv Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.820.2 Field Documentation

8.820.2.1 uint8_t unpack_dms_GetFirmwareRevisions_t::amssSize

8.820.2.2 char unpack_dms_GetFirmwareRevisions_t::AMSSString[255]

8.820.2.3 uint8_t unpack_dms_GetFirmwareRevisions_t::bootSize

8.820.2.4 char unpack_dms_GetFirmwareRevisions_t::BootString[255]

8.820.2.5 swi_uint256_t unpack_dms_GetFirmwareRevisions_t::ParamPresenceMask

8.820.2.6 uint8_t unpack_dms_GetFirmwareRevisions_t::priSize

8.820.2.7 char unpack_dms_GetFirmwareRevisions_t::PRIString[255]

8.820.2.8 uint16_t unpack_dms_GetFirmwareRevisions_t::Tlvresult

8.821 unpack_dms_GetFSN_t Struct Reference

Data Fields

- char String [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.821.1 Detailed Description

This structure used to store Factory Sequence Number parameter

Parameters

String	• Facorty Sequence Number • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.821.2 Field Documentation

8.821.2.1 swi_uint256_t unpack_dms_GetFSN_t::ParamPresenceMask

8.821.2.2 char unpack_dms_GetFSN_t::String[255]

8.821.2.3 uint16_t unpack_dms_GetFSN_t::Tlvresult

8.822 unpack_dms_GetHardwareRevision_t Struct Reference

Data Fields

- char hwVer [255]
- swi_uint256_t ParamPresenceMask

8.822.1 Detailed Description

This structure used to store unpack_dms_GetHardwareRevision parameters

Parameters

hwVer	- Hardware version • NULL-terminated string • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.822.2 Field Documentation

8.822.2.1 char unpack_dms_GetHardwareRevision_t::hwVer[255]

8.822.2.2 swi_uint256_t unpack_dms_GetHardwareRevision_t::ParamPresenceMask

8.823 unpack_dms_GetIMSI_t Struct Reference

Data Fields

- char imsi [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.823.1 Detailed Description

This structure used to store unpack_dms_GetIMSI parameters

Parameters

imsi	- IMSI no. • NULL-terminated String. • Bit to check in ParamPresenceMask - **1**
Tlvresult	- unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.823.2 Field Documentation

8.823.2.1 char unpack_dms_GetIMSI_t::imsi[255]

8.823.2.2 swi_uint256_t unpack_dms_GetIMSI_t::ParamPresenceMask

8.823.2.3 uint16_t unpack_dms_GetIMSI_t::Tlvresult

8.824 unpack_dms_GetManufacturer_t Struct Reference

Data Fields

- char manufacturer [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.824.1 Detailed Description

This structure is used to store device manufacturer information.

Parameters

manufacturer[OUT]	• NULL terminated string • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.824.2 Field Documentation

8.824.2.1 char unpack_dms_GetManufacturer_t::manufacturer[255]

8.824.2.2 swi_uint256_t unpack_dms_GetManufacturer_t::ParamPresenceMask

8.824.2.3 uint16_t unpack_dms_GetManufacturer_t::Tlvresult

8.825 unpack_dms_GetModelID_t Struct Reference

Data Fields

- char modelid [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.825.1 Detailed Description

This structure used to store unpack_dms_GetModelID parameters

Parameters

modelid	- Device model id. • NULL-terminated String. • Bit to check in ParamPresenceMask - **1**
Tlvresult	- unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.825.2 Field Documentation

8.825.2.1 char unpack_dms_GetModelID_t::modelid[255]

8.825.2.2 swi_uint256_t unpack_dms_GetModelID_t::ParamPresenceMask

8.825.2.3 uint16_t unpack_dms_GetModelID_t::Tlvresult

8.826 unpack_dms_GetNetworkTime_t Struct Reference

Data Fields

- uint16_t source
- uint64_t timestamp
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.826.1 Detailed Description

This structure used to store unpack_dms_GetNetworkTime parameters

Parameters

source	• Source of timestamp 0 - 32 kHz device clock 1 - CDMA network 2 - cdma2000 1xEV-DO network • Bit to check in ParamPresenceMask - **1**
timestamp	• Count of 1.25 ms that have elapsed from the start of GPS time (Jan 6, 1980) • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

The source of the timestamp provided specifies how the timestamp was determined. The first network time that is available will be returned. If no network time is available, the timestamp is taken from the 32 kHz slow-clock of the device.

8.826.2 Field Documentation

8.826.2.1 swi_uint256_t unpack_dms_GetNetworkTime_t::ParamPresenceMask

8.826.2.2 uint16_t unpack_dms_GetNetworkTime_t::source

8.826.2.3 uint64_t unpack_dms_GetNetworkTime_t::timestamp

8.826.2.4 uint16_t unpack_dms_GetNetworkTime_t::Tlvresult

8.827 unpack_dms_GetNetworkTimeV2_t Struct Reference

Data Fields

- uint16_t source
- uint64_t timestamp
- uint64_t * pSysTime
- uint64_t * pUsrTime
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.827.1 Detailed Description

This structure used to store unpack_dms_GetNetworkTimeV2 parameters

Parameters

source	• Source of timestamp 0 - 32 kHz device clock 1 - CDMA network 2 - cdma2000 1xEV-DO network • Bit to check in ParamPresenceMask - **1**
timestamp	• Count of 1.25 ms that have elapsed from the start of GPS time (Jan 6, 1980) • Bit to check in ParamPresenceMask - **1**
pSysTime	• Count of system time in ms that have elapsed from the start of GPS time (Jan 6, 1980) • Bit to check in ParamPresenceMask - **16**
pUsrTime	• Count of user time in ms that have elapsed from the start of GPS time (Jan 6, 1980) • Bit to check in ParamPresenceMask - **17**
Tlvresult	• Unpack Result

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

The source of the timestamp provided specifies how the timestamp was determined. The first network time that is available will be returned. If no network time is available, the timestamp is taken from the 32 kHz slow-clock of the device.

8.827.2 Field Documentation

8.827.2.1 swi_uint256_t unpack_dms_GetNetworkTimeV2_t::ParamPresenceMask

8.827.2.2 uint64_t* unpack_dms_GetNetworkTimeV2_t::pSysTime

8.827.2.3 uint64_t* unpack_dms_GetNetworkTimeV2_t::pUsrTime

8.827.2.4 uint16_t unpack_dms_GetNetworkTimeV2_t::source

8.827.2.5 uint64_t unpack_dms_GetNetworkTimeV2_t::timestamp

8.827.2.6 uint16_t unpack_dms_GetNetworkTimeV2_t::Tlvresult

8.828 unpack_dms_GetOfflineReason_t Struct Reference

Data Fields

- uint32_t * pReasonMask
- uint32_t * pbPlatform
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.828.1 Detailed Description

This structure is used to store reason why the operating mode of the device is currently offline.

Parameters

pReasonMask[-OUT]	• Optional parameter • Bitmask of offline reasons – 0x00000001 - Host image configuration issue – 0x00000002 - PRI image configuration issue – 0x00000004 - PRI version incompatible – 0x00000008 - PRI copy issue – All others - Reserved • Bit to check in ParamPresenceMask - **16**

pbPlatform[OUT]	• Optional parameter • Is the device offline due to a platform restriction? – 0 - No – 1 - Yes • Bit to check in ParamPresenceMask - **17**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.828.2 Field Documentation

8.828.2.1 swi_uint256_t unpack_dms_GetOfflineReason_t::ParamPresenceMask

8.828.2.2 uint32_t* unpack_dms_GetOfflineReason_t::pbPlatform

8.828.2.3 uint32_t* unpack_dms_GetOfflineReason_t::pReasonMask

8.828.2.4 uint16_t unpack_dms_GetOfflineReason_t::Tlvresult

8.829 unpack_dms_GetPower_t Struct Reference

Data Fields

- uint32_t OperationMode
- uint32_t OfflineReason
- uint32_t HardwareControlledMode
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.829.1 Detailed Description

This structure used to store unpack_dms_GetPower parameters

Parameters

OperationMode	- operating mode. • 0 - Online (default). • 1 - Low power (airplane) mode. • 2 - Factory test mode. • 3 - Offline. • 4 - Resetting. • 5 - Power off. • 6 - Persistent low power (airplane) mode. • 7 - Mode - only low power. • Bit to check in ParamPresenceMask - **1**

Note

Valid transitions for Power Modes

- Online to Low Power, Persistent low power, Factory test, Offline or Shut Down
- Low power to online, Persistent low power, Offline, or Shut Down
- Persistent low power to Online, Low power, Offline or Shut down
- Factory test to online
- Offline to Reset

Parameters

OfflineReason	- offline reason. • 0x0001 - Host image misconfiguration. • 0x0002 - PRI image misconfiguration. • 0x0004 - PRI version incompatible. • 0x0008 - Device memory is full,cannot copy PRI information. • Bit to check in ParamPresenceMask - **16**
Hardware-ControlledMode	- hardware restricted mode. • 0x00 - FALSE. • 0x01 - TRUE. • Bit to check in ParamPresenceMask - **17**
Tlvresult	- unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.829.2 Field Documentation

8.829.2.1 uint32_t unpack_dms_GetPower_t::HardwareControlledMode

8.829.2.2 uint32_t unpack_dms_GetPower_t::OfflineReason

8.829.2.3 uint32_t unpack_dms_GetPower_t::OperationMode

8.829.2.4 swi_uint256_t unpack_dms_GetPower_t::ParamPresenceMask

8.829.2.5 uint16_t unpack_dms_GetPower_t::Tlvresult

8.830 unpack_dms_GetPRLVersion_t Struct Reference

Data Fields

- uint8_t u8PRLPreference
- uint16_t u16PRLVersion
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.830.1 Detailed Description

This structure used to store unpack_dms_GetPRLVersion parameters

Parameters

u8PRL-Preference	• PRL Preference – 0 - Unset – 1 - Set • Bit to check in ParamPresenceMask - **17**
u16PRLVersion	• PRL version of device. • Bit to check in ParamPresenceMask - **16**
Tlvresult	• unpack Tlv Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.830.2 Field Documentation

8.830.2.1 swi_uint256_t unpack_dms_GetPRLVersion_t::ParamPresenceMask

8.830.2.2 uint16_t unpack_dms_GetPRLVersion_t::Tlvresult

8.830.2.3 uint16_t unpack_dms_GetPRLVersion_t::u16PRLVersion

8.830.2.4 uint8_t unpack_dms_GetPRLVersion_t::u8PRLPreference

8.831 unpack_dms_GetSerialNumbers_t Struct Reference

Data Fields

- char esn [255]
- char imei_no [255]
- char meid [255]
- char imeisv_svn [255]
- swi_uint256_t ParamPresenceMask

8.831.1 Detailed Description

This structure used to store unpack_dms_GetSerialNumbers parameters

Parameters

esn	- Electronic Serial Number of the device. • NULL-terminated ESN string. Empty string is returned when ESN is not supported/programmed. • Bit to check in ParamPresenceMask - **16**
imei_no	- International Mobile Equipment Identity of the device. • NULL terminated IMEI string. Empty string is returned when IMEI is not supported/programmed. • Bit to check in ParamPresenceMask - **17**

meid	- Mobile Equipment Identifier of the device. • NULL-terminated MEID string. Empty string is returned when MEID is not supported/programmed. • Bit to check in ParamPresenceMask - **18**
imeisv_svn	- NULL-terminated IMEI SVN string. Empty string is returned when IMEI SVN is not supported/programmed.imei software version revision. • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.831.2 Field Documentation

8.831.2.1 char unpack_dms_GetSerialNumbers_t::esn[255]

8.831.2.2 char unpack_dms_GetSerialNumbers_t::imei_no[255]

8.831.2.3 char unpack_dms_GetSerialNumbers_t::imeisv_svn[255]

8.831.2.4 char unpack_dms_GetSerialNumbers_t::meid[255]

8.831.2.5 swi_uint256_t unpack_dms_GetSerialNumbers_t::ParamPresenceMask

8.832 unpack_dms_GetUSBComp_t Struct Reference

Data Fields

- uint8_t USBComp
- uint8_t NumSupUSBComps
- uint8_t SupUSBComps [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.832.1 Detailed Description

This structure is used to store retrieved USB Composition

Parameters

USBComp	• Current USB Composition(optional parameter) • Values: – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM • Bit to check in ParamPresenceMask - **16**
NumSupUSB-Comps	• Number of supported USB compositions in the parameter to follow • Range - 0-255 • Bit to check in ParamPresenceMask - **17**

SupUSBComps	• Optional parameter • List of supported USB compositions(1 Byte each - Max 255) • Total length is defined by pNumSupUSBComps parameter • Values: – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM • Bit to check in ParamPresenceMask - **17**
Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.832.2 Field Documentation

8.832.2.1 uint8_t unpack_dms_GetUSBComp_t::NumSupUSBComps

8.832.2.2 swi_uint256_t unpack_dms_GetUSBComp_t::ParamPresenceMask

8.832.2.3 uint8_t unpack_dms_GetUSBComp_t::SupUSBComps[255]

8.832.2.4 uint16_t unpack_dms_GetUSBComp_t::Tlvresult

8.832.2.5 uint8_t unpack_dms_GetUSBComp_t::USBComp

8.833 unpack_dms_GetVoiceNumber_t Struct Reference

Data Fields

- uint8_t voiceNumberSize

- char VoiceNumber [255]
- uint8_t minSize
- char MIN [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.833.1 Detailed Description

Structure to store the voice number in use by the device

Parameters

voiceNumber-Size	• Maximum number of characters (including NULL terminator) that the voice number array can contain. • Bit to check in ParamPresenceMask - **1**
VoiceNumber	• Voice number string: MDN or MS ISDN • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
minSize	• Maximum number of characters (including NULL terminator) that the MIN array can contain. • Bit to check in ParamPresenceMask - **16**
MIN	• Optional Parameter • MIN string: Empty string returned when MIN is not supported/ programmed. • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **16**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.833.2 Field Documentation

8.833.2.1 char unpack_dms_GetVoiceNumber_t::MIN[255]

8.833.2.2 uint8_t unpack_dms_GetVoiceNumber_t::minSize

8.833.2.3 swi_uint256_t unpack_dms_GetVoiceNumber_t::ParamPresenceMask

8.833.2.4 uint16_t unpack_dms_GetVoiceNumber_t::Tlvresult

8.833.2.5 char unpack_dms_GetVoiceNumber_t::VoiceNumber[255]

8.833.2.6 uint8_t unpack_dms_GetVoiceNumber_t::voiceNumberSize

8.834 unpack_dms_PSMCfgChange_ind_t Struct Reference

Data Fields

- dms_PSMEnableStateIndTlv EnableState
- dms_PSMActiveTimerIndTlv ActiveTimer
- dms_PSMPeriodicUpdateTimerIndTlv PeriodicUpdateTimer
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.834.1 Detailed Description

DMS Event Report indication structure

Parameters

EnableState	• See dms_PSMEnableStateIndTlv • Bit to check in ParamPresenceMask - **16**
ActiveTimer	• See dms_PSMActiveTimerIndTlv • Bit to check in ParamPresenceMask - **17**
PeriodicUpdate-Timer	• See dms_PSMPeriodicUpdateTimerIndTlv • Bit to check in ParamPresenceMask - **18**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.834.2 Field Documentation

8.834.2.1 dms_PSMActiveTimerIndTlv unpack_dms_PSMCfgChange_ind_t::ActiveTimer

8.834.2.2 dms_PSMEnableStateIndTlv unpack_dms_PSMCfgChange_ind_t::EnableState

8.834.2.3 swi_uint256_t unpack_dms_PSMCfgChange_ind_t::ParamPresenceMask

8.834.2.4 dms_PSMPeriodicUpdateTimerIndTlv unpack_dms_PSMCfgChange_ind_t::PeriodicUpdateTimer

8.834.2.5 uint16_t unpack_dms_PSMCfgChange_ind_t::Tlvresult

8.835 unpack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.835.1 Detailed Description

This structure contains reset to factory default unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.835.2 Field Documentation

8.835.2.1 swi_uint256_t unpack_dms_ResetToFactoryDefaults_t::ParamPresenceMask

8.835.2.2 uint16_t unpack_dms_ResetToFactoryDefaults_t::Tlvresult

8.836 unpack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.836.1 Detailed Description

This structure is used to store Set Service Activation Status callback parameter unpack.

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.836.2 Field Documentation

8.836.2.1 swi_uint256_t unpack_dms_SetActivationStatusCallback_t::ParamPresenceMask

8.836.2.2 uint16_t unpack_dms_SetActivationStatusCallback_t::Tlvresult

8.837 unpack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t notused
- swi_uint256_t ParamPresenceMask

8.837.1 Detailed Description

This structure is used to store unpack_dms_SetCrashAction parameters

Parameters

notused	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.837.2 Field Documentation

8.837.2.1 uint8_t unpack_dms_SetCrashAction_t::notused

8.837.2.2 swi_uint256_t unpack_dms_SetCrashAction_t::ParamPresenceMask

8.838 unpack_dms_SetCustFeature_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.838.1 Detailed Description

This structure is used to store unpack_dms_SetCustFeature parameters

Parameters

Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.838.2 Field Documentation

8.838.2.1 swi_uint256_t unpack_dms_SetCustFeature_t::ParamPresenceMask

8.838.2.2 uint16_t unpack_dms_SetCustFeature_t::Tlvresult

8.839 unpack_dms_SetCustFeaturesV2_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.839.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.839.2 Field Documentation

8.839.2.1 swi_uint256_t unpack_dms_SetCustFeaturesV2_t::ParamPresenceMask

8.839.2.2 uint16_t unpack_dms_SetCustFeaturesV2_t::Tlvresult

8.840 unpack_dms_SetEventReport_ind_t Struct Reference

Data Fields

- dms_ActivationStatusTlv ActivationStatusTlv
- dms_OperatingModeTlv OperatingModeTlv
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.840.1 Detailed Description

DMS Event Report indication structure

Parameters

ActivationStatus-Tlv	• See dms_ActivationStatusTlv • Bit to check in ParamPresenceMask - **19**
OperatingMode-Tlv	• See dms_OperatingModeTlv • Bit to check in ParamPresenceMask - **20**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.840.2 Field Documentation

8.840.2.1 dms_ActivationStatusTlv unpack_dms_SetEventReport_ind_t::ActivationStatusTlv

8.840.2.2 dms_OperatingModeTlv unpack_dms_SetEventReport_ind_t::OperatingModeTlv

8.840.2.3 swi_uint256_t unpack_dms_SetEventReport_ind_t::ParamPresenceMask

8.840.2.4 uint16_t unpack_dms_SetEventReport_ind_t::Tlvresult

8.841 unpack_dms_SetEventReport_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.841.1 Detailed Description

This structure is used to store unpack_dms_SetEventReport parameters.

Parameters

Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.841.2 Field Documentation

8.841.2.1 swi_uint256_t unpack_dms_SetEventReport_t::ParamPresenceMask

8.841.2.2 uint16_t unpack_dms_SetEventReport_t::Tlvresult

8.842 unpack_dms_SetFirmwarePreference_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.842.1 Detailed Description

This structure is used to store unpack_dms_SetFirmwarePreference parameters

Parameters

Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.842.2 Field Documentation

8.842.2.1 swi_uint256_t unpack_dms_SetFirmwarePreference_t::ParamPresenceMask

8.842.2.2 uint16_t unpack_dms_SetFirmwarePreference_t::Tlvresult

8.843 unpack_dms_SetIndicationRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.843.1 Detailed Description

This structure is used to store unpack_dms_SetIndicationRegister parameters.

Parameters

Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.843.2 Field Documentation

8.843.2.1 swi_uint256_t unpack_dms_SetIndicationRegister_t::ParamPresenceMask

8.843.2.2 uint16_t unpack_dms_SetIndicationRegister_t::Tlvresult

8.844 unpack_dms_SetPower_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.844.1 Detailed Description

Structure to store unpack the operating mode of the device.

Parameters

Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.844.2 Field Documentation

8.844.2.1 swi_uint256_t unpack_dms_SetPower_t::ParamPresenceMask

8.844.2.2 uint16_t unpack_dms_SetPower_t::Tlvresult

8.845 unpack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.845.1 Detailed Description

This structure is used to store unpack USB composition information

Parameters

Tlvresult	• unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.845.2 Field Documentation

8.845.2.1 swi_uint256_t unpack_dms_SetUSBComp_t::ParamPresenceMask

8.845.2.2 uint16_t unpack_dms_SetUSBComp_t::Tlvresult

8.846 unpack_dms_SLQSDmsSwiGetPCInfo_t Struct Reference

Data Fields

- uint8_t opMode
- int has_LpmFlag
- uint32_t LpmFlag
- int has_Wdisable
- uint8_t Wdisable
- int has_PowerOffMode
- uint8_t PowerOffMode
- int has_PersistentLpm
- uint8_t PersistentLpm
- swi_uint256_t ParamPresenceMask

8.846.1 Detailed Description

This structure contains the TLV required to Get device power control status information.

Parameters

opMode	• Selected operating mode. • Values – 0 - Online – 1 - Low power – 2 - Factory Test mode – 3 - Offline – 4 - Resetting – 5 - Shutting down – 6 - Persistent low power – 8 - Conducting network test for GSM/WCDMA • Bit to check in ParamPresenceMask - **1**
has_LpmFlag	LPM Flag Availability. • 0 : Unavailable. • 1 : Available.

• Bit to check in ParamPresenceMask - **16**

Parameters

LpmFlag[-Optional]	• LPM Force Flags • Bitmask – bit0 - Tracks the state of W_DISABLE TLV, indicating state of the W_DISABLE switch. – bit1 - Set if LPM is requested by a host request, such as AT command, QMI or MBIM request. Cleared by a host request to return to online mode. Also set when the device is cold or warm booted in persistent LPM. – bit2 - Set when the device temperature is outside the valid operating range. Cleared if the temperature returns to the normal range. – bit3 - Set when the device voltage is outside the valid operating range. Cleared if the voltage returns to the normal range. – bit4 - Set on power up when BIOS locking is enabled. Cleared when the host has disabled the BIOS lock. – bit5 - Set if the current device configuration does not match the GOBI image preference. – bit6-31 - Additional LPM causes may be added to future products. • Bit to check in ParamPresenceMask - **16**
has_Wdisable	W-Disable Availability. • 0 : Unavailable. • 1 : Available.

• Bit to check in ParamPresenceMask - **17**

Parameters

Wdisable[-Optional]	• W_DISABLE • Values – 0 - Switch set to ON position – 1 - Switch set to OFF position • Bit to check in ParamPresenceMask - **17**
has_PowerOff-Mode	Power off mode Availability. • 0 : Unavailable. • 1 : Available.

• Bit to check in ParamPresenceMask - **18**

Parameters

PowerOffMode[-Optional]	• Power-off Mode • Action taken when W_DISABLE is switched to the OFF position – 0 - LPM – 1 - Shutdown – 2 - Ignore • Bit to check in ParamPresenceMask - **18**
has_Persistent-Lpm	Persistent LPM Availability. • 0 : Unavailable. • 1 : Available.

• Bit to check in ParamPresenceMask - **19**

Parameters

PersistentLpm[-Optional]	• LPM Persistence • Values – 0 - Non-persistent LPM – 1 - Persistent LPM • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.846.2 Field Documentation

8.846.2.1 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_LpmFlag

8.846.2.2 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_PersistentLpm

8.846.2.3 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_PowerOffMode

8.846.2.4 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_Wdisable

8.846.2.5 uint32_t unpack_dms_SLQSDmsSwiGetPCInfo_t::LpmFlag

8.846.2.6 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::opMode

8.846.2.7 swi_uint256_t unpack_dms_SLQSDmsSwiGetPCInfo_t::ParamPresenceMask

8.846.2.8 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::PersistentLpm

8.846.2.9 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::PowerOffMode

8.846.2.10 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::Wdisable

8.847 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.847.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

type[OUT]	• type of reset or power down, possible values listed below: – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down • Bit to check in ParamPresenceMask - **1**
source[OUT]	• entity which initiated the reset or power down, possible values listed below: – 0 - unknown – 1 - user requested (AT!RESET, AT!BOOTHOLD, FW/PRI download - including host-initiated image switching) – 2 - hardware switch (W_DISABLE) – 3 - temperature critical – 4 - voltage critical – 5 - configuration update (SIM-based image switching, RMA reset, NVUPs which request a reset) – 6 - LWM2M (Light Weight M2M client (internal process for LWM2M)) – 7 - OMA-DM – 8 - FOTA

- Bit to check in ParamPresenceMask - **1**

Parameters

Tlvresult[OUT]	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.847.2 Field Documentation

8.847.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::ParamPresenceMask

8.847.2.2 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::source

8.847.2.3 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::Tlvresult

8.847.2.4 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::type

8.848 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.848.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

type[OUT]	• type of reset or power down, possible values listed below: – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down • Bit to check in ParamPresenceMask - **1**

source[OUT]	• entity which initiated the reset or power down, possible values listed below: – 0 - unknown – 1 - user requested (AT!RESET, AT!BOOTHOLD, FW/PRI download - including host-initiated image switching) – 2 - hardware switch (W_DISABLE) – 3 - temperature critical – 4 - voltage critical – 5 - configuration update (SIM-based image switching, RMA reset, NVUPs which request a reset) – 6 - LWM2M (Light Weight M2M client (internal process for LWM2M)) – 7 - OMA-DM – 8 - FOTA

- Bit to check in ParamPresenceMask - **1**

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.848.2 Field Documentation

8.848.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiGetResetInfo_t::ParamPresenceMask

8.848.2.2 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::source

8.848.2.3 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_t::Tlvresult

8.848.2.4 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::type

8.849 unpack_dms_SLQSDmsSwiGetUimSelection_t Struct Reference

Data Fields

- uint8_t uimSelect
- dms_UimAutoSwitchActSlotTlv * pUimAutoSwitchActSlot
- swi_uint256_t ParamPresenceMask

8.849.1 Detailed Description

Structure contains the TLV required to Get currently selected active UIM slot.

Parameters

uimSelect	• Active selected UIM slot • Values – 0 - slot 1 (e.g. external SIM) – 1 - slot 2 (e.g. embedded SIM) – 2 - remote SIM (if supported) – 3 - SIM auto-switch activated

- Bit to check in ParamPresenceMask - **1**

Parameters

pUimAuto-SwitchActSlot	• UIM Auto Switch Active slot • see dms_UimAutoSwitchActSlotTlv for more information • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

: None

8.849.2 Field Documentation

8.849.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiGetUimSelection_t::ParamPresenceMask

8.849.2.2 dms_UimAutoSwitchActSlotTlv* unpack_dms_SLQSDmsSwiGetUimSelection_t::pUimAutoSwitchActSlot

8.849.2.3 uint8_t unpack_dms_SLQSDmsSwiGetUimSelection_t::uimSelect

8.850 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.850.1 Detailed Description

This structure contains set registration state for different indication unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.850.2 Field Documentation

8.850.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiIndicationRegister_t::ParamPresenceMask

8.850.2.2 uint16_t unpack_dms_SLQSDmsSwiIndicationRegister_t::Tlvresult

8.851 unpack_dms_SLQSGetBandCapability_t Struct Reference

Data Fields

- uint64_t bandCapability
- int is_LteBandCapability_Available
- uint64_t LteBandCapability
- int is_TdsBandCapability_Available
- uint64_t TdsBandCapability
- swi_uint256_t ParamPresenceMask

8.851.1 Detailed Description

This structure contains the Band Capabilities response.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

bandCapability[-OUT]	Bitmask of bands supported by the device • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19 • Bit 60 - WCDMA Japan 850 band • Bit 61 - WCDMA 1500 band • Bits 62 and 63 - Reserved • Bit to check in ParamPresenceMask - **1**

is_LteBand-Capability_-Available[OUT]	LTE Band Capabilities Availability. • 0 : Unavailable. • 1 : Available. • Bit to check in ParamPresenceMask - **16**

LteBand-Capability[OUT]	Bitmask of LTE bands supported by the device • Bit 0 - LTE EUTRAN Band 1 UL:1920-1980; DL: 2110-2170 • Bit 1 - LTE EUTRAN Band 2 UL:1850-1910; DL: 1930-1990 • Bit 2 - LTE EUTRAN Band 3 UL:1710-1785; DL: 1805-1880 • Bit 3 - LTE EUTRAN Band 4 UL:1710-1755; DL: 2110-2155 • Bit 4 - LTE EUTRAN Band 5 UL: 824-849; DL: 869-894 • Bit 5 - LTE EUTRAN Band 6 UL: 830-840; DL: 875-885 • Bit 6 - LTE EUTRAN Band 7 UL:2500-2570; DL: 2620-2690 • Bit 7 - LTE EUTRAN Band 8 UL: 880-915; DL: 925-960 • Bit 8 - LTE EUTRAN Band 9 UL:1749.9-1784.9; DL: 1844.9-1879.9 • Bit 9 - LTE EUTRAN Band 10 UL:1710-1770; DL: 2110-2170 • Bit 10 - LTE EUTRAN Band 11 UL:1427.9-1452.9; DL: 1475.9-1500.9 • Bit 11 - LTE EUTRAN Band 12 UL:698-716; DL: 728-746 • Bit 12 - LTE EUTRAN Band 13 UL: 777-787; DL: 746-756 • Bit 13 - LTE EUTRAN Band 14 UL: 788-798; DL: 758-768 • Bits 14 and 15 - Reserved • Bit 16 - LTE EUTRAN Band 17 UL: 704-716; DL: 734-746 • Bit 17 - LTE EUTRAN Band 18 UL: 815-830; DL: 860-875 • Bit 18 - LTE EUTRAN Band 19 UL: 830-845; DL: 875-890 • Bit 19 - LTE EUTRAN Band 20 UL: 832-862; DL: 791-821 • Bit 20 - LTE EUTRAN Band 21 UL: 1447.9-1462.9; DL: 1495.9-1510.9 • Bit 21 - Reserved • Bit 22 - LTE EUTRAN Band 23 UL: 2000-2020; DL: 2180-2200 • Bit 23 - LTE EUTRAN Band 24 UL: 1626.5-1660.5; DL: 1525-1559 • Bit 24 - LTE EUTRAN Band 25 UL: 1850-1915; DL: 1930-1995 • Bit 25 - LTE EUTRAN Band 26 UL: 814-849; DL: 859-894 • Bit 26 - Reserved • Bit 27 - LTE EUTRAN Band 28 UL: 703-748; DL: 758-803 • Bit 28 - LTE EUTRAN Band 29 UL: 1850-1910 or 1710-1755; DL: 716-728 • Bit 29 - LTE EUTRAN Band 30 UL: 2350-2360; DL: 2305-2315 • Bit 30 - Reserved • Bit 31 - LTE EUTRAN Band 32 DL: 9920-10359 • Bit 32 - LTE EUTRAN Band 33 UL: 1900-1920; DL: 1900-1920 • Bit 33 - LTE EUTRAN Band 34 UL: 2010-2025; DL: 2010-2025 • Bit 34 - LTE EUTRAN Band 35 UL: 1850-1910; DL: 1850-1910 • Bit 35 - LTE EUTRAN Band 36 UL: 1930-1990; DL: 1930-1990 • Bit 36 - LTE EUTRAN Band 37 UL: 1910-1930; DL: 1910-1930 • Bit 37 - LTE EUTRAN Band 38 UL: 2570-2620; DL: 2570-2620 • Bit 38 - LTE EUTRAN Band 39 UL: 1880-1920; DL: 1880-1920 • Bit 39 - LTE EUTRAN Band 40 UL: 2300-2400; DL: 2300-2400 • Bit 40 - LTE EUTRAN Band 41 UL: 2496-2690; DL: 2496-2690 • Bit 41 - LTE EUTRAN Band 42 UL: 3400-3600; DL: 3400-3600 • Bit 42 - LTE EUTRAN Band 43 UL: 3600-3800; DL: 3600-3800 • Bits 43 through 64 - Reserved • Bit to check in ParamPresenceMask - **16**

is_TdsBand-Capability_-Available[OUT]	TDS Band Capabilities Availability. • 0 : Unavailable. • 1 : Available. • Bit to check in ParamPresenceMask - **17**
TdsBand-Capability[OUT]	Bitmask of TDS bands supported by the device. • Bit 0 - TDS Band A 1900 to 1920 MHz, 2010 to 2020 MHz • Bit 1 - TDS Band B 1850 to 1910 MHz, 1930 to 1990 MHz • Bit 2 - TDS Band C 1910 to 1930 MHz • Bit 3 - TDS Band D 2570 to 2620 MHz • Bit 4 - TDS Band E 2300 to 2400 MHz • Bit 5 - TDS Band F 1880 to 1920 MHz • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.851.2 Field Documentation

8.851.2.1 uint64_t unpack_dms_SLQSGetBandCapability_t::bandCapability

8.851.2.2 int unpack_dms_SLQSGetBandCapability_t::is_LteBandCapability_Available

8.851.2.3 int unpack_dms_SLQSGetBandCapability_t::is_TdsBandCapability_Available

8.851.2.4 uint64_t unpack_dms_SLQSGetBandCapability_t::LteBandCapability

8.851.2.5 swi_uint256_t unpack_dms_SLQSGetBandCapability_t::ParamPresenceMask

8.851.2.6 uint64_t unpack_dms_SLQSGetBandCapability_t::TdsBandCapability

8.852 unpack_dms_SLQSGetBandCapabilityExt_t Struct Reference

Data Fields

- uint64_t bandCapability
- int is_LteBandCapability_Available
- uint64_t LteBandCapability
- int is_TdsBandCapability_Available
- uint64_t TdsBandCapability
- dms_LteBandsSupport LteBandsSupport
- swi_uint256_t ParamPresenceMask

8.852.1 Detailed Description

This structure contains the Band Capabilities response.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

bandCapability[-OUT]	Bitmask of bands supported by the device • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19 • Bit 60 - WCDMA Japan 850 band • Bit 61 - WCDMA 1500 band • Bits 62 and 63 - Reserved • Bit to check in ParamPresenceMask - **1**

is_LteBand-Capability_-Available[OUT]	LTE Band Capabilities Availability. • 0 : Unavailable. • 1 : Available. • Bit to check in ParamPresenceMask - **16**

LteBand-Capability[OUT]	Bitmask of LTE bands supported by the device • Bit 0 - LTE EUTRAN Band 1 UL:1920-1980; DL: 2110-2170 • Bit 1 - LTE EUTRAN Band 2 UL:1850-1910; DL: 1930-1990 • Bit 2 - LTE EUTRAN Band 3 UL:1710-1785; DL: 1805-1880 • Bit 3 - LTE EUTRAN Band 4 UL:1710-1755; DL: 2110-2155 • Bit 4 - LTE EUTRAN Band 5 UL: 824-849; DL: 869-894 • Bit 5 - LTE EUTRAN Band 6 UL: 830-840; DL: 875-885 • Bit 6 - LTE EUTRAN Band 7 UL:2500-2570; DL: 2620-2690 • Bit 7 - LTE EUTRAN Band 8 UL: 880-915; DL: 925-960 • Bit 8 - LTE EUTRAN Band 9 UL:1749.9-1784.9; DL: 1844.9-1879.9 • Bit 9 - LTE EUTRAN Band 10 UL:1710-1770; DL: 2110-2170 • Bit 10 - LTE EUTRAN Band 11 UL:1427.9-1452.9; DL: 1475.9-1500.9 • Bit 11 - LTE EUTRAN Band 12 UL:698-716; DL: 728-746 • Bit 12 - LTE EUTRAN Band 13 UL: 777-787; DL: 746-756 • Bit 13 - LTE EUTRAN Band 14 UL: 788-798; DL: 758-768 • Bits 14 and 15 - Reserved • Bit 16 - LTE EUTRAN Band 17 UL: 704-716; DL: 734-746 • Bit 17 - LTE EUTRAN Band 18 UL: 815-830; DL: 860-875 • Bit 18 - LTE EUTRAN Band 19 UL: 830-845; DL: 875-890 • Bit 19 - LTE EUTRAN Band 20 UL: 832-862; DL: 791-821 • Bit 20 - LTE EUTRAN Band 21 UL: 1447.9-1462.9; DL: 1495.9-1510.9 • Bit 21 - Reserved • Bit 22 - LTE EUTRAN Band 23 UL: 2000-2020; DL: 2180-2200 • Bit 23 - LTE EUTRAN Band 24 UL: 1626.5-1660.5; DL: 1525-1559 • Bit 24 - LTE EUTRAN Band 25 UL: 1850-1915; DL: 1930-1995 • Bit 25 - LTE EUTRAN Band 26 UL: 814-849; DL: 859-894 • Bit 26 - Reserved • Bit 27 - LTE EUTRAN Band 28 UL: 703-748; DL: 758-803 • Bit 28 - LTE EUTRAN Band 29 UL: 1850-1910 or 1710-1755; DL: 716-728 • Bit 29 - LTE EUTRAN Band 30 UL: 2350-2360; DL: 2305-2315 • Bit 30 - Reserved • Bit 31 - LTE EUTRAN Band 32 DL: 9920-10359 • Bit 32 - LTE EUTRAN Band 33 UL: 1900-1920; DL: 1900-1920 • Bit 33 - LTE EUTRAN Band 34 UL: 2010-2025; DL: 2010-2025 • Bit 34 - LTE EUTRAN Band 35 UL: 1850-1910; DL: 1850-1910 • Bit 35 - LTE EUTRAN Band 36 UL: 1930-1990; DL: 1930-1990 • Bit 36 - LTE EUTRAN Band 37 UL: 1910-1930; DL: 1910-1930 • Bit 37 - LTE EUTRAN Band 38 UL: 2570-2620; DL: 2570-2620 • Bit 38 - LTE EUTRAN Band 39 UL: 1880-1920; DL: 1880-1920 • Bit 39 - LTE EUTRAN Band 40 UL: 2300-2400; DL: 2300-2400 • Bit 40 - LTE EUTRAN Band 41 UL: 2496-2690; DL: 2496-2690 • Bit 41 - LTE EUTRAN Band 42 UL: 3400-3600; DL: 3400-3600 • Bit 42 - LTE EUTRAN Band 43 UL: 3600-3800; DL: 3600-3800 • Bits 43 through 64 - Reserved • Bit to check in ParamPresenceMask - **16**

is_TdsBand-Capability_-Available[OUT]	TDS Band Capabilities Availability. • 0 : Unavailable. • 1 : Available. • Bit to check in ParamPresenceMask - **17**
TdsBand-Capability[OUT]	Bitmask of TDS bands supported by the device. • Bit 0 - TDS Band A 1900 to 1920 MHz, 2010 to 2020 MHz • Bit 1 - TDS Band B 1850 to 1910 MHz, 1930 to 1990 MHz • Bit 2 - TDS Band C 1910 to 1930 MHz • Bit 3 - TDS Band D 2570 to 2620 MHz • Bit 4 - TDS Band E 2300 to 2400 MHz • Bit 5 - TDS Band F 1880 to 1920 MHz • Bit to check in ParamPresenceMask - **17**
LteBands-Support[OUT]	• See dms_LteBandsSupport for more information
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.852.2 Field Documentation

8.852.2.1 uint64_t unpack_dms_SLQSGetBandCapabilityExt_t::bandCapability

8.852.2.2 int unpack_dms_SLQSGetBandCapabilityExt_t::is_LteBandCapability_Available

8.852.2.3 int unpack_dms_SLQSGetBandCapabilityExt_t::is_TdsBandCapability_Available

8.852.2.4 uint64_t unpack_dms_SLQSGetBandCapabilityExt_t::LteBandCapability

8.852.2.5 dms_LteBandsSupport unpack_dms_SLQSGetBandCapabilityExt_t::LteBandsSupport

8.852.2.6 swi_uint256_t unpack_dms_SLQSGetBandCapabilityExt_t::ParamPresenceMask

8.852.2.7 uint64_t unpack_dms_SLQSGetBandCapabilityExt_t::TdsBandCapability

8.853 unpack_dms_SLQSGetERIFile_t Struct Reference

Data Fields

- eriDataparams eriFile
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.853.1 Detailed Description

This structure contains Get ERI file unpack

Parameters

eriFile	• Pointer to structure ERIFileparams • See eriDataparams for more information • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.853.2 Field Documentation

8.853.2.1 eriDataparams unpack_dms_SLQSGetERIFile_t::eriFile

8.853.2.2 swi_uint256_t unpack_dms_SLQSGetERIFile_t::ParamPresenceMask

8.853.2.3 uint16_t unpack_dms_SLQSGetERIFile_t::Tlvresult

8.854 unpack_dms_SLQSGetPowerSaveModeConfig_t Struct Reference

Data Fields

- dms_PSMEnableStateTlv * pPsmEnableState
- dms_PSMDurationThresholdTlv * pDurationThreshold
- dms_PSMDurationDueToOOSTlv * pDurationDueToOOS
- dms_PSMRandomizationWindowTlv * pRandomizationWindow
- dms_PSMActiveTimerTlv * pActiveTimer
- dms_PSMPeriodicUpdateTimerTlv * pPeriodicUpdateTimer
- dms_PSMEarlyWakeupTimeTlv * pEarlyWakeupTime
- swi_uint256_t ParamPresenceMask

8.854.1 Detailed Description

This structure contains the Power Save Mode (PSM) configuration parameter.

Parameters

pPsmEnable-State	[Optional] • See dms_PSMEnableStateTlv for more information • Bit to check in ParamPresenceMask - **16**
pDuration-Threshold	[Optional] • See dms_PSMDurationThresholdTlv for more information • Bit to check in ParamPresenceMask - **17**
pDurationDueTo-OOS	[Optional] • See dms_PSMDurationDueToOOSTlv for more information • Bit to check in ParamPresenceMask - **18**

pRandomization-Window	[Optional] • See dms_PSMRandomizationWindowTlv for more information • Bit to check in ParamPresenceMask - **19**
pActiveTimer	[Optional] • See dms_PSMActiveTimerTlv for more information • Bit to check in ParamPresenceMask - **20**
pPeriodic-UpdateTimer	[Optional] • See dms_PSMPeriodicUpdateTimerTlv for more information • Bit to check in ParamPresenceMask - **21**
pEarlyWakeup-Time	[Optional] • See dms_PSMEarlyWakeupTimeTlv for more information • Bit to check in ParamPresenceMask - **22**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.854.2 Field Documentation

8.854.2.1 dms_PSMActiveTimerTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pActiveTimer

8.854.2.2 swi_uint256_t unpack_dms_SLQSGetPowerSaveModeConfig_t::ParamPresenceMask

8.854.2.3 dms_PSMDurationDueToOOSTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pDurationDueToOOS

8.854.2.4 dms_PSMDurationThresholdTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pDurationThreshold

8.854.2.5 dms_PSMEarlyWakeupTimeTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pEarlyWakeupTime

8.854.2.6 dms_PSMPeriodicUpdateTimerTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pPeriodicUpdateTimer

8.854.2.7 dms_PSMEnableStateTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pPsmEnableState

8.854.2.8 dms_PSMRandomizationWindowTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pRandomization-Window

8.855 unpack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.855.1 Detailed Description

This structure is used to store unpack_dms_SLQSSetPowerSaveModeConfig parameters.

Parameters

Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.855.2 Field Documentation

8.855.2.1 swi_uint256_t unpack_dms_SLQSSetPowerSaveModeConfig_t::ParamPresenceMask

8.855.2.2 uint16_t unpack_dms_SLQSSetPowerSaveModeConfig_t::Tlvresult

8.856 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.856.1 Detailed Description

This structure contains Clear Dying GASP unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.856.2 Field Documentation

8.856.2.1 swi_uint256_t unpack_dms_SLQSSwiClearDyingGaspStatistics_t::ParamPresenceMask

8.856.2.2 uint16_t unpack_dms_SLQSSwiClearDyingGaspStatistics_t::Tlvresult

8.857 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult
- crashInfoParams crashInfoParam
- swi_uint256_t ParamPresenceMask

8.857.1 Detailed Description

This structure contains SWI get crash information unpack information

Parameters

TlvResult	• Unpack Result
crashInfoParam[-OUT]	• See crashInfoParams • crashStatus – Bit to check in ParamPresenceMask - **1** • crashInfo – Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.857.2 Field Documentation

8.857.2.1 crashInfoParams unpack_dms_SLQSSwiGetCrashInfo_t::crashInfoParam

8.857.2.2 swi_uint256_t unpack_dms_SLQSSwiGetCrashInfo_t::ParamPresenceMask

8.857.2.3 uint16_t unpack_dms_SLQSSwiGetCrashInfo_t::TlvResult

8.858 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference

Data Fields

- packgetDyingGaspCfg * pGetDyingGaspCfg
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.858.1 Detailed Description

This structure contains Get Dying GASP Config unpack

Parameters

pGetDyingGasp-Cfg	• See packgetDyingGaspCfg • pDestSMSNum. – Bit to check in ParamPresenceMask - **16** • pDestSMSContent. – Bit to check in ParamPresenceMask - **17**
TlvResult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.858.2 Field Documentation

8.858.2.1 swi_uint256_t unpack_dms_SLQSSwiGetDyingGaspCfg_t::ParamPresenceMask

8.858.2.2 packgetDyingGaspCfg* unpack_dms_SLQSSwiGetDyingGaspCfg_t::pGetDyingGaspCfg

8.858.2.3 uint16_t unpack_dms_SLQSSwiGetDyingGaspCfg_t::Tlvresult

8.859 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference

Data Fields

- packgetDyingGaspStatistics * pGetDyingGaspStatistics
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.859.1 Detailed Description

This structure contains Get Dying GASP Statistics unpack.

Parameters

pGetDyingGasp-Statistics	• See packgetDyingGaspStatistics • pTimeStamp. – Bit to check in ParamPresenceMask - **16** • pSMSAttemptedFlag. – Bit to check in ParamPresenceMask - **17**
Tlvresult	• Unpack Result

8.859.2 Field Documentation

8.859.2.1 swi_uint256_t unpack_dms_SLQSSwiGetDyingGaspStatistics_t::ParamPresenceMask

8.859.2.2 packgetDyingGaspStatistics* unpack_dms_SLQSSwiGetDyingGaspStatistics_t::pGetDyingGaspStatistics

8.859.2.3 uint16_t unpack_dms_SLQSSwiGetDyingGaspStatistics_t::Tlvresult

8.860 unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference

Data Fields

- uint8_t numEntries
- image_info_t * pCurrImgInfo
- char priver [16]
- char pkgver [16]
- char fwvers [16]
- char carrier [16]
- swi_uint256_t ParamPresenceMask

8.860.1 Detailed Description

This structure is used to store unpack_dms_SLQSSwiGetFirmwareCurr parameters

Parameters

numEntries[IN/-OUT]	• Number of entries in the image list to follow • The size of the list pCurrImgInfo must be specified when calling the API – Bit to check in ParamPresenceMask - **1**
pCurrImgInfo[O-UT]	• Currently Active Image List – Bit to check in ParamPresenceMask - **1**
priver[OUT]	• PRI version of the currently running firmware • Bit to check in ParamPresenceMask - **1**
pkgver[OUT]	• Package version of the currently running firmware • Bit to check in ParamPresenceMask - **1**
fwvers[OUT]	• firmware version of the currently running firmware • Bit to check in ParamPresenceMask - **1**
carrier[OUT]	• Carrier string of the currently running firmware • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.860.2 Field Documentation

8.860.2.1 char unpack_dms_SLQSSwiGetFirmwareCurr_t::carrier[16]

8.860.2.2 char unpack_dms_SLQSSwiGetFirmwareCurr_t::fwvers[16]

8.860.2.3 uint8_t unpack_dms_SLQSSwiGetFirmwareCurr_t::numEntries

8.860.2.4 swi_uint256_t unpack_dms_SLQSSwiGetFirmwareCurr_t::ParamPresenceMask

8.860.2.5 image_info_t* unpack_dms_SLQSSwiGetFirmwareCurr_t::pCurrImgInfo

8.860.2.6 char unpack_dms_SLQSSwiGetFirmwareCurr_t::pkgver[16]

8.860.2.7 char unpack_dms_SLQSSwiGetFirmwareCurr_t::priver[16]

8.861 unpack_dms_SLQSSwiGetFwUpdateStatus_t Struct Reference

Data Fields

- uint32_t ResCode
- uint8_t imgType
- uint32_t refData
- uint8_t refString [15]
- uint8_t logString [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.861.1 Detailed Description

This structure is used to store Firmware Update Status

Parameters

ResCode	• FW Update Result Code • Values: – 0x00000001 - Successful – 0xFFFFFFFF - Unknown (due to power off reset after firmware update) – 0x100000nn - File update errors while nn will be the exact error number: * 00 - General error – 0x200000nn - NVUP update errors while nn will be the exact error number: * 00 - General error – 0x40000nnn - FOTA update agent errors while nnn will be the exact error number: * 000 – 0FF - Insignia defined error code * 100 – 1FF - Sierra defined error code * See qaGobiApiTableFwDldErrorCodes.h for more detailed information – 0x800000nn - FDT/SSDP reported errors while nn will be the exact error number * See qaGobiApiTableFwDldErrorCodes.h for more detailed information • Bit to check in ParamPresenceMask - **16**
imgType	• Optional parameter • Firmware image type that failed the update • Bit to check in ParamPresenceMask - **17**
refData	• Optional parameter • Failed image reference data • This is normally the offset of the image that caused the failure • Bit to check in ParamPresenceMask - **18**
refString	• Optional parameter • Failed image reference string. This is normally the partition name of the image that caused the failure if applicable. • Bit to check in ParamPresenceMask - **19**

logString	• Optional parameter • Failed image reference string. This is normally the partition name of the image that caused the failure if applicable. • Bit to check in ParamPresenceMask - **20**
TlvResult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.861.2 Field Documentation

8.861.2.1 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::imgType

8.861.2.2 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::logString[255]

8.861.2.3 swi_uint256_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::ParamPresenceMask

8.861.2.4 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refData

8.861.2.5 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refString[15]

8.861.2.6 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::ResCode

8.861.2.7 uint16_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::TlvResult

8.862 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference

Data Fields

- uint16_t TlvResult
- char manString [255]
- char modelString [255]
- char swVerString [255]
- char plasmaIDString [255]
- char hostID [255]
- swi_uint256_t ParamPresenceMask

8.862.1 Detailed Description

This structure contains SWI get host device info unpack information

Parameters

TlvResult	• unpack Result

manString[OUT]	• Host Device Manufacturer Name • Null terminated ASCII String • Bit to check in ParamPresenceMask - **16**
modelString[OUT]	• Host Device Model String • Null terminated ASCII String • Bit to check in ParamPresenceMask - **17**
swVerString[OUT]	• Host Device Software Version String • Null terminated ASCII string • Bit to check in ParamPresenceMask - **18**
plasmaIDString[OUT]	• Host Device Plasma ID String • Null terminated alphanumeric ASCII String. • Bit to check in ParamPresenceMask - **19**
hostID[OUT]	• Device Host ID String • Null terminated alphanumeric ASCII String. • Bit to check in ParamPresenceMask - **20**
ParamPresenceMask[OUT]	• bitmask representation to indicate valid parameters.

8.862.2 Field Documentation

8.862.2.1 char unpack_dms_SLQSSwiGetHostDevInfo_t::hostID[255]

8.862.2.2 char unpack_dms_SLQSSwiGetHostDevInfo_t::manString[255]

8.862.2.3 char unpack_dms_SLQSSwiGetHostDevInfo_t::modelString[255]

8.862.2.4 swi_uint256_t unpack_dms_SLQSSwiGetHostDevInfo_t::ParamPresenceMask

8.862.2.5 char unpack_dms_SLQSSwiGetHostDevInfo_t::plasmaIDString[255]

8.862.2.6 char unpack_dms_SLQSSwiGetHostDevInfo_t::swVerString[255]

8.862.2.7 uint16_t unpack_dms_SLQSSwiGetHostDevInfo_t::TlvResult

8.863 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference

Data Fields

- uint16_t TlvResult
- char nameString [255]
- char versionString [255]

- swi_uint256_t ParamPresenceMask

8.863.1 Detailed Description

This structure contains SWI get host os info unpack information

Parameters

Tlvresult	• Unpack Result
nameString[OUT]	• Host operating system name • Bit to check in ParamPresenceMask - **16**
VersionString[OUT]	• Host operating system version • Bit to check in ParamPresenceMask - **17**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.863.2 Field Documentation

8.863.2.1 char unpack_dms_SLQSSwiGetOSInfo_t::nameString[255]

8.863.2.2 swi_uint256_t unpack_dms_SLQSSwiGetOSInfo_t::ParamPresenceMask

8.863.2.3 uint16_t unpack_dms_SLQSSwiGetOSInfo_t::Tlvresult

8.863.2.4 char unpack_dms_SLQSSwiGetOSInfo_t::versionString[255]

8.864 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference

Data Fields

- uint16_t Tlvresult
- char meidString [8]
- swi_uint256_t ParamPresenceMask

8.864.1 Detailed Description

This structure contains SWI get device serial number extension unpack information

Parameters

Tlvresult	• pack Result
meidString[OUT]	• optional parameter, mobile equipment identifier • Bit to check in ParamPresenceMask - **16**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.864.2 Field Documentation

8.864.2.1 char unpack_dms_SLQSSwiGetSerialNoExt_t::meidString[8]

8.864.2.2 swi_uint256_t unpack_dms_SLQSSwiGetSerialNoExt_t::ParamPresenceMask

8.864.2.3 uint16_t unpack_dms_SLQSSwiGetSerialNoExt_t::Tlvresult

8.865 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.865.1 Detailed Description

This structure contains set Dying GASP Config unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.865.2 Field Documentation

8.865.2.1 swi_uint256_t unpack_dms_SLQSSwiSetDyingGaspCfg_t::ParamPresenceMask

8.865.2.2 uint16_t unpack_dms_SLQSSwiSetDyingGaspCfg_t::Tlvresult

8.866 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.866.1 Detailed Description

This structure contains SWI set host dev info unpack

Parameters

TlvresultParamPresence-Mask	• Unpack Result
	• bitmask representation to indicate valid parameters.

8.866.2 Field Documentation

8.866.2.1 swi_uint256_t unpack_dms_SLQSSwiSetHostDevInfo_t::ParamPresenceMask

8.866.2.2 uint16_t unpack_dms_SLQSSwiSetHostDevInfo_t::Tlvresult

8.867 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.867.1 Detailed Description

This structure contains SWI set host OS info unpack

Parameters

TlvresultParamPresence-Mask	• Unpack Result
	• bitmask representation to indicate valid parameters.

8.867.2 Field Documentation

8.867.2.1 swi_uint256_t unpack_dms_SLQSSwiSetOSInfo_t::ParamPresenceMask

8.867.2.2 uint16_t unpack_dms_SLQSSwiSetOSInfo_t::Tlvresult

8.868 unpack_dms_SLQSUIMGetState_t Struct Reference

Data Fields

- uint8_t state
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.868.1 Detailed Description

This structure contains UIM get state unpack information

Parameters

state[OUT]	• UIM state • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.868.2 Field Documentation

8.868.2.1 swi_uint256_t unpack_dms_SLQSUIMGetState_t::ParamPresenceMask

8.868.2.2 uint8_t unpack_dms_SLQSUIMGetState_t::state

8.868.2.3 uint16_t unpack_dms_SLQSUIMGetState_t::Tlvresult

8.869 unpack_dms_SwiEventReportCallBack_ind_t Struct Reference

Data Fields

- dms_TemperatureTlv TempTlv
- dms_VoltageTlv VoltTlv
- dms_UimStatusTlv UimStatusTlv
- swi_uint256_t ParamPresenceMask

8.869.1 Detailed Description

Structure used to SWI event report indication parameters.

Parameters

TempTlv	• Temperature status tlv • See dms_TemperatureTlv for more information
VoltTlv	• Voltage status tlv • see dms_VoltageTlv for more information
UimStatusTlv	• UIM status tlv • see dms_UimStatusTlv for more information
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

: None

8.869.2 Field Documentation

8.869.2.1 swi_uint256_t unpack_dms_SwiEventReportCallBack_ind_t::ParamPresenceMask

8.869.2.2 dms_TemperatureTlv unpack_dms_SwiEventReportCallBack_ind_t::TempTlv

8.869.2.3 dms_UimStatusTlv unpack_dms_SwiEventReportCallBack_ind_t::UimStatusTlv

8.869.2.4 dms_VoltageTlv unpack_dms_SwiEventReportCallBack_ind_t::VoltTlv

8.870 unpack_dms_SwiSetEventReport_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.870.1 Detailed Description

This structure is used to store unpack_dms_SwiSetEventReport parameters.

Parameters

Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.870.2 Field Documentation

8.870.2.1 swi_uint256_t unpack_dms_SwiSetEventReport_t::ParamPresenceMask

8.870.2.2 uint16_t unpack_dms_SwiSetEventReport_t::Tlvresult

8.871 unpack_dms_SwiUimSelect_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.871.1 Detailed Description

This structure is used to store unpack_dms_SwiUimSelect parameters.

Parameters

Tlvresult	• unpack Tlv result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.871.2 Field Documentation

8.871.2.1 swi_uint256_t unpack_dms_SwiUimSelect_t::ParamPresenceMask

8.871.2.2 uint16_t unpack_dms_SwiUimSelect_t::Tlvresult

8.872 unpack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facilityState
- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.872.1 Detailed Description

This structure contains PIN retries status

Parameters

facilityState[OU-T]	• Control key status – 0 - Deactivated – 1 - Activated – 2 - Blocked • Bit to check in ParamPresenceMask - **1**
verifyRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. • Bit to check in ParamPresenceMask - **1**
unblockRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked i.e. UIM is unusable. • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.872.2 Field Documentation

8.872.2.1 uint8_t unpack_dms_UIMGetControlKeyStatus_t::facilityState

8.872.2.2 swi_uint256_t unpack_dms_UIMGetControlKeyStatus_t::ParamPresenceMask

8.872.2.3 uint16_t unpack_dms_UIMGetControlKeyStatus_t::Tlvresult

8.872.2.4 uint8_t unpack_dms_UIMGetControlKeyStatus_t::unblockRetriesLeft

8.872.2.5 uint8_t unpack_dms_UIMGetControlKeyStatus_t::verifyRetriesLeft

8.873 unpack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint8_t stringSize
- uint8_t String [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.873.1 Detailed Description

This structure is used to store unpack_dms_UIMGetICCID parameters.

Parameters

stringSize	• Size of String. • Bit to check in ParamPresenceMask - **1**
String	• ICCID String. • Maximum Length is 255 Bytes • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unack result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.873.2 Field Documentation

8.873.2.1 swi_uint256_t unpack_dms_UIMGetICCID_t::ParamPresenceMask

8.873.2.2 uint8_t unpack_dms_UIMGetICCID_t::String[255]

8.873.2.3 uint8_t unpack_dms_UIMGetICCID_t::stringSize

8.873.2.4 uint16_t unpack_dms_UIMGetICCID_t::Tlvresult

8.874 unpack_dms_UIMGetPINStatus_t Struct Reference

Data Fields

- uint8_t p1Status
- uint8_t p1VerifyRetriesLeft
- uint8_t p1UnblockRetriesLeft
- uint8_t p2Status
- uint8_t p2VerifyRetriesLeft
- uint8_t p2UnblockRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.874.1 Detailed Description

This structure contains Get PIN Status unpack information

Parameters

p1Status[OUT]	• PIN1 status(0xFFFFFFFF - Unknown) – 0 - PIN not initialized – 1 - PIN enabled, not verified – 2 - PIN enabled, verified – 3 - PIN disabled – 4 - PIN blocked – 5 - PIN permanently blocked • Bit to check in ParamPresenceMask - **17**
p1VerifyRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left of PIN1, after which PIN will be blocked. • Bit to check in ParamPresenceMask - **17**
p1Unblock-RetriesLeft[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left of PIN1, after which the PIN will be permanently blocked i.e. UIM is unusable. • Bit to check in ParamPresenceMask - **17**
p2Status[OUT]	• PIN2 status(0xFFFFFFFF - Unknown) – 0 - PIN not initialized – 1 - PIN enabled, not verified – 2 - PIN enabled, verified – 3 - PIN disabled – 4 - PIN blocked – 5 - PIN permanently blocked • Bit to check in ParamPresenceMask - **18**

p2VerifyRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left of PIN2, after which PIN will be blocked. • Bit to check in ParamPresenceMask - **18**
p2Unblock-RetriesLeft[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left of PIN2, after which the PIN will be permanently blocked i.e. UIM is unusable. • Bit to check in ParamPresenceMask - **18**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.874.2 Field Documentation

8.874.2.1 uint8_t unpack_dms_UIMGetPINStatus_t::p1Status

8.874.2.2 uint8_t unpack_dms_UIMGetPINStatus_t::p1UnblockRetriesLeft

8.874.2.3 uint8_t unpack_dms_UIMGetPINStatus_t::p1VerifyRetriesLeft

8.874.2.4 uint8_t unpack_dms_UIMGetPINStatus_t::p2Status

8.874.2.5 uint8_t unpack_dms_UIMGetPINStatus_t::p2UnblockRetriesLeft

8.874.2.6 uint8_t unpack_dms_UIMGetPINStatus_t::p2VerifyRetriesLeft

8.874.2.7 swi_uint256_t unpack_dms_UIMGetPINStatus_t::ParamPresenceMask

8.874.2.8 uint16_t unpack_dms_UIMGetPINStatus_t::Tlvresult

8.875 unpack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.875.1 Detailed Description

This structure contains UIM Set control key protection unpack information

Parameters

verifyRetries-Left[OUT]	• Optional parameter • number of retries left after which the control key is blocked • Bit to check in ParamPresenceMask - **16**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.875.2 Field Documentation

8.875.2.1 swi_uint256_t unpack_dms_UIMSetControlKeyProtection_t::ParamPresenceMask

8.875.2.2 uint16_t unpack_dms_UIMSetControlKeyProtection_t::Tlvresult

8.875.2.3 uint8_t unpack_dms_UIMSetControlKeyProtection_t::verifyRetriesLeft

8.876 unpack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.876.1 Detailed Description

This structure contains PIN retries status

Parameters

verifyRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. • Bit to check in ParamPresenceMask - **16**
unblockRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked i.e. UIM is unusable. • Bit to check in ParamPresenceMask - **16**
Tlvresult[OUT]	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.876.2 Field Documentation

8.876.2.1 swi_uint256_t unpack_dms_UIMSetPINProtection_t::ParamPresenceMask

8.876.2.2 uint16_t unpack_dms_UIMSetPINProtection_t::Tlvresult

8.876.2.3 uint8_t unpack_dms_UIMSetPINProtection_t::unblockRetriesLeft

8.876.2.4 uint8_t unpack_dms_UIMSetPINProtection_t::verifyRetriesLeft

8.877 unpack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t unblockRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.877.1 Detailed Description

This structure contains UIM Set control key protection unpack information

Parameters

unblockRetries-Left[OUT]	• Optional parameter • number of unblock retries left after which the control key is permanently blocked • Bit to check in ParamPresenceMask - **16**
Tlvresult[OUT]	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.877.2 Field Documentation

8.877.2.1 swi_uint256_t unpack_dms_UIMUnblockControlKey_t::ParamPresenceMask

8.877.2.2 uint16_t unpack_dms_UIMUnblockControlKey_t::Tlvresult

8.877.2.3 uint8_t unpack_dms_UIMUnblockControlKey_t::unblockRetriesLeft

8.878 unpack_dsd_SLQSGetDsdUIInfo_t Struct Reference

Data Fields

- uint8_t Apn_info_len
- DsdGlobalUIInformation_t * pDsdGlobalUIInformation
- DsdUIApnAvailableSystemInfo_t * pDsdApnAvailableSystemInfo
- DsdNullBearerReason_t * pDsdNullBearerReason
- swi_uint256_t ParamPresenceMask

8.878.1 Detailed Description

This structure contains the DSD Get UI Information response parameters.

See Also

qaGobiApiDsd.h for parameter descriptions

Parameters

Apn_info_len	• count of DsdApnAvailableSystemInfo instances.
pDsdGlobalUI-Information	• DsdUIApnAvailableSystemInfo_t.
pDsdApn-Available-SystemInfo	• DsdUIApnAvailableSystemInfo_t.
pDsdNullBearer-Reason	• DsdNullBearerReason_t.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

FOR_5G

8.878.2 Field Documentation

8.878.2.1 uint8_t unpack_dsd_SLQSGetDsdUIInfo_t::Apn_info_len

8.878.2.2 swi_uint256_t unpack_dsd_SLQSGetDsdUIInfo_t::ParamPresenceMask

8.878.2.3 DsdUIApnAvailableSystemInfo_t* unpack_dsd_SLQSGetDsdUIInfo_t::pDsdApnAvailableSystemInfo

8.878.2.4 DsdGlobalUIInformation_t* unpack_dsd_SLQSGetDsdUIInfo_t::pDsdGlobalUIInformation

8.878.2.5 DsdNullBearerReason_t* unpack_dsd_SLQSGetDsdUIInfo_t::pDsdNullBearerReason

8.879 unpack_dsd_SLQSGetSystemStatus_t Struct Reference

Data Fields

- uint8_t Avail_sys_len
- uint8_t Apn_avail_sys_info_len
- uint8_t Apn_pref_sys_len
- DsdSystemInfo_t * pDsdAvailableSystem
- DsdApnAvailableSystemInfo_t * pDsdApnAvailableSystemInfo
- DsdApnCurRecAvailableSystemInfo_t * pDsdCurRecGlobalPrefSystem
- DsdApnCurRecPreferredSystemInfo_t * pDsdApnCurRecGlobalPrefSystem
- DsdNullBearerReason_t * pDsdNullBearerReason
- swi_uint256_t ParamPresenceMask

8.879.1 Detailed Description

This structure contains the DSD Get System Status response parameters.

Parameters

Avail_sys_len	• count of DSD Available System instances.
Apn_avail_sys_info_len	• count of Dsd APN Available System Info instances.
Apn_pref_sys_len	• count of DSD APN Current and Recommended Global Preferred System instances.
pDsdAvailableSystem	• Available Systems Array of all available systems.
pDsdApnAvailableSystemInfo	• DSD APN Available System Info
pDsdCurRecGlobalPrefSystem	• Current and Recommended APN Available System Info
pDsdApnCurRecGlobalPrefSystem	• Current and Recommended Global Preferred System Info.
pDsdNullBearerReason	• Null Bearer reason.

Note

FOR_5G

8.879.2 Field Documentation

8.879.2.1 uint8_t unpack_dsd_SLQSGetSystemStatus_t::Apn_avail_sys_info_len

8.879.2.2 uint8_t unpack_dsd_SLQSGetSystemStatus_t::Apn_pref_sys_len

8.879.2.3 uint8_t unpack_dsd_SLQSGetSystemStatus_t::Avail_sys_len

8.879.2.4 swi_uint256_t unpack_dsd_SLQSGetSystemStatus_t::ParamPresenceMask

8.879.2.5 DsdApnAvailableSystemInfo_t* unpack_dsd_SLQSGetSystemStatus_t::pDsdApnAvailableSystemInfo

8.879.2.6 DsdApnCurRecPreferredSystemInfo_t* unpack_dsd_SLQSGetSystemStatus_t::pDsdApnCurRecGlobalPrefSystem

8.879.2.7 DsdSystemInfo_t* unpack_dsd_SLQSGetSystemStatus_t::pDsdAvailableSystem

8.879.2.8 DsdApnCurRecAvailableSystemInfo_t* unpack_dsd_SLQSGetSystemStatus_t::pDsdCurRecGlobalPrefSystem

8.879.2.9 DsdNullBearerReason_t* unpack_dsd_SLQSGetSystemStatus_t::pDsdNullBearerReason

8.880 unpack_dsd_SLQSSystemStatusChange_Ind_t Struct Reference

Data Fields

- uint16_t Tlvresult
- uint8_t Avail_sys_len
- uint8_t Apn_avail_sys_info_len
- uint8_t Apn_pref_sys_len
- uint8_t CurRecGlobalPrefTlvPresent
- uint8_t NullBearerReasonTlvPresent
- DsdSystemInfo_t * pDsdAvailableSystem
- DsdApnAvailableSystemInfo_t * pDsdApnAvailableSystemInfo
- DsdApnCurRecAvailableSystemInfo_t * pDsdCurRecGlobalPrefSystem
- DsdApnCurRecPreferredSystemInfo_t * pDsdApnCurRecGlobalPrefSystem
- DsdNullBearerReason_t * pDsdNullBearerReason
- swi_uint256_t ParamPresenceMask

8.880.1 Detailed Description

This structure contains the DSD System Status Change Indication response parameters.

Parameters

Tlvresult	• Unpack Result
Avail_sys_len	• count of DSD Available System instances.
Apn_avail_sys_info_len	• count of Dsd APN Available System Info instances.
Apn_pref_sys_len	• count of DSD APN Current and Recommended Global Preferred System instances.
pDsdAvailableSystem	• Available Systems Array of all available systems.
pDsdApnAvailableSystemInfo	• DSD APN Available System Info
pDsdCurRecGlobalPrefSystem	• Current and Recommended APN Available System Info
pDsdApnCurRecGlobalPrefSystem	• Current and Recommended Global Preferred System Info.
pDsdNullBearerReason	• Null Bearer reason.

Note

FOR_5G

8.880.2 Field Documentation

8.880.2.1 uint8_t unpack_dsd_SLQSSystemStatusChange_Ind_t::Apn_avail_sys_info_len

8.880.2.2 uint8_t unpack_dsd_SLQSSystemStatusChange_Ind_t::Apn_pref_sys_len

8.880.2.3 uint8_t unpack_dsd_SLQSSystemStatusChange_Ind_t::Avail_sys_len

8.880.2.4 uint8_t unpack_dsd_SLQSSystemStatusChange_Ind_t::CurRecGlobalPrefTlvPresent

8.880.2.5 uint8_t unpack_dsd_SLQSSystemStatusChange_Ind_t::NullBearerReasonTlvPresent

8.880.2.6 swi_uint256_t unpack_dsd_SLQSSystemStatusChange_Ind_t::ParamPresenceMask

8.880.2.7 DsdApnAvailableSystemInfo_t* unpack_dsd_SLQSSystemStatusChange_Ind_t::pDsdApnAvailableSystemInfo

8.880.2.8 DsdApnCurRecPreferredSystemInfo_t* unpack_dsd_SLQSSystemStatusChange_Ind_t::pDsdApnCurRecGlobalPrefSystem

8.880.2.9 DsdSystemInfo_t* unpack_dsd_SLQSSystemStatusChange_Ind_t::pDsdAvailableSystem

8.880.2.10 DsdApnCurRecAvailableSystemInfo_t* unpack_dsd_SLQSSystemStatusChange_Ind_t::pDsdCurRecGlobalPrefSystem

8.880.2.11 DsdNullBearerReason_t* unpack_dsd_SLQSSystemStatusChange_Ind_t::pDsdNullBearerReason

8.880.2.12 uint16_t unpack_dsd_SLQSSystemStatusChange_Ind_t::Tlvresult

8.881 unpack_dsd_SLQSUIInfo_Ind_t Struct Reference

Data Fields

- uint16_t Tlvresult
- uint8_t Apn_info_len
- uint8_t GlobalUIInfoTlvPresent
- uint8_t NullBearerReasonTlvPresent
- DsdGlobalUIInformation_t * pDsdGlobalUIInformation
- DsdUIApnAvailableSystemInfo_t * pDsdApnAvailableSystemInfo
- DsdNullBearerReason_t * pDsdNullBearerReason
- swi_uint256_t ParamPresenceMask

8.881.1 Detailed Description

This structure contains the DSD Get UI Info response parameters.

See Also

qaGobiApiDsd.h for parameter descriptions

Parameters

Tlvresult	• Unpack Result
Apn_info_len	• count of DsdApnAvailableSystemInfo instances.
GlobalUIInfoTlv-Present	• pDsdGlobalUIInformation Tlv Present or not.
NullBearer-ReasonTlv-Present	• pDsdNullBearerReason Tlv Present or not.
pDsdGlobalUI-Information	• DsdGlobalUIInformation.
pDsdApn-Available-SystemInfo	• DsdApnAvailableSystemInfo.
pDsdNullBearer-Reason	• DsdNullBearerReason.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

FOR_5G

8.881.2 Field Documentation

8.881.2.1 uint8_t unpack_dsd_SLQSUIInfo_Ind_t::Apn_info_len

8.881.2.2 uint8_t unpack_dsd_SLQSUIInfo_Ind_t::GlobalUIInfoTlvPresent

8.881.2.3 uint8_t unpack_dsd_SLQSUIInfo_Ind_t::NullBearerReasonTlvPresent

8.881.2.4 swi_uint256_t unpack_dsd_SLQSUIInfo_Ind_t::ParamPresenceMask

8.881.2.5 DsdUIApnAvailableSystemInfo_t* unpack_dsd_SLQSUIInfo_Ind_t::pDsdApnAvailableSystemInfo

8.881.2.6 DsdGlobalUIInformation_t* unpack_dsd_SLQSUIInfo_Ind_t::pDsdGlobalUIInformation

8.881.2.7 DsdNullBearerReason_t* unpack_dsd_SLQSUIInfo_Ind_t::pDsdNullBearerReason

8.881.2.8 uint16_t unpack_dsd_SLQSUIInfo_Ind_t::Tlvresult

8.882 unpack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageListSize
- FMSPrefImageList * pImageList

- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.882.1 Detailed Description

This structure contains the Get Image Preference information unpack

Parameters

listSize	• The number of elements in the image list
pListEntries	• Array of Image entries with size provided by previous field • See FMSImageElement • Bit to check in ParamPresenceMask - **1**
TlvResult	• Unpack result

8.882.2 Field Documentation

8.882.2.1 uint32_t unpack_fms_GetImagesPreference_t::ImageListSize

8.882.2.2 swi_uint256_t unpack_fms_GetImagesPreference_t::ParamPresenceMask

8.882.2.3 FMSPrefImageList* unpack_fms_GetImagesPreference_t::pImageList

8.882.2.4 uint16_t unpack_fms_GetImagesPreference_t::TlvResult

8.883 unpack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint32_t imagelistSize
- FMSImageList imageList
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.883.1 Detailed Description

This structure contains the Get Stored Images unpack

Parameters

listSize	• The number of elements in the image list
imageList	• Array of Image entries with size provided by previous field • See FMSImageElement • Bit to check in ParamPresenceMask - **1**

Tlvresult	• Unpack result

8.883.2 Field Documentation

8.883.2.1 FMSImageList unpack_fms_GetStoredImages_t::imageList

8.883.2.2 uint32_t unpack_fms_GetStoredImages_t::imagelistSize

8.883.2.3 swi_uint256_t unpack_fms_GetStoredImages_t::ParamPresenceMask

8.883.2.4 uint16_t unpack_fms_GetStoredImages_t::Tlvresult

8.884 unpack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageTypesSize
- uint8_t ImageTypes [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.884.1 Detailed Description

This structure contains the Set Images Preference unpack

Parameters

ImageTypesSize	• Image Type Size
ImageTypes	• Image Type • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack result

8.884.2 Field Documentation

8.884.2.1 uint8_t unpack_fms_SetImagesPreference_t::ImageTypes[255]

8.884.2.2 uint32_t unpack_fms_SetImagesPreference_t::ImageTypesSize

8.884.2.3 swi_uint256_t unpack_fms_SetImagesPreference_t::ParamPresenceMask

8.884.2.4 uint16_t unpack_fms_SetImagesPreference_t::Tlvresult

8.885 unpack_ims_SLQSGetIMSSMSConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- uint8_t * pSMSFormat
- uint8_t * pSMSOverIPNwInd
- uint8_t * pPhoneCtxtURILen
- uint8_t * pPhoneCtxtURI
- swi_uint256_t ParamPresenceMask

8.885.1 Detailed Description

This structure contains the SLQSGetIMSSMSConfig unpack parameters.

Parameters

pSettingResp	• Settings Response • Bit to check in ParamPresenceMask - **16**
pSMSFormat	• SMS format – 0 - 3GPP – 1 - 3GPP2 • Bit to check in ParamPresenceMask - **17**
pSMSOverIPNw-Ind	• SMS over IP Network Indication Flag – TRUE - Turn on mobile-originated SMS – FALSE - Turn off mobile-originated SMS • Bit to check in ParamPresenceMask - **18**
pPhoneCtxtURI-Len[IN/OUT]	• Size in bytes assigned to the Phone context Universal Resource Identifier to follow
pPhoneCtxtURI	• Phone context universal resource identifier • Length of this string must be specified in pPhoneCtxtURILen parameter • Bit to check in ParamPresenceMask - **19**

8.885.2 Field Documentation

8.885.2.1 swi_uint256_t unpack_ims_SLQSGetIMSSMSConfig_t::ParamPresenceMask

8.885.2.2 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pPhoneCtxtURI

8.885.2.3 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pPhoneCtxtURILen

8.885.2.4 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pSettingResp

8.885.2.5 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pSMSFormat

8.885.2.6 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pSMSOverIPNwInd

8.886 unpack_ims_SLQSGetIMSUserConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- uint8_t * pIMSDomainLen
- uint8_t * pIMSDomain
- swi_uint256_t ParamPresenceMask

8.886.1 Detailed Description

This structure contains the SLQSGetIMSUserConfig pack parameters.

Parameters

pSettingResp	• Settings Response • Bit to check in ParamPresenceMask - **16**
pIMSDomain-Len[IN/OUT]	• Length of IMS Domain Name to follow
pIMSDomain	• IMS domain name • Length of this string must be specified in pIMSDomainLen parameter • Bit to check in ParamPresenceMask - **17**

8.886.2 Field Documentation

8.886.2.1 swi_uint256_t unpack_ims_SLQSGetIMSUserConfig_t::ParamPresenceMask

8.886.2.2 uint8_t* unpack_ims_SLQSGetIMSUserConfig_t::pIMSDomain

8.886.2.3 uint8_t* unpack_ims_SLQSGetIMSUserConfig_t::pIMSDomainLen

8.886.2.4 uint8_t* unpack_ims_SLQSGetIMSUserConfig_t::pSettingResp

8.887 unpack_ims_SLQSGetIMSVoIPConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- uint16_t * pSessionExpiryTimer
- uint16_t * pMinSessionExpiryTimer
- uint8_t * pAmrWbEnable
- uint8_t * pScrAmrEnable
- uint8_t * pScrAmrWbEnable
- uint8_t * pAmrMode
- uint16_t * pAmrWBMode
- uint8_t * pAmrOctetAligned
- uint8_t * pAmrWBOctetAligned
- uint16_t * pRingingTimer

- uint16_t * pRingBackTimer
- uint16_t * pRTPRTCPInactTimer
- swi_uint256_t ParamPresenceMask

8.887.1 Detailed Description

This structure contains the SLQSGetIMSVoIPConfig unpack parameters.

Parameters

pSettingResp	• Settings Response. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE • Bit to check in ParamPresenceMask - **16**
pSessionExpiry-Timer	• Session duration, in seconds • Bit to check in ParamPresenceMask - **17**
pMinSession-ExpiryTimer	• Minimum allowed value for session expiry timer, in seconds • Bit to check in ParamPresenceMask - **18**
pAmrWbEnable	• Flag to enable/disable Adaptive Multirate Codec(AMR) WideBand(WB) audio • Values: – True - Enable – False - Disable • Bit to check in ParamPresenceMask - **19**
pScrAmrEnable	• Flag to enable/disable Source Control Rate(SCR) for AMR NarrowBand (NB) • Values: – True - Enable – False - Disable • Bit to check in ParamPresenceMask - **20**
pScrAmrWb-Enable	• Flag to enable/disable SCR for AMR WB Audio • Values: – True - Enable – False - Disable • Bit to check in ParamPresenceMask - **21**

pAmrMode	• BitMask for AMR NB modes allowed • Values: – 0x1 - 4.75 kbps – 0x2 - 5.15 kbps – 0x4 - 5.9 kbps – 0x8 - 6.17 kbps – 0x10 - 7.4 kbps – 0x20 - 7.95 kbps – 0x40 - 10.2 kbps – 0x80 - 12.2 kbps • Bit to check in ParamPresenceMask - **22**
pAmrWBMode	• BitMask for AMR WB modes allowed • Values: – 0x1 - 6.60 kbps – 0x2 - 8.85 kbps – 0x4 - 12.65 kbps – 0x8 - 14.25 kbps – 0x10 - 15.85 kbps – 0x20 - 18.25 kbps – 0x40 - 19.85 kbps – 0x80 - 23.05 kbps – 0x100 - 23.85 kbps • Bit to check in ParamPresenceMask - **23**
pAmrOctet-Aligned	• Flag to indicate if the octet is aligned for AMR NB Audio • Values: – True - Aligned – False - Not aligned, Bandwidth Efficient mode • Bit to check in ParamPresenceMask - **24**
pAmrWBOctet-Aligned	• Flag to indicate if the octet is aligned for AMR WB Audio • Values: – True - Aligned – False - Not aligned, Bandwidth Efficient mode • Bit to check in ParamPresenceMask - **25**
pRingingTimer	• Duration of ringing timer, in seconds. The ringing timer starts on the ringing event. If the call is not answered within the duration of this timer, the call is disconnected. • Bit to check in ParamPresenceMask - **26**
pRingBackTimer	• Duration of ringback timer, in seconds. The ringback timer starts on the ringback event. If the call is not answered within the duration of this timer, the call is disconnected. • Bit to check in ParamPresenceMask - **27**

pRTPRTCP-InactTimer	• Duration of RTP/RTCP inactivity timer, in seconds. If no RTP/RTCP packet is received prior to the expiry of this timer, the call is disconnected. • Bit to check in ParamPresenceMask - **28**

8.887.2 Field Documentation

8.887.2.1 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrMode

8.887.2.2 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrOctetAligned

8.887.2.3 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrWbEnable

8.887.2.4 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrWBMode

8.887.2.5 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrWBOctetAligned

8.887.2.6 swi_uint256_t unpack_ims_SLQSGetIMSVoIPConfig_t::ParamPresenceMask

8.887.2.7 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pMinSessionExpiryTimer

8.887.2.8 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pRingBackTimer

8.887.2.9 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pRingingTimer

8.887.2.10 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pRTPRTCPInactTimer

8.887.2.11 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pScrAmrEnable

8.887.2.12 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pScrAmrWbEnable

8.887.2.13 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pSessionExpiryTimer

8.887.2.14 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pSettingResp

8.888 unpack_ims_SLQSGetRegMgrConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- uint16_t * pPCSCFPort
- uint8_t * pPriCSCFPortNameLen
- uint8_t * pPriCSCFPortName
- uint8_t * pIMSTestMode
- swi_uint256_t ParamPresenceMask

8.888.1 Detailed Description

This structure contains the SLQSGetRegMgrConfig unpack parameters.

Parameters

pSettingResp	• Settings Response • Bit to check in ParamPresenceMask - **16**
pPCSCFPort	• Proxy call session control function port • Bit to check in ParamPresenceMask - **17**
pPriCSCFPort-NameLen(IN/O-UT)	• Size in bytes assigned to the primary CSCF Port name parameter to follow
pPriCSCFPort-Name	• Call Session control port, fully qualified domain name • Length of this string must be specified in pPriCSCFPortNameLen parameter • Bit to check in ParamPresenceMask - **18**
pIMSTestMode	• IMS Test mode Enabled. – TRUE - Enabled – FALSE - Disabled • Bit to check in ParamPresenceMask - **19**

Note

pPriCSCFPortNameLen must be set to a valid value during API call to retrieve pPriCSCFPortName.

8.888.2 Field Documentation

8.888.2.1 swi_uint256_t unpack_ims_SLQSGetRegMgrConfig_t::ParamPresenceMask

8.888.2.2 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pIMSTestMode

8.888.2.3 uint16_t* unpack_ims_SLQSGetRegMgrConfig_t::pPCSCFPort

8.888.2.4 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pPriCSCFPortName

8.888.2.5 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pPriCSCFPortNameLen

8.888.2.6 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pSettingResp

8.889 unpack_ims_SLQSGetSIPConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- uint16_t * pSIPLocalPort
- uint32_t * pTimerSIPReg
- uint32_t * pSubscribeTimer
- uint32_t * pTimerT1
- uint32_t * pTimerT2
- uint32_t * pTimerTf

- uint8_t * pSigCompEnabled
- swi_uint256_t ParamPresenceMask

8.889.1 Detailed Description

This structure contains the SLQSGetSIPConfig unpack parameters.

Parameters

pSettingResp	• Settings Response • Bit to check in ParamPresenceMask - **16**
pSIPLocalPort	• Primary call session control function SIP port number • Bit to check in ParamPresenceMask - **17**
pTimerSIPReg	• Initial SIP registration duration from the User equipment, in seconds • Bit to check in ParamPresenceMask - **18**
pSubscribeTimer	• Duration of the subscription by the UE for IMS registration notifications, in seconds • Bit to check in ParamPresenceMask - **19**
pTimerT1	• RTT estimate, in milliseconds • Bit to check in ParamPresenceMask - **20**
pTimerT2	• The maximum retransmit interval for non-invite requests and invite responses, in milliseconds • Bit to check in ParamPresenceMask - **21**
pTimerTf	• Non-invite transaction timeout timer, in milliseconds • Bit to check in ParamPresenceMask - **22**
pSigComp-Enabled	• Sig Comp Status – TRUE - Sig Comp Enabled – FALSE - Sig Comp Disabled • Bit to check in ParamPresenceMask - **23**

8.889.2 Field Documentation

8.889.2.1 swi_uint256_t unpack_ims_SLQSGetSIPConfig_t::ParamPresenceMask

8.889.2.2 uint8_t* unpack_ims_SLQSGetSIPConfig_t::pSettingResp

8.889.2.3 uint8_t* unpack_ims_SLQSGetSIPConfig_t::pSigCompEnabled

8.889.2.4 uint16_t* unpack_ims_SLQSGetSIPConfig_t::pSIPLocalPort

8.889.2.5 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pSubscribeTimer

8.889.2.6 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerSIPReg

8.889.2.7 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerT1

8.889.2.8 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerT2

8.889.2.9 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerTf

8.890 unpack_ims_SLQSRegMgrCfgCallBack_ind_t Struct Reference

Data Fields

- ims_PCSCFPortInfo PCTlv
- ims_CSCFPortNameInfo PNTlv
- ims_IMSTestModeInfo TMTlv
- swi_uint256_t ParamPresenceMask

8.890.1 Detailed Description

Structure used to store Reg Mgr Config Indication Parameters.

Parameters

PCTlv	• Primary PCSCF Port info • See ims_PCSCFPortInfo for more information • Bit to check in ParamPresenceMask - **16**
PNTlv	• CSCF Port info • see ims_CSCFPortNameInfo for more information • Bit to check in ParamPresenceMask - **17**
TMTlv	• IMS test mode info • see ims_IMSTestModeInfo for more information • Bit to check in ParamPresenceMask - **18**

Note

: None

8.890.2 Field Documentation

8.890.2.1 swi_uint256_t unpack_ims_SLQSRegMgrCfgCallBack_ind_t::ParamPresenceMask

8.890.2.2 ims_PCSCFPortInfo unpack_ims_SLQSRegMgrCfgCallBack_ind_t::PCTlv

8.890.2.3 ims_CSCFPortNameInfo unpack_ims_SLQSRegMgrCfgCallBack_ind_t::PNTlv

8.890.2.4 ims_IMSTestModeInfo unpack_ims_SLQSRegMgrCfgCallBack_ind_t::TMTlv

8.891 unpack_ims_SLQSSetIMSSMSConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.891.1 Detailed Description

This structure contains the SLQSSetIMSSMSConfig unpack parameters.

Parameters

pSettingResp	• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE • Bit to check in ParamPresenceMask - **16**

8.891.2 Field Documentation

8.891.2.1 swi_uint256_t unpack_ims_SLQSSetIMSSMSConfig_t::ParamPresenceMask

8.891.2.2 uint8_t* unpack_ims_SLQSSetIMSSMSConfig_t::pSettingResp

8.892 unpack_ims_SLQSSetIMSUserConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.892.1 Detailed Description

This structure contains the SLQSSetIMSUserConfig unpack parameters.

Parameters

pSettingResp	• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE • Bit to check in ParamPresenceMask - **16**

8.892.2 Field Documentation

8.892.2.1 swi_uint256_t unpack_ims_SLQSSetIMSUserConfig_t::ParamPresenceMask

8.892.2.2 uint8_t* unpack_ims_SLQSSetIMSUserConfig_t::pSettingResp

8.893 unpack_ims_SLQSSetIMSVoIPConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.893.1 Detailed Description

This structure contains the SLQSSetIMSVoIPConfig unpack parameters.

Parameters

pSettingResp	• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE • Bit to check in ParamPresenceMask - **16**

8.893.2 Field Documentation

8.893.2.1 swi_uint256_t unpack_ims_SLQSSetIMSVoIPConfig_t::ParamPresenceMask

8.893.2.2 uint8_t* unpack_ims_SLQSSetIMSVoIPConfig_t::pSettingResp

8.894 unpack_ims_SLQSSetRegMgrConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.894.1 Detailed Description

This structure contains the SLQSSetRegMgrConfig unpack parameters.

Parameters

pSettingResp	• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE • Bit to check in ParamPresenceMask - **16**

8.894.2 Field Documentation

8.894.2.1 swi_uint256_t unpack_ims_SLQSSetRegMgrConfig_t::ParamPresenceMask

8.894.2.2 uint8_t* unpack_ims_SLQSSetRegMgrConfig_t::pSettingResp

8.895 unpack_ims_SLQSSetSIPConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.895.1 Detailed Description

This structure contains the SLQSSetSIPConfig unpack parameters.

Parameters

pSettingResp	• Settings standard response type. A settings specific error code is returned when the standard response error type is QMI_ERR_CAUSE_CODE • Bit to check in ParamPresenceMask - **16**

8.895.2 Field Documentation

8.895.2.1 swi_uint256_t unpack_ims_SLQSSetSIPConfig_t::ParamPresenceMask

8.895.2.2 uint8_t* unpack_ims_SLQSSetSIPConfig_t::pSettingResp

8.896 unpack_ims_SLQSSIPCfgCallBack_ind_t Struct Reference

Data Fields

- ims_SIPPortInfo SPTlv
- ims_SIPRegnTmrInfo SRTlv
- ims_SubscrTmrInfo STTlv
- ims_TmrT1Info TT1Tlv
- ims_TmrT2Info TT2Tlv
- ims_TmrTfInfo TTfTlv
- ims_SigCompEnInfo SCTlv
- swi_uint256_t ParamPresenceMask

8.896.1 Detailed Description

Structure used to store Reg Mgr Config Indication Parameters.

Parameters

SPTlv	• SIP port tlv • See ims_SIPPortInfo for more information • Bit to check in ParamPresenceMask - **16**
SRTlv	• SIP Registration Timer tlv • see ims_SIPRegnTmrInfo for more information • Bit to check in ParamPresenceMask - **17**

STTlv	• Subscriber timer tlv • see ims_SubscrTmrInfo for more information • Bit to check in ParamPresenceMask - **18**
TT1Tlv	• Timer T1 tlv • see ims_TmrT1Info for more information • Bit to check in ParamPresenceMask - **19**
TT2Tlv	• Timer T2 tlv • see ims_TmrT2Info for more information • Bit to check in ParamPresenceMask - **20**
TTfTlv	• Timer Tf tlv • see ims_TmrTfInfo for more information • Bit to check in ParamPresenceMask - **21**
SCTlv	• SigComp Status Tlv • see ims_SigCompEnInfo for more information • Bit to check in ParamPresenceMask - **22**

Note

: None

8.896.2 Field Documentation

8.896.2.1 swi_uint256_t unpack_ims_SLQSSIPCfgCallBack_ind_t::ParamPresenceMask

8.896.2.2 ims_SigCompEnInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::SCTlv

8.896.2.3 ims_SIPPortInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::SPTlv

8.896.2.4 ims_SIPRegnTmrInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::SRTlv

8.896.2.5 ims_SubscrTmrInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::STTlv

8.896.2.6 ims_TmrT1Info unpack_ims_SLQSSIPCfgCallBack_ind_t::TT1Tlv

8.896.2.7 ims_TmrT2Info unpack_ims_SLQSSIPCfgCallBack_ind_t::TT2Tlv

8.896.2.8 ims_TmrTfInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::TTfTlv

8.897 unpack_ims_SLQSSMSCfgCallBack_ind_t Struct Reference

Data Fields

- ims_SMSFmtInfo SFTlv

- ims_SMSoIPNwInfo SINTlv
- ims_PhCtxtURIInfo PCURTlv
- swi_uint256_t ParamPresenceMask

8.897.1 Detailed Description

Structure used to store SMS Config Indication Parameters.

Parameters

SFTlv	• SMS format info • See ims_SMSFmtInfo for more information • Bit to check in ParamPresenceMask - **16**
SINTlv	• SMS over IP network info • see ims_SMSoIPNwInfo for more information • Bit to check in ParamPresenceMask - **17**
PCURTlv	• Phone context URI info • see ims_PhCtxtURIInfo for more information • Bit to check in ParamPresenceMask - **18**

Note

: None

8.897.2 Field Documentation

8.897.2.1 swi_uint256_t unpack_ims_SLQSSMSCfgCallBack_ind_t::ParamPresenceMask

8.897.2.2 ims_PhCtxtURIInfo unpack_ims_SLQSSMSCfgCallBack_ind_t::PCURTlv

8.897.2.3 ims_SMSFmtInfo unpack_ims_SLQSSMSCfgCallBack_ind_t::SFTlv

8.897.2.4 ims_SMSoIPNwInfo unpack_ims_SLQSSMSCfgCallBack_ind_t::SINTlv

8.898 unpack_ims_SLQSUserCfgCallBack_ind_t Struct Reference

Data Fields

- ims_IMSDomainInfo IDTlv
- swi_uint256_t ParamPresenceMask

8.898.1 Detailed Description

Structure used to store SMS Config Indication Parameters.

Parameters

IDTlv	• IMS domain info • See ims_IMSDomainInfo for more information • Bit to check in ParamPresenceMask - **16**

Note

: None

8.898.2 Field Documentation

8.898.2.1 ims_IMSDomainInfo unpack_ims_SLQSUserCfgCallBack_ind_t::IDTlv

8.898.2.2 swi_uint256_t unpack_ims_SLQSUserCfgCallBack_ind_t::ParamPresenceMask

8.899 unpack_ims_SLQSVoIPCfgCallBack_ind_t Struct Reference

Data Fields

- ims_SessDurInfo SDTlv
- ims_MinSessExpInfo MSETlv
- ims_EnabAMRWBInfo EAWTlv
- ims_EnabSCRAMRInfo ESATlv
- ims_EnabSCRAMRWBInfo ESAWTlv
- ims_AMRModeInfo AMTlv
- ims_AMRWBModeInfo AWMTlv
- ims_AMROctAlgnInfo AOATlv
- ims_AMRWBOctAlgnInfo AWOATlv
- ims_RngTmrInfo RTTlv
- ims_RngBkTmrInfo RBTTlv
- ims_RTPRTCPInactTmrDurInfo RTIDTlv
- swi_uint256_t ParamPresenceMask

8.899.1 Detailed Description

Structure used to store VOIP Config Indication Parameters.

Parameters

SDTlv	• Session Duration tlv • See ims_SessDurInfo for more information • Bit to check in ParamPresenceMask - **16**
MSETlv	• Minimum Session Timer tlv • see ims_MinSessExpInfo for more information • Bit to check in ParamPresenceMask - **17**

EAWTlv	• Enable AMR WB tlv • see ims_EnabAMRWBInfo for more information • Bit to check in ParamPresenceMask - **18**
ESATlv	• Enable SCR AMR NB tlv • see ims_EnabSCRAMRInfo for more information • Bit to check in ParamPresenceMask - **19**
ESAWTlv	• Enable SCR AMR WB tlv • see ims_EnabSCRAMRWBInfo for more information • Bit to check in ParamPresenceMask - **20**
AMTlv	• AMR NB Mode tlv • see ims_AMRModeInfo for more information • Bit to check in ParamPresenceMask - **21**
AWMTlv	• AMR WB Mode Tlv • see ims_AMRWBModeInfo for more information • Bit to check in ParamPresenceMask - **22**
AOATlv	• AMR NB Octet Aligned tlv • see ims_AMROctAlgnInfo for more information • Bit to check in ParamPresenceMask - **23**
AWOATlv	• AMR WB Octet Aligned tlv • see ims_AMRWBOctAlgnInfo for more information • Bit to check in ParamPresenceMask - **24**
RTTlv	• Ringing Timer Duration tlv • see ims_RngTmrInfo for more information • Bit to check in ParamPresenceMask - **25**
RBTTlv	• Ringback Duration tlv • see ims_RngBkTmrInfo for more information • Bit to check in ParamPresenceMask - **26**
RTIDTlv	• RTP/RTCP Inactivity Timer Duration Tlv • see ims_RTPRTCPInactTmrDurInfo for more information • Bit to check in ParamPresenceMask - **27**

Note

: None

8.899.2 Field Documentation

8.899.2.1 ims_AMRModeInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AMTlv

8.899.2.2 ims_AMROctAlgnInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AOATlv

8.899.2.3 ims_AMRWBModeInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AWMTlv

8.899.2.4 ims_AMRWBOctAlgnInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AWOATlv

8.899.2.5 ims_EnabAMRWBInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::EAWTlv

8.899.2.6 ims_EnabSCRAMRInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::ESATlv

8.899.2.7 ims_EnabSCRAMRWBInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::ESAWTlv

8.899.2.8 ims_MinSessExpInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::MSETlv

8.899.2.9 swi_uint256_t unpack_ims_SLQSVoIPCfgCallBack_ind_t::ParamPresenceMask

8.899.2.10 ims_RngBkTmrInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::RBTTlv

8.899.2.11 ims_RTPRTCPInactTmrDurInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::RTIDTlv

8.899.2.12 ims_RngTmrInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::RTTlv

8.899.2.13 ims_SessDurInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::SDTlv

8.900 unpack_imsa_SLQSGetIMSARegStatus_t Struct Reference

Data Fields

- uint8_t ImsRegStatus
- uint16_t ImsRegErrCode
- uint32_t NewImsRegStatus
- swi_uint256_t ParamPresenceMask

8.900.1 Detailed Description

This structure contains response parameters of registration status.

Parameters

ImsRegStatus	• IMS Registration Status (Deprecated). • Values – TRUE - UE is registered on the IMS network – FALSE - UE is not registered on the IMS network • Bit to check in ParamPresenceMask - **16**

ImsRegErrCode	• IMS Registration Error Code. • An error code is returned when the IMS registration status is IMSA_STATUS_NOT_REGISTERED. -Values – 3xx - Redirection responses – 4xx - Client failure responses – 5xx - Server failure responses – 6xx - Global failure responses • Bit to check in ParamPresenceMask - **17**
NewImsRegStatus	• New IMS Registration Status • Values – 0 - Not registered for IMS – 1 - Registering for IMS – 2 - Registered for IMS • Bit to check in ParamPresenceMask - **18**

8.900.2 Field Documentation

8.900.2.1 uint16_t unpack_imsa_SLQSGetIMSARegStatus_t::ImsRegErrCode

8.900.2.2 uint8_t unpack_imsa_SLQSGetIMSARegStatus_t::ImsRegStatus

8.900.2.3 uint32_t unpack_imsa_SLQSGetIMSARegStatus_t::NewImsRegStatus

8.900.2.4 swi_uint256_t unpack_imsa_SLQSGetIMSARegStatus_t::ParamPresenceMask

8.901 unpack_imsa_SLQSGetIMSAServiceStatus_t Struct Reference

Data Fields

- uint32_t SmsServiceStatus
- uint32_t VoipServiceStatus
- uint32_t VtServiceStatus
- uint32_t SmsServiceRat
- uint32_t VoipServiceRat
- uint32_t VtServiceRat
- uint32_t UtServiceStatus
- uint32_t UtServiceRat
- uint32_t VsServiceStatus
- uint32_t VsServiceRat
- swi_uint256_t ParamPresenceMask

8.901.1 Detailed Description

This structure contains response parameters of service status for various IMS services.

Parameters

SmsService-Status	• SMS Service Status. • Values – 0 - IMS SMS service is not available – 1 - IMS SMS is in limited service – 2 - IMS SMS is in full service • Bit to check in ParamPresenceMask - **16**
VoipSeriveStatus	• VoIP Service Status. • Values – 0 - IMS VoIP service is not available – 2 - IMS VoIP is in full service • Bit to check in ParamPresenceMask - **17**
VtServiceStatus	• VT Service Status • Values – 0 - IMS VT service is not available – 2 - IMS VT is in full service • Bit to check in ParamPresenceMask - **18**
SmsServiceRat	• SMS service RAT • Values – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN • Bit to check in ParamPresenceMask - **19**
VoipServiceRat	• VoIP service RAT. • Values – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN • Bit to check in ParamPresenceMask - **20**
VtServiceRat	• VT service RAT. • Values – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN • Bit to check in ParamPresenceMask - **21**

UtServiceStatus	• UT service Status. • Values – 0 - IMS UT service is not available – 2 - IMS UT is in full service • Bit to check in ParamPresenceMask - **22**
UtServiceRat	• UT service RAT. • Values – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN • Bit to check in ParamPresenceMask - **23**
VsServiceStatus	• VS service Status. • Values – 0 - IMS UT service is not available – 2 - IMS UT is in full service • Bit to check in ParamPresenceMask - **24**
VsServiceRat	• VS service RAT. • Values – 0 - IMS service is registered on WLAN – 1 - IMS service is registered on WWAN – 2 - IMS service is registered on interworking WLAN • Bit to check in ParamPresenceMask - **25**

8.901.2 Field Documentation

8.901.2.1 swi_uint256_t unpack_imsa_SLQSGetIMSAServiceStatus_t::ParamPresenceMask

8.901.2.2 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::SmsServiceRat

8.901.2.3 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::SmsServiceStatus

8.901.2.4 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::UtServiceRat

8.901.2.5 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::UtServiceStatus

8.901.2.6 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VoipServiceRat

8.901.2.7 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VoipServiceStatus

8.901.2.8 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VsServiceRat

8.901.2.9 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VsServiceStatus

8.901.2.10 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VtServiceRat

8.901.2.11 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VtServiceStatus

8.902 unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t Struct Reference

Data Fields

- uint8_t PdpConnState
- imsa_IMSFailErrCodeTlv FailErrCode
- swi_uint256_t ParamPresenceMask

8.902.1 Detailed Description

Structure used to store IMSA PDP status indication Parameters.

Parameters

PdpConnState	• Values: – 0- not connected – 1- connected • Bit to check in ParamPresenceMask - **1**
FailErrCode	• Fail error code when PDP is not connected • See imsa_IMSFailErrCodeTlv for more information • Bit to check in ParamPresenceMask - **16**

Note

: None

8.902.2 Field Documentation

8.902.2.1 imsa_IMSFailErrCodeTlv unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t::FailErrCode

8.902.2.2 swi_uint256_t unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t::ParamPresenceMask

8.902.2.3 uint8_t unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t::PdpConnState

8.903 unpack_imsa_SLQSImsaRatStatusCallBack_ind_t Struct Reference

Data Fields

- imsa_RatHandoverStatusInfo RatHandover
- swi_uint256_t ParamPresenceMask

8.903.1 Detailed Description

Structure used to store IMSA RAT status indication Parameters.

Parameters

RatHandover	• RAT handover status info • See imsa_RatHandoverStatusInfo for more information • Bit to check in ParamPresenceMask - **16**

Note

: None

8.903.2 Field Documentation

8.903.2.1 swi_uint256_t unpack_imsa_SLQSImsaRatStatusCallBack_ind_t::ParamPresenceMask

8.903.2.2 imsa_RatHandoverStatusInfo unpack_imsa_SLQSImsaRatStatusCallBack_ind_t::RatHandover

8.904 unpack_imsa_SLQSImsaRegStatusCallBack_ind_t Struct Reference

Data Fields

- imsa_IMSRegStatusInfo IMSRegistration
- imsa_IMSRegStatusErrorCodeInfo IMSRegistrationError
- imsa_NewIMSRegStatusInfo NewIMSRegistration
- swi_uint256_t ParamPresenceMask

8.904.1 Detailed Description

Structure used to store IMSA registration status indication Parameters.

Parameters

IMSRegistration	• IMS registration info • See imsa_IMSRegStatusInfo for more information • Bit to check in ParamPresenceMask - **1**
IMSRegistration-Error	• IMS registration error • see imsa_IMSRegStatusErrorCodeInfo for more information • Bit to check in ParamPresenceMask - **16**
NewIMS-Registration	• New IMS registration info • see imsa_NewIMSRegStatusInfo for more information • Bit to check in ParamPresenceMask - **17**

Note

: None

8.904.2 Field Documentation

8.904.2.1 imsa_IMSRegStatusInfo unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::IMSRegistration

8.904.2.2 imsa_IMSRegStatusErrorCodeInfo unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::IMSRegistrationError

8.904.2.3 imsa_NewIMSRegStatusInfo unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::NewIMSRegistration

8.904.2.4 swi_uint256_t unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::ParamPresenceMask

8.905 unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t Struct Reference

Data Fields

- imsa_SmsSvcStatusInfo SmsService
- imsa_VoipSvcStatusInfo VoipService
- imsa_VtSvcStatusInfo VtService
- imsa_SmsRatInfo SmsRat
- imsa_VoipRatInfo VoipRat
- imsa_VtRatInfo VtRat
- imsa_UtSvcStatusInfo UtService
- imsa_UtRatInfo UtRat
- swi_uint256_t ParamPresenceMask

8.905.1 Detailed Description

Structure used to store IMSA service status indication Parameters.

Parameters

SmsService	• SMS service info • See imsa_SmsSvcStatusInfo for more information • Bit to check in ParamPresenceMask - **16**
VoipService	• VOIP service info • see imsa_VoipSvcStatusInfo for more information • Bit to check in ParamPresenceMask - **17**
VtService	• VT service info • see imsa_VtSvcStatusInfo for more information • Bit to check in ParamPresenceMask - **18**
SmsRat	• SMS RAT info • see imsa_SmsRatInfo for more information • Bit to check in ParamPresenceMask - **19**
VoipRat	• VOIP RAT info • see imsa_VoipRatInfo for more information • Bit to check in ParamPresenceMask - **20**

VtRat	• VT RAT info • see imsa_VtRatInfo for more information • Bit to check in ParamPresenceMask - **21**
UtService	• UT service info • see imsa_UtSvcStatusInfo for more information • Bit to check in ParamPresenceMask - **22**
UtRat	• UT RAT info • see imsa_UtRatInfo for more information • Bit to check in ParamPresenceMask - **23**

Note

: None

8.905.2 Field Documentation

8.905.2.1 swi_uint256_t unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::ParamPresenceMask

8.905.2.2 imsa_SmsRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::SmsRat

8.905.2.3 imsa_SmsSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::SmsService

8.905.2.4 imsa_UtRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::UtRat

8.905.2.5 imsa_UtSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::UtService

8.905.2.6 imsa_VoipRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VoipRat

8.905.2.7 imsa_VoipSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VoipService

8.905.2.8 imsa_VtRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VtRat

8.905.2.9 imsa_VtSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VtService

8.906 unpack_loc_BestAvailPos_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint32_t * pXid
- uint64_t * pLatitude
- uint64_t * pLongitude
- uint32_t * pHorUncCircular
- uint32_t * pAltitudeWrtEllipsoid
- uint32_t * pVertUnc
- uint64_t * pTimestampUtc
- uint32_t * pTimeUnc

- uint32_t * pHorUncEllipseSemiMinor
- uint32_t * pHorUncEllipseSemiMajor
- uint32_t * pHorUncEllipseOrientAzimuth
- uint8_t * pHorCirConf
- uint8_t * pHorEllpConf
- uint32_t * pHorReliability
- uint32_t * pSpeedHorizontal
- uint32_t * pSpeedUnc
- uint32_t * pAltitudeWrtMeanSeaLevel
- uint8_t * pVertConfidence
- uint32_t * pVertReliability
- uint32_t * pSpeedVertical
- uint32_t * pSpeedVerticalUnc
- uint32_t * pHeading
- uint32_t * pHeadingUnc
- uint32_t * pMagneticDeviation
- uint32_t * pTechnologyMask
- loc_precisionDilution * pPrecisionDilution
- loc_gpsTime * pGpsTime
- uint32_t * pTimeSrc
- loc_sensorDataUsage * pSensorDataUsage
- loc_svUsedforFix * pSvUsedforFix
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.906.1 Detailed Description

This structure contains Best Available Position

Parameters

status	• Valid values: – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
xid	Transaction ID that was specified in the Get Best Available Position request. • Bit to check in ParamPresenceMask - **16**

pLatitude	• Latitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude • Bit to check in ParamPresenceMask - **17**
pLongitude	• Longitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude • Bit to check in ParamPresenceMask - **18**
pHorUncCircular	• Horizontal position uncertainty. • Units - Meters • Bit to check in ParamPresenceMask - **19**
pAltitudeWrt-Ellipsoid	• Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883 • Bit to check in ParamPresenceMask - **20**
pVertUnc	• Vertical uncertainty. • Units - Meters • Bit to check in ParamPresenceMask - **21**
pTimestampUtc	• UTC timestamp • Units - Milliseconds since Jan. 1, 1970 • Bit to check in ParamPresenceMask - **22**
pTimeUnc	• Time uncertainty. • Units - Milliseconds • Bit to check in ParamPresenceMask - **23**
pHorUncEllipse-SemiMinor	• Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters • Bit to check in ParamPresenceMask - **24**

pHorUncEllipse-SemiMajor	• Semi-major axis of horizontal elliptical uncertainty. • Units: Meters • Bit to check in ParamPresenceMask - **25**
pHorUncEllipse-OrientAzimuth	• Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180 • Bit to check in ParamPresenceMask - **26**
pHorCirConf	• Horizontal circular uncertainty confidence • Units: Precent • Range: 0 to 99 • Bit to check in ParamPresenceMask - **27**
pHorEllpConf	• Horizontal elliptical uncertainty confidence • Units: Precent • Range: 0 to 99 • Bit to check in ParamPresenceMask - **28**
pHorReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed • Bit to check in ParamPresenceMask - **29**
pSpeed-Horizontal	• Horizontal speed. • Units - Meters/second • Bit to check in ParamPresenceMask - **30**
pSpeedUnc	• 3-D Speed uncertainty. • Units - Meters/second. • Bit to check in ParamPresenceMask - **31**
pAltitudeWrt-MeanSeaLevel	• Altitude With Respect to Sea Level. • Units - Meters • Bit to check in ParamPresenceMask - **32**

pVertConfidence	• Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99 • Bit to check in ParamPresenceMask - **33**
pVertReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed • Bit to check in ParamPresenceMask - **34**
pSpeedVertical	• Vertical speed. • Units - Meters/second • Bit to check in ParamPresenceMask - **35**
pSpeedVertical-Unc	• Veritical speed • Units: Meters/second • Bit to check in ParamPresenceMask - **36**
pHeading	• Heading. • Units - Degree • Range 0 to 359.999 • Bit to check in ParamPresenceMask - **37**
pHeadingUnc	• Heading uncertainty. • Units - Degree • Range 0 to 359.999 • Bit to check in ParamPresenceMask - **38**
pMagnetic-Deviation	• Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north. • Bit to check in ParamPresenceMask - **39**

pTechnology-Mask	• Values – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix • Bit to check in ParamPresenceMask - **40**
-pPrecision-Dilution	• See loc_precisionDilution for more information • Bit to check in ParamPresenceMask - **41**
-pGpsTime	• See loc_gpsTime for more information • Bit to check in ParamPresenceMask - **42**
pTimeSrc	• Values – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites. • Bit to check in ParamPresenceMask - **43**
-pSensorData-Usage	• See loc_sensorDataUsage for more information • Bit to check in ParamPresenceMask - **44**
-pSvUsedforFix	• See loc_svUsedforFix for more information • Bit to check in ParamPresenceMask - **45**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.906.2 Field Documentation

8.906.2.1 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtEllipsoid

8.906.2.2 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtMeanSeaLevel

8.906.2.3 swi_uint256_t unpack_loc_BestAvailPos_Ind_t::ParamPresenceMask

8.906.2.4 loc_gpsTime* unpack_loc_BestAvailPos_Ind_t::pGpsTime

8.906.2.5 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeading

8.906.2.6 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeadingUnc

8.906.2.7 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorCirConf

8.906.2.8 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorEllpConf

8.906.2.9 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorReliability

8.906.2.10 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncCircular

8.906.2.11 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseOrientAzimuth

8.906.2.12 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMajor

8.906.2.13 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMinor

8.906.2.14 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLatitude

8.906.2.15 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLongitude

8.906.2.16 uint32_t* unpack_loc_BestAvailPos_Ind_t::pMagneticDeviation

8.906.2.17 loc_precisionDilution* unpack_loc_BestAvailPos_Ind_t::pPrecisionDilution

8.906.2.18 loc_sensorDataUsage* unpack_loc_BestAvailPos_Ind_t::pSensorDataUsage

8.906.2.19 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedHorizontal

8.906.2.20 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedUnc

8.906.2.21 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVertical

8.906.2.22 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVerticalUnc

8.906.2.23 loc_svUsedforFix* unpack_loc_BestAvailPos_Ind_t::pSvUsedforFix

8.906.2.24 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTechnologyMask

8.906.2.25 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeSrc

8.906.2.26 uint64_t* unpack_loc_BestAvailPos_Ind_t::pTimestampUtc

8.906.2.27 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeUnc

8.906.2.28 uint8_t* unpack_loc_BestAvailPos_Ind_t::pVertConfidence

8.906.2.29 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertReliability

8.906.2.30 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertUnc

8.906.2.31 uint32_t* unpack_loc_BestAvailPos_Ind_t::pXid

8.906.2.32 uint32_t unpack_loc_BestAvailPos_Ind_t::status

8.906.2.33 uint16_t unpack_loc_BestAvailPos_Ind_t::Tlvresult

8.907 unpack_loc_CradleMountCallback_Ind_t Struct Reference

Data Fields

- uint32_t cradleMountConfigStatus
- swi_uint256_t ParamPresenceMask

8.907.1 Detailed Description

This structure contains LOC Cradle Mount Config Status

Parameters

cradleMount-ConfigStatus	• Values – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.907.2 Field Documentation

8.907.2.1 uint32_t unpack_loc_CradleMountCallback_Ind_t::cradleMountConfigStatus

8.907.2.2 swi_uint256_t unpack_loc_CradleMountCallback_Ind_t::ParamPresenceMask

8.908 unpack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.908.1 Detailed Description

This structure contains LOC delete assist data unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.908.2 Field Documentation

8.908.2.1 swi_uint256_t unpack_loc_Delete_Assist_Data_t::ParamPresenceMask

8.908.2.2 uint16_t unpack_loc_Delete_Assist_Data_t::Tlvresult

8.909 unpack_loc_DeleteAssistData_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.909.1 Detailed Description

Contain the parameters passed for SetLocDeleteAssistDataCallback by the device.

Parameters

status	• Status of the Delete Assist Data request • Valid values: – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
TlvResult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.909.2 Field Documentation

8.909.2.1 swi_uint256_t unpack_loc_DeleteAssistData_Ind_t::ParamPresenceMask

8.909.2.2 uint32_t unpack_loc_DeleteAssistData_Ind_t::status

8.909.2.3 uint16_t unpack_loc_DeleteAssistData_Ind_t::TlvResult

8.910 unpack_loc_EngineState_Ind_t Struct Reference

Data Fields

- uint32_t engineState
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.910.1 Detailed Description

This structure contains LOC Engine State field.

Parameters

engineState	• Location engine state. • Valid values – 1 - Location engine is on – 2 - Loction engine is off • Bit to check in ParamPresenceMask - **1**
TlvResult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.910.2 Field Documentation

8.910.2.1 uint32_t unpack_loc_EngineState_Ind_t::engineState

8.910.2.2 swi_uint256_t unpack_loc_EngineState_Ind_t::ParamPresenceMask

8.910.2.3 uint16_t unpack_loc_EngineState_Ind_t::TlvResult

8.911 unpack_loc_EventNMEA_Ind_t Struct Reference

Data Fields

- char NMEAData [201]
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.911.1 Detailed Description

Contain the parameters passed for SetLocEventMaskNMEACallback by the device.

Parameters

NMEAData	• NMEA string • Type - NULL terminated string • Maximum string length (including NULL terminator) - 201 • Bit to check in ParamPresenceMask - **1**
TlvResult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.911.2 Field Documentation

8.911.2.1 char unpack_loc_EventNMEA_Ind_t::NMEAData[201]

8.911.2.2 swi_uint256_t unpack_loc_EventNMEA_Ind_t::ParamPresenceMask

8.911.2.3 uint16_t unpack_loc_EventNMEA_Ind_t::Tlvresult

8.912 unpack_loc_EventRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.912.1 Detailed Description

This structure contains Event Register unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.912.2 Field Documentation

8.912.2.1 swi_uint256_t unpack_loc_EventRegister_t::ParamPresenceMask

8.912.2.2 uint16_t unpack_loc_EventRegister_t::Tlvresult

8.913 unpack_loc_EventTimeSyncCallback_Ind_t Struct Reference

Data Fields

- uint32_t timeSyncRefCounter
- swi_uint256_t ParamPresenceMask

8.913.1 Detailed Description

This structure contains LOC Event Time Sync Reference COunter

Parameters

timeSyncRef-Counter	• Sent by the location engine when it needs to synchronize location engine and control point (sensor processor) times. • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.913.2 Field Documentation

8.913.2.1 swi_uint256_t unpack_loc_EventTimeSyncCallback_Ind_t::ParamPresenceMask

8.913.2.2 uint32_t unpack_loc_EventTimeSyncCallback_Ind_t::timeSyncRefCounter

8.914 unpack_loc_FixCriteria_Ind_t Struct Reference

Data Fields

- loc_FixCriteriaStatusTlv loc_fixCriteriaStatus
- loc_HorAccuracyLvlTlv loc_horAccuracy
- loc_IntermediateRptStateTlv loc_intermediateRptState
- loc_MinIntervalTlv loc_minInterval
- loc_AppProviderInfoTlv loc_appProviderInfo
- swi_uint256_t ParamPresenceMask

8.914.1 Detailed Description

Contain the parameters passed for SetLocGetFixCriteriaCallback by the device.

Parameters

fixCriteriaStatus	• Status of the get fix criteria request • See loc_FixCriteriaStatusTlv for more information.
horAccuracyLvl	• Horizontal accuracy level • See loc_HorAccuracyLvlTlv for more information.
intermediateRpt-State	• Intermediate Report state (ON, OFF). • See loc_IntermediateRptStateTlv for more information.
minInterval	• Minimum Interval Between fixes • See loc_MinIntervalTlv for more information.
appProviderInfo	• ID of the Application that Sent the Position Request Application provider, name, and version. – See loc_AppProviderInfoTlv for more information.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.914.2 Field Documentation

8.914.2.1 loc_AppProviderInfoTlv unpack_loc_FixCriteria_Ind_t::loc_appProviderInfo

8.914.2.2 loc_FixCriteriaStatusTlv unpack_loc_FixCriteria_Ind_t::loc_fixCriteriaStatus

8.914.2.3 loc_HorAccuracyLvlTlv unpack_loc_FixCriteria_Ind_t::loc_horAccuracy

8.914.2.4 loc_IntermediateRptStateTlv unpack_loc_FixCriteria_Ind_t::loc_intermediateRptState

8.914.2.5 loc_MinIntervalTlv unpack_loc_FixCriteria_Ind_t::loc_minInterval

8.914.2.6 swi_uint256_t unpack_loc_FixCriteria_Ind_t::ParamPresenceMask

8.915 unpack_loc_GetOpMode_Ind_t Struct Reference

Data Fields

- uint32_t Status
- uint32_t * pMode
- swi_uint256_t ParamPresenceMask

8.915.1 Detailed Description

Contain the parameters passed for SetLocGetOpModeCallback by the device.

Parameters

Status	• Status of the Get Server request. • Valid values – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure – 0xffffffff - Invalid data. • Bit to check in ParamPresenceMask - **1**

pMode	• Current operation mode. • Valid values: – eQMI_LOC_OPER_MODE_DEFAULT(1) - Use the default engine mode – eQMI_LOC_OPER_MODE_MSB (2)- Use the MS-based mode – eQMI_LOC_OPER_MODE_MSA (3)- Use the MS-assisted mode – eQMI_LOC_OPER_MODE_STANDALONE (4) - Use Standalone mode – eQMI_LOC_OPER_MODE_CELL_ID (5) - Use cell ID; this mode is only valid for GSM/UMTS networks – eQMI_LOC_OPER_MODE_WWAN(6) - Use WWAN measurements to calculate the position; if this mode is set, AFLT will be used for 1X networks and OTDOA will be used for LTE networks • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.915.2 Field Documentation

8.915.2.1 swi_uint256_t unpack_loc_GetOpMode_Ind_t::ParamPresenceMask

8.915.2.2 uint32_t* unpack_loc_GetOpMode_Ind_t::pMode

8.915.2.3 uint32_t unpack_loc_GetOpMode_Ind_t::Status

8.916 unpack_loc_GetServer_Ind_t Struct Reference

Data Fields

- uint32_t serverStatus
- uint32_t serverType
- loc_IPv4Info * pIPv4AddrInfo
- loc_IPv6Info * pIPv6AddrInfo
- loc_urlAddr * pURL
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.916.1 Detailed Description

Contain the parameters passed for SetLocGetServerCallback by the device.

Parameters

serverStatus	• Status of the Get Server request. • Valid values – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
serverType	• Type of server. • Valid values – eQMI_LOC_SERVER_TYPE_CDMA_PDE (1) - Server type is CDMA PDE – eQMI_LOC_SERVER_TYPE_CDMA_MPC (2) - Server type is CDMA MPC – eQMI_LOC_SERVER_TYPE_UMTS_SLP (3) - Server type is UMTS SLP – eQMI_LOC_SERVER_TYPE_CUSTOM_PDE (4) - Server type is custom PDE • Bit to check in ParamPresenceMask - **2**
pIPv4AddrInfo	• See loc_IPv4Info for more information. • Bit to check in ParamPresenceMask - **16**
pIPv6AddrInfo	• See loc_IPv6Info for more information. • Bit to check in ParamPresenceMask - **17**
pURL	• See loc_urlAddr for more information. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.916.2 Field Documentation

8.916.2.1 swi_uint256_t unpack_loc_GetServer_Ind_t::ParamPresenceMask

8.916.2.2 loc_IPv4Info* unpack_loc_GetServer_Ind_t::pIPv4AddrInfo

8.916.2.3 loc_IPv6Info* unpack_loc_GetServer_Ind_t::pIPv6AddrInfo

8.916.2.4 loc_urlAddr* unpack_loc_GetServer_Ind_t::pURL

8.916.2.5 uint32_t unpack_loc_GetServer_Ind_t::serverStatus

8.916.2.6 uint32_t unpack_loc_GetServer_Ind_t::serverType

8.916.2.7 uint16_t unpack_loc_GetServer_Ind_t::Tlvresult

8.917 unpack_loc_GnssSvInfo_Ind_t Struct Reference

Data Fields

- uint8_t altitudeAssumed
- loc_satelliteInfo * pSatelliteInfo
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.917.1 Detailed Description

Contain the parameters passed for SetLocGnssSvInfoCallback by the device.

Parameters

altitudeAssumed	• Indicates whether altitude is assumed or calculated – 0x00 (FALSE) - Valid altitude is calculated – 0x01 (TRUE) - Valid altitude is assumed; there may not be enough satellites to determine precise altitude • Bit to check in ParamPresenceMask - **1**
pSatelliteInfo	• See loc_satelliteInfo for more information. • Bit to check in ParamPresenceMask - **16**
Tlvresult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.917.2 Field Documentation

8.917.2.1 uint8_t unpack_loc_GnssSvInfo_Ind_t::altitudeAssumed

8.917.2.2 swi_uint256_t unpack_loc_GnssSvInfo_Ind_t::ParamPresenceMask

8.917.2.3 loc_satelliteInfo* unpack_loc_GnssSvInfo_Ind_t::pSatelliteInfo

8.917.2.4 uint16_t unpack_loc_GnssSvInfo_Ind_t::Tlvresult

8.918 unpack_loc_InjectPositionCallback_Ind_t Struct Reference

Data Fields

- uint32_t status
- swi_uint256_t ParamPresenceMask

8.918.1 Detailed Description

Contain the parameters passed for LOC Inject Position indication by the device.

Parameters

status	• Position Injection Status • Valid values: – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.918.2 Field Documentation

8.918.2.1 swi_uint256_t unpack_loc_InjectPositionCallback_Ind_t::ParamPresenceMask

8.918.2.2 uint32_t unpack_loc_InjectPositionCallback_Ind_t::status

8.919 unpack_loc_InjectSensorDataCallback_Ind_t Struct Reference

Data Fields

- uint32_t injectSensorDataStatus
- uint32_t * pOpaqueIdentifier
- uint8_t * pAccelSamplesAccepted
- uint8_t * pGyroSamplesAccepted
- uint8_t * pAccelTempSamplesAccepted

- uint8_t * pGyroTempSamplesAccepted
- swi_uint256_t ParamPresenceMask

8.919.1 Detailed Description

This structure contains LOC Inject Sensor Data

Parameters

injectSensorDataStatus	• Values – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
pOpaqueIdentifier[OPTIONAL]	• Sent in by the client echoed so the client can relate the indication to the request. • NULL when this TLV not present • Bit to check in ParamPresenceMask - **16**
pAccelSamplesAccepted[OPTIONAL]	• Lets the client know how many 3-axis accelerometer samples were accepted. • This field is present only if the accelerometer samples were sent in the request. • NULL when this TLV not present • Bit to check in ParamPresenceMask - **17**
pGyroSamplesAccepted[OPTIONAL]	• Lets the client know how many 3-axis gyroscope samples were accepted. • This field is present only if the gyroscope samples were sent in the request. • NULL when this TLV not present • Bit to check in ParamPresenceMask - **18**
pAccelTempSamplesAccepted[OPTIONAL]	• Lets the client know how many accelerometer temperature samples were accepted. • This field is present only if the accelerometer temperature samples were sent in the request. • NULL when this TLV not present • Bit to check in ParamPresenceMask - **19**
pGyroTempSamplesAccepted[OPTIONAL]	• Lets the client know how many gyroscope temperature samples were accepted. • This field is present only if the gyroscope temperature samples were sent in the request. • NULL when this TLV not present • Bit to check in ParamPresenceMask - **20**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.919.2 Field Documentation

8.919.2.1 uint32_t unpack_loc_InjectSensorDataCallback_Ind_t::injectSensorDataStatus

8.919.2.2 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pAccelSamplesAccepted

8.919.2.3 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pAccelTempSamplesAccepted

8.919.2.4 swi_uint256_t unpack_loc_InjectSensorDataCallback_Ind_t::ParamPresenceMask

8.919.2.5 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pGyroSamplesAccepted

8.919.2.6 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pGyroTempSamplesAccepted

8.919.2.7 uint32_t* unpack_loc_InjectSensorDataCallback_Ind_t::pOpaqueIdentifier

8.920 unpack_loc_InjectTimeSyncDataCallback_Ind_t Struct Reference

Data Fields

- uint32_t injectTimeSyncStatus
- swi_uint256_t ParamPresenceMask

8.920.1 Detailed Description

This structure contains LOC Inject Time Sync Data Status

Parameters

injectTimeSync-Status	• Values – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure

- Bit to check in ParamPresenceMask - **1**

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.920.2 Field Documentation

8.920.2.1 uint32_t unpack_loc_InjectTimeSyncDataCallback_Ind_t::injectTimeSyncStatus

8.920.2.2 swi_uint256_t unpack_loc_InjectTimeSyncDataCallback_Ind_t::ParamPresenceMask

8.921 unpack_loc_InjectUTCTimeCallback_Ind_t Struct Reference

Data Fields

- uint32_t status
- swi_uint256_t ParamPresenceMask

8.921.1 Detailed Description

Contain the parameters passed for Inject UTC time indication by the device.

Parameters

status	• Status of the UTC Time Injection request • Valid values: – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.921.2 Field Documentation

8.921.2.1 swi_uint256_t unpack_loc_InjectUTCTimeCallback_Ind_t::ParamPresenceMask

8.921.2.2 uint32_t unpack_loc_InjectUTCTimeCallback_Ind_t::status

8.922 unpack_loc_PositionRpt_Ind_t Struct Reference

Data Fields

- uint32_t sessionStatus
- uint8_t sessionId
- uint64_t * pLatitude
- uint64_t * pLongitude
- uint32_t * pHorUncCircular
- uint32_t * pHorUncEllipseSemiMinor
- uint32_t * pHorUncEllipseSemiMajor
- uint32_t * pHorUncEllipseOrientAzimuth
- uint8_t * pHorConfidence
- uint32_t * pHorReliability
- uint32_t * pSpeedHorizontal
- uint32_t * pSpeedUnc
- uint32_t * pAltitudeWrtEllipsoid
- uint32_t * pAltitudeWrtMeanSeaLevel
- uint32_t * pVertUnc
- uint8_t * pVertConfidence
- uint32_t * pVertReliability
- uint32_t * pSpeedVertical
- uint32_t * pHeading
- uint32_t * pHeadingUnc
- uint32_t * pMagneticDeviation
- uint32_t * pTechnologyMask
- loc_precisionDilution * pPrecisionDilution
- uint64_t * pTimestampUtc
- uint8_t * pLeapSeconds
- loc_gpsTime * pGpsTime
- uint32_t * pTimeUnc
- uint32_t * pTimeSrc
- loc_sensorDataUsage * pSensorDataUsage
- uint32_t * pFixId
- loc_svUsedforFix * pSvUsedforFix
- uint8_t * pAltitudeAssumed
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.922.1 Detailed Description

This structure contains Event Position Report Indication unpack

Parameters

sessionStatus	• Values – 0 - Session was successful – 1 - Session is still in progress; further position reports will be generated until either the fix criteria specified by the client are met or the client response timeout occurs. – 2 - Session failed.. – 3 - Fix request failed because the session timed out. – 4 - Fix request failed because the session was ended by the user. – 5 - Fix request failed due to bad parameters in the request. – 6 - Fix request failed because the phone is offline. – 7 - Fix request failed because the engine is locked • Bit to check in ParamPresenceMask - **1**
sessionId	• ID of the session that was specified in the Start request • Range - 0 to 255 • Bit to check in ParamPresenceMask - **2**
pLatitude	• Latitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude • Bit to check in ParamPresenceMask - **16**
pLongitude	• Longitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude • Bit to check in ParamPresenceMask - **17**
pHorUncCircular	• Horizontal position uncertainty. • Units - Meters • Bit to check in ParamPresenceMask - **18**
pHorUncEllipse-SemiMinor	• Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters • Bit to check in ParamPresenceMask - **19**
pHorUncEllipse-SemiMajor	• Semi-major axis of horizontal elliptical uncertainty. • Units: Meters • Bit to check in ParamPresenceMask - **20**

pHorUncEllipse-OrientAzimuth	• Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180 • Bit to check in ParamPresenceMask - **21**
pHorConfidence	• Horizontal uncertainty confidence. • If both elliptical and horizontal uncertainties are specified in this message, the confidence corresponds to the elliptical uncertainty. • Units - Percentage • Range 0-99 • Bit to check in ParamPresenceMask - **22**
pHorReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed

• Bit to check in ParamPresenceMask - **23**

Parameters

pSpeed-Horizontal	• Horizontal speed. • Units - Meters/second • Bit to check in ParamPresenceMask - **24**
pSpeedUnc	• 3-D Speed uncertainty. • Units - Meters/second. • Bit to check in ParamPresenceMask - **25**
pAltitudeWrt-Ellipsoid	• Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883 • Bit to check in ParamPresenceMask - **26**
pAltitudeWrt-MeanSeaLevel	• Altitude With Respect to Sea Level. • Units - Meters • Bit to check in ParamPresenceMask - **27**
pVertUnc	• Vertical uncertainty. • Units - Meters • Bit to check in ParamPresenceMask - **28**

pVertConfidence	• Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99 • Bit to check in ParamPresenceMask - **29**
pVertReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed – Bit to check in ParamPresenceMask - **30**
pSpeedVertical	• Vertical speed. • Units - Meters/second • Bit to check in ParamPresenceMask - **31**
pHeading	• Heading. • Units - Degree • Range 0 to 359.999 • Bit to check in ParamPresenceMask - **32**
pHeadingUnc	• Heading uncertainty. • Units - Degree • Range 0 to 359.999 • Bit to check in ParamPresenceMask - **33**
pMagnetic-Deviation	• Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north. • Bit to check in ParamPresenceMask - **34**
pTechnology-Mask	• Values – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix • Bit to check in ParamPresenceMask - **35**

pPrecision-Dilution	• See loc_precisionDilution for more information • Bit to check in ParamPresenceMask - **36**
pTimestampUtc	• UTC timestamp • Units - Milliseconds since Jan. 1, 1970 • Bit to check in ParamPresenceMask - **37**
pLeapSeconds	• Leap second information. If leapSeconds is not available, timestampUtc is calculated based on a hard-coded value for leap seconds. • Units - Seconds • Bit to check in ParamPresenceMask - **38**
pGpsTime	• See loc_gpsTime for more information • Bit to check in ParamPresenceMask - **39**
pTimeUnc	• Time uncertainty. • Units - Milliseconds • Bit to check in ParamPresenceMask - **40**
pTimeSrc	• Values – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites.

- Bit to check in ParamPresenceMask - **41**

Parameters

pSensorData-Usage	• See loc_sensorDataUsage for more information • Bit to check in ParamPresenceMask - **42**
pFixId	• Fix count for the session. Starts with 0 and increments by one for each successive position report for a particular session. • Bit to check in ParamPresenceMask - **43**
pSvUsedforFix	• See loc_svUsedforFix for more information • Bit to check in ParamPresenceMask - **44**
pAltitude-Assumed	• Indicates whether altitude is assumed or calculated. • Value – 0x00 - Altitude is calculated – 0x01 - Altitude is assumed • Bit to check in ParamPresenceMask - **45**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.922.2 Field Documentation

8.922.2.1 uint8_t* unpack_loc_PositionRpt_Ind_t::pAltitudeAssumed

8.922.2.2 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtEllipsoid

8.922.2.3 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtMeanSeaLevel

8.922.2.4 swi_uint256_t unpack_loc_PositionRpt_Ind_t::ParamPresenceMask

8.922.2.5 uint32_t* unpack_loc_PositionRpt_Ind_t::pFixId

8.922.2.6 loc_gpsTime* unpack_loc_PositionRpt_Ind_t::pGpsTime

8.922.2.7 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeading

8.922.2.8 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeadingUnc

8.922.2.9 uint8_t* unpack_loc_PositionRpt_Ind_t::pHorConfidence

8.922.2.10 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorReliability

8.922.2.11 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncCircular

8.922.2.12 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseOrientAzimuth

8.922.2.13 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMajor

8.922.2.14 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMinor

8.922.2.15 uint64_t* unpack_loc_PositionRpt_Ind_t::pLatitude

8.922.2.16 uint8_t* unpack_loc_PositionRpt_Ind_t::pLeapSeconds

8.922.2.17 uint64_t* unpack_loc_PositionRpt_Ind_t::pLongitude

8.922.2.18 uint32_t* unpack_loc_PositionRpt_Ind_t::pMagneticDeviation

8.922.2.19 loc_precisionDilution* unpack_loc_PositionRpt_Ind_t::pPrecisionDilution

8.922.2.20 loc_sensorDataUsage* unpack_loc_PositionRpt_Ind_t::pSensorDataUsage

8.922.2.21 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedHorizontal

8.922.2.22 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedUnc

8.922.2.23 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedVertical

8.922.2.24 loc_svUsedforFix* unpack_loc_PositionRpt_Ind_t::pSvUsedforFix

8.922.2.25 uint32_t* unpack_loc_PositionRpt_Ind_t::pTechnologyMask

8.922.2.26 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeSrc

8.922.2.27 uint64_t* unpack_loc_PositionRpt_Ind_t::pTimestampUtc

8.922.2.28 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeUnc

8.922.2.29 uint8_t* unpack_loc_PositionRpt_Ind_t::pVertConfidence

8.922.2.30 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertReliability

8.922.2.31 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertUnc

8.922.2.32 uint8_t unpack_loc_PositionRpt_Ind_t::sessionId

8.922.2.33 uint32_t unpack_loc_PositionRpt_Ind_t::sessionStatus

8.922.2.34 uint16_t unpack_loc_PositionRpt_Ind_t::Tlvresult

8.923 unpack_loc_SensorStreamingCallback_Ind_t Struct Reference

Data Fields

- loc_accelAcceptReady * pAccelAcceptReady
- loc_gyroAcceptReady * pGyroAcceptReady
- loc_accelTempAcceptReady * pAccelTempAcceptReady
- loc_gyroTempAcceptReady * pGyroTempAcceptReady
- swi_uint256_t ParamPresenceMask

8.923.1 Detailed Description

This structure contains LOC Event Sensor Streaming Ready Status

Parameters

-pAccelAccept-Ready[OPTION-AL]	• See loc_accelAcceptReady for more information • Bit to check in ParamPresenceMask - **16**
-pGyroAccept-Ready[OPTION-AL]	• See loc_gyroAcceptReady for more information • Bit to check in ParamPresenceMask - **17**
-pAccelTemp-AcceptReady[O-PTIONAL]	• See loc_accelTempAcceptReady for more information • Bit to check in ParamPresenceMask - **18**
-pGyroTemp-AcceptReady[O-PTIONAL]	• See loc_gyroTempAcceptReady for more information • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.923.2 Field Documentation

8.923.2.1 loc_accelAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pAccelAcceptReady

8.923.2.2 loc_accelTempAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pAccelTempAcceptReady

8.923.2.3 swi_uint256_t unpack_loc_SensorStreamingCallback_Ind_t::ParamPresenceMask

8.923.2.4 loc_gyroAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pGyroAcceptReady

8.923.2.5 loc_gyroTempAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pGyroTempAcceptReady

8.924 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.924.1 Detailed Description

This structure contains LOC Set External Power Configure status field.

Parameters

status	• Valid values – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
TlvResult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.924.2 Field Documentation

8.924.2.1 swi_uint256_t unpack_loc_SetExtPowerConfig_Ind_t::ParamPresenceMask

8.924.2.2 uint32_t unpack_loc_SetExtPowerConfig_Ind_t::status

8.924.2.3 uint16_t unpack_loc_SetExtPowerConfig_Ind_t::TlvResult

8.925 unpack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.925.1 Detailed Description

This structure contains Set Ext Power State unpack

Parameters

TlvResult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.925.2 Field Documentation

8.925.2.1 swi_uint256_t unpack_loc_SetExtPowerState_t::ParamPresenceMask

8.925.2.2 uint16_t unpack_loc_SetExtPowerState_t::Tlvresult

8.926 unpack_loc_SetOperationMode_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.926.1 Detailed Description

This structure contains LOC Set External Power Configure status field.

Parameters

status	• Status of the Set Operation Mode request. • Valid values: – 0 - Request was completed successfully – 1 - Request failed because of a general failure – 2 - Request failed because it is not supported – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - Request failed because the engine could not allocate sufficient memory for the request – 9 - Request failed because the maximum number of Geofences are already programmed – 10 - Location service failed because of an XTRA version-based file format check failure • Bit to check in ParamPresenceMask - **1**
Tlvresult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.926.2 Field Documentation

8.926.2.1 swi_uint256_t unpack_loc_SetOperationMode_Ind_t::ParamPresenceMask

8.926.2.2 uint32_t unpack_loc_SetOperationMode_Ind_t::status

8.926.2.3 uint16_t unpack_loc_SetOperationMode_Ind_t::Tlvresult

8.927 unpack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.927.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.927.2 Field Documentation

8.927.2.1 swi_uint256_t unpack_loc_SetOperationMode_t::ParamPresenceMask

8.927.2.2 uint16_t unpack_loc_SetOperationMode_t::Tlvresult

8.928 unpack_loc_SetServer_Ind_t Struct Reference

Data Fields

- uint32_t serverStatus
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.928.1 Detailed Description

Contain the parameters passed for SetLocSetServerCallback by the device.

Parameters

serverStatus	• Status of the Get Server request. • Valid values – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure – 0xffffffff - Invalid data. • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.928.2 Field Documentation

8.928.2.1 swi_uint256_t unpack_loc_SetServer_Ind_t::ParamPresenceMask

8.928.2.2 uint32_t unpack_loc_SetServer_Ind_t::serverStatus

8.928.2.3 uint16_t unpack_loc_SetServer_Ind_t::TlvResult

8.929 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.929.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

TlvResult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.929.2 Field Documentation

8.929.2.1 swi_uint256_t unpack_loc_SLQSLOCGetBestAvailPos_t::ParamPresenceMask

8.929.2.2 uint16_t unpack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.930 unpack_loc_SLQSLOCGetOpMode_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.930.1 Detailed Description

This structure contains Start LOC unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.930.2 Field Documentation

8.930.2.1 swi_uint256_t unpack_loc_SLQSLOCGetOpMode_t::ParamPresenceMask

8.930.2.2 uint16_t unpack_loc_SLQSLOCGetOpMode_t::Tlvresult

8.931 unpack_loc_Start_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.931.1 Detailed Description

This structure contains Start LOC unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.931.2 Field Documentation

8.931.2.1 swi_uint256_t unpack_loc_Start_t::ParamPresenceMask

8.931.2.2 uint16_t unpack_loc_Start_t::Tlvresult

8.932 unpack_loc_Stop_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.932.1 Detailed Description

This structure contains Stop LOC unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.932.2 Field Documentation

8.932.2.1 swi_uint256_t unpack_loc_Stop_t::ParamPresenceMask

8.932.2.2 uint16_t unpack_loc_Stop_t::Tlvresult

8.933 unpack_nas_GetACCOLC_t Struct Reference

Data Fields

- uint8_t * pAccolc
- swi_uint256_t ParamPresenceMask

8.933.1 Detailed Description

Retrieves information about the access overload class unpack structure

Parameters

pAccolc	• An 8-bit integer representation of the ACCOLC. – Range: 0 to 15 (0x00 to 0x0F). • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.933.2 Field Documentation

8.933.2.1 uint8_t* unpack_nas_GetACCOLC_t::pAccolc

8.933.2.2 swi_uint256_t unpack_nas_GetACCOLC_t::ParamPresenceMask

8.934 unpack_nas_GetANAAAAuthenticationStatus_t Struct Reference

Data Fields

- uint32_t * pAuthStatus
- swi_uint256_t ParamPresenceMask

8.934.1 Detailed Description

AN-AAA authentication status of the device

Parameters

AN-AAA	Authentication Status. • Status of the last AN-AAA authentication request, if any, for the current 1xEV-DO session. – 0 - AAA_STATUS_FAILED - Authentication failed – 1 - AAA_STATUS_SUCCESS - Authentication success – 2 - AAA_STATUS_NO_REQUEST - No authentication requested • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.934.2 Field Documentation

8.934.2.1 swi_uint256_t unpack_nas_GetANAAAAuthenticationStatus_t::ParamPresenceMask

8.934.2.2 uint32_t* unpack_nas_GetANAAAAuthenticationStatus_t::pAuthStatus

8.935 unpack_nas_GetCDMANetworkParameters_t Struct Reference

Data Fields

- uint8_t SCI
- uint8_t SCM
- uint8_t RegHomeSID
- uint8_t RegForeignSID
- uint8_t RegForeignNID
- uint8_t ForceRev0
- uint8_t CustomSCP
- uint32_t Protocol
- uint32_t Broadcast
- uint32_t Application
- uint32_t Roaming
- swi_uint256_t ParamPresenceMask

8.935.1 Detailed Description

Gets the current CDMA network parameters

Parameters

SCI	• Slot cycle index • Bit to check in ParamPresenceMask - **17**
SCM	• Station class mark • Bit to check in ParamPresenceMask - **18**
RegHomeSID	• Register on home SID – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **19**
RegForeignSID	• Register on foreign SID – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **19**
RegForeignNID	• Register on foreign NID – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **19**
ForceRev0	• Force CDMA 1x-EV-DO Rev. 0 mode – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **20**

CustomSCP	• Use a custom config for CDMA 1x-EV-DO SCP – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **21**
Protocol	• Protocol mask for custom SCP config – 0x00000001 - Subtype 2 Physical Layer – 0x00000002 - Enhanced CCMAC – 0x00000004 - Enhanced ACMAC – 0x00000008 - Enhanced FTCMAC – 0x00000010 - Subtype 3 RTCMAC – 0x00000020 - Subsystem 1 RTCMAC – 0x00000040 - Enhanced Idle – 0x00000080 - Generic Multimode Capable Disc Port • Bit to check in ParamPresenceMask - **21**
Broadcast	• Broadcast mask for custom SCP config – 0x00000001 - Generic broadcast enabled • Bit to check in ParamPresenceMask - **21**
Application	• Application mask for custom SCP config – 0x00000001 - SN Multiflow Packet Application – 0x00000002 - SN Enhanced Multiflow Packet Application • Bit to check in ParamPresenceMask - **21**
Roaming	• Roaming preference – 0 - Automatic – 1 - Home Only – 2 - Affiliated Roaming Only – 3 - Home and Affiliated Roaming • Bit to check in ParamPresenceMask - **22**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.935.2 Field Documentation

8.935.2.1 uint32_t unpack_nas_GetCDMANetworkParameters_t::Application

8.935.2.2 uint32_t unpack_nas_GetCDMANetworkParameters_t::Broadcast

8.935.2.3 uint8_t unpack_nas_GetCDMANetworkParameters_t::CustomSCP

8.935.2.4 uint8_t unpack_nas_GetCDMANetworkParameters_t::ForceRev0

8.935.2.5 swi_uint256_t unpack_nas_GetCDMANetworkParameters_t::ParamPresenceMask

8.935.2.6 uint32_t unpack_nas_GetCDMANetworkParameters_t::Protocol

8.935.2.7 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignNID

8.935.2.8 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignSID

8.935.2.9 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegHomeSID

8.935.2.10 uint32_t unpack_nas_GetCDMANetworkParameters_t::Roaming

8.935.2.11 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCI

8.935.2.12 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCM

8.936 unpack_nas_GetHomeNetwork3GPP2_t Struct Reference

Data Fields

- uint16_t * pMCC
- uint16_t * pMNC
- uint8_t nameSize
- uint8_t * pName
- uint16_t * pSID
- uint16_t * pNID
- uint16_t * pNw2MCC
- uint16_t * pNw2MNC
- uint8_t * pNw2DescDisp
- uint8_t * pNw2DescEnc
- uint8_t * pNw2DescLen
- uint8_t * pNw2Name
- swi_uint256_t ParamPresenceMask

8.936.1 Detailed Description

Structure for HomeNetwork3GPP2 unpack.

Parameters

pMCC	• Mobile country code (UMTS only). • Bit to check in ParamPresenceMask - **1**
pMNC	• Mobile network code (UMTS only). • Bit to check in ParamPresenceMask - **1**
nameSize	• Maximum number of characters (including NULL terminator) that network name array can contain (UMTS only). • Bit to check in ParamPresenceMask - **1**

pName	• Network name or description represented as a NULL terminated string (empty string returned when unknown) (UMTS only). • Bit to check in ParamPresenceMask - **1**
pSID	• Home network system ID – 0xFFFF - Unknown. – Only applies to cdma2000 • Bit to check in ParamPresenceMask - **16**
pNID	• Home network ID – 0xFFFF - Unknown. – Only applies to cdma2000 • Bit to check in ParamPresenceMask - **16**
pNw2MCC	• Mobile country code (3GPP2 only). • Range : 0 to 999 • Bit to check in ParamPresenceMask - **17**
pNw2MNC	• Mobile network code (3GPP2 only). • Range : 0 to 999 • Bit to check in ParamPresenceMask - **17**
pNw2DescDisp	• Network Name Display (3GPP2 only). -Valid Value – 0x00 - Do not display – 0x01 - Display – 0xFF - Unknown • Bit to check in ParamPresenceMask - **17**
pNw2DescEnc	• Encoding of the network description (3GPP2 only). • Valid Value – 0x00 - Octet, unspecified – 0x02 - 7-bit ASCII (liteqmi_helper_decode7bitAsciiEncString) – 0x04 - Unicode – 0x09 - GSM 7-bit default • Bit to check in ParamPresenceMask - **17**
pNw2DescLen	• Network Description Length (3GPP2 only). • Bit to check in ParamPresenceMask - **17**
pNw2Name	• Network Name (3GPP2 only). • Bit to check in ParamPresenceMask - **17**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.936.2 Field Documentation

8.936.2.1 uint8_t unpack_nas_GetHomeNetwork3GPP2_t::nameSize

8.936.2.2 swi_uint256_t unpack_nas_GetHomeNetwork3GPP2_t::ParamPresenceMask

8.936.2.3 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMCC

8.936.2.4 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMNC

8.936.2.5 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pName

8.936.2.6 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNID

8.936.2.7 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescDisp

8.936.2.8 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescEnc

8.936.2.9 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescLen

8.936.2.10 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MCC

8.936.2.11 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MNC

8.936.2.12 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2Name

8.936.2.13 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pSID

8.937 unpack_nas_GetHomeNetwork_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- char name [255]
- uint16_t sid
- uint16_t nid
- swi_uint256_t ParamPresenceMask

8.937.1 Detailed Description

This structure contains unpack get get home network parameters.

Parameters

mcc	• Mobile country code (UMTS only). • A 16-bit integer representation of MCC. Range: 0 to 999. • Bit to check in ParamPresenceMask - **1**

mnc	• Mobile network code (UMTS only). • A 16-bit integer representation of MNC. Range: 0 to 999. • Bit to check in ParamPresenceMask - **1**
name	• Network name or description represented as a NULL terminated string (empty string returned when unknown) (UMTS only). • Bit to check in ParamPresenceMask - **1**
sid	• Home network system ID – 0xFFFF - Unknown. – Only applies to cdma2000 • Bit to check in ParamPresenceMask - **16**
nid	• Home network ID – 0xFFFF - Unknown. – Only applies to cdma2000 • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.937.2 Field Documentation

8.937.2.1 uint16_t unpack_nas_GetHomeNetwork_t::mcc

8.937.2.2 uint16_t unpack_nas_GetHomeNetwork_t::mnc

8.937.2.3 char unpack_nas_GetHomeNetwork_t::name[255]

8.937.2.4 uint16_t unpack_nas_GetHomeNetwork_t::nid

8.937.2.5 swi_uint256_t unpack_nas_GetHomeNetwork_t::ParamPresenceMask

8.937.2.6 uint16_t unpack_nas_GetHomeNetwork_t::sid

8.938 unpack_nas_GetNetworkPreference_t Struct Reference

Data Fields

- uint32_t ActiveTechPref
- uint32_t Duration
- uint32_t PersistentTechPref
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.938.1 Detailed Description

Returns the network registration preference. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use unpack_nas_SLQSGetSys-SelectionPref() for new firmware versions and new modules

Parameters

TechnologyPref[-OUT]	• Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved • Bit to check in ParamPresenceMask - **1**
Duration[OUT]	• Duration of active preference – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call • Bit to check in ParamPresenceMask - **1**
Persistent-TechnologyPref[-OUT]	• Bit field representing persistent radio technology preference – Same representation as the pTechnologyPref parameter • Bit to check in ParamPresenceMask - **16**
Tlvresult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.938.2 Field Documentation

8.938.2.1 uint32_t unpack_nas_GetNetworkPreference_t::ActiveTechPref

8.938.2.2 uint32_t unpack_nas_GetNetworkPreference_t::Duration

8.938.2.3 swi_uint256_t unpack_nas_GetNetworkPreference_t::ParamPresenceMask

8.938.2.4 uint32_t unpack_nas_GetNetworkPreference_t::PersistentTechPref

8.938.2.5 uint16_t unpack_nas_GetNetworkPreference_t::TlvResult

8.939 unpack_nas_GetRFInfo_t Struct Reference

Data Fields

- uint8_t instancesSize
- RFBandInfoElements RFBandInfoElements [255]
- swi_uint256_t ParamPresenceMask

8.939.1 Detailed Description

Structur to strore the RFInfoList

Parameters

instanceSize	• Upon input, maximum number of elements that the RF info instances array can contain. • Upon successful output, actual number of elements in RF info instances array.
RFBandInfo-Elements	• RF info instances array – See RFBandInfoElements for more information • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.939.2 Field Documentation

8.939.2.1 uint8_t unpack_nas_GetRFInfo_t::instancesSize

8.939.2.2 swi_uint256_t unpack_nas_GetRFInfo_t::ParamPresenceMask

8.939.2.3 RFBandInfoElements unpack_nas_GetRFInfo_t::RFBandInfoElements[255]

8.940 unpack_nas_GetServingNetwork_t Struct Reference

Data Fields

- uint32_t RegistrationState
- uint32_t CSDomain
- uint32_t PSDomain
- uint32_t RAN
- uint8_t RadioIfacesSize
- uint8_t RadioIfaces [255]
- uint32_t Roaming
- uint16_t MCC
- uint16_t MNC
- uint8_t nameSize
- uint8_t Name [255]
- uint8_t DataCapsLen

- uint8_t DataCaps [255]
- swi_uint256_t ParamPresenceMask

8.940.1 Detailed Description

This structure contains unpack get get serving network parameters.

Parameters

Registration-State	• Registration state: – 0 - Not registered – 1 - Registered – 2 - Searching/Not Registered – 3 - Registration Denied – 4 - Unknown • Bit to check in ParamPresenceMask - **1**
CSDomain	• Circuit switch domain status: – 0 - Unknown/Not Applicable – 1 - Attached – 2 - Detached • Bit to check in ParamPresenceMask - **1**
PSDomain	• Packet switch domain status – 0 - Unknown/Not Applicable – 1 - Attached – 2 - Detached • Bit to check in ParamPresenceMask - **1**
RAN	• Type of radio access network on which mobile is registered: – 0 - Unknown – 1 - cdma2000 network – 2 - UMTS network • Bit to check in ParamPresenceMask - **1**
RadioIfacesSize	• Upon input, maximum number of elements that the radio interface array contain. • Upon successful output, actual number of elements in the radio interface array. • Bit to check in ParamPresenceMask - **1**
RadioIfaces	• An array of Radio Interface Technology – See qaGobiApiTableRadioInterfaces.h for the Radio Interface Technologies • Bit to check in ParamPresenceMask - **1**
Roaming	• Roaming indicator • Bit to check in ParamPresenceMask - **16**

MCC	• Mobile country code • Bit to check in ParamPresenceMask - **18**
MNC	• Mobile network code • Bit to check in ParamPresenceMask - **18**
nameSize	• Maximum number of characters (including NULL terminator) that network name array can contain; applicable only for UMTS networks • Bit to check in ParamPresenceMask - **18**
Name	• Network name or description represented as a NULL terminated string; empty string is returned when unknown; applicable only for UMTS networks • Bit to check in ParamPresenceMask - **18**
DataCapsLen[IN/OUT]	• Upon input, the maximum number of elements the data capabilities array can contain. • Upon output, the actual number of elements in the data capabilities array. • Bit to check in ParamPresenceMask - **17**
DataCaps[OUT]	• Data capabilities array of unsigned long type – 1 - GPRS – 2 - EDGE – 3 - HSDPA – 4 - HSUPA – 5 - WCDMA – 6 - CDMA 1xRTT – 7 - CDMA 1xEV-DO Rev 0 – 8 - CDMA 1xEV-DO Rev. A – 9 - GSM – 10 - EVDO Rev. B – 11 - LTE – 12 - HSDPA Plus – 13 - Dual Carrier HSDPA Plus • Bit to check in ParamPresenceMask - **17**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.940.2 Field Documentation

8.940.2.1 uint32_t unpack_nas_GetServingNetwork_t::CSDomain

8.940.2.2 uint8_t unpack_nas_GetServingNetwork_t::DataCaps[255]

8.940.2.3 uint8_t unpack_nas_GetServingNetwork_t::DataCapsLen

8.940.2.4 uint16_t unpack_nas_GetServingNetwork_t::MCC

8.940.2.5 uint16_t unpack_nas_GetServingNetwork_t::MNC

8.940.2.6 uint8_t unpack_nas_GetServingNetwork_t::Name[255]

8.940.2.7 uint8_t unpack_nas_GetServingNetwork_t::nameSize

8.940.2.8 swi_uint256_t unpack_nas_GetServingNetwork_t::ParamPresenceMask

8.940.2.9 uint32_t unpack_nas_GetServingNetwork_t::PSDomain

8.940.2.10 uint8_t unpack_nas_GetServingNetwork_t::RadioIfaces[255]

8.940.2.11 uint8_t unpack_nas_GetServingNetwork_t::RadioIfacesSize

8.940.2.12 uint32_t unpack_nas_GetServingNetwork_t::RAN

8.940.2.13 uint32_t unpack_nas_GetServingNetwork_t::RegistrationState

8.940.2.14 uint32_t unpack_nas_GetServingNetwork_t::Roaming

8.941 unpack_nas_GetServingNetworkCapabilities_t Struct Reference

Data Fields

- uint8_t DataCapsLen
- uint8_t DataCaps [255]
- swi_uint256_t ParamPresenceMask

8.941.1 Detailed Description

This structure contains unpack get serving network capabilities parameters.

Parameters

DataCapsLen	• Upon input, the maximum number of elements the data capabilities array can contain. • Upon output, the actual number of elements in the data capabilities array. • Bit to check in ParamPresenceMask - **17**
DataCaps[OUT]	• Data capabilities array of unsigned long type – 1 - GPRS – 2 - EDGE – 3 - HSDPA – 4 - HSUPA – 5 - WCDMA – 6 - CDMA 1xRTT – 7 - CDMA 1xEV-DO Rev 0 – 8 - CDMA 1xEV-DO Rev. A – 9 - GSM – 10 - EVDO Rev. B – 11 - LTE – 12 - HSDPA Plus – 13 - Dual Carrier HSDPA Plus • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.941.2 Field Documentation

8.941.2.1 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCaps[255]

8.941.2.2 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCapsLen

8.941.2.3 swi_uint256_t unpack_nas_GetServingNetworkCapabilities_t::ParamPresenceMask

8.942 unpack_nas_GetSignalStrengths_t Struct Reference

Data Fields

- uint32_t len
- signed char rssi [8]
- uint32_t radio [8]
- swi_uint256_t ParamPresenceMask

8.942.1 Detailed Description

This structure contains unpack get signal strengths parameters.

Parameters

len	• number of rssi and radio items following
rssi	• signal strength array • Bit to check in ParamPresenceMask - **16**
radio	• radio interface array • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.942.2 Field Documentation

8.942.2.1 uint32_t unpack_nas_GetSignalStrengths_t::len

8.942.2.2 swi_uint256_t unpack_nas_GetSignalStrengths_t::ParamPresenceMask

8.942.2.3 uint32_t unpack_nas_GetSignalStrengths_t::radio[8]

8.942.2.4 signed char unpack_nas_GetSignalStrengths_t::rssi[8]

8.943 unpack_nas_PerformNetworkScan_t Struct Reference

Data Fields

- uint8_t * p3GppNetworkInstanceSize
- nas_QmiNas3GppNetworkInfo * p3GppNetworkInfoInstances
- uint8_t * pRATInstanceSize
- nas_QmiNas3GppNetworkRAT * pRATINstance
- uint8_t * pPCSInstanceSize
- nas_QmisNasPcsDigit * pPCSInstance
- uint32_t * pScanResult
- nas_QmisNasSlqsNasPCIInfo * pPCIInfo
- nas_lteOpModeTlv * pLteOpModeTlv
- swi_uint256_t ParamPresenceMask

8.943.1 Detailed Description

Contain the network scan information.

Parameters

p3GppNetwork-InstanceSize	• Upon input, maximum number of elements that the network info instance array can contain. • Upon successful output, the actual number of elements in the network info instance array. • Bit to check in ParamPresenceMask - **16**

p3GppNetwork-InfoInstances	• Network info instance array – See nas_QmiNas3GppNetworkInfo for more information • Bit to check in ParamPresenceMask - **16**
pRATInstance-Size	• Upon input, maximum number of elements that the RAT info instance array can contain. • Upon successful output, the actual number of elements in the RAT info instance array. • Bit to check in ParamPresenceMask - **17**
pRATINstance	• RAT info instance array – See nas_QmiNas3GppNetworkRAT for more information • Bit to check in ParamPresenceMask - **17**
pPCSInstance-Size	• Upon input, maximum number of elements that the PCS Digit info instance array can contain. • Upon successful output, the actual number of elements in the PCS Digit info instance array. • Bit to check in ParamPresenceMask - **18**
pPCSInstance	• PCS Digit info instance array – See nas_QmisNasPcsDigit for more information • Bit to check in ParamPresenceMask - **18**
pScanResult	• status of network scan • 0x00 - scan successful • 0x01 - scan was aborted • 0x02 - scan did not complete due to a radio link failure recovery in progress • Bit to check in ParamPresenceMask - **19**
pPCIInfo	• PCI Information – See nas_QmisNasSlqsNasPCIInfo for more information • Bit to check in ParamPresenceMask - **23**
pLteOpMode-Tlv[OUT]	• LTE Operational Mode. • See nas_lteOpModeTlv for more information. • Bit to check in ParamPresenceMask - **26**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.943.2 Field Documentation

8.943.2.1 nas_QmiNas3GppNetworkInfo* unpack_nas_PerformNetworkScan_t::p3GppNetworkInfoInstances

8.943.2.2 uint8_t* unpack_nas_PerformNetworkScan_t::p3GppNetworkInstanceSize

8.943.2.3 swi_uint256_t unpack_nas_PerformNetworkScan_t::ParamPresenceMask

8.943.2.4 nas_lteOpModeTlv* unpack_nas_PerformNetworkScan_t::pLteOpModeTlv

8.943.2.5 nas_QmisNasSlqsNasPCIInfo* unpack_nas_PerformNetworkScan_t::pPCIInfo

8.943.2.6 nas_QmisNasPcsDigit* unpack_nas_PerformNetworkScan_t::pPCSInstance

8.943.2.7 uint8_t* unpack_nas_PerformNetworkScan_t::pPCSInstanceSize

8.943.2.8 nas_QmiNas3GppNetworkRAT* unpack_nas_PerformNetworkScan_t::pRATINstance

8.943.2.9 uint8_t* unpack_nas_PerformNetworkScan_t::pRATInstanceSize

8.943.2.10 uint32_t* unpack_nas_PerformNetworkScan_t::pScanResult

8.944 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference

Data Fields

- uint8_t dataCapsSize
- uint8_t dataCaps [255]
- swi_uint256_t ParamPresenceMask

8.944.1 Detailed Description

This structure contains unpack set data capabilities callback indication parameters.

Parameters

dataCapsSize	• Number of Data Capabilities • Bit to check in ParamPresenceMask - **17**
dataCaps	• Data Capabilities • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.944.2 Field Documentation

8.944.2.1 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCaps[255]

8.944.2.2 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCapsSize

8.944.2.3 swi_uint256_t unpack_nas_SetDataCapabilitiesCallback_ind_t::ParamPresenceMask

8.945 unpack_nas_SetEventReportInd_t Struct Reference

Data Fields

- nas_SignalStrengthTlv SSTlv
- nas_RFInfoTlv RFTlv
- nas_RejectReasonTlv RRTlv
- nas_SLQSSignalStrengthsTlv SLQSSSTlv
- swi_uint256_t ParamPresenceMask

8.945.1 Detailed Description

This structure contains unpack set event report indication parameters.

Parameters

SSTlv	• signal strength tlv • Bit to check in ParamPresenceMask - **16**
RFTlv	• RF tlv • Bit to check in ParamPresenceMask - **17**
RRTlv	• RR tlv • Bit to check in ParamPresenceMask - **18**
SLQSSSTlv	• signal strength complete info tlv • Bit to check in ParamPresenceMask - **19** • Bit to check in ParamPresenceMask - **20** • Bit to check in ParamPresenceMask - **21** • Bit to check in ParamPresenceMask - **22** • Bit to check in ParamPresenceMask - **23** • Bit to check in ParamPresenceMask - **24** • Bit to check in ParamPresenceMask - **25** • Bit to check in ParamPresenceMask - **26**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.945.2 Field Documentation

8.945.2.1 swi_uint256_t unpack_nas_SetEventReportInd_t::ParamPresenceMask

8.945.2.2 nas_RFInfoTlv unpack_nas_SetEventReportInd_t::RFTlv

8.945.2.3 nas_RejectReasonTlv unpack_nas_SetEventReportInd_t::RRTlv

8.945.2.4 nas_SLQSSignalStrengthsTlv unpack_nas_SetEventReportInd_t::SLQSSSTlv

8.945.2.5 nas_SignalStrengthTlv unpack_nas_SetEventReportInd_t::SSTlv

8.946 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference

Data Fields

- nas_PhyCaAggScellIndType sPhyCaAggScellIndType
- nas_PhyCaAggScellDlBw sPhyCaAggScellDlBw
- nas_PhyCaAggScellInfo sPhyCaAggScellInfo
- nas_PhyCaAggPcellInfo sPhyCaAggPcellInfo
- nas_PhyCaAggScellIndex sPhyCaAggScellIndex
- nas_PhyCaAggDlBW sPhyCaAggDlBW
- nas_NumScellsConfig sNumScellsConfig
- swi_uint256_t ParamPresenceMask

8.946.1 Detailed Description

Structure for storing the LTE PHY CA indication parameters.

Parameters

pPhyCaAggScellIndType	• See nas_PhyCaAggScellIndType for more information. • Bit to check in ParamPresenceMask - **1**
sPhyCaAggScellDlBw	• See nas_PhyCaAggScellDlBw for more information. • Bit to check in ParamPresenceMask - **16**
sPhyCaAggScellInfo	• See nas_PhyCaAggScellInfo for more information. • Bit to check in ParamPresenceMask - **17**
sPhyCaAggPcellInfo	• See nas_PhyCaAggPcellInfo for more information. • Bit to check in ParamPresenceMask - **18**
sPhyCaAggScellIndex	• See nas_PhyCaAggScellIndex for more information. • Bit to check in ParamPresenceMask - **19**
sPhyCaAggDlBW	• See nas_PhyCaAggDlBW for more information. • Bit to check in ParamPresenceMask - **20**
sNumScellsConfig	• See nas_NumScellsConfig for more information. • Bit to check in ParamPresenceMask - **21**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.946.2 Field Documentation

8.946.2.1 swi_uint256_t unpack_nas_SetNasLTECphyCaIndCallback_ind_t::ParamPresenceMask

8.946.2.2 nas_NumScellsConfig unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sNumScellsConfig

8.946.2.3 nas_PhyCaAggDlBW unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggDlBW

8.946.2.4 nas_PhyCaAggPcellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggPcellInfo

8.946.2.5 nas_PhyCaAggScellDlBw unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellDlBw

8.946.2.6 nas_PhyCaAggScellIndex unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndex

8.946.2.7 nas_PhyCaAggScellIndType unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndType

8.946.2.8 nas_PhyCaAggScellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellInfo

8.947 unpack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.947.1 Detailed Description

This structure contains unpack set network preference parameters.

Parameters

Tlvresult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.947.2 Field Documentation

8.947.2.1 swi_uint256_t unpack_nas_SetNetworkPreference_t::ParamPresenceMask

8.947.2.2 uint16_t unpack_nas_SetNetworkPreference_t::Tlvresult

8.948 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference

Data Fields

- uint8_t roaming
- swi_uint256_t ParamPresenceMask

8.948.1 Detailed Description

This structure contains unpack set roaming indicator callback indication parameters.

Parameters

roaming	• Roaming Indication – 0 - Roaming – 1 - Home – 2 - Roaming partner – >2 - Operator defined values • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.948.2 Field Documentation

8.948.2.1 swi_uint256_t unpack_nas_SetRoamingIndicatorCallback_ind_t::ParamPresenceMask

8.948.2.2 uint8_t unpack_nas_SetRoamingIndicatorCallback_ind_t::roaming

8.949 unpack_nas_SetServingSystemCallback_ind_t Struct Reference

Data Fields

- NASServingSystemInfo SSInfo
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.949.1 Detailed Description

This structure contains unpack set serving system callback indication parameters.

Parameters

SSInfo	• Serving system parameters information – See NASServingSystemInfo for more details • Bit to check in ParamPresenceMask - **1**
Tlvresult	• unpack result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.949.2 Field Documentation

8.949.2.1 swi_uint256_t unpack_nas_SetServingSystemCallback_ind_t::ParamPresenceMask

8.949.2.2 NASServingSystemInfo unpack_nas_SetServingSystemCallback_ind_t::SSInfo

8.949.2.3 uint16_t unpack_nas_SetServingSystemCallback_ind_t::TlvResult

8.950 unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t Struct Reference

Data Fields

- nas_DDTMSettings ddtm_Settings
- swi_uint256_t ParamPresenceMask

8.950.1 Detailed Description

This structure contains the DDTM Settings Indication parameters.

Parameters

ddtm_Settings	• See nas_DDTMSettings for more information. • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.950.2 Field Documentation

8.950.2.1 nas_DDTMSettings unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t::ddtm_Settings

8.950.2.2 swi_uint256_t unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t::ParamPresenceMask

8.951 unpack_nas_SLQSGetErrorRate_t Struct Reference

Data Fields

- uint16_t * pCDMAFrameErrRate
- uint16_t * pHDRPackErrRate
- uint8_t * pGSMBER
- uint8_t * pWCDMABER
- swi_uint256_t ParamPresenceMask

8.951.1 Detailed Description

Structure for Error Rate unpack parameters.

Parameters

pCDMAFrame-ErrRate	• CDMA Frame Error Rate • Valid error rate values between 1 and 10000 are returned to indicate the percentage, e.g., a value of 300 means the error rate is 3%. • A value of 0xFFFF indicates that the error rate is unknown/unavailable. • Bit to check in ParamPresenceMask - **16**

pHDRPackErr-Rate	• HDR Packet Error Rate • Valid error rate values between 1 and 10000 are returned to indicate the percentage, e.g., a value of 300 means the error rate is 3%. • A value of 0xFFFF indicates that the error rate is unknown/unavailable. • Bit to check in ParamPresenceMask - **17**
pGSMBER	• GSM Bit Error Rate • Valid error rate values between 1 and 100 are returned to indicate the percentage value. • A 0% block error rate (BLER) indicates No Data. • Bit to check in ParamPresenceMask - **18**
pWCDMABER	• WCDMA Block Error Rate • Valid error rate values between 1 and 100 are returned to indicate the percentage value. • A value of 0xFF indicates that the error rate is unknown/unavailable. • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.951.2 Field Documentation

8.951.2.1 swi_uint256_t unpack_nas_SLQSGetErrorRate_t::ParamPresenceMask

8.951.2.2 uint16_t* unpack_nas_SLQSGetErrorRate_t::pCDMAFrameErrRate

8.951.2.3 uint8_t* unpack_nas_SLQSGetErrorRate_t::pGSMBER

8.951.2.4 uint16_t* unpack_nas_SLQSGetErrorRate_t::pHDRPackErrRate

8.951.2.5 uint8_t* unpack_nas_SLQSGetErrorRate_t::pWCDMABER

8.952 unpack_nas_SLQSGetHomeNetwork_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- char name [255]
- uint16_t sid
- uint16_t nid
- nas_homeNwMNC3GppTlv * pHomeNwMNC3Gpp
- nas_nwNameSrc3GppTlv * pNwNameSrc3Gpp
- swi_uint256_t ParamPresenceMask

8.952.1 Detailed Description

This structure contains unpack get get home network parameters.

Parameters

mcc	• Mobile country code (UMTS only). • A 16-bit integer representation of MCC. Range: 0 to 999. • Bit to check in ParamPresenceMask - **1**
mnc	• Mobile network code (UMTS only). • A 16-bit integer representation of MNC. Range: 0 to 999. • Bit to check in ParamPresenceMask - **1**
name	• Network name or description represented as a NULL terminated string (empty string returned when unknown) (UMTS only). • Bit to check in ParamPresenceMask - **1**
sid	• Home network system ID – 0xFFFF - Unknown. – Only applies to cdma2000 • Bit to check in ParamPresenceMask - **16**
nid	• Home network ID – 0xFFFF - Unknown. – Only applies to cdma2000 • Bit to check in ParamPresenceMask - **16**
pHomeNwMNC3Gpp[OUT]	• 3GPP Home Network MNC • See nas_homeNwMNC3GppTlv for more information • Bit to check in ParamPresenceMask - **18**
pNwNameSrc3Gpp[OUT]	• 3GPP Network Name Source • See nas_nwNameSrc3GppTlv for more information • Bit to check in ParamPresenceMask - **19**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.952.2 Field Documentation

8.952.2.1 uint16_t unpack_nas_SLQSGetHomeNetwork_t::mcc

8.952.2.2 uint16_t unpack_nas_SLQSGetHomeNetwork_t::mnc

8.952.2.3 char unpack_nas_SLQSGetHomeNetwork_t::name[255]

8.952.2.4 uint16_t unpack_nas_SLQSGetHomeNetwork_t::nid

8.952.2.5 swi_uint256_t unpack_nas_SLQSGetHomeNetwork_t::ParamPresenceMask

8.952.2.6 nas_homeNwMNC3GppTlv* unpack_nas_SLQSGetHomeNetwork_t::pHomeNwMNC3Gpp

8.952.2.7 nas_nwNameSrc3GppTlv* unpack_nas_SLQSGetHomeNetwork_t::pNwNameSrc3Gpp

8.952.2.8 uint16_t unpack_nas_SLQSGetHomeNetwork_t::sid

8.953 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference

Data Fields

- NasGetLTECphyCaInfo LTECphyCaInfo
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.953.1 Detailed Description

This structure contains unpack get carrier aggregation event information parameters.

Parameters

LTECphyCa	• Carrier aggregation event information – See NasGetLTECphyCaInfo for more details
Tlvresult	• unpack result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.953.2 Field Documentation

8.953.2.1 NasGetLTECphyCaInfo unpack_nas_SlqsGetLTECphyCAInfo_t::LTECphyCaInfo

8.953.2.2 swi_uint256_t unpack_nas_SlqsGetLTECphyCAInfo_t::ParamPresenceMask

8.953.2.3 uint16_t unpack_nas_SlqsGetLTECphyCAInfo_t::Tlvresult

8.954 unpack_nas_SLQSGetNetworkTime_t Struct Reference

Data Fields

- nas_timeInfo * p3GPP2TimeInfo
- nas_timeInfo * p3GPPTimeInfo
- swi_uint256_t ParamPresenceMask

8.954.1 Detailed Description

This structure contains information about the GetNetworkTime response parameters.

Parameters

p3GPP2TimeInfo	[Optional] • See nas_timeInfo for more information • Bit to check in ParamPresenceMask - **16**
p3GPPTimeInfo	[Optional] • See nas_timeInfo for more information • Bit to check in ParamPresenceMask - **17**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.954.2 Field Documentation

8.954.2.1 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPP2TimeInfo

8.954.2.2 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPPTimeInfo

8.954.2.3 swi_uint256_t unpack_nas_SLQSGetNetworkTime_t::ParamPresenceMask

8.955 unpack_nas_SLQSGetOperatorNameData_t Struct Reference

Data Fields

- nas_serviceProviderName * pSrvcProviderName
- nas_operatorPLMNList * pOperatorPLMNList
- nas_PLMNNetworkName * pPLMNNetworkName
- nas_operatorNameString * pOperatorNameString
- nas_PLMNNetworkNameData * pNITZInformation
- swi_uint256_t ParamPresenceMask

8.955.1 Detailed Description

Structure for Operator Name unpack.

Parameters

pSrvcProviderName	• Refer nas_serviceProviderName for details (Optional). • Can provide NULL if this parameter is not required. • Bit to check in ParamPresenceMask - **16**
pOperatorPLMNList	• Refer nas_operatorPLMNList for details (Optional). • Can provide NULL if this parameter is not required. • Bit to check in ParamPresenceMask - **17**

pPLMNNetwork-Name	• Refer nas_PLMNNetworkName for details (Optional). • Can provide NULL if this parameter is not required. • Bit to check in ParamPresenceMask - **18**
pOperatorName-String	• Refer nas_operatorNameString for details (Optional). • Can provide NULL if this parameter is not required. • Bit to check in ParamPresenceMask - **19**
pNITZ-Information	• Refer nas_PLMNNetworkNameData for details (Optional). • Can provide NULL if this parameter is not required. • Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.955.2 Field Documentation

8.955.2.1 swi_uint256_t unpack_nas_SLQSGetOperatorNameData_t::ParamPresenceMask

8.955.2.2 nas_PLMNNetworkNameData* unpack_nas_SLQSGetOperatorNameData_t::pNITZInformation

8.955.2.3 nas_operatorNameString* unpack_nas_SLQSGetOperatorNameData_t::pOperatorNameString

8.955.2.4 nas_operatorPLMNList* unpack_nas_SLQSGetOperatorNameData_t::pOperatorPLMNList

8.955.2.5 nas_PLMNNetworkName* unpack_nas_SLQSGetOperatorNameData_t::pPLMNNetworkName

8.955.2.6 nas_serviceProviderName* unpack_nas_SLQSGetOperatorNameData_t::pSrvcProviderName

8.956 unpack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint8_t spnEncoding
- uint8_t spnLength
- char spn [255]
- uint8_t shortNameEn
- uint8_t shortNameCI
- uint8_t shortNameSB
- char shortNameLen
- uint8_t shortName [255]
- uint8_t longNameEn
- uint8_t longNameCI
- uint8_t longNameSB
- uint8_t longNameLen
- char longName [255]
- swi_uint256_t ParamPresenceMask

8.956.1 Detailed Description

This structure contains unpack get operator name for specificed network parameters.

Parameters

spnEncoding	• Coding scheme used for service provider name. This value is ignored if spn_len is zero Values: – 0x00 - SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 0 – 0x01 - UCS2 (16 bit, little-endian) 3GPP TS 23.038 – Note: This value is ignored if spnLength is zero. • Bit to check in ParamPresenceMask - **16**
spnLength	• Length of SPN which follows • Note: This SPN value is ignored if spnLength is zero and spnEncoding is 0xff. • Bit to check in ParamPresenceMask - **16**
spn	• Service Provider name string • Bit to check in ParamPresenceMask - **16**
shortNameEn	• Coding scheme used for PLMN short name. This value is ignored if PLMN short name length is zero Values: – 0x00 - SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 0 – 0x01 - UCS2 (16 bit, little-endian) 3GPP TS 23.038 – Note: This value is ignored if shortNameLen is zero. • Bit to check in ParamPresenceMask - **16**
shortNameCI	• Indicates whether the country initials are to be added to the shortName. This value is ignored if shortNameLen is zero. Values: – 0x00 - Do not add the letters for the countrys initials to the name – 0x01 - Add the countrys initials and a text string to the name – 0xFF - Not specified – Note: This value is ignored if shortNameLen is zero. • Bit to check in ParamPresenceMask - **16**
shortNameSB	• PLMN short name spare bits. This value is ignored if shortNameLen is zero. Values: – 0x01 - Bit 8 is spare and set to 0 in octet – 0x02 - Bits 7 and 8 are spare and set to 0 in octet n – 0x03 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n – 0x04 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n – 0x05 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n – 0x06 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n – 0x07 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n – 0x00 - Carries no information about the number of spare bits in octet n – Note: This value is ignored if shortNameLen is zero. • Bit to check in ParamPresenceMask - **16**

shortNameLen	• Length of shortName which follows • Note: This shortName value is ignored if shortNameLen is zero and shortNameEn is 0xff. • Bit to check in ParamPresenceMask - **16**
shortName	• PLMN short name • Bit to check in ParamPresenceMask - **16**
longNameEn	• Coding scheme used for PLMN long name. This value is ignored if PLMN long name length is zero Values: – 0x00 - SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 0 – 0x01 - UCS2 (16 bit, little-endian) 3GPP TS 23.038 – Note: This value is ignored if shortNameLen is zero. • Bit to check in ParamPresenceMask - **16**
longNameCI	• Indicates whether the country initials are to be added to the longName. This value is ignored if longNameLen is zero. Values: – 0x00 - Do not add the letters for the countrys initials to the name – 0x01 - Add the countrys initials and a text string to the name – 0xFF - Not specified – Note: This value is ignored if shortNameLen is zero. • Bit to check in ParamPresenceMask - **16**
longNameSB	• PLMN long name spare bits. This value is ignored if longNameLen is zero. Values: – 0x01 - Bit 8 is spare and set to 0 in octet – 0x02 - Bits 7 and 8 are spare and set to 0 in octet n – 0x03 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n – 0x04 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n – 0x05 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n – 0x06 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n – 0x07 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n – 0x00 - Carries no information about the number of spare bits in octet n – Note: This value is ignored if shortNameLen is zero. • Bit to check in ParamPresenceMask - **16**
longNameLen	• Length of longName which follows • Note: This longName value is ignored if longNameLen is zero and longNameEn is 0xff. • Bit to check in ParamPresenceMask - **16**
longName	• PLMN long name • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.956.2 Field Documentation

8.956.2.1 char unpack_nas_SLQSGetPLMNName_t::longName[255]

8.956.2.2 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameCI

8.956.2.3 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameEn

8.956.2.4 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameLen

8.956.2.5 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameSB

8.956.2.6 swi_uint256_t unpack_nas_SLQSGetPLMNName_t::ParamPresenceMask

8.956.2.7 uint8_t unpack_nas_SLQSGetPLMNName_t::shortName[255]

8.956.2.8 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameCI

8.956.2.9 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameEn

8.956.2.10 char unpack_nas_SLQSGetPLMNName_t::shortNameLen

8.956.2.11 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameSB

8.956.2.12 char unpack_nas_SLQSGetPLMNName_t::spn[255]

8.956.2.13 uint8_t unpack_nas_SLQSGetPLMNName_t::spnEncoding

8.956.2.14 uint8_t unpack_nas_SLQSGetPLMNName_t::spnLength

8.957 unpack_nas_SLQSGetServingSystem_t Struct Reference

Data Fields

- nas_servSystem ServingSystem
- uint8_t RoamIndicatorVal
- nas_dataSrvCapabilities DataSrvCapabilities
- nas_currentPLMN CurrentPLMN
- uint16_t SystemID
- uint16_t NetworkID
- uint16_t BasestationID
- uint32_t BasestationLatitude
- uint32_t BasestationLongitude
- nas_roamIndList RoamingIndicatorList
- uint8_t DefaultRoamInd
- nas_qaQmi3Gpp2TimeZone Gpp2TimeZone
- uint8_t CDMA_P_Rev
- uint8_t GppTimeZone
- uint8_t GppNetworkDSTAdjustment
- uint16_t Lac
- uint32_t CellID
- uint8_t ConcSvcInfo
- uint8_t PRLInd
- uint8_t DTMInd
- nas_detailSvcInfo DetailedSvcInfo

- nas_CDMASysInfoExt CDMASystemInfoExt
- uint8_t HdrPersonality
- uint16_t TrackAreaCode
- nas_callBarStatus CallBarStatus
- swi_uint256_t ParamPresenceMask

8.957.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

ServingSystem	• Serving System • See nas_servSystem for more information • Bit to check in ParamPresenceMask - **1**
RoamIndicator-Val	• Optional parameter indicating Roaming Indicator value • Values: – 0x00 - Roaming – 0x01 - Home – 0x02 - Flashing – 0x03 and above - Operator defined values • Bit to check in ParamPresenceMask - **16**
DataSrv-Capabilities	• Optional parameter indicating Data services capability • See nas_dataSrvCapabilities for more information • Bit to check in ParamPresenceMask - **17**
CurrentPLMN	• Optional parameter indicating Current PLMN • See nas_currentPLMN for more information • Bit to check in ParamPresenceMask - **18**
SystemID	• Optional parameter indicating System ID • Bit to check in ParamPresenceMask - **19**
NetworkID	• Optional parameter indicating Network ID • Bit to check in ParamPresenceMask - **19**
BaseStationID	• Optional parameter indicating Base Station Identification Number • Bit to check in ParamPresenceMask - **20**

BaseStation-Latitude	• Optional parameter indicating Base station latitude in units of 0.25 sec,expressed as a two's complement signed number with positive numbers signifying North latitude • Bit to check in ParamPresenceMask - **20**
Basestation-Longitude	• Optional parameter indicating Base station longitude in units of 0.25 sec, expressed as a Two's complement signed number with positive numbers signifying East longitude • Bit to check in ParamPresenceMask - **20**
Roaming-IndicatorList	• Optional parameter indicating Roaming Indicator List • See nas_roamIndList for more information • Bit to check in ParamPresenceMask - **21**
DefaultRoamInd	• Optional parameter indicating Default Roaming Indicator • Values: – 0x00 - Roaming – 0x01 - Home • Bit to check in ParamPresenceMask - **22**
Gpp2TimeZone	• Optional parameter indicating 3GPP2 Time Zone • See nas_qaQmi3Gpp2TimeZone for more information • Bit to check in ParamPresenceMask - **23**
CDMA_P_Rev	• Optional parameter indicating CDMA P_Rev in use • Bit to check in ParamPresenceMask - **24**
GppTimeZone	• Optional parameter indicating Offset from Universal time, i.e., difference between local time and Universal time, in increments of 15 min. (signed value). • Bit to check in ParamPresenceMask - **26**
GppNetworkDS-TAdjustment	• Optional parameter indicating 3GPP network daylight saving adjustment • Values: – 0x00 - No adjustment for Daylight Saving Time – 0x01 - 1 hr adjustment for Daylight Saving Time – 0x02 - 2 hr adjustment for Daylight Saving Time • Bit to check in ParamPresenceMask - **27**
Lac	• Optional parameter indicating 3GPP Location Area Code • Bit to check in ParamPresenceMask - **28**
CellID	• Optional parameter indicating 3GPP Cell ID • Bit to check in ParamPresenceMask - **29**

ConcSvcInfo	• Optional parameter indicating 3GPP2 concurrent service Info • Values: – 0x00 - Concurrent service not available – 0x01 - Concurrent service available • Bit to check in ParamPresenceMask - **30**
PRLInd	• Optional parameter indicating 3GPP2 PRL Indicator • Values: – 0x00 - System not in PRL – 0x01 - System is in PRL • Bit to check in ParamPresenceMask - **31**
DTMInd	• Optional parameter indicating Dual Transfer Mode Indication(GSM Only) • Values: – 0x00 - DTM not supported – 0x01 - DTM supported • Bit to check in ParamPresenceMask - **32**
DetailedSvcInfo	• Optional parameter indicating Detailed service information • See nas_detailSvcInfo for more information • Bit to check in ParamPresenceMask - **33**
CDMASystem-InfoExt	• Optional parameter indicating CDMA System Info Ext • See nas_CDMASysInfoExt for more information • Bit to check in ParamPresenceMask - **34**
HdrPersonality	• Optional parameter indicating HDR Personality Information • Values: – 0x00 - Unknown – 0x01 - HRPD – 0x02 - eHRPD • Bit to check in ParamPresenceMask - **35**
TrackAreaCode	• Optional parameter indicating Tracking area code information for LTE • Bit to check in ParamPresenceMask - **36**
CallBarStatus	• Optional parameter indicating Call Barring Status • See nas_callBarStatus for more information • Bit to check in ParamPresenceMask - **37**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.957.2 Field Documentation

8.957.2.1 uint16_t unpack_nas_SLQSGetServingSystem_t::BasestationID

8.957.2.2 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLatitude

8.957.2.3 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLongitude

8.957.2.4 **nas_callBarStatus** unpack_nas_SLQSGetServingSystem_t::CallBarStatus

8.957.2.5 uint8_t unpack_nas_SLQSGetServingSystem_t::CDMA_P_Rev

8.957.2.6 **nas_CDMASysInfoExt** unpack_nas_SLQSGetServingSystem_t::CDMASystemInfoExt

8.957.2.7 uint32_t unpack_nas_SLQSGetServingSystem_t::CellID

8.957.2.8 uint8_t unpack_nas_SLQSGetServingSystem_t::ConcSvcInfo

8.957.2.9 **nas_currentPLMN** unpack_nas_SLQSGetServingSystem_t::CurrentPLMN

8.957.2.10 **nas_dataSrvCapabilities** unpack_nas_SLQSGetServingSystem_t::DataSrvCapabilities

8.957.2.11 uint8_t unpack_nas_SLQSGetServingSystem_t::DefaultRoamInd

8.957.2.12 **nas_detailSvcInfo** unpack_nas_SLQSGetServingSystem_t::DetailedSvcInfo

8.957.2.13 uint8_t unpack_nas_SLQSGetServingSystem_t::DTMInd

8.957.2.14 **nas_qaQmi3Gpp2TimeZone** unpack_nas_SLQSGetServingSystem_t::Gpp2TimeZone

8.957.2.15 uint8_t unpack_nas_SLQSGetServingSystem_t::GppNetworkDSTAdjustment

8.957.2.16 uint8_t unpack_nas_SLQSGetServingSystem_t::GppTimeZone

8.957.2.17 uint8_t unpack_nas_SLQSGetServingSystem_t::HdrPersonality

8.957.2.18 uint16_t unpack_nas_SLQSGetServingSystem_t::Lac

8.957.2.19 uint16_t unpack_nas_SLQSGetServingSystem_t::NetworkID

8.957.2.20 **swi_uint256_t** unpack_nas_SLQSGetServingSystem_t::ParamPresenceMask

8.957.2.21 uint8_t unpack_nas_SLQSGetServingSystem_t::PRLInd

8.957.2.22 uint8_t unpack_nas_SLQSGetServingSystem_t::RoamIndicatorVal

8.957.2.23 **nas_roamIndList** unpack_nas_SLQSGetServingSystem_t::RoamingIndicatorList

8.957.2.24 **nas_servSystem** unpack_nas_SLQSGetServingSystem_t::ServingSystem

8.957.2.25 uint16_t unpack_nas_SLQSGetServingSystem_t::SystemID

8.957.2.26 uint16_t unpack_nas_SLQSGetServingSystem_t::TrackAreaCode

8.958 unpack_nas_SLQSGetServingSystemV2_t Struct Reference

Data Fields

- nas_servSystem ServingSystem
- uint8_t RoamIndicatorVal
- nas_dataSrvCapabilities DataSrvCapabilities
- nas_currentPLMN CurrentPLMN
- uint16_t SystemID
- uint16_t NetworkID
- uint16_t BasestationID
- uint32_t BasestationLatitude
- uint32_t BasestationLongitude
- nas_roamIndList RoamingIndicatorList
- uint8_t DefaultRoamInd
- nas_qaQmi3Gpp2TimeZone Gpp2TimeZone
- uint8_t CDMA_P_Rev
- uint8_t GppTimeZone
- uint8_t GppNetworkDSTAdjustment
- uint16_t Lac
- uint32_t CellID
- uint8_t ConcSvcInfo
- uint8_t PRLInd
- uint8_t DTMInd
- nas_detailSvcInfo DetailedSvcInfo
- nas_CDMASysInfoExt CDMASystemInfoExt
- uint8_t HdrPersonality
- uint16_t TrackAreaCode
- nas_callBarStatus CallBarStatus
- nas_MNCPCSDigitStatus MNCPCSDigitStatTlv
- swi_uint256_t ParamPresenceMask

8.958.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

ServingSystem	• Serving System • See nas_servSystem for more information • Bit to check in ParamPresenceMask - **1**
RoamIndicator-Val	• Optional parameter indicating Roaming Indicator value • Values: – 0x00 - Roaming – 0x01 - Home – 0x02 - Flashing – 0x03 and above - Operator defined values • Bit to check in ParamPresenceMask - **16**

DataSrv-Capabilities	• Optional parameter indicating Data services capability • See nas_dataSrvCapabilities for more information • Bit to check in ParamPresenceMask - **17**
CurrentPLMN	• Optional parameter indicating Current PLMN • See nas_currentPLMN for more information • Bit to check in ParamPresenceMask - **18**
SystemID	• Optional parameter indicating System ID • Bit to check in ParamPresenceMask - **19**
NetworkID	• Optional parameter indicating Network ID • Bit to check in ParamPresenceMask - **19**
BaseStationID	• Optional parameter indicating Base Station Identification Number • Bit to check in ParamPresenceMask - **20**
BaseStation-Latitude	• Optional parameter indicating Base station latitude in units of 0.25 sec,expressed as a two's complement signed number with positive numbers signifying North latitude • Bit to check in ParamPresenceMask - **20**
Basestation-Longitude	• Optional parameter indicating Base station longitude in units of 0.25 sec, expressed as a Two's complement signed number with positive numbers signifying East longitude • Bit to check in ParamPresenceMask - **20**
Roaming-IndicatorList	• Optional parameter indicating Roaming Indicator List • See nas_roamIndList for more information • Bit to check in ParamPresenceMask - **21**
DefaultRoamInd	• Optional parameter indicating Default Roaming Indicator • Values: – 0x00 - Roaming – 0x01 - Home • Bit to check in ParamPresenceMask - **22**
Gpp2TimeZone	• Optional parameter indicating 3GPP2 Time Zone • See nas_qaQmi3Gpp2TimeZone for more information • Bit to check in ParamPresenceMask - **23**

CDMA_P_Rev	• Optional parameter indicating CDMA P_Rev in use • Bit to check in ParamPresenceMask - **24**
GppTimeZone	• Optional parameter indicating Offset from Universal time, i.e., difference between local time and Universal time, in increments of 15 min. (signed value). • Bit to check in ParamPresenceMask - **26**
GppNetworkDS-TAdjustment	• Optional parameter indicating 3GPP network daylight saving adjustment • Values: – 0x00 - No adjustment for Daylight Saving Time – 0x01 - 1 hr adjustment for Daylight Saving Time – 0x02 - 2 hr adjustment for Daylight Saving Time • Bit to check in ParamPresenceMask - **27**
Lac	• Optional parameter indicating 3GPP Location Area Code • Bit to check in ParamPresenceMask - **28**
CellID	• Optional parameter indicating 3GPP Cell ID • Bit to check in ParamPresenceMask - **29**
ConcSvcInfo	• Optional parameter indicating 3GPP2 concurrent service Info • Values: – 0x00 - Concurrent service not available – 0x01 - Concurrent service available • Bit to check in ParamPresenceMask - **30**
PRLInd	• Optional parameter indicating 3GPP2 PRL Indicator • Values: – 0x00 - System not in PRL – 0x01 - System is in PRL • Bit to check in ParamPresenceMask - **31**
DTMInd	• Optional parameter indicating Dual Transfer Mode Indication(GSM Only) • Values: – 0x00 - DTM not supported – 0x01 - DTM supported • Bit to check in ParamPresenceMask - **32**
DetailedSvcInfo	• Optional parameter indicating Detailed service information • See nas_detailSvcInfo for more information • Bit to check in ParamPresenceMask - **33**

CDMASystem-InfoExt	• Optional parameter indicating CDMA System Info Ext • See nas_CDMASysInfoExt for more information • Bit to check in ParamPresenceMask - **34**
HdrPersonality	• Optional parameter indicating HDR Personality Information • Values: – 0x00 - Unknown – 0x01 - HRPD – 0x02 - eHRPD • Bit to check in ParamPresenceMask - **35**
TrackAreaCode	• Optional parameter indicating Tracking area code information for LTE • Bit to check in ParamPresenceMask - **36**
CallBarStatus	• Optional parameter indicating Call Barring Status • See nas_callBarStatus for more information • Bit to check in ParamPresenceMask - **37**
MNCPCSDigit-Tlv	• Optional parameter indicating MNC PCS Digit Include Status • See nas_MNCPCSDigitStatus for more information • Bit to check in ParamPresenceMask - **39**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.958.2 Field Documentation

8.958.2.1 uint16_t unpack_nas_SLQSGetServingSystemV2_t::BasestationID

8.958.2.2 uint32_t unpack_nas_SLQSGetServingSystemV2_t::BasestationLatitude

8.958.2.3 uint32_t unpack_nas_SLQSGetServingSystemV2_t::BasestationLongitude

8.958.2.4 nas_callBarStatus unpack_nas_SLQSGetServingSystemV2_t::CallBarStatus

8.958.2.5 uint8_t unpack_nas_SLQSGetServingSystemV2_t::CDMA_P_Rev

8.958.2.6 nas_CDMASysInfoExt unpack_nas_SLQSGetServingSystemV2_t::CDMASystemInfoExt

8.958.2.7 uint32_t unpack_nas_SLQSGetServingSystemV2_t::CellID

8.958.2.8 uint8_t unpack_nas_SLQSGetServingSystemV2_t::ConcSvcInfo

8.958.2.9 nas_currentPLMN unpack_nas_SLQSGetServingSystemV2_t::CurrentPLMN

8.958.2.10 nas_dataSrvCapabilities unpack_nas_SLQSGetServingSystemV2_t::DataSrvCapabilities

8.958.2.11 uint8_t unpack_nas_SLQSGetServingSystemV2_t::DefaultRoamInd

8.958.2.12 nas_detailSvcInfo unpack_nas_SLQSGetServingSystemV2_t::DetailedSvcInfo

8.958.2.13 uint8_t unpack_nas_SLQSGetServingSystemV2_t::DTMInd

8.958.2.14 nas_qaQmi3Gpp2TimeZone unpack_nas_SLQSGetServingSystemV2_t::Gpp2TimeZone

8.958.2.15 uint8_t unpack_nas_SLQSGetServingSystemV2_t::GppNetworkDSTAdjustment

8.958.2.16 uint8_t unpack_nas_SLQSGetServingSystemV2_t::GppTimeZone

8.958.2.17 uint8_t unpack_nas_SLQSGetServingSystemV2_t::HdrPersonality

8.958.2.18 uint16_t unpack_nas_SLQSGetServingSystemV2_t::Lac

8.958.2.19 nas_MNCPCSDigitStatus unpack_nas_SLQSGetServingSystemV2_t::MNCPCSDigitStatTlv

8.958.2.20 uint16_t unpack_nas_SLQSGetServingSystemV2_t::NetworkID

8.958.2.21 swi_uint256_t unpack_nas_SLQSGetServingSystemV2_t::ParamPresenceMask

8.958.2.22 uint8_t unpack_nas_SLQSGetServingSystemV2_t::PRLInd

8.958.2.23 uint8_t unpack_nas_SLQSGetServingSystemV2_t::RoamIndicatorVal

8.958.2.24 nas_roamIndList unpack_nas_SLQSGetServingSystemV2_t::RoamingIndicatorList

8.958.2.25 nas_servSystem unpack_nas_SLQSGetServingSystemV2_t::ServingSystem

8.958.2.26 uint16_t unpack_nas_SLQSGetServingSystemV2_t::SystemID

8.958.2.27 uint16_t unpack_nas_SLQSGetServingSystemV2_t::TrackAreaCode

8.959 unpack_nas_SLQSGetSignalStrength_t Struct Reference

Data Fields

- uint16_t signalStrengthReqMask
- uint16_t rxSignalStrengthListLen
- nas_rxSignalStrengthListElement rxSignalStrengthList [18]
- uint16_t ecioListLen
- nas_ecioListElement ecioList [18]
- int32_t Io
- uint8_t sinr
- uint16_t errorRateListLen
- nas_errorRateListElement errorRateList [18]
- nas_rsrqInformation rsrqInfo
- int16_t ltesnr
- int16_t ltersrp
- swi_uint256_t ParamPresenceMask

8.959.1 Detailed Description

This structure contains the Signal Strength Information

Parameters

signalStrength-ReqMask	• Request Mask – Request additional signal information for: Bit 0 - RSSI Information bit Valid values are: 0 - Do Not Request Additional Info for RSSI 1 - Request Additional Info for RSSI Bit 1 - ECIO Information bit Valid values are: 0 - Do Not Request Additional Info for ECIO 1 - Request Additional Info for ECIO Bit 2 - IO Information bit Valid values are: 0 - Do Not Request Additional Info for IO 1 - Request Additional Info for IO Bit 3 - SINR Information bit Valid values are: 0 - Do Not Request Additional Info for SINR 1 - Request Additional Info for SINR Bit 4 - ERROR RATE Information bit Valid values are: 0 - Do Not Request Additional Info for Error Rate 1 - Request Additional Info for Error Rate Bit 5 - RSRQ Information bit Valid values are: 0 - Do Not Request Additional Info for RSRQ 1 - Request Additional Info for RSRQ Bit 6 - LTE SNR information bit Valid values are: 0 - Do not request additional information for LTE SNR 1 - Request additional information for LTE SNR Bit 7 - LTE RSRP Information bit Valid values are: 0 - Do not request additional information for LTE RSRP 1 - Request additional information for LTE RSRP • Bit to check in ParamPresenceMask - **1**
rxSignal-StrengthListLen	• Number of elements in Receive Signal Strength List • Bit to check in ParamPresenceMask - **17**
rxSignal-StrengthList	• See nas_rxSignalStrengthListElement for more information • Bit to check in ParamPresenceMask - **17**
ecioListLen	• Number of elements in ECIO List • Bit to check in ParamPresenceMask - **18**
ecioList	• See nas_ecioListElement for more information • Bit to check in ParamPresenceMask - **18**

Io	• Received Io in dBm; IO is only applicable for 1xEV-DO • Bit to check in ParamPresenceMask - **19**
sinr	• SINR level – SINR is only applicable for 1xEV-DO; valid levels are 0 to 8 where maximum value for 0 - SINR_LEVEL_0 is -9 dB 1 - SINR_LEVEL_1 is -6 dB 2 - SINR_LEVEL_2 is -4.5 dB 3 - SINR_LEVEL_3 is -3 dB 4 - SINR_LEVEL_4 is -2 dB 5 - SINR_LEVEL_5 is +1 dB 6 - SINR_LEVEL_6 is +3 dB 7 - SINR_LEVEL_7 is +6 dB 8 - SINR_LEVEL_8 is +9 dB • Bit to check in ParamPresenceMask - **20**
errorRateListLen	• Number of elements in Error Rate List • Bit to check in ParamPresenceMask - **21**
errorRateList	• See nas_errorRateListElement for more information • Bit to check in ParamPresenceMask - **21**
rsrqInfo	• See nas_rsrqInformation for more information • Bit to check in ParamPresenceMask - **22**
ltesnr	• LTE SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246. LTE SNR is included only when the current serving system is LTE • Bit to check in ParamPresenceMask - **23**
ltersrp	• LTE SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246. LTE SNR is included only when the current serving system is LTE • Bit to check in ParamPresenceMask - **24**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.959.2 Field Documentation

8.959.2.1 nas_ecioListElement unpack_nas_SLQSGetSignalStrength_t::ecioList[18]

8.959.2.2 uint16_t unpack_nas_SLQSGetSignalStrength_t::ecioListLen

8.959.2.3 nas_errorRateListElement unpack_nas_SLQSGetSignalStrength_t::errorRateList[18]

8.959.2.4 uint16_t unpack_nas_SLQSGetSignalStrength_t::errorRateListLen

8.959.2.5 int32_t unpack_nas_SLQSGetSignalStrength_t::Io

8.959.2.6 int16_t unpack_nas_SLQSGetSignalStrength_t::ltersrp

8.959.2.7 int16_t unpack_nas_SLQSGetSignalStrength_t::ltesnr

8.959.2.8 swi_uint256_t unpack_nas_SLQSGetSignalStrength_t::ParamPresenceMask

8.959.2.9 nas_rsrqInformation unpack_nas_SLQSGetSignalStrength_t::rsrqInfo

8.959.2.10 nas_rxSignalStrengthListElement unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthList[18]

8.959.2.11 uint16_t unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthListLen

8.959.2.12 uint16_t unpack_nas_SLQSGetSignalStrength_t::signalStrengthReqMask

8.959.2.13 uint8_t unpack_nas_SLQSGetSignalStrength_t::sinr

8.960 unpack_nas_SLQSGetSysInfo_t Struct Reference

Data Fields

- nas_SrvStatusInfo * pCDMASrvStatusInfo
- nas_SrvStatusInfo * pHDRSrvStatusInfo
- nas_GSMSrvStatusInfo * pGSMSrvStatusInfo
- nas_GSMSrvStatusInfo * pWCDMASrvStatusInfo
- nas_GSMSrvStatusInfo * pLTESrvStatusInfo
- nas_CDMASysInfo * pCDMASysInfo
- nas_HDRSysInfo * pHDRSysInfo
- nas_GSMSysInfo * pGSMSysInfo
- nas_WCDMASysInfo * pWCDMASysInfo
- nas_LTESysInfo * pLTESysInfo
- nas_AddCDMASysInfo * pAddCDMASysInfo
- uint16_t * pAddHDRSysInfo
- nas_AddSysInfo * pAddGSMSysInfo
- nas_AddSysInfo * pAddWCDMASysInfo
- uint16_t * pAddLTESysInfo
- nas_CallBarringSysInfo * pGSMCallBarringSysInfo
- nas_CallBarringSysInfo * pWCDMACallBarringSysInfo
- uint8_t * pLTEVoiceSupportSysInfo
- uint8_t * pGSMCipherDomainSysInfo
- uint8_t * pWCDMACipherDomainSysInfo
- uint32_t * pCampedCiotLteOpMode
- swi_uint256_t ParamPresenceMask

8.960.1 Detailed Description

This structure contains unpack get system information parameters.

Parameters

pCDMASrv-StatusInfo	• See nas_SrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **16**
pHDRSrvStatus-Info	• See nas_SrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **17**

pGSMSrvStatus-Info	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **18**
pWCDMASrv-StatusInfo	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **19**
pLTESrvStatus-Info	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **20**
pCDMASysInfo	• See nas_CDMASysInfo for more information. • Bit to check in ParamPresenceMask - **21**
pHDRSysInfo	• See nas_HDRSysInfo for more information. • Bit to check in ParamPresenceMask - **22**
pGSMSysInfo	• See nas_GSMSysInfo for more information. • Bit to check in ParamPresenceMask - **23**
pWCDMASys-Info	• See nas_WCDMASysInfo for more information. • Bit to check in ParamPresenceMask - **24**
pLTESysInfo	• See nas_LTESysInfo for more information. • Bit to check in ParamPresenceMask - **25**
pAddCDMASys-Info	• See nas_AddCDMASysInfo for more information. • Bit to check in ParamPresenceMask - **26**
pAddHDRSys-Info	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used. • Bit to check in ParamPresenceMask - **27**
pAddGSMSys-Info	• See nas_AddSysInfo for more information. • Bit to check in ParamPresenceMask - **28**
pAddWCDMA-SysInfo	• See nas_AddSysInfo for more information. • Bit to check in ParamPresenceMask - **29**

pAddLTESysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used. • Bit to check in ParamPresenceMask - **30**
pGSMCall-BarringSysInfo	• See nas_CallBarringSysInfo for more information. • Bit to check in ParamPresenceMask - **31**
pWCDMACall-BarringSysInfo	• See nas_CallBarringSysInfo for more information. • Bit to check in ParamPresenceMask - **32**
pLTEVoice-SupportSysInfo	• Indicates voice support status on LTE. – 0x00 - Voice is not supported – 0x01 - Voice is supported • Bit to check in ParamPresenceMask - **33**
pGSMCipher-DomainSysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched • Bit to check in ParamPresenceMask - **34**
pWCDMA-CipherDomain-SysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched • Bit to check in ParamPresenceMask - **35**
pCampedCiot-LteOpMode	• Camped CIOT LTE Operational Mode. • Values – NAS_CIOT_SYS_MODE_NO_SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_NB1 (0x03) - Camped on LTE NB1 • Bit to check in ParamPresenceMask - **73**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.960.2 Field Documentation

8.960.2.1 **nas_AddCDMASysInfo*** unpack_nas_SLQSGetSysInfo_t::pAddCDMASysInfo

8.960.2.2 **nas_AddSysInfo*** unpack_nas_SLQSGetSysInfo_t::pAddGSMSysInfo

8.960.2.3 **uint16_t*** unpack_nas_SLQSGetSysInfo_t::pAddHDRSysInfo

8.960.2.4 **uint16_t*** unpack_nas_SLQSGetSysInfo_t::pAddLTESysInfo

8.960.2.5 **nas_AddSysInfo*** unpack_nas_SLQSGetSysInfo_t::pAddWCDMASysInfo

8.960.2.6 **swi_uint256_t** unpack_nas_SLQSGetSysInfo_t::ParamPresenceMask

8.960.2.7 **uint32_t*** unpack_nas_SLQSGetSysInfo_t::pCampedCiotLteOpMode

8.960.2.8 **nas_SrvStatusInfo*** unpack_nas_SLQSGetSysInfo_t::pCDMASrvStatusInfo

8.960.2.9 **nas_CDMASysInfo*** unpack_nas_SLQSGetSysInfo_t::pCDMASysInfo

8.960.2.10 **nas_CallBarringSysInfo*** unpack_nas_SLQSGetSysInfo_t::pGSMCallBarringSysInfo

8.960.2.11 **uint8_t*** unpack_nas_SLQSGetSysInfo_t::pGSMCipherDomainSysInfo

8.960.2.12 **nas_GSMSrvStatusInfo*** unpack_nas_SLQSGetSysInfo_t::pGSMSrvStatusInfo

8.960.2.13 **nas_GSMSysInfo*** unpack_nas_SLQSGetSysInfo_t::pGSMSysInfo

8.960.2.14 **nas_SrvStatusInfo*** unpack_nas_SLQSGetSysInfo_t::pHDRSrvStatusInfo

8.960.2.15 **nas_HDRSysInfo*** unpack_nas_SLQSGetSysInfo_t::pHDRSysInfo

8.960.2.16 **nas_GSMSrvStatusInfo*** unpack_nas_SLQSGetSysInfo_t::pLTESrvStatusInfo

8.960.2.17 **nas_LTESysInfo*** unpack_nas_SLQSGetSysInfo_t::pLTESysInfo

8.960.2.18 **uint8_t*** unpack_nas_SLQSGetSysInfo_t::pLTEVoiceSupportSysInfo

8.960.2.19 **nas_CallBarringSysInfo*** unpack_nas_SLQSGetSysInfo_t::pWCDMACallBarringSysInfo

8.960.2.20 **uint8_t*** unpack_nas_SLQSGetSysInfo_t::pWCDMACipherDomainSysInfo

8.960.2.21 **nas_GSMSrvStatusInfo*** unpack_nas_SLQSGetSysInfo_t::pWCDMASrvStatusInfo

8.960.2.22 **nas_WCDMASysInfo*** unpack_nas_SLQSGetSysInfo_t::pWCDMASysInfo

8.961 unpack_nas_SLQSGetSysInfoV2_t Struct Reference

Data Fields

- nas_SrvStatusInfo * pCDMASrvStatusInfo
- nas_SrvStatusInfo * pHDRSrvStatusInfo
- nas_GSMSrvStatusInfo * pGSMSrvStatusInfo
- nas_GSMSrvStatusInfo * pWCDMASrvStatusInfo
- nas_GSMSrvStatusInfo * pLTESrvStatusInfo
- nas_CDMASysInfo * pCDMASysInfo
- nas_HDRSysInfo * pHDRSysInfo
- nas_GSMSysInfo * pGSMSysInfo

- nas_WCDMASysInfo * pWCDMASysInfo
- nas_LTESysInfo * pLTESysInfo
- nas_AddCDMASysInfo * pAddCDMASysInfo
- uint16_t * pAddHDRSysInfo
- nas_AddSysInfo * pAddGSMSysInfo
- nas_AddSysInfo * pAddWCDMASysInfo
- uint16_t * pAddLTESysInfo
- nas_CallBarringSysInfo * pGSMCallBarringSysInfo
- nas_CallBarringSysInfo * pWCDMACallBarringSysInfo
- uint8_t * pLTEVoiceSupportSysInfo
- uint8_t * pGSMCipherDomainSysInfo
- uint8_t * pWCDMACipherDomainSysInfo
- uint32_t * pCampedCiotLteOpMode
- nas_LteEmbmsCoverageTlv * pLteEmbmsCoverage
- nas_SimRejInfoTlv * pSimRejInfo
- nas_ImsVoiceSupportLteTlv * pImsVoiceSupportLte
- nas_LteVoiceDomainTlv * pLteVoiceDomain
- nas_SrvRegRestrictionTlv * pSrvRegRestriction
- nas_LteRegDomainTlv * pLteRegDomain
- nas_LteEmbmsTraceIdTlv * pLteEmbmsTraceId
- nas_NR5GSrvStatusTlv * pNR5GSrvStatusinfo
- nas_NR5GSysInfoTlv * pNR5GSysInfo
- nas_NR5GCellStatusInfoTlv * pNR5GCellStatusInfo
- swi_uint256_t ParamPresenceMask

8.961.1 Detailed Description

This structure contains unpack get system information parameters.

Parameters

pCDMASrvStatusInfo	• See nas_SrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **16**
pHDRSrvStatusInfo	• See nas_SrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **17**
pGSMSrvStatusInfo	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **18**
pWCDMASrvStatusInfo	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **19**
pLTESrvStatusInfo	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **20**
pCDMASysInfo	• See nas_CDMASysInfo for more information. • Bit to check in ParamPresenceMask - **21**

pHDRSysInfo	• See nas_HDRSysInfo for more information. • Bit to check in ParamPresenceMask - **22**
pGSMSysInfo	• See nas_GSMSysInfo for more information. • Bit to check in ParamPresenceMask - **23**
pWCDMASysInfo	• See nas_WCDMASysInfo for more information. • Bit to check in ParamPresenceMask - **24**
pLTESysInfo	• See nas_LTESysInfo for more information. • Bit to check in ParamPresenceMask - **25**
pAddCDMASysInfo	• See nas_AddCDMASysInfo for more information. • Bit to check in ParamPresenceMask - **26**
pAddHDRSysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used. • Bit to check in ParamPresenceMask - **27**
pAddGSMSysInfo	• See nas_AddSysInfo for more information. • Bit to check in ParamPresenceMask - **28**
pAddWCDMASysInfo	• See nas_AddSysInfo for more information. • Bit to check in ParamPresenceMask - **29**
pAddLTESysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used. • Bit to check in ParamPresenceMask - **30**
pGSMCallBarringSysInfo	• See nas_CallBarringSysInfo for more information. • Bit to check in ParamPresenceMask - **31**
pWCDMACallBarringSysInfo	• See nas_CallBarringSysInfo for more information. • Bit to check in ParamPresenceMask - **32**

pLTEVoice-SupportSysInfo	• Indicates voice support status on LTE. – 0x00 - Voice is not supported – 0x01 - Voice is supported • Bit to check in ParamPresenceMask - **33**
pGSMCipher-DomainSysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched • Bit to check in ParamPresenceMask - **34**
pWCDMA-CipherDomain-SysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched • Bit to check in ParamPresenceMask - **35**
pCampedCiot-LteOpMode	• Camped CIOT LTE Operational Mode. • Values – NAS_CIOT_SYS_MODE_NO_SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_NB1 (0x03) - Camped on LTE NB1 • Bit to check in ParamPresenceMask - **73**
pLteEmbms-Coverage	• See nas_LteEmbmsCoverageTlv for more information. • Bit to check in ParamPresenceMask - **38**
pSimRejInfo	• See nas_SimRejInfoTlv for more information. • Bit to check in ParamPresenceMask - **39**
pImsVoice-SupportLte	• See nas_ImsVoiceSupportLteTlv for more information. • Bit to check in ParamPresenceMask - **41**
pLteVoice-Domain	• See nas_LteVoiceDomainTlv for more information. • Bit to check in ParamPresenceMask - **42**
pSrvReg-Restriction	• See nas_SrvRegRestrictionTlv for more information. • Bit to check in ParamPresenceMask - **47**

pLteRegDomain	• See nas_LteRegDomainTlv for more information. • Bit to check in ParamPresenceMask - **49**
pLteEmbms-TraceId	• See nas_LteEmbmsTraceIdTlv for more information. • Bit to check in ParamPresenceMask - **52**
pNR5GSrv-Statusinfo	• See nas_NR5GSrvStatusTlv for more information. • Bit to check in ParamPresenceMask - **74**
pNR5gSysInfo	• See nas_NR5GSysInfoTlv for more information. • Bit to check in ParamPresenceMask - **75**
pNR5GCell-StatusInfo	• See nas_NR5GCellStatusInfoTlv for more information. • Bit to check in ParamPresenceMask - **76**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.961.2 Field Documentation

8.961.2.1 nas_AddCDMASysInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pAddCDMASysInfo

8.961.2.2 nas_AddSysInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pAddGSMSysInfo

8.961.2.3 uint16_t∗ unpack_nas_SLQSGetSysInfoV2_t::pAddHDRSysInfo

8.961.2.4 uint16_t∗ unpack_nas_SLQSGetSysInfoV2_t::pAddLTESysInfo

8.961.2.5 nas_AddSysInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pAddWCDMASysInfo

8.961.2.6 swi_uint256_t unpack_nas_SLQSGetSysInfoV2_t::ParamPresenceMask

8.961.2.7 uint32_t∗ unpack_nas_SLQSGetSysInfoV2_t::pCampedCiotLteOpMode

8.961.2.8 nas_SrvStatusInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pCDMASrvStatusInfo

8.961.2.9 nas_CDMASysInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pCDMASysInfo

8.961.2.10 nas_CallBarringSysInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pGSMCallBarringSysInfo

8.961.2.11 uint8_t∗ unpack_nas_SLQSGetSysInfoV2_t::pGSMCipherDomainSysInfo

8.961.2.12 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pGSMSrvStatusInfo

8.961.2.13 nas_GSMSysInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pGSMSysInfo

8.961.2.14 nas_SrvStatusInfo∗ unpack_nas_SLQSGetSysInfoV2_t::pHDRSrvStatusInfo

8.961.2.15 **nas_HDRSysInfo**∗ unpack_nas_SLQSGetSysInfoV2_t::pHDRSysInfo

8.961.2.16 **nas_ImsVoiceSupportLteTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pImsVoiceSupportLte

8.961.2.17 **nas_LteEmbmsCoverageTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pLteEmbmsCoverage

8.961.2.18 **nas_LteEmbmsTraceIdTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pLteEmbmsTraceId

8.961.2.19 **nas_LteRegDomainTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pLteRegDomain

8.961.2.20 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSGetSysInfoV2_t::pLTESrvStatusInfo

8.961.2.21 **nas_LTESysInfo**∗ unpack_nas_SLQSGetSysInfoV2_t::pLTESysInfo

8.961.2.22 **nas_LteVoiceDomainTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pLteVoiceDomain

8.961.2.23 uint8_t∗ unpack_nas_SLQSGetSysInfoV2_t::pLTEVoiceSupportSysInfo

8.961.2.24 **nas_NR5GCellStatusInfoTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pNR5GCellStatusInfo

8.961.2.25 **nas_NR5GSrvStatusTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pNR5GSrvStatusinfo

8.961.2.26 **nas_NR5GSysInfoTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pNR5GSysInfo

8.961.2.27 **nas_SimRejInfoTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pSimRejInfo

8.961.2.28 **nas_SrvRegRestrictionTlv**∗ unpack_nas_SLQSGetSysInfoV2_t::pSrvRegRestriction

8.961.2.29 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSGetSysInfoV2_t::pWCDMACallBarringSysInfo

8.961.2.30 uint8_t∗ unpack_nas_SLQSGetSysInfoV2_t::pWCDMACipherDomainSysInfo

8.961.2.31 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSGetSysInfoV2_t::pWCDMASrvStatusInfo

8.961.2.32 **nas_WCDMASysInfo**∗ unpack_nas_SLQSGetSysInfoV2_t::pWCDMASysInfo

8.962 unpack_nas_SLQSGetSysSelectionPref_t Struct Reference

Data Fields

- uint8_t ∗ pEmerMode
- uint16_t ∗ pModePref
- uint64_t ∗ pBandPref
- uint16_t ∗ pPRLPref
- uint16_t ∗ pRoamPref
- uint64_t ∗ pLTEBandPref
- uint8_t ∗ pNetSelPref
- uint32_t ∗ pSrvDomainPref
- uint32_t ∗ pGWAcqOrderPref
- swi_uint256_t ParamPresenceMask

8.962.1 Detailed Description

This structure contains unpack get get system selection preferences parameters.

Parameters

pEmerMode	• Optional parameter specifying the emergency Mode • Values: – 0 - OFF (normal) – 1 - ON (Emergency) • Bit to check in ParamPresenceMask - **16**
pModePref	• Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE • Bit to check in ParamPresenceMask - **17**

pBandPref	• Optional parameter • Bit mask representing the band preference • Bit values: – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved • Bit to check in ParamPresenceMask - **18**

pPRLPref	• Optional parameter indicating the CDMA PRL Preference • Values: – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems • Bit to check in ParamPresenceMask - **19**
pRoamPref	• Optional parameter indicating the roaming Preference • Values: – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator • Bit to check in ParamPresenceMask - **20**

pLTEBandPref	• Optional parameter • Bit mask representing the LTE band preference • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – All other bits are reserved • Bit to check in ParamPresenceMask - **21**
pNetSelPref	• Optional parameter indicating network selection preference • Values: – 0x00 - Automatic network selection – 0x01 - Manual network selection. • Bit to check in ParamPresenceMask - **22**

pSrvDomainPref	• Optional parameter indicating Service domain preference • Values: – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach • Bit to check in ParamPresenceMask - **24**
pGWAcqOrder-Pref	• Optional parameter indicating GSM/WCDMA Acquisition order Preference • Values: – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM • Bit to check in ParamPresenceMask - **25**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.962.2 Field Documentation

8.962.2.1 swi_uint256_t unpack_nas_SLQSGetSysSelectionPref_t::ParamPresenceMask

8.962.2.2 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pBandPref

8.962.2.3 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pEmerMode

8.962.2.4 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pGWAcqOrderPref

8.962.2.5 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pLTEBandPref

8.962.2.6 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pModePref

8.962.2.7 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pNetSelPref

8.962.2.8 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pPRLPref

8.962.2.9 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pRoamPref

8.962.2.10 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pSrvDomainPref

8.963 unpack_nas_SLQSGetSysSelectionPrefExt_t Struct Reference

Data Fields

- nas_EmerModeTlv * pEmerMode
- nas_ModePrefTlv * pModePref
- nas_BandPrefTlv * pBandPref
- nas_PRLPrefTlv * pPRLPref

- nas_RoamPrefTlv * pRoamPref
- nas_LTEBandPrefTlv * pLTEBandPref
- nas_NetSelPrefTlv * pNetSelPref
- nas_SrvDomainPrefTlv * pSrvDomainPref
- nas_GWAcqOrderPrefTlv * pGWAcqOrderPref
- nas_AcqOrderPrefTlv * pAcqOrderPref
- nas_RatDisabledMaskTlv * pRatDisabledMask
- nas_CiotLteOpModePrefTlv * pCiotLteOpModePref
- nas_LteM1BandPrefTlv * pLteM1BandPref
- nas_LteNb1BandPrefTlv * pLteNb1BandPref
- nas_CiotAcqOrderPrefTlv * pCiotAcqOrderPref
- nas_NR5gBandPrefTlv * pNr5gBandPref
- nas_LTEBandPrefExtTlv * pLTEBandPrefExt

8.963.1 Detailed Description

Structure for storing the current preferred system selection settings for the device.

Parameters

pEmerMode	• Optional parameter specifying the emergency Mode • See nas_EmerModeTlv for more information
pModePref	• Optional parameterindicating the radio technology mode preference • See nas_ModePrefTlv for more information
pBandPref	• Optional parameter representing the band preference • See nas_BandPrefTlv for more information
pPRLPref	• Optional parameter indicating the CDMA PRL Preference • See nas_PRLPrefTlv for more information
pRoamPref	• Optional parameter indicating the roaming Preference • See nas_RoamPrefTlv for more information
pLTEBandPref	• Optional parameter representing the LTE band preference • See nas_LTEBandPrefTlv for more information
pNetSelPref	• Optional parameter indicating network selection preference • See nas_NetSelPrefTlv for more information
pSrvDomainPref	• Optional parameter indicating Service domain preference • See nas_SrvDomainPrefTlv for more information

pGWAcqOrderPref	• Optional parameter indicating GSM/WCDMA Acquisition order Preference • See nas_GWAcqOrderPrefTlv for more information
pAcqOrderPref	• Optional parameter • Acquisition Order Preference • See nas_AcqOrderPrefTlv for more information
pRatDisabledMask	• Optional parameter representing the radio technologies that are disabled. • See nas_RatDisabledMaskTlv for more information
pCiotLteOpModePref	• Optional parameter • CIOT LTE Operational Mode Preference • See nas_CiotLteOpModePrefTlv for more information
pLteM1BandPref	• Optional parameter representing the LTE M1 band preference • See nas_LteM1BandPrefTlv for more information
pLteNb1BandPref	• Optional parameter representing the LTE NB1 band preference. • See nas_LteNb1BandPrefTlv for more information
pCiotAcqOrderPref	• Optional parameter indicating CIOT Acquisition Order Preference • See nas_CiotAcqOrderPrefTlv for more information
pNr5gBandPref	• Optional parameter indicating NR5G Band Preference • See nas_NR5gBandPrefTlv for more information
pLTEBandPrefExt	• Optional parameter indicating LTE Band Preference Extended • See nas_LTEBandPrefExtTlv for more information

8.963.2 Field Documentation

8.963.2.1 nas_AcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pAcqOrderPref

8.963.2.2 nas_BandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pBandPref

8.963.2.3 nas_CiotAcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pCiotAcqOrderPref

8.963.2.4 nas_CiotLteOpModePrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pCiotLteOpModePref

8.963.2.5 nas_EmerModeTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pEmerMode

8.963.2.6 nas_GWAcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pGWAcqOrderPref

8.963.2.7 **nas_LTEBandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLTEBandPref

8.963.2.8 **nas_LTEBandPrefExtTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLTEBandPrefExt

8.963.2.9 **nas_LteM1BandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLteM1BandPref

8.963.2.10 **nas_LteNb1BandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLteNb1BandPref

8.963.2.11 **nas_ModePrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pModePref

8.963.2.12 **nas_NetSelPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pNetSelPref

8.963.2.13 **nas_NR5gBandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pNr5gBandPref

8.963.2.14 **nas_PRLPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pPRLPref

8.963.2.15 **nas_RatDisabledMaskTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pRatDisabledMask

8.963.2.16 **nas_RoamPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pRoamPref

8.963.2.17 **nas_SrvDomainPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pSrvDomainPref

8.964 unpack_nas_SLQSGetSysSelectionPrefExtV2_t Struct Reference

Data Fields

- nas_EmerModeTlv ∗ pEmerMode
- nas_ModePrefTlv ∗ pModePref
- nas_BandPrefTlv ∗ pBandPref
- nas_PRLPrefTlv ∗ pPRLPref
- nas_RoamPrefTlv ∗ pRoamPref
- nas_LTEBandPrefTlv ∗ pLTEBandPref
- nas_NetSelPrefTlv ∗ pNetSelPref
- nas_SrvDomainPrefTlv ∗ pSrvDomainPref
- nas_GWAcqOrderPrefTlv ∗ pGWAcqOrderPref
- nas_AcqOrderPrefTlv ∗ pAcqOrderPref
- nas_RatDisabledMaskTlv ∗ pRatDisabledMask
- nas_CiotLteOpModePrefTlv ∗ pCiotLteOpModePref
- nas_LteM1BandPrefTlv ∗ pLteM1BandPref
- nas_LteNb1BandPrefTlv ∗ pLteNb1BandPref
- nas_CiotAcqOrderPrefTlv ∗ pCiotAcqOrderPref
- nas_NR5gBandPrefTlv ∗ pNr5gBandPref
- nas_LTEBandPrefExtTlv ∗ pLTEBandPrefExt
- nas_TDSCDMABandPrefTlv ∗ pTDSCDMABandPref
- nas_SrvRegRestricTlv ∗ pSrvRegRestric
- nas_UsageSettingTlv ∗ pUsageSetting
- nas_VoiceDomainPrefTlv ∗ pVoiceDomainPref
- swi_uint256_t ParamPresenceMask

8.964.1 Detailed Description

Structure for storing the current preferred system selection settings for the device.

Parameters

pEmerMode	• Optional parameter specifying the emergency Mode • See nas_EmerModeTlv for more information • Bit to check in ParamPresenceMask - **16**
pModePref	• Optional parameterindicating the radio technology mode preference • See nas_ModePrefTlv for more information • Bit to check in ParamPresenceMask - **17**
pBandPref	• Optional parameter representing the band preference • See nas_BandPrefTlv for more information • Bit to check in ParamPresenceMask - **18**
pPRLPref	• Optional parameter indicating the CDMA PRL Preference • See nas_PRLPrefTlv for more information • Bit to check in ParamPresenceMask - **19**
pRoamPref	• Optional parameter indicating the roaming Preference • See nas_RoamPrefTlv for more information • Bit to check in ParamPresenceMask - **20**
pLTEBandPref	• Optional parameter representing the LTE band preference • See nas_LTEBandPrefTlv for more information • Bit to check in ParamPresenceMask - **21**
pNetSelPref	• Optional parameter indicating network selection preference • See nas_NetSelPrefTlv for more information • Bit to check in ParamPresenceMask - **22**
pSrvDomainPref	• Optional parameter indicating Service domain preference • See nas_SrvDomainPrefTlv for more information • Bit to check in ParamPresenceMask - **24**
pGWAcqOrder-Pref	• Optional parameter indicating GSM/WCDMA Acquisition order Preference • See nas_GWAcqOrderPrefTlv for more information • Bit to check in ParamPresenceMask - **25**
pAcqOrderPref	• Optional parameter • Acquisition Order Preference • See nas_AcqOrderPrefTlv for more information • Bit to check in ParamPresenceMask - **28**

pRatDisabled-Mask	• Optional parameter representing the radio technologies that are disabled. • See nas_RatDisabledMaskTlv for more information • Bit to check in ParamPresenceMask - **34**
pCiotLteOp-ModePref	• Optional parameter • CIOT LTE Operational Mode Preference • See nas_CiotLteOpModePrefTlv for more information • Bit to check in ParamPresenceMask - **36**
pLteM1BandPref	• Optional parameter representing the LTE M1 band preference • See nas_LteM1BandPrefTlv for more information • Bit to check in ParamPresenceMask - **37**
pLteNb1Band-Pref	• Optional parameter representing the LTE NB1 band preference. • See nas_LteNb1BandPrefTlv for more information • Bit to check in ParamPresenceMask - **38**
pCiotAcqOrder-Pref	• Optional parameter indicating CIOT Acquisition Order Preference • See nas_CiotAcqOrderPrefTlv for more information • Bit to check in ParamPresenceMask - **39**
pNr5gBandPref	• Optional parameter indicating NR5G Band Preference • See nas_NR5gBandPrefTlv for more information • Bit to check in ParamPresenceMask - **40**
pLTEBandPref-Ext	• Optional parameter indicating LTE Band Preference Extended • See nas_LTEBandPrefExtTlv for more information • Bit to check in ParamPresenceMask - **35**
pTDSCDMA-BandPref	• Optional parameter indicating TDSCDMA Preference • See nas_TDSCDMABandPrefTlv for more information • Bit to check in ParamPresenceMask - **26**
pSrvRegRestric	• Optional parameter indicating SRV Registration Restriction • See nas_SrvRegRestricTlv for more information • Bit to check in ParamPresenceMask - **29**
pUsageSetting	• Optional parameter indicating Usage settings • See nas_UsageSettingTlv for more information • Bit to check in ParamPresenceMask - **31**

pVoiceDomain-Pref	• Optional parameter indicating Voice Domain Preference • See nas_VoiceDomainPrefTlv for more information • Bit to check in ParamPresenceMask - **32**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.964.2 Field Documentation

8.964.2.1 nas_AcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pAcqOrderPref

8.964.2.2 swi_uint256_t unpack_nas_SLQSGetSysSelectionPrefExtV2_t::ParamPresenceMask

8.964.2.3 nas_BandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pBandPref

8.964.2.4 nas_CiotAcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pCiotAcqOrderPref

8.964.2.5 nas_CiotLteOpModePrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pCiotLteOpModePref

8.964.2.6 nas_EmerModeTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pEmerMode

8.964.2.7 nas_GWAcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pGWAcqOrderPref

8.964.2.8 nas_LTEBandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pLTEBandPref

8.964.2.9 nas_LTEBandPrefExtTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pLTEBandPrefExt

8.964.2.10 nas_LteM1BandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pLteM1BandPref

8.964.2.11 nas_LteNb1BandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pLteNb1BandPref

8.964.2.12 nas_ModePrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pModePref

8.964.2.13 nas_NetSelPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pNetSelPref

8.964.2.14 nas_NR5gBandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pNr5gBandPref

8.964.2.15 nas_PRLPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pPRLPref

8.964.2.16 nas_RatDisabledMaskTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pRatDisabledMask

8.964.2.17 nas_RoamPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pRoamPref

8.964.2.18 nas_SrvDomainPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pSrvDomainPref

8.964.2.19 nas_SrvRegRestricTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pSrvRegRestric

8.964.2.20 nas_TDSCDMABandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pTDSCDMABandPref

8.964.2.21 nas_UsageSettingTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pUsageSetting

8.964.2.22 nas_VoiceDomainPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExtV2_t::pVoiceDomainPref

8.965 unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t Struct Reference

Data Fields

- nas_EdrxEnableType edrxEnableTypeTlv
- nas_EdrxCycleLength edrxCycleLengthTlv
- nas_EdrxPagingTimeWindow edrxPagingTimeWindowTlv
- nas_EdrxRatType edrxRatTypeTlv
- nas_EdrxCiotLteMode edrxCiotLteModeTlv
- swi_uint256_t ParamPresenceMask

8.965.1 Detailed Description

Structure for storing the eDRX Change Info indication parameters.

Parameters

edrxEnableTypeTlv	• See nas_EdrxEnableType for more information. • Bit to check in ParamPresenceMask - **16**
edrxCycleLengthTlv	• See nas_EdrxCycleLength for more information. • Bit to check in ParamPresenceMask - **17**
edrxPagingTimeWindowTlv	• See nas_EdrxPagingTimeWindow for more information. • Bit to check in ParamPresenceMask - **18**
edrxRatTypeTlv	• See nas_EdrxRatType for more information. • Bit to check in ParamPresenceMask - **19**
edrxCiotLteModeTlv	• See nas_EdrxCiotLteMode for more information. • Bit to check in ParamPresenceMask - **20**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.965.2 Field Documentation

8.965.2.1 nas_EdrxCiotLteMode unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxCiotLteModeTlv

8.965.2.2 nas_EdrxCycleLength unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxCycleLengthTlv

8.965.2.3 nas_EdrxEnableType unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxEnableTypeTlv

8.965.2.4 nas_EdrxPagingTimeWindow unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxPagingTimeWindowTlv

8.965.2.5 nas_EdrxRatType unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxRatTypeTlv

8.965.2.6 swi_uint256_t unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::ParamPresenceMask

8.966 unpack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference

Data Fields

- nas_namName * pNAMNameInfo
- nas_dirNum * pDirNum
- nas_homeSIDNID * pHomeSIDNID
- nas_minBasedIMSI * pMinBasedIMSI
- nas_trueIMSI * pTrueIMSI
- nas_CDMAChannel * pCDMAChannel
- nas_Mdn * pMdn
- swi_uint256_t ParamPresenceMask

8.966.1 Detailed Description

Structure for 3GPP2Subscription unpack.

Parameters

pNAMNameInfo	[Optional] • See nas_namName for more information • Bit to check in ParamPresenceMask - **16**
pDirNum	[Optional] • See nas_dirNum for more information • Bit to check in ParamPresenceMask - **17**
pHomeSIDNID	[Optional] • See nas_homeSIDNID for more information • Bit to check in ParamPresenceMask - **18**
pMinBasedIMSI	[Optional] • See nas_minBasedIMSI for more information • Bit to check in ParamPresenceMask - **19**
pTrueIMSI	[Optional] • See nas_trueIMSI for more information • Bit to check in ParamPresenceMask - **20**
pCDMAChannel	[Optional] • See nas_CDMAChannel for more information • Bit to check in ParamPresenceMask - **21**
pMdn	[Optional] • See nas_Mdn for more information • Bit to check in ParamPresenceMask - **22**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.966.2 Field Documentation

8.966.2.1 swi_uint256_t unpack_nas_SLQSNasGet3GPP2Subscription_t::ParamPresenceMask

8.966.2.2 nas_CDMAChannel* unpack_nas_SLQSNasGet3GPP2Subscription_t::pCDMAChannel

8.966.2.3 nas_dirNum* unpack_nas_SLQSNasGet3GPP2Subscription_t::pDirNum

8.966.2.4 nas_homeSIDNID* unpack_nas_SLQSNasGet3GPP2Subscription_t::pHomeSIDNID

8.966.2.5 nas_Mdn* unpack_nas_SLQSNasGet3GPP2Subscription_t::pMdn

8.966.2.6 nas_minBasedIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pMinBasedIMSI

8.966.2.7 nas_namName* unpack_nas_SLQSNasGet3GPP2Subscription_t::pNAMNameInfo

8.966.2.8 nas_trueIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pTrueIMSI

8.967 unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference

Data Fields

- nas_GERANInfo * pGERANInfo
- nas_UMTSInfo * pUMTSInfo
- nas_CDMAInfo * pCDMAInfo
- nas_LTEInfoIntrafreq * pLTEInfoIntrafreq
- nas_LTEInfoInterfreq * pLTEInfoInterfreq
- nas_LTEInfoNeighboringGSM * pLTEInfoNeighboringGSM
- nas_LTEInfoNeighboringWCDMA * pLTEInfoNeighboringWCDMA
- uint32_t * pUMTSCellID
- nas_WCDMAInfoLTENeighborCell * pWCDMAInfoLTENeighborCell
- swi_uint256_t ParamPresenceMask

8.967.1 Detailed Description

This structure contains information about the Get Cell Location response parameters.

Parameters

pGERANInfo	• See nas_GERANInfo for more information. • Bit to check in ParamPresenceMask - **16**
pUMTSInfo	• See nas_UMTSInfo for more information. • Bit to check in ParamPresenceMask - **17**
pCDMAInfo	• See nas_CDMAInfo for more information. • Bit to check in ParamPresenceMask - **18**
pLTEInfo-Intrafreq	• See nas_LTEInfoIntrafreq for more information. • Bit to check in ParamPresenceMask - **19**

pLTEInfo-Interfreq	• See nas_LTEInfoInterfreq for more information. • Bit to check in ParamPresenceMask - **20**
pLTEInfo-NeighboringGS-M	• See nas_LTEInfoNeighboringGSM for more information. • Bit to check in ParamPresenceMask - **21**
pLTEInfo-NeighboringWC-DMA	• See nas_LTEInfoNeighboringWCDMA for more information. • Bit to check in ParamPresenceMask - **22**
pUMTSCellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present. • Bit to check in ParamPresenceMask - **23**
pWCDMAInfoLT-ENeighborCell	• See nas_WCDMAInfoLTENeighborCell for more information. • Bit to check in ParamPresenceMask - **24**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.967.2 Field Documentation

8.967.2.1 swi_uint256_t unpack_nas_SLQSNasGetCellLocationInfo_t::ParamPresenceMask

8.967.2.2 nas_CDMAInfo* unpack_nas_SLQSNasGetCellLocationInfo_t::pCDMAInfo

8.967.2.3 nas_GERANInfo* unpack_nas_SLQSNasGetCellLocationInfo_t::pGERANInfo

8.967.2.4 nas_LTEInfoInterfreq* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoInterfreq

8.967.2.5 nas_LTEInfoIntrafreq* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoIntrafreq

8.967.2.6 nas_LTEInfoNeighboringGSM* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringGSM

8.967.2.7 nas_LTEInfoNeighboringWCDMA* unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringWCDMA

8.967.2.8 uint32_t* unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSCellID

8.967.2.9 nas_UMTSInfo* unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSInfo

8.967.2.10 nas_WCDMAInfoLTENeighborCell* unpack_nas_SLQSNasGetCellLocationInfo_t::pWCDMAInfoLTENeighbor-Cell

8.968 unpack_nas_SLQSNasGetCellLocationInfoV2_t Struct Reference

Data Fields

- nas_GERANInfo * pGERANInfo

- nas_UMTSInfo * pUMTSInfo
- nas_CDMAInfo * pCDMAInfo
- nas_LTEInfoIntrafreq * pLTEInfoIntrafreq
- nas_LTEInfoInterfreq * pLTEInfoInterfreq
- nas_LTEInfoNeighboringGSM * pLTEInfoNeighboringGSM
- nas_LTEInfoNeighboringWCDMA * pLTEInfoNeighboringWCDMA
- uint32_t * pUMTSCellID
- nas_WCDMAInfoLTENeighborCell * pWCDMAInfoLTENeighborCell
- nas_WCDMACellInfoExt * pWCDMACellInfoExt
- nas_UMTSExtInfo * pUMTSExtInfo
- nas_LteEarfcnInfo * pLteEarfcnInfo
- swi_uint256_t ParamPresenceMask

8.968.1 Detailed Description

This structure contains information about the Get Cell Location response parameters.

Parameters

pGERANInfo	• See nas_GERANInfo for more information. • Bit to check in ParamPresenceMask - **16**
pUMTSInfo	• See nas_UMTSInfo for more information. • Bit to check in ParamPresenceMask - **17**
pCDMAInfo	• See nas_CDMAInfo for more information. • Bit to check in ParamPresenceMask - **18**
pLTEInfo-Intrafreq	• See nas_LTEInfoIntrafreq for more information. • Bit to check in ParamPresenceMask - **19**
pLTEInfo-Interfreq	• See nas_LTEInfoInterfreq for more information. • Bit to check in ParamPresenceMask - **20**
pLTEInfo-NeighboringGS-M	• See nas_LTEInfoNeighboringGSM for more information. • Bit to check in ParamPresenceMask - **21**
pLTEInfo-NeighboringWC-DMA	• See nas_LTEInfoNeighboringWCDMA for more information. • Bit to check in ParamPresenceMask - **22**
pUMTSCellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present. • Bit to check in ParamPresenceMask - **23**

pWCDMAInfoLTENeighborCell	• See nas_WCDMAInfoLTENeighborCell for more information. • Bit to check in ParamPresenceMask - **24**
pWCDMACellInfoExt	• See nas_WCDMACellInfoExt for more information. • Bit to check in ParamPresenceMask - **28**
pUMTSExtInfo	• See nas_UMTSExtInfo for more information. • Bit to check in ParamPresenceMask - **34**
pLteEarfcnInfo	• See nas_LteEarfcnInfo for more information. • Bit to check in ParamPresenceMask - **41**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.968.2 Field Documentation

8.968.2.1 swi_uint256_t unpack_nas_SLQSNasGetCellLocationInfoV2_t::ParamPresenceMask

8.968.2.2 nas_CDMAInfo* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pCDMAInfo

8.968.2.3 nas_GERANInfo* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pGERANInfo

8.968.2.4 nas_LteEarfcnInfo* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pLteEarfcnInfo

8.968.2.5 nas_LTEInfoInterfreq* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pLTEInfoInterfreq

8.968.2.6 nas_LTEInfoIntrafreq* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pLTEInfoIntrafreq

8.968.2.7 nas_LTEInfoNeighboringGSM* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pLTEInfoNeighboringGSM

8.968.2.8 nas_LTEInfoNeighboringWCDMA* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pLTEInfoNeighboringWCDMA

8.968.2.9 uint32_t* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pUMTSCellID

8.968.2.10 nas_UMTSExtInfo* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pUMTSExtInfo

8.968.2.11 nas_UMTSInfo* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pUMTSInfo

8.968.2.12 nas_WCDMACellInfoExt* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pWCDMACellInfoExt

8.968.2.13 nas_WCDMAInfoLTENeighborCell* unpack_nas_SLQSNasGetCellLocationInfoV2_t::pWCDMAInfoLTENeighborCell

8.969 unpack_nas_SLQSNASGeteDRXParams_t Struct Reference

Data Fields

- uint8_t * pCycleLen
- uint8_t * pPagingTimeWindow
- uint8_t * pEdrxEnable
- swi_uint256_t ParamPresenceMask

8.969.1 Detailed Description

This structure contains the SLQSNASGeteDRXParams unpack parameters.

Parameters

pCycleLen	[Optional] • eDRX cycle length • Value is per 3GPP TS 24.008 Rel-13 Section 10.5.5.32 • Bit to check in ParamPresenceMask - **16**
pPagingTime-Window	[Optional] • eDRX paging time window • Value is per 3GPP TS 24.008 Rel-13 Section10.5.5.32 • Bit to check in ParamPresenceMask - **17**
pEdrxEnable	[Optional] • eDRX Enable • Values – 0x00 - Disable – 0x01 - Enable • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.969.2 Field Documentation

8.969.2.1 swi_uint256_t unpack_nas_SLQSNASGeteDRXParams_t::ParamPresenceMask

8.969.2.2 uint8_t* unpack_nas_SLQSNASGeteDRXParams_t::pCycleLen

8.969.2.3 uint8_t* unpack_nas_SLQSNASGeteDRXParams_t::pEdrxEnable

8.969.2.4 uint8_t* unpack_nas_SLQSNASGeteDRXParams_t::pPagingTimeWindow

8.970 unpack_nas_SLQSNASGeteDRXParamsExt_t Struct Reference

Data Fields

- uint8_t * pCycleLen
- uint8_t * pPagingTimeWindow
- uint8_t * pEdrxEnable
- uint8_t * pEdrxRAT
- uint32_t * pLteOpMode

- swi_uint256_t ParamPresenceMask

8.970.1 Detailed Description

This structure contains the SLQSNASGeteDRXParams unpack parameters.

Parameters

pCycleLen	[Optional] • eDRX cycle length • Value is per 3GPP TS 24.008 Rel-13 Section 10.5.5.32 • Bit to check in ParamPresenceMask - **16**
pPagingTime-Window	[Optional] • eDRX paging time window • Value is per 3GPP TS 24.008 Rel-13 Section10.5.5.32 • Bit to check in ParamPresenceMask - **17**
pEdrxEnable	[Optional] • eDRX Enable • Values – 0x00 - Disable – 0x01 - Enable • Bit to check in ParamPresenceMask - **18**
pEdrxRAT	[Optional] • eDRX Radio Access Technology • Values – NAS_RADIO_IF_NO_SVC (0x00) - None (no service) – NAS_RADIO_IF_CDMA_1X (0x01) - cdma2000 ® 1X – NAS_RADIO_IF_CDMA_1XEVDO (0x02) - cdma2000 ® HRPD (1xEV-DO) – NAS_RADIO_IF_AMPS (0x03) - AMPS – NAS_RADIO_IF_GSM (0x04) - GSM – NAS_RADIO_IF_UMTS (0x05) - UMTS – NAS_RADIO_IF_WLAN (0x06) - WLAN – NAS_RADIO_IF_GPS (0x07) - GPS – NAS_RADIO_IF_LTE (0x08) - LTE – NAS_RADIO_IF_TDSCDMA (0x09) - TD-SCDMA – NAS_RADIO_IF_LTE_M1 (0x0a) - LTE-M1 – NAS_RADIO_IF_LTE_NB1 (0x0b) - LTE-NB1 – NAS_RADIO_IF_NO_CHANGE (-1) - No change • Bit to check in ParamPresenceMask - **19**
pLteOpMode	[Optional] • LTE Operational Mode • Values – NAS_CIOT_SYS_MODE_NO_ SRV (0x00) - No service – NAS_CIOT_SYS_MODE_LTE_ WB (0x01) - Camped on LTE wideband – NAS_CIOT_SYS_MODE_LTE_ M1 (0x02) - Camped on LTE M1 – NAS_CIOT_SYS_MODE_LTE_ NB1 (0x03) - Camped on LTE NB1 • Bit to check in ParamPresenceMask - **20**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.970.2 Field Documentation

8.970.2.1 swi_uint256_t unpack_nas_SLQSNASGeteDRXParamsExt_t::ParamPresenceMask

8.970.2.2 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pCycleLen

8.970.2.3 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pEdrxEnable

8.970.2.4 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pEdrxRAT

8.970.2.5 uint32_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pLteOpMode

8.970.2.6 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pPagingTimeWindow

8.971 unpack_nas_SLQSNASGetForbiddenNetworks_t Struct Reference

Data Fields

- nas_ForbiddenNetworks3GPP * pForbiddenNetworks3GPP
- swi_uint256_t ParamPresenceMask

8.971.1 Detailed Description

This structure contains the SLQSNASGeteDRXParams request parameters.

Parameters

pForbidden-Networks3GPP	[Optional] • See nas_ForbiddenNetworks3GPP for more information. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.971.2 Field Documentation

8.971.2.1 swi_uint256_t unpack_nas_SLQSNASGetForbiddenNetworks_t::ParamPresenceMask

8.971.2.2 nas_ForbiddenNetworks3GPP* unpack_nas_SLQSNASGetForbiddenNetworks_t::pForbiddenNetworks3GPP

8.972 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference

Data Fields

- uint8_t * pColorCode
- swi_uint256_t ParamPresenceMask

8.972.1 Detailed Description

Structure for storing the current preferred system selection settings for the device.

Parameters

pColorCode	[Optional] • Color code value • Color code corresponding to the sector to which the AT is sending the access probe • See 3GPP2 C.S0024-B V3.0, Section 7.11.6.2.1 for more information. – 0xFF - Not Available • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.972.2 Field Documentation

8.972.2.1 swi_uint256_t unpack_nas_SLQSNasGetHDRColorCode_t::ParamPresenceMask

8.972.2.2 uint8_t* unpack_nas_SLQSNasGetHDRColorCode_t::pColorCode

8.973 unpack_nas_SLQSNasGetRFInfo_t Struct Reference

Data Fields

- nas_RfBandInfoList rfbandInfoList
- nas_RfDedicatedBandInfo * pRfDedicatedBandInfo
- nas_RfBandInfoExtFormat * pRfBandInfoExtFormat
- nas_RfBandwidthInfo * pRfBandwidthInfo
- nas_LTEOperationMode * pLTEOperationMode
- uint16_t Tlvresult

8.973.1 Detailed Description

This structure contains the SLQSNasGetRFInfo response parameters.

Parameters

rfbandInfoList	• See nas_RfBandInfoList for more information
pRfDedicated-BandInfo	[Optional] • See nas_RfDedicatedBandInfo for more information
pRfBandInfoExt-Format	[Optional] • See nas_RfBandInfoExtFormat for more information
pRfBandwidth-Info	[Optional] • See nas_RfBandwidthInfo for more information
pLTEOpration-Mode	[Optional] • See nas_LTEOperationMode for more information

8.973.2 Field Documentation

8.973.2.1 nas_LTEOperationMode∗ unpack_nas_SLQSNasGetRFInfo_t::pLTEOperationMode

8.973.2.2 nas_RfBandInfoExtFormat∗ unpack_nas_SLQSNasGetRFInfo_t::pRfBandInfoExtFormat

8.973.2.3 nas_RfBandwidthInfo∗ unpack_nas_SLQSNasGetRFInfo_t::pRfBandwidthInfo

8.973.2.4 nas_RfDedicatedBandInfo∗ unpack_nas_SLQSNasGetRFInfo_t::pRfDedicatedBandInfo

8.973.2.5 nas_RfBandInfoList unpack_nas_SLQSNasGetRFInfo_t::rfbandInfoList

8.973.2.6 uint16_t unpack_nas_SLQSNasGetRFInfo_t::TlvResult

8.974 unpack_nas_SLQSNasGetSigInfo_t Struct Reference

Data Fields

- cdmaSSInfo CDMASSInfo
- hdrSSInfo HDRSSInfo
- int8_t GSMSSInfo
- cdmaSSInfo WCDMASSInfo
- lteSSInfo LTESSInfo
- swi_uint256_t ParamPresenceMask

8.974.1 Detailed Description

This structure contains the SLQSNasGetSigInfo response parameters.

Parameters

CDMASSInfo	• See cdmaSSInfo for more information • Bit to check in ParamPresenceMask - **16**
HDRSSInfo	• See hdrSSInfo for more information • Bit to check in ParamPresenceMask - **17**
GSMSSInfo	• GSM signal strength is the RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal. • Bit to check in ParamPresenceMask - **18**
WCDMASSInfo	• See cdmaSSInfo for more information • Bit to check in ParamPresenceMask - **19**
LTESSInfo	• See lteSSInfo for more information • Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.974.2 Field Documentation

8.974.2.1 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::CDMASSInfo

8.974.2.2 int8_t unpack_nas_SLQSNasGetSigInfo_t::GSMSSInfo

8.974.2.3 hdrSSInfo unpack_nas_SLQSNasGetSigInfo_t::HDRSSInfo

8.974.2.4 lteSSInfo unpack_nas_SLQSNasGetSigInfo_t::LTESSInfo

8.974.2.5 swi_uint256_t unpack_nas_SLQSNasGetSigInfo_t::ParamPresenceMask

8.974.2.6 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::WCDMASSInfo

8.975 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- nas_rxInfo * pRXChain0Info
- nas_rxInfo * pRXChain1Info
- nas_txInfo * pTXInfo
- swi_uint256_t ParamPresenceMask

8.975.1 Detailed Description

Structure for TxRx Info unpack

Parameters

pRXChain0Info	[Optional] • See nas_rxInfo for more information. • Bit to check in ParamPresenceMask - **16**
pRXChain1Info	[Optional] • See nas_rxInfo for more information. • Bit to check in ParamPresenceMask - **17**
pTXInfo	[Optional] • See nas_txInfo for more information. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.975.2 Field Documentation

8.975.2.1 swi_uint256_t unpack_nas_SLQSNasGetTxRxInfo_t::ParamPresenceMask

8.975.2.2 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain0Info

8.975.2.3 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain1Info

8.975.2.4 nas_txInfo∗ unpack_nas_SLQSNasGetTxRxInfo_t::pTXInfo

8.976 unpack_nas_SLQSNasNetworkRejectCallback_Ind_t Struct Reference

Data Fields

- uint8_t radioIf
- LITE_TYPE_OF_SERVICE_DOMAIN serviceDomain
- uint8_t rejectCause
- nas_PlmnID ∗ pPlmnId
- nas_CsgId ∗ pCsgId
- nas_LteOpMode ∗ pLteOpMode
- swi_uint256_t ParamPresenceMask

8.976.1 Detailed Description

Structure for storing the network reject indication parameters.

Parameters

radioIf	• Radio interface from which to get the information. Values: – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TDSCDMA • Bit to check in ParamPresenceMask - **1**
serviceDomain	• Type of service domain in which the registration is rejected. Values: – 0x00 - LITE_SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - LITE_SYS_SRV_DOMAIN_CS_ONLY - Circuit-switched only – 0x02 - LITE_SYS_SRV_DOMAIN_PS_ONLY - Packet-switched only – 0x03 - LITE_SYS_SRV_DOMAIN_CS_PS - Circuit-switched and packet-switched – 0x04 - LITE_SYS_SRV_DOMAIN_CAMPED - Camped • Bit to check in ParamPresenceMask - **2**

rejectCause	• Reject cause values. – 2 - IMSI unknown in HLR – 3 - Illegal MS – 4 - IMSI unknown in VLR – 5 - IMEI not accepted – 6 - Illegal ME – 7 - EPS services not allowed – 8 - EPS services and non-EPS services not allowed – 9 - UE identity cannot be derived by the network – 10 - Implicitly detached – 11 - PLMN not allowed – 12 - Location Area not allowed – 13 - Roaming not allowed in this location area – 14 - EPS services not allowed in this PLMN – 15 - No Suitable Cells In Location Area – 16 - MSC temporarily not reachable – 17 - Network failure – 18 - CS domain not available – 19 - ESM failure – 20 - MAC failure – 21 - Synch failure – 22 - Congestion – 23 - GSM authentication unacceptable – 24 - Security mode rejected, unspecified – 25 - Not authorized for this CSG – 26 - Non-EPS authentication unacceptable – 32 - Service option not supported – 33 - Requested service option not subscribed – 34 - Service option temporarily out of order – 38 - Call cannot be identified – 48 to 63 - retry upon entry into a new cell – 95 - Semantically incorrect message – 96 - Invalid mandatory information – 97 - Message type non-existent or not implemented – 98 - Message type not compatible with the protocol state – 99 - Information element non-existent or not implemented – 100 - Conditional IE error – 101 - Message not compatible with the protocol state – 111 - Protocol error, unspecified • Bit to check in ParamPresenceMask - **3**

Note

Any other value received by the mobile station shall be treated as 34, 'Service option temporarily out of order'.. Any other value received by the network shall be treated as 111, 'Protocol error, unspecified'.
See 3GPP TS 24.008, Section 4.4.4.7 and Section 10.5.3.6 See qaGobiApiTableCallEndReasons.h for Call End reasons

Parameters

pPlmnId	(optional) • See nas_PlmnID for more information • Bit to check in ParamPresenceMask - **16**
pCsgId	(optional) • See nas_CsgId for more information. • Bit to check in ParamPresenceMask - **17**
pLteOpMode	(optional) • See nas_LteOpMode for more information. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.976.2 Field Documentation

8.976.2.1 swi_uint256_t unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::ParamPresenceMask

8.976.2.2 nas_CsgId* unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::pCsgId

8.976.2.3 nas_LteOpMode* unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::pLteOpMode

8.976.2.4 nas_PlmnID* unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::pPlmnId

8.976.2.5 uint8_t unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::radioIf

8.976.2.6 uint8_t unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::rejectCause

8.976.2.7 LITE_TYPE_OF_SERVICE_DOMAIN unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::serviceDomain

8.977 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference

Data Fields

- nas_UniversalTime universalTime
- uint8_t * pTimeZone
- uint8_t * pDayltSavAdj
- uint8_t * pRadioInterface
- swi_uint256_t ParamPresenceMask

8.977.1 Detailed Description

Structure for storing the NAS Network Time indication parameters.

Parameters

universalTime	• See nas_UniversalTime for more information. • Bit to check in ParamPresenceMask - **1**

pTimeZone	• Time Zone. • Offset from Universal time, i.e., the difference between local time and Universal time, in increments of 15 min (signed value). • Bit to check in ParamPresenceMask - **16**
pDayltSavAdj	• Daylight Saving Adjustment. • Daylight saving adjustment in hr. – Possible values: 0, 1, and 2. • Bit to check in ParamPresenceMask - **17**
pRadioInterface	• Radio interface from which the information comes • Values – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA -TD-SCDMA • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.977.2 Field Documentation

8.977.2.1 swi_uint256_t unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::ParamPresenceMask

8.977.2.2 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pDayltSavAdj

8.977.2.3 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pRadioInterface

8.977.2.4 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pTimeZone

8.977.2.5 nas_UniversalTime unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::universalTime

8.978 unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t Struct Reference

Data Fields

- nas_NetworkScanStatusTlv networkScanStatus
- nas_NetworkScan3GPPInfoTlv network3GPPScanInfo
- nas_CSGInfoTlv CSGInfo
- nas_NetworkNameSrcTlv networkNameSrc
- nas_SignalQualityTlv signalQuality
- nas_SignalStrngthTlv signalStrength
- swi_uint256_t ParamPresenceMask

8.978.1 Detailed Description

Contain the parameters passed for SLQSNasPerformIncNetworkScanCallback by the device.

Parameters

networkScanStatus	• See nas_NetworkScanStatusTlv for more information. • Bit to check in ParamPresenceMask - **1**
3GPPNetworkScanInfo	• See nas_NetworkScan3GPPInfoTlv for more information. • Bit to check in ParamPresenceMask - **16**
CSGInfo	• See nas_CSGInfoTlv for more information. • Bit to check in ParamPresenceMask - **17**
networkNameSrc	• See nas_NetworkNameSrcTlv for more information. • Bit to check in ParamPresenceMask - **18**
signalQuality	• See nas_SignalQualityTlv for more information. • Bit to check in ParamPresenceMask - **19**
signalStrength	• See nas_SignalStrngthTlv for more information. • Bit to check in ParamPresenceMask - **20**

Note

None

8.978.2 Field Documentation

8.978.2.1 nas_CSGInfoTlv unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t::CSGInfo

8.978.2.2 nas_NetworkScan3GPPInfoTlv unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t::network3GPPScanInfo

8.978.2.3 nas_NetworkNameSrcTlv unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t::networkNameSrc

8.978.2.4 nas_NetworkScanStatusTlv unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t::networkScanStatus

8.978.2.5 swi_uint256_t unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t::ParamPresenceMask

8.978.2.6 nas_SignalQualityTlv unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t::signalQuality

8.978.2.7 nas_SignalStrngthTlv unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t::signalStrength

8.979 unpack_nas_SLQSNasRFBandInfoCallback_Ind_t Struct Reference

Data Fields

- nas_RFBandInfoTlv rfBandInfo
- nas_RFDedicatedBandInfoTlv rfDedicatedBandInfo
- nas_RFBandInfoExtTlv rfBandInfoExt
- nas_RFBandwidthInfoTlv rfBandwidthInfo
- nas_LTEOperationalModeTlv lteOperationalMode
- swi_uint256_t ParamPresenceMask

8.979.1 Detailed Description

Contain the parameters passed for unpack_nas_SLQSNasRFBandInfoCallback_Ind by the device.

Parameters

rfBandInfo	• See nas_RFBandInfoTlv for more information. • Bit to check in ParamPresenceMask - **1**
rfDedicated-BandInfo	• See nas_RFDedicatedBandInfoTlv for more information.

- Bit to check in ParamPresenceMask - **16**

Parameters

rfBandInfoExt	• See nas_RFBandInfoExtTlv for more information. • Bit to check in ParamPresenceMask - **17**
rfBandwidthInfo	• See nas_RFBandwidthInfoTlv for more information. • Bit to check in ParamPresenceMask - **18**
lteOperational-Mode	• See nas_LTEOperationalModeTlv for more information. • Bit to check in ParamPresenceMask - **19**

Note

None

8.979.2 Field Documentation

8.979.2.1 nas_LTEOperationalModeTlv unpack_nas_SLQSNasRFBandInfoCallback_Ind_t::lteOperationalMode

8.979.2.2 swi_uint256_t unpack_nas_SLQSNasRFBandInfoCallback_Ind_t::ParamPresenceMask

8.979.2.3 nas_RFBandInfoTlv unpack_nas_SLQSNasRFBandInfoCallback_Ind_t::rfBandInfo

8.979.2.4 nas_RFBandInfoExtTlv unpack_nas_SLQSNasRFBandInfoCallback_Ind_t::rfBandInfoExt

8.979.2.5 nas_RFBandwidthInfoTlv unpack_nas_SLQSNasRFBandInfoCallback_Ind_t::rfBandwidthInfo

8.979.2.6 nas_RFDedicatedBandInfoTlv unpack_nas_SLQSNasRFBandInfoCallback_Ind_t::rfDedicatedBandInfo

8.980 unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference

Data Fields

- cdmaSSInfo * pCDMASigInfo
- hdrSSInfo * pHDRSigInfo
- int8_t * pGSMSigInfo
- cdmaSSInfo * pWCDMASigInfo
- lteSSInfo * pLTESigInfo
- int8_t * pRscp
- tdscdmaSigInfoExt * pTDSCDMASigInfoExt
- swi_uint256_t ParamPresenceMask

8.980.1 Detailed Description

Structure for storing the nasSigInfo indication parameters.

Parameters

pCDMASigInfo	• See cdmaSSInfo for more information. • Bit to check in ParamPresenceMask - **16**
pHDRSigInfo	• See hdrSSInfo for more information. • Bit to check in ParamPresenceMask - **17**
pGSMSigInfo	• one byte value, GSM signal strength is the RSSI in dBm (signed value). A value of -125 dBm or lower is used to indicate No Signal • Bit to check in ParamPresenceMask - **18**
pWCDMASigInfo	• See cdmaSSInfo for more information. • Bit to check in ParamPresenceMask - **19**
pLTESigInfo	• See lteSSInfo for more information. • Bit to check in ParamPresenceMask - **20**
pRscp	• RSCP of the Primary Common Control Physical Channel (PCCPCH) in dBm. Measurement range: -120 dBm to -25 dBm. • Bit to check in ParamPresenceMask - **21**
pTDSCDMASig-InfoExt	• See tdscdmaSigInfoExt for more information. • Bit to check in ParamPresenceMask - **22**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.980.2 Field Documentation

8.980.2.1 swi_uint256_t unpack_nas_SLQSNasSigInfoCallback_ind_t::ParamPresenceMask

8.980.2.2 cdmaSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pCDMASigInfo

8.980.2.3 int8_t* unpack_nas_SLQSNasSigInfoCallback_ind_t::pGSMSigInfo

8.980.2.4 hdrSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pHDRSigInfo

8.980.2.5 lteSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pLTESigInfo

8.980.2.6 int8_t* unpack_nas_SLQSNasSigInfoCallback_ind_t::pRscp

8.980.2.7 tdscdmaSigInfoExt* unpack_nas_SLQSNasSigInfoCallback_ind_t::pTDSCDMASigInfoExt

8.980.2.8 cdmaSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pWCDMASigInfo

8.981 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN * pWcdmaUARFCN
- nas_lteEARFCN * pLteEARFCN
- nas_ltePCI * pLtePCI
- swi_uint256_t ParamPresenceMask

8.981.1 Detailed Description

Structure for Get Channel Lock unpack.

Parameters

pWcdmaUARF-CN	[Optional] • See nas_wcdmaUARFCN for more information • Bit to check in ParamPresenceMask - **16**
pLteEARFCN	[Optional] • See nas_lteEARFCN for more information • Bit to check in ParamPresenceMask - **17**
pLtePCI	[Optional] • See nas_ltePCI for more information • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.981.2 Field Documentation

8.981.2.1 swi_uint256_t unpack_nas_SLQSNASSwiGetChannelLock_t::ParamPresenceMask

8.981.2.2 nas_lteEARFCN* unpack_nas_SLQSNASSwiGetChannelLock_t::pLteEARFCN

8.981.2.3 nas_ltePCI* unpack_nas_SLQSNASSwiGetChannelLock_t::pLtePCI

8.981.2.4 nas_wcdmaUARFCN* unpack_nas_SLQSNASSwiGetChannelLock_t::pWcdmaUARFCN

8.982 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference

Data Fields

- nas_CommInfo commonInfo
- nas_LTEInfo * pLTEInfo
- swi_uint256_t ParamPresenceMask

8.982.1 Detailed Description

Structure for storing the SLQS Nas Swi Modem Status response parameters.

Parameters

commonInfo	(mandatory) • See nas_CommInfo for more information • Bit to check in ParamPresenceMask - **1**
pLTEInfo	(optional) • See nas_LTEInfo for more information • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.982.2 Field Documentation

8.982.2.1 nas_CommInfo unpack_nas_SLQSNasSwiModemStatus_t::commonInfo

8.982.2.2 swi_uint256_t unpack_nas_SLQSNasSwiModemStatus_t::ParamPresenceMask

8.982.2.3 nas_LTEInfo* unpack_nas_SLQSNasSwiModemStatus_t::pLTEInfo

8.983 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSwiOTAMessageInd Info
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.983.1 Detailed Description

This structure contains unpack OTA message indication parameters.

Parameters

Info	• Structure used to store all QMI Notification Info. – See NASQmiCbkNasSwiOTAMessageInd for more details
TlvResult	• unpack result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.983.2 Field Documentation

8.983.2.1 NASQmiCbkNasSwiOTAMessageInd unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Info

8.983.2.2 swi_uint256_t unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::ParamPresenceMask

8.983.2.3 uint16_t unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::TlvResult

8.984 unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference

Data Fields

- char t3396_apn [101]
- uint8_t t3396_plmn_id [3]
- uint32_t t3396_val
- swi_uint256_t ParamPresenceMask

8.984.1 Detailed Description

Structure for Network Timer indication parameters.

Parameters

t3396_apn	• apn • Bit to check in ParamPresenceMask - **16**
t3396_plmn_id	• plmn id • Bit to check in ParamPresenceMask - **16**
t3396_val	• timer value • Bit to check in ParamPresenceMask - **16**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.984.2 Field Documentation

8.984.2.1 swi_uint256_t unpack_nas_SLQSNasTimerCallback_ind_t::ParamPresenceMask

8.984.2.2 char unpack_nas_SLQSNasTimerCallback_ind_t::t3396_apn[101]

8.984.2.3 uint8_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_plmn_id[3]

8.984.2.4 uint32_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_val

8.985 unpack_nas_SLQSPerformNetworkScanV2_t Struct Reference

Data Fields

- uint8_t * p3GppNetworkInstanceSize
- nas_QmiNas3GppNetworkInfo * p3GppNetworkInfoInstances
- uint8_t * pRATInstanceSize
- nas_QmiNas3GppNetworkRAT * pRATINstance
- uint8_t * pPCSInstanceSize
- nas_QmisNasPcsDigit * pPCSInstance
- uint32_t * pScanResult
- nas_QmisNasSlqsNasPCIInfo * pPCIInfo
- nas_lteOpModeTlv * pLteOpModeTlv
- nas_networkNameSrcTlv * pNetworkNameSrcTlv
- swi_uint256_t ParamPresenceMask

8.985.1 Detailed Description

Contain the network scan information.

Parameters

p3GppNetwork-InstanceSize	• Upon input, maximum number of elements that the network info instance array can contain. • Upon successful output, the actual number of elements in the network info instance array. • Bit to check in ParamPresenceMask - **16**
p3GppNetwork-InfoInstances	• Network info instance array – See nas_QmiNas3GppNetworkInfo for more information • Bit to check in ParamPresenceMask - **16**
pRATInstance-Size	• Upon input, maximum number of elements that the RAT info instance array can contain. • Upon successful output, the actual number of elements in the RAT info instance array. • Bit to check in ParamPresenceMask - **17**

pRATINstance	• RAT info instance array – See nas_QmiNas3GppNetworkRAT for more information • Bit to check in ParamPresenceMask - **17**
pPCSInstance-Size	• Upon input, maximum number of elements that the PCS Digit info instance array can contain. • Upon successful output, the actual number of elements in the PCS Digit info instance array. • Bit to check in ParamPresenceMask - **18**
pPCSInstance	• PCS Digit info instance array – See nas_QmisNasPcsDigit for more information • Bit to check in ParamPresenceMask - **18**
pScanResult	• status of network scan • 0x00 - scan successful • 0x01 - scan was aborted • 0x02 - scan did not complete due to a radio link failure recovery in progress • Bit to check in ParamPresenceMask - **19**
pPCIInfo	• PCI Information – See nas_QmisNasSlqsNasPCIInfo for more information • Bit to check in ParamPresenceMask - **23**
pLteOpMode-Tlv[OUT]	• LTE Operational Mode. • See nas_lteOpModeTlv for more information. • Bit to check in ParamPresenceMask - **26**
pNetworkName-SrcTlv[OUT]	• Network Name Source. • See nas_networkNameSrcTlv for more information. • Bit to check in ParamPresenceMask - **22**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.985.2 Field Documentation

8.985.2.1 nas_QmiNas3GppNetworkInfo* unpack_nas_SLQSPerformNetworkScanV2_t::p3GppNetworkInfoInstances

8.985.2.2 uint8_t* unpack_nas_SLQSPerformNetworkScanV2_t::p3GppNetworkInstanceSize

8.985.2.3 swi_uint256_t unpack_nas_SLQSPerformNetworkScanV2_t::ParamPresenceMask

8.985.2.4 nas_lteOpModeTlv* unpack_nas_SLQSPerformNetworkScanV2_t::pLteOpModeTlv

8.985.2.5 nas_networkNameSrcTlv∗ unpack_nas_SLQSPerformNetworkScanV2_t::pNetworkNameSrcTlv

8.985.2.6 nas_QmisNasSlqsNasPCIInfo∗ unpack_nas_SLQSPerformNetworkScanV2_t::pPCIInfo

8.985.2.7 nas_QmisNasPcsDigit∗ unpack_nas_SLQSPerformNetworkScanV2_t::pPCSInstance

8.985.2.8 uint8_t∗ unpack_nas_SLQSPerformNetworkScanV2_t::pPCSInstanceSize

8.985.2.9 nas_QmiNas3GppNetworkRAT∗ unpack_nas_SLQSPerformNetworkScanV2_t::pRATInstance

8.985.2.10 uint8_t∗ unpack_nas_SLQSPerformNetworkScanV2_t::pRATInstanceSize

8.985.2.11 uint32_t∗ unpack_nas_SLQSPerformNetworkScanV2_t::pScanResult

8.986 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSystemSelPrefInd Info
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.986.1 Detailed Description

This structure contains unpack set system selection preference callback indication parameters.

Parameters

Info	• Structure used to store all QMI Notification Info. – See NASQmiCbkNasSystemSelPrefInd for more details
Tlvresult	• unpack result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.986.2 Field Documentation

8.986.2.1 NASQmiCbkNasSystemSelPrefInd unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Info

8.986.2.2 swi_uint256_t unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::ParamPresenceMask

8.986.2.3 uint16_t unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Tlvresult

8.987 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference

Data Fields

- uint16_t ∗ pCurrentPersonality

- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtocolSubtypeElement
- swi_uint256_t ParamPresenceMask

8.987.1 Detailed Description

Structure for HDRPersonality unpack.

Parameters

pCurrentPersonality[Out]	• Current active personality index. • Bit to check in ParamPresenceMask - **16**
pPersonalityListLength[In/Out]	• Number of Personality Protocol Subtype contains in this response. • maximum input value is 3 • Bit to check in ParamPresenceMask - **17**
pProtocolSubtypeElement[Out]	• See nas_protocolSubtypeElement for more information. • Bit to check in ParamPresenceMask - **17**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.987.2 Field Documentation

8.987.2.1 swi_uint256_t unpack_nas_SLQSSwiGetHDRPersonality_t::ParamPresenceMask

8.987.2.2 uint16_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pCurrentPersonality

8.987.2.3 uint8_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pPersonalityListLength

8.987.2.4 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRPersonality_t::pProtocolSubtypeElement

8.988 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference

Data Fields

- uint16_t * pCurrentPrsnlty
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtoSubTypElmnt
- uint64_t * pAppSubType
- swi_uint256_t ParamPresenceMask

8.988.1 Detailed Description

Structure for HDRProtSubtype unpack.

Parameters

pCurrent-Personality	• Current active personality index. • Bit to check in ParamPresenceMask - **16**
pPersonalityList-Length	• Number of Personality Protocol Subtype contains in this response. • maximum input value is 4 • Bit to check in ParamPresenceMask - **17**
pProtocol-SubtypeElement	• See nas_protocolSubtypeElement for more information. • Bit to check in ParamPresenceMask - **17**
pAppSubType	• Stream application subtype • Application subtype for each stream • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.988.2 Field Documentation

8.988.2.1 uint64_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pAppSubType

8.988.2.2 swi_uint256_t unpack_nas_SLQSSwiGetHDRProtSubtype_t::ParamPresenceMask

8.988.2.3 uint16_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pCurrentPrsnlty

8.988.2.4 uint8_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pPersonalityListLength

8.988.2.5 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pProtoSubTypElmnt

8.989 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference

Data Fields

- nas_DRCParams * pDRCParams
- uint8_t * pUATI
- nas_PilotSetData * pPilotSetData
- swi_uint256_t ParamPresenceMask

8.989.1 Detailed Description

Structure for HRPD Stats unpack.

Parameters

pDRCParams	• See nas_DRCParams for more information. • Bit to check in ParamPresenceMask - **16**

pUATI	• A 128-bit address that includes the access terminal identifier and subnet ID • Size must be 16 bytes • Bit to check in ParamPresenceMask - **17**
pPilotSetData	• See nas_PilotSetData for more information. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.989.2 Field Documentation

8.989.2.1 swi_uint256_t unpack_nas_SLQSSwiGetHRPDStats_t::ParamPresenceMask

8.989.2.2 nas_DRCParams* unpack_nas_SLQSSwiGetHRPDStats_t::pDRCParams

8.989.2.3 nas_PilotSetData* unpack_nas_SLQSSwiGetHRPDStats_t::pPilotSetData

8.989.2.4 uint8_t* unpack_nas_SLQSSwiGetHRPDStats_t::pUATI

8.990 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference

Data Fields

- uint8_t ValidityCW0
- uint8_t CQIValueCW0
- uint8_t ValidityCW1
- uint8_t CQIValueCW1
- swi_uint256_t ParamPresenceMask

8.990.1 Detailed Description

This structure contains paramaters unpack fetch CQI parameters for LTE data session.

Parameters

ValidityCW0[OUT]	• Values – 0- Invalid. – 1- Valid. • Bit to check in ParamPresenceMask - **1**
CQIValueCW0[OUT]	• Values – Range 0 – 15 • Bit to check in ParamPresenceMask - **1**

ValidityCW1[OUT]	• Values – 0- Invalid. – 1- Valid. • Bit to check in ParamPresenceMask - **1**
CQIValueCW1[OUT]	• Values – Range 0 – 15 • Bit to check in ParamPresenceMask - **1**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.990.2 Field Documentation

8.990.2.1 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW0

8.990.2.2 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW1

8.990.2.3 swi_uint256_t unpack_nas_SLQSSwiGetLteCQI_t::ParamPresenceMask

8.990.2.4 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW0

8.990.2.5 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW1

8.991 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference

Data Fields

- nas_SccRxInfo * pSccRxInfo
- swi_uint256_t ParamPresenceMask

8.991.1 Detailed Description

This structure contains unpack get LTE Secondary carrier Rx signal level information parameters.

Parameters

pSccRxInfo	Secondary carrier Rx signal level info • See nas_SccRxInfo – Bit to check in ParamPresenceMask - **16**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.991.2 Field Documentation

8.991.2.1 swi_uint256_t unpack_nas_SLQSSwiGetLteSccRxInfo_t::ParamPresenceMask

8.991.2.2 nas_SccRxInfo* unpack_nas_SLQSSwiGetLteSccRxInfo_t::pSccRxInfo

8.992 unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t Struct Reference

Data Fields

- nas_HDRPersonality_Ind_Data * pHDRPersInd
- swi_uint256_t ParamPresenceMask

8.992.1 Detailed Description

Structure for HDR Personality indication unpack.

Parameters

pHDRPersInd	• See nas_HDRPersonality_Ind_Data for more information.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.992.2 Field Documentation

8.992.2.1 swi_uint256_t unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t::ParamPresenceMask

8.992.2.2 nas_HDRPersonality_Ind_Data* unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t::pHDRPersInd

8.993 unpack_nas_SLQSSwiNetworkDebug_t Struct Reference

Data Fields

- uint8_t * pObjectVer
- nas_NetworkStat1x * pNetworkStat1x
- nas_NetworkStatEVDO * pNetworkStatEVDO
- nas_DeviceConfigDetail * pDeviceConfigDetail
- nas_DataStatusDetail * pDataStatusDetail
- swi_uint256_t ParamPresenceMask

8.993.1 Detailed Description

Structure for SwiNetworkDebug unpack.

Parameters

pObjectVer	• Object's version number for the host to handle – 0xFF - NA – Others - shows in decimal • Bit to check in ParamPresenceMask - **16**

pNetworkStat1x	• See nas_NetworkStat1x for more information • Bit to check in ParamPresenceMask - **17**
pNetworkStatE-VDO	• See nas_NetworkStatEVDO for more information. • Bit to check in ParamPresenceMask - **18**
pDeviceConfig-Detail	• See nas_DeviceConfigDetail for more information. • Bit to check in ParamPresenceMask - **19**
pDataStatus-Detail	• See nas_DataStatusDetail for more information. • Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.993.2 Field Documentation

8.993.2.1 swi_uint256_t unpack_nas_SLQSSwiNetworkDebug_t::ParamPresenceMask

8.993.2.2 nas_DataStatusDetail* unpack_nas_SLQSSwiNetworkDebug_t::pDataStatusDetail

8.993.2.3 nas_DeviceConfigDetail* unpack_nas_SLQSSwiNetworkDebug_t::pDeviceConfigDetail

8.993.2.4 nas_NetworkStat1x* unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStat1x

8.993.2.5 nas_NetworkStatEVDO* unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStatEVDO

8.993.2.6 uint8_t* unpack_nas_SLQSSwiNetworkDebug_t::pObjectVer

8.994 unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t Struct Reference

Data Fields

- nas_RankIndicatorTlv rankIndicatorTlv
- swi_uint256_t ParamPresenceMask

8.994.1 Detailed Description

Structure for Rank Indicator indication unpack.

Parameters

rankIndicatorTlv	• See nas_RankIndicatorTlv for more information. • Bit to check in ParamPresenceMask - **1**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.994.2 Field Documentation

8.994.2.1 swi_uint256_t unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t::ParamPresenceMask

8.994.2.2 nas_RankIndicatorTlv unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t::rankIndicatorTlv

8.995 unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference

Data Fields

- nas_SrvStatusInfo * pCDMASrvStatusInfo
- nas_SrvStatusInfo * pHDRSrvStatusInfo
- nas_GSMSrvStatusInfo * pGSMSrvStatusInfo
- nas_GSMSrvStatusInfo * pWCDMASrvStatusInfo
- nas_GSMSrvStatusInfo * pLTESrvStatusInfo
- nas_CDMASysInfo * pCDMASysInfo
- nas_HDRSysInfo * pHDRSysInfo
- nas_GSMSysInfo * pGSMSysInfo
- nas_WCDMASysInfo * pWCDMASysInfo
- nas_LTESysInfo * pLTESysInfo
- nas_AddCDMASysInfo * pAddCDMASysInfo
- uint16_t * pAddHDRSysInfo
- nas_AddSysInfo * pAddGSMSysInfo
- nas_AddSysInfo * pAddWCDMASysInfo
- uint16_t * pAddLTESysInfo
- nas_CallBarringSysInfo * pGSMCallBarringSysInfo
- nas_CallBarringSysInfo * pWCDMACallBarringSysInfo
- uint8_t * pLTEVoiceSupportSysInfo
- uint8_t * pGSMCipherDomainSysInfo
- uint8_t * pWCDMACipherDomainSysInfo
- uint8_t * pSysInfoNoChange
- nas_LteCiotOpModeTlv * pLteCiotOpModeTlv
- nas_NR5GSerStatTlv * pNR5GSerStatTlv
- nas_NR5GSystemInfoTlv * pNR5GSystemInfoTlv
- nas_NR5GCellStatusTlv * pNR5GCellStatus
- swi_uint256_t ParamPresenceMask

8.995.1 Detailed Description

This structure contains unpack system information callback indication parameters.

Parameters

pCDMASrv-StatusInfo	• See nas_SrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **16**

pHDRSrvStatus-Info	• See nas_SrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **17**
pGSMSrvStatus-Info	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **18**
pWCDMASrv-StatusInfo	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **19**
pLTESrvStatus-Info	• See nas_GSMSrvStatusInfo for more information. • Bit to check in ParamPresenceMask - **20**
pCDMASysInfo	• See nas_CDMASysInfo for more information. • Bit to check in ParamPresenceMask - **21**
pHDRSysInfo	• See nas_HDRSysInfo for more information. • Bit to check in ParamPresenceMask - **22**
pGSMSysInfo	• See nas_GSMSysInfo for more information. • Bit to check in ParamPresenceMask - **23**
pWCDMASys-Info	• See nas_WCDMASysInfo for more information. • Bit to check in ParamPresenceMask - **24**
pLTESysInfo	• See nas_LTESysInfo for more information. • Bit to check in ParamPresenceMask - **25**
pAddCDMASys-Info	• See nas_AddCDMASysInfo for more information. • Bit to check in ParamPresenceMask - **26**
pAddHDRSys-Info	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used. • Bit to check in ParamPresenceMask - **27**
pAddGSMSys-Info	• See nas_AddSysInfo for more information. • Bit to check in ParamPresenceMask - **28**

pAddWCDMA-SysInfo	• See nas_AddSysInfo for more information. • Bit to check in ParamPresenceMask - **29**
pAddLTESysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used. • Bit to check in ParamPresenceMask - **30**
pGSMCall-BarringSysInfo	• See nas_CallBarringSysInfo for more information. • Bit to check in ParamPresenceMask - **31**
pWCDMACall-BarringSysInfo	• See nas_CallBarringSysInfo for more information. • Bit to check in ParamPresenceMask - **32**
pLTEVoice-SupportSysInfo	• Indicates voice support status on LTE. – 0x00 - Voice is not supported – 0x01 - Voice is supported • Bit to check in ParamPresenceMask - **33**
pGSMCipher-DomainSysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched • Bit to check in ParamPresenceMask - **34**
pWCDMA-CipherDomain-SysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched • Bit to check in ParamPresenceMask - **35**
pSysInfoNo-Change	• System Info No Change. • Flag used to notify clients that a request to select a network ended with no change in the PLMN. – 0x01 - No change in system information • Bit to check in ParamPresenceMask - **36**
pLteCiotOp-ModeTlv	• See nas_LteCiotOpModeTlv for more information. • Bit to check in ParamPresenceMask - **75**

pNR5GSerStat-Tlv	• See nas_NR5GSerStatTlv for more information. • Bit to check in ParamPresenceMask - **76**
pNR5GSystem-InfoTlv	• See nas_NR5GSystemInfoTlv for more information. • Bit to check in ParamPresenceMask - **77**
pNR5GCell-StatusTlv	• See nas_NR5GCellStatusTlv for more information. • Bit to check in ParamPresenceMask - **78**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.995.2 Field Documentation

8.995.2.1 nas_AddCDMASysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddCDMASysInfo

8.995.2.2 nas_AddSysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddGSMSysInfo

8.995.2.3 uint16_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddHDRSysInfo

8.995.2.4 uint16_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddLTESysInfo

8.995.2.5 nas_AddSysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddWCDMASysInfo

8.995.2.6 swi_uint256_t unpack_nas_SLQSSysInfoCallback_ind_t::ParamPresenceMask

8.995.2.7 nas_SrvStatusInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASrvStatusInfo

8.995.2.8 nas_CDMASysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASysInfo

8.995.2.9 nas_CallBarringSysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCallBarringSysInfo

8.995.2.10 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCipherDomainSysInfo

8.995.2.11 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSrvStatusInfo

8.995.2.12 nas_GSMSysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSysInfo

8.995.2.13 nas_SrvStatusInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSrvStatusInfo

8.995.2.14 nas_HDRSysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSysInfo

8.995.2.15 nas_LteCiotOpModeTlv∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLteCiotOpModeTlv

8.995.2.16 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTESrvStatusInfo

8.995.2.17 nas_LTESysInfo∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTESysInfo

8.995.2.18 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTEVoiceSupportSysInfo

8.995.2.19 **nas_NR5GCellStatusTlv**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pNR5GCellStatus

8.995.2.20 **nas_NR5GSerStatTlv**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pNR5GSerStatTlv

8.995.2.21 **nas_NR5GSystemInfoTlv**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pNR5GSystemInfoTlv

8.995.2.22 **uint8_t**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pSysInfoNoChange

8.995.2.23 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACallBarringSysInfo

8.995.2.24 **uint8_t**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACipherDomainSysInfo

8.995.2.25 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASrvStatusInfo

8.995.2.26 **nas_WCDMASysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASysInfo

8.996 unpack_omaDmConfigTlv_t Struct Reference

Data Fields

- uint8_t state
- uint8_t userInputReq
- uint16_t userInputTimeout
- uint16_t alertmsglength
- uint8_t alertmsg [256]

8.996.1 Detailed Description

This structure will hold the SwiOmaDmConfig session parameters information.

Parameters

state	• 0x01 - OMA-DM Read Request • 0x02 - OMA-DM Change Request • 0x03 - OMA-DM Config Complete
userInputReq	- Bit mask of available user inputs • 0x00 - No user input required.Informational indication • 0x01 - Accept • 0x02 - Reject
userInput-Timeout	• Timeout for user input in minutes. A value of 0 means no time-out
alertmsglength	• Length of Alert message string in bytes
alertmsg	• Alert message in UCS2 (Max 256 characters)

8.996.2 Field Documentation

8.996.2.1 uint8_t unpack_omaDmConfigTlv_t::alertmsg[256]

8.996.2.2 uint16_t unpack_omaDmConfigTlv_t::alertmsglength

8.996.2.3 uint8_t unpack_omaDmConfigTlv_t::state

8.996.2.4 uint8_t unpack_omaDmConfigTlv_t::userInputReq

8.996.2.5 uint16_t unpack_omaDmConfigTlv_t::userInputTimeout

8.997 unpack_omaDmFotaTlv_t Struct Reference

Data Fields

- uint8_t state
- uint8_t userInputReq
- uint16_t userInputTimeout
- uint32_t fwdloadsize
- uint32_t fwloadComplete
- uint16_t updateCompleteStatus
- uint8_t severity
- uint16_t versionlength
- uint8_t version [256]
- uint16_t namelength
- uint8_t package_name [256]
- uint16_t descriptionlength
- uint8_t description [256]
- uint8_t sessionType

8.997.1 Detailed Description

This structure will hold the SwiOmaDmFota session parameters information.

Parameters

state	• 0x01 - No Firmware available • 0x02 - Query Firmware Download • 0x03 - Firmware Downloading • 0x04 - Firmware downloaded • 0x05 - Query Firmware Update • 0x06 - Firmware updating • 0x07 - Firmware updated
userInputReq	- Bit mask of available user inputs • 0x00 - No user input required.Informational indication • 0x01 - Accept • 0x02 - Reject
userInput-Timeout	• Timeout for user input in minutes. A value of 0 means no time-out

fwdloadsize	• The size (in bytes) of the firmware update package
fwloadComplete	• The number of bytes downloaded. Need to determine how often to send this message for progress bar notification. Every 500ms or 5% increment.
update-CompleteStatus	• This field should be looked at only when the OMADM session is complete. • See qaGobiApiTableSwiOMADMUpdateCompleteStatus.h for update complete status.
severity	• 0x01 - Mandatory • 0x02 - Optional
versionlength	• Length of FW Version string in bytes
version	• FW Version string in ASCII (Max 256 characters)
namelength	• Length Package Name string in bytes
package_name	• Package Name in UCS2 (Max 256 characters)
descriptionlength	• Length of description in bytes
description	• Description of Update Package in USC2 (Max 256 characters)
sessionType	• 0x00 - Client initiated • 0x01 - Network initiated

8.997.2 Field Documentation

8.997.2.1 uint8_t unpack_omaDmFotaTlv_t::description[256]

8.997.2.2 uint16_t unpack_omaDmFotaTlv_t::descriptionlength

8.997.2.3 uint32_t unpack_omaDmFotaTlv_t::fwdloadsize

8.997.2.4 uint32_t unpack_omaDmFotaTlv_t::fwloadComplete

8.997.2.5 uint16_t unpack_omaDmFotaTlv_t::namelength

8.997.2.6 uint8_t unpack_omaDmFotaTlv_t::package_name[256]

8.997.2.7 uint8_t unpack_omaDmFotaTlv_t::sessionType

8.997.2.8 uint8_t unpack_omaDmFotaTlv_t::severity

8.997.2.9 uint8_t unpack_omaDmFotaTlv_t::state

8.997.2.10 uint16_t unpack_omaDmFotaTlv_t::updateCompleteStatus

8.997.2.11 uint8_t unpack_omaDmFotaTlv_t::userInputReq

8.997.2.12 uint16_t unpack_omaDmFotaTlv_t::userInputTimeout

8.997.2.13 uint8_t unpack_omaDmFotaTlv_t::version[256]

8.997.2.14 uint16_t unpack_omaDmFotaTlv_t::versionlength

8.998 unpack_omaDmNotificationsTlv_t Struct Reference

Data Fields

- uint8_t notification
- uint16_t sessionStatus

8.998.1 Detailed Description

This structure will hold the SwiOmaDmConfig session notification parameters information.

Parameters

notification	• 0x01 - GPS settings change • 0x02 - Device reset • 0x03 - Device factory reset • 0x04 - CI-DC Session start • 0x05 - CI-DC Session end • 0x06 - CI-PRL Session start • 0x07 - CI-PRL Session end • 0x08 - CI-FUMO Session start • 0x09 - CI-FUMO session end • 0x0A - HFA-DC Session start • 0x0B - HFA-DC Session end • 0x0C - HFA-PRL Session start • 0x0D - HFA-PRL Session end • 0x0E - HFA-FUMO Session start • 0x0F - HFA-FUMO session end • 0x10 - NI Session start • 0x11 - NI session end
sessionStatus	• This field will set to the session status for notifications that occur at the end of a session, zero for all other notifications • See qaGobiApiTableSwiOMADMSessionStatus.h for session completion code.

8.998.2 Field Documentation

8.998.2.1 uint8_t unpack_omaDmNotificationsTlv_t::notification

8.998.2.2 uint16_t unpack_omaDmNotificationsTlv_t::sessionStatus

8.999 unpack_pds_GetPDSDefaults_t Struct Reference

Data Fields

- uint32_t * pOperation
- uint8_t * pTimeout
- uint32_t * pInterval
- uint32_t * pAccuracy
- swi_uint256_t ParamPresenceMask

8.999.1 Detailed Description

Structure contain session configuration parameters.

Parameters

pOperation	• Current session operating mode – 0 - Standalone – 1 - MS based – 2 - MS assisted • Bit to check in ParamPresenceMask - **1**
pTimeout	• Maximum amount of time (seconds) to work on each fix, maximum is 255 • Bit to check in ParamPresenceMask - **1**
pInterval	• Interval (seconds) between fix requests • Bit to check in ParamPresenceMask - **1**
pAccuracy	• Preferred accuracy threshold (meters) • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.999.2 Field Documentation

8.999.2.1 uint32_t* unpack_pds_GetPDSDefaults_t::pAccuracy

8.999.2.2 swi_uint256_t unpack_pds_GetPDSDefaults_t::ParamPresenceMask

8.999.2.3 uint32_t* unpack_pds_GetPDSDefaults_t::pInterval

8.999.2.4 uint32_t* unpack_pds_GetPDSDefaults_t::pOperation

8.999.2.5 uint8_t* unpack_pds_GetPDSDefaults_t::pTimeout

8.1000 unpack_pds_GetPDSState_t Struct Reference

Data Fields

- uint32_t * pEnabledStatus
- uint32_t * pTrackingStatus
- swi_uint256_t ParamPresenceMask

8.1000.1 Detailed Description

Structure contain parameters of current PDS state.

Parameters

pEnabledStatus	• Current PDS state – 0 - disable – 1 - enable • Bit to check in ParamPresenceMask - **1**
pTrackingStatus	• Current PDS tracking session state • Values: – 0x00 - Unknown – 0x01 - Inactive – 0x02 - Active • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1000.2 Field Documentation

8.1000.2.1 swi_uint256_t unpack_pds_GetPDSState_t::ParamPresenceMask

8.1000.2.2 uint32_t* unpack_pds_GetPDSState_t::pEnabledStatus

8.1000.2.3 uint32_t* unpack_pds_GetPDSState_t::pTrackingStatus

8.1001 unpack_pds_GetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t * pbAuto
- swi_uint256_t ParamPresenceMask

8.1001.1 Detailed Description

Structure contain the parameter of automatic tracking configuration for the NMEA COM port.

Parameters

pbAuto	• Automatic tracking enabled for NMEA COM port – 0x00 - Disabled – 0x01 - Enabled – Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1001.2 Field Documentation

8.1001.2.1 swi_uint256_t unpack_pds_GetPortAutomaticTracking_t::ParamPresenceMask

8.1001.2.2 uint32_t* unpack_pds_GetPortAutomaticTracking_t::pbAuto

8.1002 unpack_pds_GetServiceAutomaticTracking_t Struct Reference

Data Fields

- uint32_t * pbAuto
- swi_uint256_t ParamPresenceMask

8.1002.1 Detailed Description

Structure contain the parameter for automatic tracking state.

Parameters

pbAuto	• Automatic tracking session started for service – 0x00 - Disabled – 0x01 - Enabled • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1002.2 Field Documentation

8.1002.2.1 swi_uint256_t unpack_pds_GetServiceAutomaticTracking_t::ParamPresenceMask

8.1002.2.2 uint32_t* unpack_pds_GetServiceAutomaticTracking_t::pbAuto

8.1003 unpack_pds_GetXTRAAutomaticDownload_t Struct Reference

Data Fields

- uint32_t * pbEnabled
- uint16_t * pInterval
- swi_uint256_t ParamPresenceMask

8.1003.1 Detailed Description

Structure contain the parameter of automatic tracking configuration for the NMEA COM port.

Parameters

pbEnabled	• Automatic XTRA download status – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **16**
pInterval	• Interval (hours) between XTRA downloads • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1003.2 Field Documentation

8.1003.2.1 swi_uint256_t unpack_pds_GetXTRAAutomaticDownload_t::ParamPresenceMask

8.1003.2.2 uint32_t* unpack_pds_GetXTRAAutomaticDownload_t::pbEnabled

8.1003.2.3 uint16_t* unpack_pds_GetXTRAAutomaticDownload_t::pInterval

8.1004 unpack_pds_GetXTRANetwork_t Struct Reference

Data Fields

- uint32_t * pPreference
- swi_uint256_t ParamPresenceMask

8.1004.1 Detailed Description

Structure contain the parameter for XTRA WWAN network preference

Parameters

pPreference	• XTRA WWAN network preference – 0x00 - None (any available network) – 0x01 - Home-only, only when on home systems – 0x02 - Roam-only, only when on non-home systems • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1004.2 Field Documentation

8.1004.2.1 swi_uint256_t unpack_pds_GetXTRANetwork_t::ParamPresenceMask

8.1004.2.2 uint32_t* unpack_pds_GetXTRANetwork_t::pPreference

8.1005 unpack_pds_GetXTRAValidity_t Struct Reference

Data Fields

- uint16_t * pGPSWeek
- uint16_t * pGPSWeekOffset
- uint16_t * pDuration
- swi_uint256_t ParamPresenceMask

8.1005.1 Detailed Description

Structure contain the parameter for XTRA database validity period

Parameters

pGPSWeek	• Starting GPS week of validity period • Bit to check in ParamPresenceMask - **19**
pGPSWeek-Offset	• Starting GPS week offset (minutes) of validity period • Bit to check in ParamPresenceMask - **19**
pDuration	• Length of validity period (hours) • NULL pointer - Invalid data. • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1005.2 Field Documentation

8.1005.2.1 swi_uint256_t unpack_pds_GetXTRAValidity_t::ParamPresenceMask

8.1005.2.2 uint16_t* unpack_pds_GetXTRAValidity_t::pDuration

8.1005.2.3 uint16_t* unpack_pds_GetXTRAValidity_t::pGPSWeek

8.1005.2.4 uint16_t* unpack_pds_GetXTRAValidity_t::pGPSWeekOffset

8.1006 unpack_pds_SetEventReport_Ind_t Struct Reference

Data Fields

- uint8_t has_PositionDataNMEA
- char PositionDataNMEA [200]
- uint8_t has_dLongitude
- double dLongitude
- uint8_t has_dLatitude
- double dLatitude
- uint8_t has_SessionStatus
- uint8_t SessionStatus
- uint8_t has_posSrc
- uint32_t posSrc
- swi_uint256_t ParamPresenceMask

8.1006.1 Detailed Description

Structure contain the parameter for the Set event report Indication. Please check has_<Param_Name> field for presence of optional parameters

Parameters

PositionDataNMEA	• Position Data NMEA. • String containing the position data in NMEA sentence format (maximum 200 bytes). • Bit to check in ParamPresenceMask - **16**
dLongitude	• Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of the Greenwich Meridian. • Units - Decimal degrees • Range - -180 to +180 • Note - Value is in double float format • Bit to check in ParamPresenceMask - **19**
dLatitude	• Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. • Units: Decimal degrees • Range: -90 to +90 • Note - Value is in double float format • Bit to check in ParamPresenceMask - **19**

session_status	• Position Session Status • Status of current session. • Values – 0x00 - Success – 0x01 - In progress – 0x02 - General failure – 0x03 - Timeout – 0x04 - User ended the session – 0x05 - Bad parameter – 0x06 - Phone is offline – 0x07 - Engine is locked – 0x08 - E911 session in progress • Bit to check in ParamPresenceMask - **18** • Note - This TLV is always sent while a fix is being generated or if it halts due to an error. The Parsed Position Data TLV (0x13) is included if the session status is Success (0x00) or In Progress (0x01); otherwise this TLV is passed alone.
posSrc	• Position Source • This TLV is always sent with the Parsed Position Data TLV (0x13) when the latitude/longitude is marked as valid. Source of the position (bitmask). • Values – 0x00000001 - GPS – 0x00000002 - Cell ID – 0x00000004 - GLONASS – 0x00000008 - Network – 0x00000010 - External position injection • Bit to check in ParamPresenceMask - **28**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1006.2 Field Documentation

8.1006.2.1 double unpack_pds_SetEventReport_Ind_t::dLatitude

8.1006.2.2 double unpack_pds_SetEventReport_Ind_t::dLongitude

8.1006.2.3 uint8_t unpack_pds_SetEventReport_Ind_t::has_dLatitude

8.1006.2.4 uint8_t unpack_pds_SetEventReport_Ind_t::has_dLongitude

8.1006.2.5 uint8_t unpack_pds_SetEventReport_Ind_t::has_PositionDataNMEA

8.1006.2.6 uint8_t unpack_pds_SetEventReport_Ind_t::has_posSrc

8.1006.2.7 uint8_t unpack_pds_SetEventReport_Ind_t::has_SessionStatus

8.1006.2.8 swi_uint256_t unpack_pds_SetEventReport_Ind_t::ParamPresenceMask

8.1006.2.9 char unpack_pds_SetEventReport_Ind_t::PositionDataNMEA[200]

8.1006.2.10 uint32_t unpack_pds_SetEventReport_Ind_t::posSrc

8.1006.2.11 uint8_t unpack_pds_SetEventReport_Ind_t::SessionStatus

8.1007 unpack_pds_SetPdsState_Ind_t Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t EnabledStatus
- uint32_t TrackingStatus
- swi_uint256_t ParamPresenceMask

8.1007.1 Detailed Description

Structure contain the parameter for the Set PDS State Indication.

Parameters

EnabledStatus	• GPS service state. • Values – 0x00 - Disable – 0x01 - Enable • Bit to check in ParamPresenceMask - **1**
TrackingStatus	• Tracking session state. • Values – 0x00 - Unknown – 0x01 - Inactive – 0x02 - Active • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1007.2 Field Documentation

8.1007.2.1 uint32_t unpack_pds_SetPdsState_Ind_t::EnabledStatus

8.1007.2.2 swi_uint256_t unpack_pds_SetPdsState_Ind_t::ParamPresenceMask

8.1007.2.3 uint16_t unpack_pds_SetPdsState_Ind_t::TlvPresent

8.1007.2.4 uint32_t unpack_pds_SetPdsState_Ind_t::TrackingStatus

8.1008 unpack_pds_SLQSGetAGPSConfig_t Struct Reference

Data Fields

- uint32_t * pServerAddress
- uint32_t * pServerPort
- uint8_t * pServerURL
- uint8_t * pServerURLLength
- swi_uint256_t ParamPresenceMask

8.1008.1 Detailed Description

Structure contain the parameter for PDS AGPS (MS-based) configuration unpack.

Parameters

pServerAddress	• IPv4 address of AGPS server. “0” if not set • Bit to check in ParamPresenceMask - **16**
pServerPort	• Port number of AGPS server. “0” if not set • Bit to check in ParamPresenceMask - **16**
pServerURL	• URL of the AGPS server. “0” if not set • Bit to check in ParamPresenceMask - **17**
pServerURL-Length	• URL length of AGPS server. “0” if not set • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1008.2 Field Documentation

8.1008.2.1 swi_uint256_t unpack_pds_SLQSGetAGPSConfig_t::ParamPresenceMask

8.1008.2.2 uint32_t* unpack_pds_SLQSGetAGPSConfig_t::pServerAddress

8.1008.2.3 uint32_t* unpack_pds_SLQSGetAGPSConfig_t::pServerPort

8.1008.2.4 uint8_t* unpack_pds_SLQSGetAGPSConfig_t::pServerURL

8.1008.2.5 uint8_t* unpack_pds_SLQSGetAGPSConfig_t::pServerURLLength

8.1009 unpack_pds_SLQSGetGPSStateInfo_t Struct Reference

Data Fields

- uint8_t EngineState
- uint32_t ValidMask
- uint64_t Latitude
- uint64_t Longitude

- uint32_t HorizontalUncertainty
- uint32_t Altitude
- uint32_t VerticalUncertainty
- uint32_t TimeStmp_tow_ms
- uint16_t TimeStmp_gps_week
- uint32_t Time_uncert_ms
- uint8_t Iono_valid
- uint32_t gps_ephemeris_sv_msk
- uint32_t gps_almanac_sv_msk
- uint32_t gps_health_sv_msk
- uint32_t gps_visible_sv_msk
- uint32_t glo_ephemeris_sv_msk
- uint32_t glo_almanac_sv_msk
- uint32_t glo_health_sv_msk
- uint32_t glo_visible_sv_msk
- uint32_t sbas_ephemeris_sv_msk
- uint32_t sbas_almanac_sv_msk
- uint32_t sbas_health_sv_msk
- uint32_t sbas_visible_sv_msk
- uint16_t xtra_start_gps_week
- uint16_t xtra_start_gps_minutes
- uint16_t xtra_valid_duration_hours
- swi_uint256_t ParamPresenceMask

8.1009.1 Detailed Description

Structure contain the parameter for GPS state Info.

Parameters

EngineState	• Values: – 0 - OFF – 1 - ON • This field is always valid • Bit to check in ParamPresenceMask - **16**
ValidMask	• Mask of valid state information data. • Values: – 0x00000001 - Position(latitude/longitude/horizontal uncertainty) – 0x00000002 - Altitude and vertical uncertainty – 0x00000004 - Time ms – 0x00000008 - Time week number – 0x00000010 - Time uncertainty – 0x00000020 - Iono validity – 0x00000040 - GPS ephemeris – 0x00000080 - GPS almanac – 0x00000100 - GPS health – 0x00000200 - GPS visible SVs – 0x00000400 - GLONASS ephemeris – 0x00000800 - GLONASS almanac – 0x00001000 - GLONASS health – 0x00002000 - GLONASS visible SVs – 0x00004000 - SBAS ephemeris – 0x00008000 - SBAS almanac – 0x00010000 - SBAS health – 0x00020000 - SBAS visible SVs – 0x00040000 - XTRA information • Bit to check in ParamPresenceMask - **16**
Latitude	• Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. • Units: Decimal degrees • Range: -90 to +90 degrees. • Value is in double float format (refer to IEEE Std 754-1985) • Bit to check in ParamPresenceMask - **16**
Longitude	• Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of Greenwich meridian. • Units: Decimal degrees • Range: -180 to +180 degrees • Value is in double float format (refer to IEEE Std 754-1985) • Bit to check in ParamPresenceMask - **16**

Horizontal-Uncertainty	• Circular horizontal uncertainty (in meters). The uncertainty is provided at 63 percent confidence. • Value is in single float format (refer to IEEE Std 754-1985) • Bit to check in ParamPresenceMask - **16**
Altitude	• Height above the WGS-84 reference ellipsoid. Value conveys height (in meters) plus 500 m • Range -500 to 15883 • Value in single float format (refer to IEEE Std 754-1985) • Bit to check in ParamPresenceMask - **16**
Vertical-Uncertainty	• Vertical uncertainty (in meters). The uncertainty is provided at 68 percent confidence. • Value in single float format (refer to IEEE Std 754-1985) • Bit to check in ParamPresenceMask - **16**
TimeStmp_tow_ms	• Time stamp in GPS time of week(in milliseconds) • Bit to check in ParamPresenceMask - **16**
TimeStmp_gps_week	• GPS week number • Bit to check in ParamPresenceMask - **16**
Time_uncert_ms	• Time uncertainty (in milliseconds). The uncertainty is provided at 99 percent confidence. • Bit to check in ParamPresenceMask - **16**
Iono_valid	• Iono validity. • Values: – 0 - Invalid – 1 - Valid • Bit to check in ParamPresenceMask - **16**
gps_ephemeris_sv_msk	• GPS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available. • Bit to check in ParamPresenceMask - **16**
gps_almanac_sv_msk	• GPS SV mask for almanac; if the bit is set, almanac for that SV is available. • Bit to check in ParamPresenceMask - **16**
gps_health_sv_msk	• GPS SV mask for health; if the bit is set, health for that SV is available. • Bit to check in ParamPresenceMask - **16**
gps_visible_sv_msk	• GPS SV mask for visible Svs; if the bit is set, the SV is available. • Bit to check in ParamPresenceMask - **16**

glo_ephemeris_sv_msk	• GLONASS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available. • Bit to check in ParamPresenceMask - **16**
glo_almanac_sv_msk	• GLONASS SV mask for almanac; if the bit is set, almanac for that SV is available. • Bit to check in ParamPresenceMask - **16**
glo_health_sv_msk	• GLONASS SV mask for health; if the bit is set, health for that SV is available. • Bit to check in ParamPresenceMask - **16**
glo_visible_sv_msk	• GLONASS SV mask for visible SVs; if the bit is set, the SV is available. • Bit to check in ParamPresenceMask - **16**
sbas_ephemeris_sv_msk	• SBAS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available. • Bit to check in ParamPresenceMask - **16**
sbas_almanac_sv_msk	• SBAS SV mask for almanac; if the bit is set, almanac for that SV is available. • Bit to check in ParamPresenceMask - **16**
sbas_health_sv_msk	• SBAS SV mask for health; if the bit is set, health for that SV is available. • Bit to check in ParamPresenceMask - **16**
sbas_visible_sv_msk	• SBAS SV mask for visible SVs; if the bit is set, the SV is available. • Bit to check in ParamPresenceMask - **16**
xtra_start_gps_week	• Current XTRA information is valid starting from this GPS week number • Bit to check in ParamPresenceMask - **16**
xtra_start_gps_minutes	• Current XTRA information is valid starting from the GPS minutes with the GPS week • Bit to check in ParamPresenceMask - **16**
xtra_valid_duration_hours	• XTRA information is valid for this many hours starting from the specified GPS week/minutes • Bit to check in ParamPresenceMask - **16**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1009.2 Field Documentation

8.1009.2.1 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::Altitude

8.1009.2.2 uint8_t unpack_pds_SLQSGetGPSStateInfo_t::EngineState

8.1009.2.3 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_almanac_sv_msk

8.1009.2.4 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_ephemeris_sv_msk

8.1009.2.5 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_health_sv_msk

8.1009.2.6 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_visible_sv_msk

8.1009.2.7 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_almanac_sv_msk

8.1009.2.8 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_ephemeris_sv_msk

8.1009.2.9 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_health_sv_msk

8.1009.2.10 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_visible_sv_msk

8.1009.2.11 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::HorizontalUncertainty

8.1009.2.12 uint8_t unpack_pds_SLQSGetGPSStateInfo_t::Iono_valid

8.1009.2.13 uint64_t unpack_pds_SLQSGetGPSStateInfo_t::Latitude

8.1009.2.14 uint64_t unpack_pds_SLQSGetGPSStateInfo_t::Longitude

8.1009.2.15 swi_uint256_t unpack_pds_SLQSGetGPSStateInfo_t::ParamPresenceMask

8.1009.2.16 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_almanac_sv_msk

8.1009.2.17 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_ephemeris_sv_msk

8.1009.2.18 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_health_sv_msk

8.1009.2.19 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_visible_sv_msk

8.1009.2.20 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::Time_uncert_ms

8.1009.2.21 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::TimeStmp_gps_week

8.1009.2.22 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::TimeStmp_tow_ms

8.1009.2.23 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::ValidMask

8.1009.2.24 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::VerticalUncertainty

8.1009.2.25 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_start_gps_minutes

8.1009.2.26 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_start_gps_week

8.1009.2.27 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_valid_duration_hours

8.1010 unpack_qmi_t Struct Reference

Data Fields

- enum msgtype type
- uint16_t msgid
- uint16_t xid

8.1010.1 Detailed Description

qmi response context

Parameters

out	*type*	message type
out	*msgid*	message id
out	*xid*	transaction id

8.1010.2 Field Documentation

8.1010.2.1 uint16_t unpack_qmi_t::msgid

8.1010.2.2 enum msgtype unpack_qmi_t::type

8.1010.2.3 uint16_t unpack_qmi_t::xid

8.1011 unpack_qos_BindDataPort_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1011.1 Detailed Description

This structure contains unpack Qos BindDataPort.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1011.2 Field Documentation

8.1011.2.1 swi_uint256_t unpack_qos_BindDataPort_t::ParamPresenceMask

8.1012 unpack_qos_dataRate_t Struct Reference

Data Fields

- uint32_t dataRateMax
- uint32_t guaranteedRate

8.1012.1 Detailed Description

This structure contains the IP flow data rate min max

Parameters

dataRateMax	Maximum required data rate (bits per second)
guaranteedRate	Minimum guaranteed data rate (bits per second)

8.1012.2 Field Documentation

8.1012.2.1 uint32_t unpack_qos_dataRate_t::dataRateMax

8.1012.2.2 uint32_t unpack_qos_dataRate_t::guaranteedRate

8.1013 unpack_qos_IPv4Addr_t Struct Reference

Data Fields

- uint32_t addr
- uint32_t subnetMask

8.1013.1 Detailed Description

This structure contains the IPv4 filter address

Parameters

addr	IPv4 address
subnetMask	A packet matches if: • (addr and subnetMask) == (IP pkt addr & subnetMask) Callers to set up a filter with a range of source addresses, if needed; subnet mask of all 1s (255.255.255.255) specifies a single address value

8.1013.2 Field Documentation

8.1013.2.1 uint32_t unpack_qos_IPv4Addr_t::addr

8.1013.2.2 uint32_t unpack_qos_IPv4Addr_t::subnetMask

8.1014 unpack_qos_IPv6Addr_t Struct Reference

Data Fields

- uint8_t addr [16]
- uint8_t prefixLen

8.1014.1 Detailed Description

This structure contains the IPv6 filter address

Parameters

addr	IPv6 address (in network byte order); this is a 16-byte byte array (in Big-endian format)
prefixLen	IPv6 filter prefix length; can take a value between 0 and 128 Note: A packet matches if the IPv6 source address bytes until the prefix lengths are equal. Therefore prefix length can be used to set a filter with a range of source addresses. A prefix length of 128 specifies a single address value.

8.1014.2 Field Documentation

8.1014.2.1 uint8_t unpack_qos_IPv6Addr_t::addr[16]

8.1014.2.2 uint8_t unpack_qos_IPv6Addr_t::prefixLen

8.1015 unpack_qos_IPv6TrafCls_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.1015.1 Detailed Description

This structure contains the IPv6 filter traffic class

Parameters

val	The traffic class value
mask	The packet matches the traffic class filter if: (IPv6_filter_traffic_class_val and IPv6_filter_traffic_class_mask) == (Traffic class value in the IP packet > IPv6_filter_traffic_class_mask) Example: • IPv6_filter_tc_val = 00101000 • IPv6_filter_tc_mask = 11111100 Filter will compare only the first 6 bits in IPv6_filter_traffic_class with the first 6 bits in the traffic class field of the IP packet; first 6 bits in the traffic class field of the IP packet must be 001010 to match filter; last 2 bits can be anything, since they are ignored by filtering

8.1015.2 Field Documentation

8.1015.2.1 uint8_t unpack_qos_IPv6TrafCls_t::mask

8.1015.2.2 uint8_t unpack_qos_IPv6TrafCls_t::val

8.1016 unpack_qos_pktErrRate_t Struct Reference

Data Fields

- uint16_t multiplier
- uint16_t exponent

8.1016.1 Detailed Description

This structure contains the IP flow packet error rate

Parameters

multiplier	Factor m in calculating packet error rate: E = m∗10∗∗(-p)
exponent	Factor p in calculating packet error rate (see above)

8.1016.2 Field Documentation

8.1016.2.1 uint16_t unpack_qos_pktErrRate_t::exponent

8.1016.2.2 uint16_t unpack_qos_pktErrRate_t::multiplier

8.1017 unpack_qos_Port_t Struct Reference

Data Fields

- uint16_t port
- uint16_t range

8.1017.1 Detailed Description

This structure contains the Port Filter

Parameters

port	port value of the filter
range	range specifies the number of ports to be included in the filter starting from port; filter will match if port in the IP packet lies between port and (port + range) Range value of 0 implies that only one value of the port is valid, as specified by the port

8.1017.2 Field Documentation

8.1017.2.1 uint16_t unpack_qos_Port_t::port

8.1017.2.2 uint16_t unpack_qos_Port_t::range

8.1018 unpack_qos_QosFlowInfo_t Struct Reference

Data Fields

- unpack_qos_QosFlowInfoState_t QFlowState
- uint8_t is_TxQFlowGranted_Available
- unpack_qos_swiQosFlow_t TxQFlowGranted
- uint8_t is_RxQFlowGranted_Available
- unpack_qos_swiQosFlow_t RxQFlowGranted
- uint8_t NumTxFilters
- unpack_qos_swiQosFilter_t TxQFilter [25]
- uint8_t NumRxFilters
- unpack_qos_swiQosFilter_t RxQFilter [25]
- uint8_t BearerID

8.1018.1 Detailed Description

Structure with QoS flow details.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

QFlowState	• QoS flow state information, please check unpack_qos_QosFlowInfoState_t for more information
is_TxQFlowGranted_Available	• TRUE if optional TxQFlowGranted is available
TxQFlowGranted	• The Tx Qos flow granted, please check unpack_qos_swiQosFlow_t for more information
is_RxQFlowGranted_Available	• TRUE if optional RxQFlowGranted is available
RxQFlowGranted	• The Rx Qos flow granted, please check unpack_qos_swiQosFlow_t for more information
NumTxFilters	• Number of Tx filters available
TxQFilter	• The Tx Qos filter, please check unpack_qos_swiQosFilter_t for more information • See LITEQMI_MAX_QOS_FILTERS for more information
NumRxFilters	• Number of Tx filters available
RxQFilter	• The Rx Qos filter, please check unpack_qos_swiQosFilter_t for more information • See LITEQMI_MAX_QOS_FILTERS for more information
BearerID	• The bearer ID • Bearer ID or Radio Link Protocol (RLP) ID of the activated flow. • Valid Values - 0 to 16 • 0xFF - Invalid value.

8.1018.2 Field Documentation

8.1018.2.1 uint8_t unpack_qos_QosFlowInfo_t::BearerID

8.1018.2.2 uint8_t unpack_qos_QosFlowInfo_t::is_RxQFlowGranted_Available

8.1018.2.3 uint8_t unpack_qos_QosFlowInfo_t::is_TxQFlowGranted_Available

8.1018.2.4 uint8_t unpack_qos_QosFlowInfo_t::NumRxFilters

8.1018.2.5 uint8_t unpack_qos_QosFlowInfo_t::NumTxFilters

8.1018.2.6 unpack_qos_QosFlowInfoState_t unpack_qos_QosFlowInfo_t::QFlowState

8.1018.2.7 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::RxQFilter[25]

8.1018.2.8 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::RxQFlowGranted

8.1018.2.9 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::TxQFilter[25]

8.1018.2.10 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::TxQFlowGranted

8.1019 unpack_qos_QosFlowInfoState_t Struct Reference

Data Fields

- uint32_t id
- uint8_t isNewFlow
- uint8_t state

8.1019.1 Detailed Description

This structure contains QoS flow state

Parameters

id	QoS identifier
isNewFlow	• 1 - Newly added flow • 0 - Existing flow
state	This indicates that the flow that was added/modified/deleted: • 0x01 - Flow activated • 0x02 - Flow modified • 0x03 - Flow deleted • 0x04 - Flow suspended • 0x05 - Flow enabled • 0x06 - Flow disabled

8.1019.2 Field Documentation

8.1019.2.1 uint32_t unpack_qos_QosFlowInfoState_t::id

8.1019.2.2 uint8_t unpack_qos_QosFlowInfoState_t::isNewFlow

8.1019.2.3 uint8_t unpack_qos_QosFlowInfoState_t::state

8.1020 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference

Data Fields

- uint8_t NWQoSStatus
- swi_uint256_t ParamPresenceMask

8.1020.1 Detailed Description

Structure that contains the response to get NW QoS status command

Parameters

NWQoSStatus	Network QoS support status • 0 - No QoS support in network • 1 - Network supports QoS • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1020.2 Field Documentation

8.1020.2.1 uint8_t unpack_qos_SLQSQosGetNetworkStatus_t::NWQoSStatus

8.1020.2.2 swi_uint256_t unpack_qos_SLQSQosGetNetworkStatus_t::ParamPresenceMask

8.1021 unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference

Data Fields

- uint32_t apnId
- uint8_t ambr_ul
- uint8_t ambr_dl
- uint8_t ambr_ul_ext
- uint8_t ambr_dl_ext
- uint8_t ambr_ul_ext2
- uint8_t ambr_dl_ext2
- swi_uint256_t ParamPresenceMask

8.1021.1 Detailed Description

Structure that contains extra APN parameters

Parameters

apnId	• APN id • ID identifing the APN that the client would like to query the AMBR params • Bit to check in ParamPresenceMask - **3**
ambr_ul	• APN AMBR uplink • APN AMBR uplink values from 1 kbps to 8640 kbps • Bit to check in ParamPresenceMask - **3**
ambr_dl	• APN AMBR downlink • APN AMBR downlink values from 1 kbps to 8640 kbps • Bit to check in ParamPresenceMask - **3**

ambr_ul_ext	• Extended APN AMBR uplink • APN AMBR uplink values from 8700 kbps to 256 Mbps • Bit to check in ParamPresenceMask - **3**
ambr_dl_ext	• Extended APN AMBR downlink • APN AMBR downlink values from 8700 kbps to 256 Mbps • Bit to check in ParamPresenceMask - **3**
ambr_ul_ext2	• Second extended APN AMBR uplink • APN AMBR uplink values from 256 Mbps to 65280 Mbps • Bit to check in ParamPresenceMask - **3**
ambr_dl_ext2	• Second extended APN AMBR downlink • APN AMBR downlink values from 256 Mbps to 65280 Mbps • Bit to check in ParamPresenceMask - **3**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1021.2 Field Documentation

8.1021.2.1 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl

8.1021.2.2 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext

8.1021.2.3 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext2

8.1021.2.4 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul

8.1021.2.5 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext

8.1021.2.6 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext2

8.1021.2.7 uint32_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.1021.2.8 swi_uint256_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ParamPresenceMask

8.1022 unpack_qos_SLQSQosSwiReadDataStats_t Struct Reference

Data Fields

- uint32_t apnId
- uint32_t total_tx_pkt
- uint32_t total_tx_pkt_drp
- uint32_t total_rx_pkt
- uint64_t total_tx_bytes
- uint64_t total_tx_bytes_drp

- uint64_t total_rx_bytes
- uint32_t numQosFlow
- unpack_QosFlowStat_t qosFlow [10]
- swi_uint256_t ParamPresenceMask

8.1022.1 Detailed Description

Structure that contains APN data statistics

Parameters

apnId	• APN id • ID identifing the connected APN that the client would like to query the data statistic for • Bit to check in ParamPresenceMask - **3**
total_tx_pkt	• sum of all packets sent • Bit to check in ParamPresenceMask - **3**
total_tx_pkt_drp	• sum of all(TX) packets dropped • Bit to check in ParamPresenceMask - **3**
total_rx_pkt	• sum of all packets received • Bit to check in ParamPresenceMask - **3**
total_tx_bytes	• sum of all bytes sent • Bit to check in ParamPresenceMask - **3**
total_tx_bytes_drp	• sum of all(TX) bytes dropped • Bit to check in ParamPresenceMask - **3**
total_rx_bytes	• number of received bytes for the QoS flow ID • Bit to check in ParamPresenceMask - **3**
numQosFlow	• pointer to number of QoS flow Stat • Bit to check in ParamPresenceMask - **4**
qosFlow[LITEQMI_MAX_QOS_FLOW_PER_APN_STATS]	• Data statistic per QoS flow • See unpack_QosFlowStat_t for more information • See LITEQMI_MAX_QOS_FLOW_PER_APN_STATS for more information • Bit to check in ParamPresenceMask - **4**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1022.2 Field Documentation

8.1022.2.1 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::apnId

8.1022.2.2 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::numQosFlow

8.1022.2.3 swi_uint256_t unpack_qos_SLQSQosSwiReadDataStats_t::ParamPresenceMask

8.1022.2.4 unpack_QosFlowStat_t unpack_qos_SLQSQosSwiReadDataStats_t::qosFlow[10]

8.1022.2.5 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_bytes

8.1022.2.6 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_pkt

8.1022.2.7 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes

8.1022.2.8 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes_drp

8.1022.2.9 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt

8.1022.2.10 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt_drp

8.1023 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference

Data Fields

- uint8_t NumFlows
- unpack_qos_QosFlowInfo_t QosFlowInfo [8]
- swi_uint256_t ParamPresenceMask

8.1023.1 Detailed Description

Structure with QoS event details

Parameters

NumFlows	• Number of QoS flows available • Bit to check in ParamPresenceMask - **16**
QosFlowInfo	• The Qos flow details, please check unpack_qos_QosFlowInfo_t for more information • See LITEQMI_MAX_QOS_FLOWS for more information • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1023.2 Field Documentation

8.1023.2.1 uint8_t unpack_qos_SLQSSetQosEventCallback_ind_t::NumFlows

8.1023.2.2 swi_uint256_t unpack_qos_SLQSSetQosEventCallback_ind_t::ParamPresenceMask

8.1023.2.3 unpack_qos_QosFlowInfo_t unpack_qos_SLQSSetQosEventCallback_ind_t::QosFlowInfo[8]

8.1024 unpack_qos_SLQSSetQosEventCallback_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1024.1 Detailed Description

This structure contains unpack QOS SLQSSetQosEventCallback. param ParamPresenceMask

- bitmask representation to indicate valid parameters.

8.1024.2 Field Documentation

8.1024.2.1 swi_uint256_t unpack_qos_SLQSSetQosEventCallback_t::ParamPresenceMask

8.1025 unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference

Data Fields

- uint8_t status
- swi_uint256_t ParamPresenceMask

8.1025.1 Detailed Description

Structure with network's QoS status

Parameters

status	Network QoS support status • 0x00 - Current network does not support QoS • 0x01 - Current network supports QoS • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

- Technology Supported: CDMA

8.1025.2 Field Documentation

8.1025.2.1 swi_uint256_t unpack_qos_SLQSSetQosNWStatusCallback_ind_t::ParamPresenceMask

8.1025.2.2 uint8_t unpack_qos_SLQSSetQosNWStatusCallback_ind_t::status

8.1026 unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference

Data Fields

- uint16_t event
- swi_uint256_t ParamPresenceMask

8.1026.1 Detailed Description

Structure with QoS primary flow events

Parameters

event	Event which causes this indication: • 0x0001 - Primary flow QoS modify operation success • 0x0002 - Primary flow QoS modify operation failure • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1026.2 Field Documentation

8.1026.2.1 uint16_t unpack_qos_SLQSSetQosPriEventCallback_ind_t::event

8.1026.2.2 swi_uint256_t unpack_qos_SLQSSetQosPriEventCallback_ind_t::ParamPresenceMask

8.1027 unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference

Data Fields

- uint32_t id
- uint8_t status
- uint8_t event
- uint8_t reason
- swi_uint256_t ParamPresenceMask

8.1027.1 Detailed Description

Structure with QoS status indication details

Parameters

id	• Index identifying the QoS flow whose status is being reported • Bit to check in ParamPresenceMask - **1**
status	Current QoS flow status: • 0x01 - QMI_QOS_STATUS_ACTIVATED • 0x02 - QMI_QOS_STATUS_SUSPENDED • 0x03 - QMI_QOS_STATUS_GONE • Bit to check in ParamPresenceMask - **1**

event	• 0x01 - QMI_QOS_ACTIVATED_EV • 0x02 - QMI_QOS_SUSPENDED_EV • 0x03 - QMI_QOS_GONE_EV • 0x04 - QMI_QOS_MODIFY_ACCEPTED_EV • 0x05 - QMI_QOS_MODIFY_REJECTED_EV • 0x06 - QMI_QOS_INFO_CODE_UPDATED_EV • Bit to check in ParamPresenceMask - **1**
reason	• 0x01 - QMI_QOS_INVALID_PARAMS • 0x02 - QMI_QOS_INTERNAL_CALL_ENDED • 0x03 - QMI_QOS_INTERNAL_ERROR • 0x04 - QMI_QOS_INSUFFICIENT_LOCAL_Resources • 0x05 - QMI_QOS_TIMED_OUT_OPERATION • 0x06 - QMI_QOS_INTERNAL_UNKNOWN_CAUSE_CODE • 0x07 - QMI_QOS_INTERNAL_MODIFY_IN_PROGRESS • 0x08 - QMI_QOS_NOT_SUPPORTED • 0x09 - QMI_QOS_NOT_AVAILABLE • 0x0A - QMI_QOS_NOT_GUARANTEED • 0x0B - QMI_QOS_INSUFFICIENT_NETWORK_RESOURCES • 0x0C - QMI_QOS_AWARE_SYSTEM • 0x0D - QMI_QOS_UNAWARE_SYSTEM • 0x0E - QOS_REJECTED_OPERATION • 0x0F - QMI_QOS_WILL_GRANT_WHEN_QOS_RESUMED • 0x10 - QMI_QOS_NETWORK_CALL_ENDED • 0x11 - QMI_QOS_NETWORK_SERVICE_NOT_AVAILABLE • 0x12 - QMI_QOS_NETWORK_L2_LINK_RELEASED • 0x13 - QMI_QOS_NETWORK_L2_LINK_REESTAB_REJ • 0x14 - QMI_QOS_NETWORK_L2_LINK_REESTAB_IND • 0x15 - QMI_QOS_NETWORK_UNKNOWN_CAUSE_CODE • 0x16 - QMI_NETWORK_BUSY • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1027.2 Field Documentation

8.1027.2.1 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::event

8.1027.2.2 uint32_t unpack_qos_SLQSSetQosStatusCallback_ind_t::id

8.1027.2.3 swi_uint256_t unpack_qos_SLQSSetQosStatusCallback_ind_t::ParamPresenceMask

8.1027.2.4 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::reason

8.1027.2.5 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::status

8.1028 unpack_qos_swiQosFilter_t Struct Reference

Data Fields

- uint8_t index
- uint8_t version
- uint8_t is_IPv4SrcAddr_Available
- unpack_qos_IPv4Addr_t IPv4SrcAddr
- uint8_t is_IPv4DstAddr_Available
- unpack_qos_IPv4Addr_t IPv4DstAddr
- uint8_t is_NxtHdrProto_Available
- uint8_t NxtHdrProto
- uint8_t is_IPv4Tos_Available
- unpack_qos_Tos_t IPv4Tos
- uint8_t is_IPv6SrcAddr_Available
- unpack_qos_IPv6Addr_t IPv6SrcAddr
- uint8_t is_IPv6DstAddr_Available
- unpack_qos_IPv6Addr_t IPv6DstAddr
- uint8_t is_IPv6TrafCls_Available
- unpack_qos_IPv6TrafCls_t IPv6TrafCls
- uint8_t is_IPv6Label_Available
- uint32_t IPv6Label
- uint8_t is_TCPSrcPort_Available
- unpack_qos_Port_t TCPSrcPort
- uint8_t is_TCPDstPort_Available
- unpack_qos_Port_t TCPDstPort
- uint8_t is_UDPSrcPort_Available
- unpack_qos_Port_t UDPSrcPort
- uint8_t is_UDPDstPort_Available
- unpack_qos_Port_t UDPDstPort
- uint8_t is_EspSpi_Available
- uint32_t EspSpi
- uint8_t is_Precedence_Available
- uint16_t Precedence
- uint8_t is_Id_Available
- uint16_t Id
- uint8_t is_TranSrcPort_Available
- unpack_qos_Port_t TranSrcPort
- uint8_t is_TranDstPort_Available
- unpack_qos_Port_t TranDstPort

8.1028.1 Detailed Description

This structure contains the QoS Filter Request.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

index	Mandatory parameter IP filter index Integer that uniquely identifies each filter instance This TLV must be present in the request
version	Mandatory parameter IP filter version Identifies whether the filter is associated with IPv4 or IPv6; value specified also implies that only TLVs defined for that IP version, i.e., TLVs with IPv4 or IPv6 in the name, can be specified • 0x04 - IPv4 • 0x06 - Ipv6
IPv4SrcAddr	IPv4 filter soruce address See unpack_qos_IPv4Addr_t for more information • Implemented only for unsolicited indication
IPv4DstAddr	IPv4 filter destination address See unpack_qos_IPv4Addr_t for more information • Implemented only for unsolicited indication
NxtHdrProto	IP filter next header protocol This TLV must be present if any non-IP filter TLV(s) are provided If this field is specified, only IP packets belonging to specified higher layer protocol are considered when filtering The following protocols may be specified: • 0x01 = ICMP • 0x06 = TCP • 0x11 = UDP • 0x32 = ESP Note: The next header protocol field will be set to 0xFD (TCP & UDP) if a TFT is received specifying a source or destination port number, but IP next header type is not specified.
IPv4Tos	IPv4 filter type of service See unpack_qos_Tos_t for more information
IPv6SrcAddr	IPv6 filter soruce address See unpack_qos_IPv6Addr_t for more information • Implemented only for unsolicited indication
IPv6DstAddr	IPv6 filter destination address See unpack_qos_IPv6Addr_t for more information • Implemented only for unsolicited indication
IPv6TrafCls	IPv6 filter traffic class See unpack_qos_IPv6TrafCls_t for more information
IPv6Label	IPv6 flow label Packet matches the IPv6 flow label filter if: (∗pIPv6Label == flow label in the IPv6 header) • Implemented only for unsolicited indication
TCPSrcPort	TCP filter source port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
TCPDstPort	TCP filter destination port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
UDPSrcPort	UDP filter source port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
UDPDstPort	UDP filter destination port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
EspSpi	ESP filter security policy index Security policy index to uniquely identify each IP flow for filtering encrypted packets for encapsulating security payload • Implemented only for unsolicited indication
Precedence	Filter Precedence Specifies the order in which filters are applied; lower numerical value has higher precedence Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored

Id	Filter ID Unique identifier for each filter;filter ID is assigned by the modem Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored
TranSrcPort	Transport protocolfilter source port See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
UDPDstPort	Transport protocol filter destination port See unpack_qos_Port_t for more information • Implemented only for unsolicited indication

8.1028.2 Field Documentation

8.1028.2.1 uint32_t unpack_qos_swiQosFilter_t::EspSpi

8.1028.2.2 uint16_t unpack_qos_swiQosFilter_t::Id

8.1028.2.3 uint8_t unpack_qos_swiQosFilter_t::index

8.1028.2.4 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4DstAddr

8.1028.2.5 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4SrcAddr

8.1028.2.6 unpack_qos_Tos_t unpack_qos_swiQosFilter_t::IPv4Tos

8.1028.2.7 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6DstAddr

8.1028.2.8 uint32_t unpack_qos_swiQosFilter_t::IPv6Label

8.1028.2.9 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6SrcAddr

8.1028.2.10 unpack_qos_IPv6TrafCls_t unpack_qos_swiQosFilter_t::IPv6TrafCls

8.1028.2.11 uint8_t unpack_qos_swiQosFilter_t::is_EspSpi_Available

8.1028.2.12 uint8_t unpack_qos_swiQosFilter_t::is_Id_Available

8.1028.2.13 uint8_t unpack_qos_swiQosFilter_t::is_IPv4DstAddr_Available

8.1028.2.14 uint8_t unpack_qos_swiQosFilter_t::is_IPv4SrcAddr_Available

8.1028.2.15 uint8_t unpack_qos_swiQosFilter_t::is_IPv4Tos_Available

8.1028.2.16 uint8_t unpack_qos_swiQosFilter_t::is_IPv6DstAddr_Available

8.1028.2.17 uint8_t unpack_qos_swiQosFilter_t::is_IPv6Label_Available

8.1028.2.18 uint8_t unpack_qos_swiQosFilter_t::is_IPv6SrcAddr_Available

8.1028.2.19 uint8_t unpack_qos_swiQosFilter_t::is_IPv6TrafCls_Available

8.1028.2.20 uint8_t unpack_qos_swiQosFilter_t::is_NxtHdrProto_Available

8.1028.2.21 uint8_t unpack_qos_swiQosFilter_t::is_Precedence_Available

8.1028.2.22 uint8_t unpack_qos_swiQosFilter_t::is_TCPDstPort_Available

8.1028.2.23 uint8_t unpack_qos_swiQosFilter_t::is_TCPSrcPort_Available

8.1028.2.24 uint8_t unpack_qos_swiQosFilter_t::is_TranDstPort_Available

8.1028.2.25 uint8_t unpack_qos_swiQosFilter_t::is_TranSrcPort_Available

8.1028.2.26 uint8_t unpack_qos_swiQosFilter_t::is_UDPDstPort_Available

8.1028.2.27 uint8_t unpack_qos_swiQosFilter_t::is_UDPSrcPort_Available

8.1028.2.28 uint8_t unpack_qos_swiQosFilter_t::NxtHdrProto

8.1028.2.29 uint16_t unpack_qos_swiQosFilter_t::Precedence

8.1028.2.30 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPDstPort

8.1028.2.31 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPSrcPort

8.1028.2.32 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranDstPort

8.1028.2.33 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranSrcPort

8.1028.2.34 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPDstPort

8.1028.2.35 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPSrcPort

8.1028.2.36 uint8_t unpack_qos_swiQosFilter_t::version

8.1029 unpack_qos_swiQosFlow_t Struct Reference

Data Fields

- uint8_t index
- uint8_t is_ProfileId3GPP2_Available
- uint16_t ProfileId3GPP2
- uint8_t is_val_3GPP2Pri_Available
- uint8_t val_3GPP2Pri
- uint8_t is_TrafficClass_Available
- uint8_t TrafficClass
- uint8_t is_DataRate_Available
- unpack_qos_dataRate_t DataRate
- uint8_t is_TokenBucket_Available
- unpack_qos_tokenBucket_t TokenBucket
- uint8_t is_Latency_Available
- uint32_t Latency
- uint8_t is_Jitter_Available
- uint32_t Jitter
- uint8_t is_PktErrRate_Available
- unpack_qos_pktErrRate_t PktErrRate
- uint8_t is_MinPolicedPktSz_Available
- uint32_t MinPolicedPktSz
- uint8_t is_MaxAllowedPktSz_Available
- uint32_t MaxAllowedPktSz
- uint8_t is_val_3GPPResResidualBER_Available
- uint16_t val_3GPPResResidualBER

- uint8_t is_val_3GPPTraHdlPri_Available
- uint8_t val_3GPPTraHdlPri
- uint8_t is_val_3GPPImCn_Available
- uint8_t val_3GPPImCn
- uint8_t is_val_3GPPSigInd_Available
- uint8_t val_3GPPSigInd
- uint8_t is_LteQci_Available
- uint8_t LteQci

8.1029.1 Detailed Description

This structure contains the QoS Flow Request.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

index	• Mandatory parameter • IP flow index • Integer that uniquely identifies each flow instance • Unique index must be assigned by the control point to every flow_spec instance
ProfileId3GPP2	• IP flow 3GPP2 profile ID • A profile ID is shorthand for a defined set of QoS flow parameters specified by the network; to be present while requesting QoS for a CDMA device
val_3GPP2Pri	• IP flow 3GPP2 flow priority • Flow priority used by the network in case of contention between flows with same QoS; this parameter applies for CDMA devices
TrafficClass	• IP flow traffic class • Integer that designates the requested traffic class: • 0 - Conversational • 1 - Streaming • 2 - Interactive • 3 - Background
DataRate	• IP flow data rate min max • See unpack_qos_dataRate_t for more information
TokenBucket	• IP flow data rate token bucket • See unpack_qos_tokenBucket_t for more information
Latency	• IP flow latency • Maximum delay (in milliseconds) that can be tolerated by an IP packet during transfer through the wireless link

Jitter	• IP flow jitter • Difference between the maximum and minimum latency (in milliseconds) that can be tolerated by an IP packet during the transfer through the wireless link
PktErrRate	• IP flow packet error rate • See unpack_qos_pktErrRate_t for more information
MinPolicedPktSz	• IP flow minimum policed packet size • Integer that defines the minimum packet size (in bytes) that will be policed for QoS guarantees; any IP packets that are smaller than the minimum specified policed size may not receive requested QoS
MaxAllowedPktSz	• IP flow maximum allowed packet size • Integer that defines the maximum packet size (in bytes) allowed in the IP flow; any IP packets greater in size than the maximum allowed packet size are not queued for transmission
val_3GPPResResidualBER	• IP flow 3GPP residual bit error rate • residual_bit_error_rate • 0 = 5∗10-2 residual BER • 1 = 1∗10-2 residual BER • 2 = 5∗10-3 residual BER • 3 = 4∗10-3 residual BER • 4 = 1∗10-3 residual BER • 5 = 1∗10-4 residual BER • 6 = 1∗10-5 residual BER • 7 = 1∗10-6 residual BER • 8 = 6∗10-8 residual BER • Integer that indicates the undetected BER for each IP flow in the delivered packets; Tapplies only to 3GPP networks
val_3GPPTraHdlPri	• 3GPP traffic handling priority • 0 - Relative traffic handling priority 1 • 1 - Relative traffic handling priority 2 • 2 - Relative traffic handling priority 3 • Defines the relative priority of the flow; applies only to 3GPP networks
val_3GPPImCn	• IP flow 3GPP IM CN flag • IM CN subsystem signaling flag: • 0x00 - FALSE • 0x01 - TRUE • This parameter applies only to 3GPP networks

val_3GPPSigInd	• IP flow 3GPP signaling indication • 0x00 - FALSE • 0x01 - TRUE • This parameter applies only to 3GPP networks
LteQci	• LTE QoS Class Identifier • QoS Class Identifier(QCI) is a required parameter to request QoS in LTE • QCI values: – QCI value 0 requests the network to assign the appropriate QCI value – QCI values 1-4 are associated with guaranteed bitrates – QCI values 5-9 are associated with nonguaranteed bitrates, so the values specified as guaranteed and maximum bitrates are ignored

8.1029.2 Field Documentation

8.1029.2.1 unpack_qos_dataRate_t unpack_qos_swiQosFlow_t::DataRate

8.1029.2.2 uint8_t unpack_qos_swiQosFlow_t::index

8.1029.2.3 uint8_t unpack_qos_swiQosFlow_t::is_DataRate_Available

8.1029.2.4 uint8_t unpack_qos_swiQosFlow_t::is_Jitter_Available

8.1029.2.5 uint8_t unpack_qos_swiQosFlow_t::is_Latency_Available

8.1029.2.6 uint8_t unpack_qos_swiQosFlow_t::is_LteQci_Available

8.1029.2.7 uint8_t unpack_qos_swiQosFlow_t::is_MaxAllowedPktSz_Available

8.1029.2.8 uint8_t unpack_qos_swiQosFlow_t::is_MinPolicedPktSz_Available

8.1029.2.9 uint8_t unpack_qos_swiQosFlow_t::is_PktErrRate_Available

8.1029.2.10 uint8_t unpack_qos_swiQosFlow_t::is_ProfileId3GPP2_Available

8.1029.2.11 uint8_t unpack_qos_swiQosFlow_t::is_TokenBucket_Available

8.1029.2.12 uint8_t unpack_qos_swiQosFlow_t::is_TrafficClass_Available

8.1029.2.13 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPP2Pri_Available

8.1029.2.14 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPImCn_Available

8.1029.2.15 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPResResidualBER_Available

8.1029.2.16 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPSigInd_Available

8.1029.2.17 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPTraHdlPri_Available

8.1029.2.18 uint32_t unpack_qos_swiQosFlow_t::Jitter

8.1029.2.19 uint32_t unpack_qos_swiQosFlow_t::Latency

8.1029.2.20 uint8_t unpack_qos_swiQosFlow_t::LteQci

8.1029.2.21 uint32_t unpack_qos_swiQosFlow_t::MaxAllowedPktSz

8.1029.2.22 uint32_t unpack_qos_swiQosFlow_t::MinPolicedPktSz

8.1029.2.23 unpack_qos_pktErrRate_t unpack_qos_swiQosFlow_t::PktErrRate

8.1029.2.24 uint16_t unpack_qos_swiQosFlow_t::ProfileId3GPP2

8.1029.2.25 unpack_qos_tokenBucket_t unpack_qos_swiQosFlow_t::TokenBucket

8.1029.2.26 uint8_t unpack_qos_swiQosFlow_t::TrafficClass

8.1029.2.27 uint8_t unpack_qos_swiQosFlow_t::val_3GPP2Pri

8.1029.2.28 uint8_t unpack_qos_swiQosFlow_t::val_3GPPImCn

8.1029.2.29 uint16_t unpack_qos_swiQosFlow_t::val_3GPPResResidualBER

8.1029.2.30 uint8_t unpack_qos_swiQosFlow_t::val_3GPPSigInd

8.1029.2.31 uint8_t unpack_qos_swiQosFlow_t::val_3GPPTraHdlPri

8.1030 unpack_qos_tokenBucket_t Struct Reference

Data Fields

- uint32_t peakRate
- uint32_t tokenRate
- uint32_t bucketSz

8.1030.1 Detailed Description

This structure contains the TP flow data rate token bucket

Parameters

peakRate	Maximum rate at which data can be transmitted when the token bucket is full (bits per second)
tokenRate	Rate at which tokens will be put in the token bucket (bits per second); a token is required to be present in the bucket to send a byte of data
bucketSz	Maximum number of tokens that can be accumulated at any instance (bytes); controls the size of the burst that is allowed at any given time

8.1030.2 Field Documentation

8.1030.2.1 uint32_t unpack_qos_tokenBucket_t::bucketSz

8.1030.2.2 uint32_t unpack_qos_tokenBucket_t::peakRate

8.1030.2.3 uint32_t unpack_qos_tokenBucket_t::tokenRate

8.1031 unpack_qos_Tos_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.1031.1 Detailed Description

This structure contains the IPv4 filter type of service

Parameters

val	Type of service value
mask	Packet matches the TOS filter if: (IPv4_filter_tos_val and IPv4_filter_tos_mask) == (TOS value in the IP packet & IPv4_filter_tos_mask) Example: • IPv4_filter_tos_val = 00101000 • IPv4_filter_tos_mask = 11111100 The filter will compare only the first 6 bits in the IPv4_filter_type_of_service with the first 6 bits in the TOS field of the IP packet. The first 6 bits in the TOS field of the IP packet must be 001010 to match the filter. The last 2 bits can be anything since they are ignored by filtering.

8.1031.2 Field Documentation

8.1031.2.1 uint8_t unpack_qos_Tos_t::mask

8.1031.2.2 uint8_t unpack_qos_Tos_t::val

8.1032 unpack_QosFlowStat_t Struct Reference

Data Fields

- uint32_t bearerId
- uint32_t tx_pkt
- uint32_t tx_pkt_drp
- uint64_t tx_bytes
- uint64_t tx_bytes_drp

8.1032.1 Detailed Description

This structure contains the Data statistic per QoS flow

Parameters

bearerId	• Bearer ID
tx_pkt	• number of sent packets for the QoS flow ID
tx_pkt_drp	• number of dropped(TX) packets for the QoS flow ID

tx_bytes	• number of sent bytes for the QoS flow ID
tx_bytes_drp	• number of dropped(TX) bytes for the QoS flow ID

8.1032.2 Field Documentation

8.1032.2.1 uint32_t unpack_QosFlowStat_t::bearerId

8.1032.2.2 uint64_t unpack_QosFlowStat_t::tx_bytes

8.1032.2.3 uint64_t unpack_QosFlowStat_t::tx_bytes_drp

8.1032.2.4 uint32_t unpack_QosFlowStat_t::tx_pkt

8.1032.2.5 uint32_t unpack_QosFlowStat_t::tx_pkt_drp

8.1033 unpack_result_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1033.1 Detailed Description

unpack result struct

Parameters

out	*Tlvresult*	
out	*ParamPresence-Mask*	Parameter presence bit mask

8.1033.2 Field Documentation

8.1033.2.1 swi_uint256_t unpack_result_t::ParamPresenceMask

8.1033.2.2 uint16_t unpack_result_t::Tlvresult

8.1034 unpack_rms_GetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask
- swi_uint256_t ParamPresenceMask

8.1034.1 Detailed Description

Unpack get SMS wake parameters

Parameters

enabled	• SMS wake functionality enabled – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **16**
wake_mask	• SMS wake mask to search for incoming messages (only relevant when enabled) • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1034.2 Field Documentation

8.1034.2.1 uint32_t unpack_rms_GetSMSWake_t::enabled

8.1034.2.2 swi_uint256_t unpack_rms_GetSMSWake_t::ParamPresenceMask

8.1034.2.3 uint32_t unpack_rms_GetSMSWake_t::wake_mask

8.1035 unpack_rms_SetSMSWake_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1035.1 Detailed Description

This structure contains unpack Set SMS Wake.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1035.2 Field Documentation

8.1035.2.1 swi_uint256_t unpack_rms_SetSMSWake_t::ParamPresenceMask

8.1036 unpack_RMTransferStatistics_ind_t Struct Reference

Data Fields

- wds_DataULongTlv TxOkConutTlv

- wds_DataULongTlv RxOkConutTlv
- wds_DataULongLongTlv TxOkByteCountTlv
- wds_DataULongLongTlv RxOkByteCountTlv
- wds_DataULongTlv TxDropConutTlv
- wds_DataULongTlv RxDropConutTlv
- swi_uint256_t ParamPresenceMask

8.1036.1 Detailed Description

WDS packet RM Transfer Statistics data structure for individual session

Parameters

TxOkConutTlv	• Tx Ok Packet Tlv Value. • Bit to check in ParamPresenceMask - **16**
RxOkConutTlv	• Rx Ok Packet Tlv Value. • Bit to check in ParamPresenceMask - **17**
TxOkByteCount-Tlv	• Tx Ok Byte Count Packet Tlv Value. • Bit to check in ParamPresenceMask - **25**
RxOkByteCount-Tlv	• Rx Ok Byte Count Packet Tlv Value. • Bit to check in ParamPresenceMask - **26**
TxDropConutTlv	• Tx Drop Count Packet Tlv Value. • Bit to check in ParamPresenceMask - **37**
RxDropConutTlv	• Rx Drop Count Packet Tlv Value. • Bit to check in ParamPresenceMask - **38**

8.1036.2 Field Documentation

8.1036.2.1 swi_uint256_t unpack_RMTransferStatistics_ind_t::ParamPresenceMask

8.1036.2.2 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxDropConutTlv

8.1036.2.3 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::RxOkByteCountTlv

8.1036.2.4 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxOkConutTlv

8.1036.2.5 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxDropConutTlv

8.1036.2.6 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::TxOkByteCountTlv

8.1036.2.7 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxOkConutTlv

8.1037 unpack_sar_SLQSGetRfSarState_t Struct Reference

Data Fields

- uint32_t * pSarRFState
- swi_uint256_t ParamPresenceMask

8.1037.1 Detailed Description

This structure contains unpack get RF SAR state parameter.

Parameters

pSarRFState	• SAR RF State – QMI_SAR_RF_STATE_DEFAULT – QMI_SAR_RF_STATE_1 – QMI_SAR_RF_STATE_2 – QMI_SAR_RF_STATE_3 – QMI_SAR_RF_STATE_4 – QMI_SAR_RF_STATE_5 – QMI_SAR_RF_STATE_6 – QMI_SAR_RF_STATE_7 – QMI_SAR_RF_STATE_8 • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1037.2 Field Documentation

8.1037.2.1 swi_uint256_t unpack_sar_SLQSGetRfSarState_t::ParamPresenceMask

8.1037.2.2 uint32_t* unpack_sar_SLQSGetRfSarState_t::pSarRFState

8.1038 unpack_sms_GetSMSCAddress_t Struct Reference

Data Fields

- uint8_t addressSize
- uint8_t * pSMSCAddress
- uint8_t typeSize
- uint8_t * pSMSCType
- swi_uint256_t ParamPresenceMask

8.1038.1 Detailed Description

Structure contains Gets the SMS center address parameters.

Parameters

addressSize	• The maximum number of characters (including NULL terminator) that the SMS center address array can contain. • Bit to check in ParamPresenceMask - **1**
pSMSCAddress	• The SMS center address represented as a NULL terminated string. • Bit to check in ParamPresenceMask - **1**
typeSize	• The maximum number of characters (including NULL terminator) that the SMS center address type array can contain. • Bit to check in ParamPresenceMask - **1**
pSMSCType	• The SMS center address type represented as a NULL terminated string. • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1038.2 Field Documentation

8.1038.2.1 uint8_t unpack_sms_GetSMSCAddress_t::addressSize

8.1038.2.2 swi_uint256_t unpack_sms_GetSMSCAddress_t::ParamPresenceMask

8.1038.2.3 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCAddress

8.1038.2.4 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCType

8.1038.2.5 uint8_t unpack_sms_GetSMSCAddress_t::typeSize

8.1039 unpack_sms_SaveSMS_t Struct Reference

Data Fields

- uint32_t * pMessageIndex
- swi_uint256_t ParamPresenceMask

8.1039.1 Detailed Description

This structure contains unpack save SMS parameter.

Parameters

pMessageIndex	• The message index assigned by the device • Bit to check in ParamPresenceMask - **1**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1039.2 Field Documentation

8.1039.2.1 swi_uint256_t unpack_sms_SaveSMS_t::ParamPresenceMask

8.1039.2.2 uint32_t* unpack_sms_SaveSMS_t::pMessageIndex

8.1040 unpack_sms_SendSMS_t Struct Reference

Data Fields

- uint16_t messageID
- uint32_t messageFailureCode
- swi_uint256_t ParamPresenceMask

8.1040.1 Detailed Description

This structure contains unpack send SMS parameters.

Parameters

messageID	• WMS message ID • Bit to check in ParamPresenceMask - **1**
messageFailure-Code	• pointer to message failure code. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1040.2 Field Documentation

8.1040.2.1 uint32_t unpack_sms_SendSMS_t::messageFailureCode

8.1040.2.2 uint16_t unpack_sms_SendSMS_t::messageID

8.1040.2.3 swi_uint256_t unpack_sms_SendSMS_t::ParamPresenceMask

8.1041 unpack_sms_SetNewSMSCallback_ind_t Struct Reference

Data Fields

- newMTMessageTlv NewMMTlv
- transferRouteMessageTlv TRMessageTlv
- messageModeTlv MMTlv

- sMSEtwsMessageTlv ETWSTlv
- eTWSPLMNInfoTlv ETWSPLMNTlv
- sMSCAddressTlv SMSCTlv
- sMSOnIMSTlv IMSTlv
- swi_uint256_t ParamPresenceMask

8.1041.1 Detailed Description

This structure contains unpack new SMS callback indication.

Parameters

NewMMTlv	• MT message • See newMTMessageTlv for more information • Bit to check in ParamPresenceMask - **16**
TRMessageTlv	• Transfer Route MT Message • See transferRouteMessageTlv for more information • Bit to check in ParamPresenceMask - **17**
MMTlv	• Message mode • See messageModeTlv for more information • Bit to check in ParamPresenceMask - **18**
ETWSTlv	• ETWS Message • See sMSEtwsMessageTlv for more information • Bit to check in ParamPresenceMask - **19**
ETWSPLMNTlv	• ETWS PLMN Information • See eTWSPLMNInfoTlv for more information • Bit to check in ParamPresenceMask - **20**
SMSCTlv	• SMSC Address • See sMSCAddressTlv for more information • Bit to check in ParamPresenceMask - **21**
IMSTlv	• SMS on IMS • See sMSOnIMSTlv for more information • Bit to check in ParamPresenceMask - **22**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1041.2 Field Documentation

8.1041.2.1 eTWSPLMNInfoTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSPLMNTlv

8.1041.2.2 sMSEtwsMessageTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSTlv

8.1041.2.3 sMSOnIMSTlv unpack_sms_SetNewSMSCallback_ind_t::IMSTlv

8.1041.2.4 messageModeTlv unpack_sms_SetNewSMSCallback_ind_t::MMTlv

8.1041.2.5 newMTMessageTlv unpack_sms_SetNewSMSCallback_ind_t::NewMMTlv

8.1041.2.6 swi_uint256_t unpack_sms_SetNewSMSCallback_ind_t::ParamPresenceMask

8.1041.2.7 sMSCAddressTlv unpack_sms_SetNewSMSCallback_ind_t::SMSCTlv

8.1041.2.8 transferRouteMessageTlv unpack_sms_SetNewSMSCallback_ind_t::TRMessageTlv

8.1042 unpack_sms_SetNewSMSCallback_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1042.1 Detailed Description

This structure contains unpack set new SMS callback.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1042.2 Field Documentation

8.1042.2.1 swi_uint256_t unpack_sms_SetNewSMSCallback_t::ParamPresenceMask

8.1043 unpack_sms_SetSMSCAddress_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1043.1 Detailed Description

This structure contains unpack Set SMS Address parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1043.2 Field Documentation

8.1043.2.1 swi_uint256_t unpack_sms_SetSMSCAddress_t::ParamPresenceMask

8.1044 unpack_sms_SLQSDeleteSMS_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1044.1 Detailed Description

This structure contains unpack delete SMS parameters.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1044.2 Field Documentation

8.1044.2.1 swi_uint256_t unpack_sms_SLQSDeleteSMS_t::ParamPresenceMask

8.1045 unpack_sms_SLQSGetIndicationRegister_t Struct Reference

Data Fields

- sms_getIndicationReg * pGetIndicationRegInfo
- swi_uint256_t ParamPresenceMask

8.1045.1 Detailed Description

Structure containing Parameters that provides registration state of different WMS indications.

Parameters

pGetIndication-RegInfo	• Pointer to structure of getIndicationReg – See sms_getIndicationReg for more information – pRegTransLayerInfoEvt * Bit to check in ParamPresenceMask - **16** – pRegTransNWRegInfoEvt * Bit to check in ParamPresenceMask - **17** – pRegCallStatInfoEvt * Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1045.2 Field Documentation

8.1045.2.1 swi_uint256_t unpack_sms_SLQSGetIndicationRegister_t::ParamPresenceMask

8.1045.2.2 sms_getIndicationReg* unpack_sms_SLQSGetIndicationRegister_t::pGetIndicationRegInfo

8.1046 unpack_sms_SLQSGetMessageWaiting_t Struct Reference

Data Fields

- sms_getMsgWaitingInfo * pGetMsgWaitingInfoResp
- swi_uint256_t ParamPresenceMask

8.1046.1 Detailed Description

Structure contain Parameter that provide information about the message waiting information.

Parameters

pGetMsg-WaitingInfoResp	• Pointer to structure of getMsgWaitingInfoResp – See sms_getMsgWaitingInfo for more information • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1046.2 Field Documentation

8.1046.2.1 swi_uint256_t unpack_sms_SLQSGetMessageWaiting_t::ParamPresenceMask

8.1046.2.2 sms_getMsgWaitingInfo* unpack_sms_SLQSGetMessageWaiting_t::pGetMsgWaitingInfoResp

8.1047 unpack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t messageTag
- uint32_t messageFormat
- uint32_t messageSize
- uint8_t message [2048]
- swi_uint256_t ParamPresenceMask

8.1047.1 Detailed Description

This sturcture contains unpack get SMS parameters.

Parameters

messageTag	• Message tag – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent

- Bit to check in ParamPresenceMask - **1**

Parameters

messageFormat	• Message format – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP

- Bit to check in ParamPresenceMask - **1**

Parameters

messageSize	• Upon input the maximum number of bytes that can be written to the message array. • Upon successful output the actual number of bytes written to the message array. • Bit to check in ParamPresenceMask - **1**
message	• The message contents array
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1047.2 Field Documentation

8.1047.2.1 uint8_t unpack_sms_SLQSGetSMS_t::message[2048]

8.1047.2.2 uint32_t unpack_sms_SLQSGetSMS_t::messageFormat

8.1047.2.3 uint32_t unpack_sms_SLQSGetSMS_t::messageSize

8.1047.2.4 uint32_t unpack_sms_SLQSGetSMS_t::messageTag

8.1047.2.5 swi_uint256_t unpack_sms_SLQSGetSMS_t::ParamPresenceMask

8.1048 unpack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- sms_qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig
- sms_qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig
- swi_uint256_t ParamPresenceMask

8.1048.1 Detailed Description

Structure contain parameters that Provides Information about the SMS BroadcastConfiguration

Parameters

pBroadcast-Config	• The data for 3GPP Broadcast Information(Optional). • Bit to check in ParamPresenceMask - **16**
pCDMA-BroadcastConfig	• The data for 3GPP2 Broadcast Information(Optional). • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1048.2 Field Documentation

8.1048.2.1 swi_uint256_t unpack_sms_SLQSGetSmsBroadcastConfig_t::ParamPresenceMask

8.1048.2.2 sms_qaQmi3GPPBroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pBroadcastConfig

8.1048.2.3 sms_qaQmi3GPP2BroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pCDMABroadcast-Config

8.1049 unpack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t messageListSize
- qmiSmsMessageList messageList [255]
- swi_uint256_t ParamPresenceMask

8.1049.1 Detailed Description

This structure contains unpack get SMS list parameters.

Parameters

messageListSize	• Upon input the maximum number of elements that the message list array can contain. • Upon successful output the actual number of elements in the message list array. • Bit to check in ParamPresenceMask - **1**

messageList	• Message List • See qmiSmsMessageList for more information • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1049.2 Field Documentation

8.1049.2.1 qmiSmsMessageList unpack_sms_SLQSGetSMSList_t::messageList[255]

8.1049.2.2 uint32_t unpack_sms_SLQSGetSMSList_t::messageListSize

8.1049.2.3 swi_uint256_t unpack_sms_SLQSGetSMSList_t::ParamPresenceMask

8.1050 unpack_sms_SLQSGetTransLayerInfo_t Struct Reference

Data Fields

- sms_getTransLayerInfo * pGetTransLayerInfo
- swi_uint256_t ParamPresenceMask

8.1050.1 Detailed Description

Structure contain parameters that gives information about the transport layer.

Parameters

pGetTransLayer-Info	• Pointer to structure of getTransLayerInfo – See sms_getTransLayerInfo for more information – pGetTransLayerInfo->pRegInd * Bit to check in ParamPresenceMask - **16** – pGetTransLayerInfo->pTransLayerInfo * Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1050.2 Field Documentation

8.1050.2.1 swi_uint256_t unpack_sms_SLQSGetTransLayerInfo_t::ParamPresenceMask

8.1050.2.2 sms_getTransLayerInfo* unpack_sms_SLQSGetTransLayerInfo_t::pGetTransLayerInfo

8.1051 unpack_sms_SLQSGetTransNWRegInfo_t Struct Reference

Data Fields

- sms_getTransNWRegInfo * pGetTransNWRegInfo
- swi_uint256_t ParamPresenceMask

8.1051.1 Detailed Description

Structure containing parameters that provides transport layer network registration info.

Parameters

pGetTransNW-RegInfo	• Pointer to structure of getTransNWRegInfo – See sms_getTransNWRegInfo for more information – Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1051.2 Field Documentation

8.1051.2.1 swi_uint256_t unpack_sms_SLQSGetTransNWRegInfo_t::ParamPresenceMask

8.1051.2.2 sms_getTransNWRegInfo* unpack_sms_SLQSGetTransNWRegInfo_t::pGetTransNWRegInfo

8.1052 unpack_sms_SLQSModifySMSStatus_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1052.1 Detailed Description

This structure contains unpack modify SMS status parameters.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1052.2 Field Documentation

8.1052.2.1 swi_uint256_t unpack_sms_SLQSModifySMSStatus_t::ParamPresenceMask

8.1053 unpack_sms_SLQSNWRegInfoCallback_ind_t Struct Reference

Data Fields

- uint8_t NWRegStat
- swi_uint256_t ParamPresenceMask

8.1053.1 Detailed Description

Contains the parameters passed for indication about transport network registration change by the device.

Parameters

NWRegStat	• provides the transport network registration information • Values: – 0x00 - No Service – 0x01 - In Progress – 0x02 - Failed – 0x03 - Limited Service – 0x04 - Full Service • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.1053.2 Field Documentation

8.1053.2.1 uint8_t unpack_sms_SLQSNWRegInfoCallback_ind_t::NWRegStat

8.1053.2.2 swi_uint256_t unpack_sms_SLQSNWRegInfoCallback_ind_t::ParamPresenceMask

8.1054 unpack_sms_SLQSSendAsyncSMS_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1054.1 Detailed Description

This structure contains unpack SLQSSendAsyncSMS parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1054.2 Field Documentation

8.1054.2.1 swi_uint256_t unpack_sms_SLQSSendAsyncSMS_t::ParamPresenceMask

8.1055 unpack_sms_SLQSSetIndicationRegister_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1055.1 Detailed Description

This structure contains unpack SLQSSetIndicationRegister parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1055.2 Field Documentation

8.1055.2.1 swi_uint256_t unpack_sms_SLQSSetIndicationRegister_t::ParamPresenceMask

8.1056 unpack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1056.1 Detailed Description

This structure contains unpack SLQSSetSmsBroadcastActivation parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1056.2 Field Documentation

8.1056.2.1 swi_uint256_t unpack_sms_SLQSSetSmsBroadcastActivation_t::ParamPresenceMask

8.1057 unpack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1057.1 Detailed Description

This structure contains unpack SLQSSetSmsBroadcastConfig parameters.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1057.2 Field Documentation

8.1057.2.1 swi_uint256_t unpack_sms_SLQSSetSmsBroadcastConfig_t::ParamPresenceMask

8.1058 unpack_sms_SLQSSetSmsStorage_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1058.1 Detailed Description

This structure contains unpack SLQSSetSmsStorage parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1058.2 Field Documentation

8.1058.2.1 swi_uint256_t unpack_sms_SLQSSetSmsStorage_t::ParamPresenceMask

8.1059 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeResp * pMaxStorageSizeResp
- swi_uint256_t ParamPresenceMask

8.1059.1 Detailed Description

Structure contain parameters that provides the maximum number of messages that can be stored in the specified memory storage. Also it provides the number of slots currently available

Parameters

pMaxStorage-SizeResp	• Response parameters for SmsSLQSGetMaxStorageSize
	  – See sms_maxStorageSizeResp for more information
	• pMaxStorageSizeResp->maxStorageSize
	  – Bit to check in ParamPresenceMask - **1**
	• pMaxStorageSizeResp->freeSlots
	  – Bit to check in ParamPresenceMask - **16**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1059.2 Field Documentation

8.1059.2.1 swi_uint256_t unpack_sms_SLQSSmsGetMaxStorageSize_t::ParamPresenceMask

8.1059.2.2 sms_maxStorageSizeResp* unpack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeResp

8.1060 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference

Data Fields

- sms_msgProtocolResp * pMessageProtocol
- swi_uint256_t ParamPresenceMask

8.1060.1 Detailed Description

Structure contain Parameters that get the message protocol currently in use for the WMS client.

Parameters

pMessage-Protocol	• Pointer to smsMsgprotocolResp – See sms_msgProtocolResp for more information • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1060.2 Field Documentation

8.1060.2.1 swi_uint256_t unpack_sms_SLQSSmsGetMessageProtocol_t::ParamPresenceMask

8.1060.2.2 sms_msgProtocolResp* unpack_sms_SLQSSmsGetMessageProtocol_t::pMessageProtocol

8.1061 unpack_sms_SLQSSmsSetRoutes_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1061.1 Detailed Description

This structure contains unpack SLQSSmsSetRoutes parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1061.2 Field Documentation

8.1061.2.1 swi_uint256_t unpack_sms_SLQSSmsSetRoutes_t::ParamPresenceMask

8.1062 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference

Data Fields

- uint32_t * pSmsStorage
- swi_uint256_t ParamPresenceMask

8.1062.1 Detailed Description

Structure contain Parameters that return current SMS configuration that is applied to all incoming and outgoing messages.

Parameters

pSmsStorage	• Values: – 0x01 - device's permanent memory – 0x02 - UICC • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1062.2 Field Documentation

8.1062.2.1 swi_uint256_t unpack_sms_SLQSSwiGetSMSStorage_t::ParamPresenceMask

8.1062.2.2 uint32_t* unpack_sms_SLQSSwiGetSMSStorage_t::pSmsStorage

8.1063 unpack_sms_SLQSTransLayerInfoCallback_ind_t Struct Reference

Data Fields

- uint8_t regInd
- sms_transLayerInfo * pTransLayerInfo
- swi_uint256_t ParamPresenceMask

8.1063.1 Detailed Description

Contains the parameters passed for the indication about change in transport layer information

Parameters

regInd	• Indicates whether the transport layer is registered or not • Values: – 0x00 - Transport layer is not registered – 0x01 - Transport layer is registered • Bit to check in ParamPresenceMask - **1**

pTransLayerInfo	• Optional parameter • See sms_transLayerInfo for more information • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.1063.2 Field Documentation

8.1063.2.1 swi_uint256_t unpack_sms_SLQSTransLayerInfoCallback_ind_t::ParamPresenceMask

8.1063.2.2 sms_transLayerInfo* unpack_sms_SLQSTransLayerInfoCallback_ind_t::pTransLayerInfo

8.1063.2.3 uint8_t unpack_sms_SLQSTransLayerInfoCallback_ind_t::regInd

8.1064 unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t Struct Reference

Data Fields

- uint16_t sendStatus
- uint16_t messageID
- uint16_t causeCode
- uint8_t errorClass
- uint16_t RPCause
- uint8_t TPCause
- uint8_t msgDelFailureType
- uint8_t msgDelFailureCause
- uint8_t alphaIDLen
- uint8_t * pAlphaID
- uint32_t userData
- swi_uint256_t ParamPresenceMask

8.1064.1 Detailed Description

This structure contains SMS parameters for indication of RAW ASYNC SEND

Parameters

sendStatus	• Send Status • Values: – QMI_ERR_NONE - No error in the request – QMI_ERR_CAUSE_CODE - SMS cause code – QMI_ERR_MESSAGE_DELIVERY_FAILURE - Message could not be delivered – QMI_ERR_NO_MEMORY - Device could not allocate memory to formulate a response • Bit to check in ParamPresenceMask - **1**

messageID	• Unique ID assigned by WMS for non-retry messages. • Bit to check in ParamPresenceMask - **16**
causeCode	• WMS cause code • Bit to check in ParamPresenceMask - **17**
errorClass	• Error Class • Values: – 0x00 - ERROR_CLASS_TEMPORARY – 0x01 - ERROR_CLASS_PERMANENT • Bit to check in ParamPresenceMask - **18**
RPCause	• GW RP cause • Bit to check in ParamPresenceMask - **19**
TPCause	• GW TP Cause • Bit to check in ParamPresenceMask - **19**
msgDelFailure-Type	• Message delivery failure type • Values: – 0x00 - WMS_MESSAGE_DELIVERY_FAILURE_TEMPORARY – 0x01 - WMS_MESSAGE_DELIVERY_FAILURE_PERMANENT • Bit to check in ParamPresenceMask - **20**
msgDelFailure-Cause	• Message delivery failure cause • Values: – 0x00 - WMS_MESSAGE_BLOCKED_DUE_TO_CALL_CONTROL • Bit to check in ParamPresenceMask - **21**
alphaIDLen	• Number of sets of the pAlphaID • Bit to check in ParamPresenceMask - **22**
pAlphaID	• Alpha ID • Bit to check in ParamPresenceMask - **22**
userData	• Identifies the request associated with this indication. • Bit to check in ParamPresenceMask - **23**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1064.2 Field Documentation

8.1064.2.1 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::alphaIDLen

8.1064.2.2 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::causeCode

8.1064.2.3 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::errorClass

8.1064.2.4 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::messageID

8.1064.2.5 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::msgDelFailureCause

8.1064.2.6 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::msgDelFailureType

8.1064.2.7 uint8_t* unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::pAlphaID

8.1064.2.8 swi_uint256_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::ParamPresenceMask

8.1064.2.9 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::RPCause

8.1064.2.10 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::sendStatus

8.1064.2.11 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::TPCause

8.1064.2.12 uint32_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::userData

8.1065 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference

Data Fields

- uint8_t storageType
- uint8_t messageMode
- swi_uint256_t ParamPresenceMask

8.1065.1 Detailed Description

This structure contains unpack SMS memory full callback indication.

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV • Bit to check in ParamPresenceMask - **1**
messageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS) • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1065.2 Field Documentation

8.1065.2.1 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::messageMode

8.1065.2.2 swi_uint256_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::ParamPresenceMask

8.1065.2.3 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::storageType

8.1066 unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t Struct Reference

Data Fields

- uint8_t numInstances
- sms_messageWaitingInfoContent msgWaitInfo [0xFF]
- swi_uint256_t ParamPresenceMask

8.1066.1 Detailed Description

This structure holds information related to message waiting information indication

Parameters

numInstances	• Number of sets of the elements in structure sms_messageWaitingInfoContent • Bit to check in ParamPresenceMask - **1**
msgWaitInfo	• Pointer to structure of sms_messageWaitingInfoContent. – See sms_messageWaitingInfoContent for more information. • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1066.2 Field Documentation

8.1066.2.1 sms_messageWaitingInfoContent unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t::msgWaitInfo[0xFF]

8.1066.2.2 uint8_t unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t::numInstances

8.1066.2.3 swi_uint256_t unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t::ParamPresenceMask

8.1067 unpack_swiaudio_SLQSGetM2MAudioProfile_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t EarMute
- uint8_t MicMute
- uint8_t Generator
- uint8_t Volume

- uint8_t CwtMute
- swi_uint256_t ParamPresenceMask

8.1067.1 Detailed Description

This structure contains the SLQSGetM2MAudioProfile unpack parameters.

Parameters

Profile	• Audio Profile – 0-5 • Bit to check in ParamPresenceMask - **1**
EarMute	• Ear Mute – 0 - Mute – 1 - Unmute • Bit to check in ParamPresenceMask - **2**
MicMute	• MIC Mute – 0 - Mute – 1 - Unmute • Bit to check in ParamPresenceMask - **3**
Generator	• Generator – 0 - Voice • Bit to check in ParamPresenceMask - **4**
Volume	• RX volume level – 0-5 • Bit to check in ParamPresenceMask - **5**
CwtMute	• Call waiting tone Mute – 0 - Mute – 1 - Unmute • Bit to check in ParamPresenceMask - **6**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1067.2 Field Documentation

8.1067.2.1 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::CwtMute

8.1067.2.2 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::EarMute

8.1067.2.3 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::Generator

8.1067.2.4 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::MicMute

8.1067.2.5 swi_uint256_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::ParamPresenceMask

8.1067.2.6 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::Profile

8.1067.2.7 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::Volume

8.1068 unpack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference

Data Fields

- uint8_t Level
- swi_uint256_t ParamPresenceMask

8.1068.1 Detailed Description

This structure contains the SLQSGetM2MAudioVolume unpack parameters.

Parameters

Level	• The RX Volume Level – 0-5 • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1068.2 Field Documentation

8.1068.2.1 uint8_t unpack_swiaudio_SLQSGetM2MAudioVolume_t::Level

8.1068.2.2 swi_uint256_t unpack_swiaudio_SLQSGetM2MAudioVolume_t::ParamPresenceMask

8.1069 unpack_swiaudio_SLQSGetM2MAVMute_t Struct Reference

Data Fields

- uint8_t EarMute
- uint8_t MicMute
- uint8_t CwtMute
- swi_uint256_t ParamPresenceMask

8.1069.1 Detailed Description

This structure contains the SLQSGetM2MAVMute unpack parameters.

Parameters

pEarMute	• Ear Mute – 0-Mute – 1-UnMute • Bit to check in ParamPresenceMask - **1**
pMicMute	• Mic Mute – 0-Mute – 1-unmute • Bit to check in ParamPresenceMask - **2**
CwtMute	• Waiting tone Mute – 0-5 • Bit to check in ParamPresenceMask - **3**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1069.2 Field Documentation

8.1069.2.1 uint8_t unpack_swiaudio_SLQSGetM2MAVMute_t::CwtMute

8.1069.2.2 uint8_t unpack_swiaudio_SLQSGetM2MAVMute_t::EarMute

8.1069.2.3 uint8_t unpack_swiaudio_SLQSGetM2MAVMute_t::MicMute

8.1069.2.4 swi_uint256_t unpack_swiaudio_SLQSGetM2MAVMute_t::ParamPresenceMask

8.1070 unpack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference

Data Fields

- uint16_t Value
- swi_uint256_t ParamPresenceMask

8.1070.1 Detailed Description

This structure contains the SLQSGetM2MSpkrGain unpack parameters.

Parameters

Value	• RX speakerphone gain – 0x0 - 0x7fff • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1070.2 Field Documentation

8.1070.2.1 swi_uint256_t unpack_swiaudio_SLQSGetM2MSpkrGain_t::ParamPresenceMask

8.1070.2.2 uint16_t unpack_swiaudio_SLQSGetM2MSpkrGain_t::Value

8.1071 unpack_swiavms_SLQSAVMSEventReportInd_t Struct Reference

Data Fields

- UnpackSwiAvmsEventReportBinaryUpdateSessionInfo * pBinaryUpdateSessionInfo
- UnpackSwiAvmsEventReportConfig * pConfig
- UnpackSwiAvmsEventReportNotification * pNotification
- UnpackSwiAvmsEventReportPackageID * pPackageID
- UnpackSwiAvmsEventReportConnectionRequest * pConnectionRequest
- UnpackSwiAvmsEventReportWAMSParamChange * pWAMSParaChanged
- UnpackSwiAvmsEventReportRegStatus * pRegStatus
- UnpackSwiAvmsEventReportDataSessionStatus * pDataSessionStatus
- UnpackSwiAvmsEventReportSessionType * pSessionType
- UnpackSwiAvmsEventReportHTTPStatus * pHTTPStatus
- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1071.1 Detailed Description

Structure that contains the session type response for AVMS event report indication command

Parameters

pBinaryUpdateSessionInfo[OUT]	• See UnpackSwiAvmsEventReportBinaryUpdateSessionInfo for more information • Bit to check in ParamPresenceMask - **16**
pConfig[OUT]	• See UnpackSwiAvmsEventReportConfig for more information • Bit to check in ParamPresenceMask - **17**
pNotification[OUT]	• See UnpackSwiAvmsEventReportNotification for more information • Bit to check in ParamPresenceMask - **18**
pPackageID[OUT]	• See UnpackSwiAvmsEventReportPackageID for more information • Bit to check in ParamPresenceMask - **21**
pConnectionRequest[OUT]	• See UnpackSwiAvmsEventReportConnectionRequest for more information • Bit to check in ParamPresenceMask - **19**
pWAMSParaChanged[OUT]	• See UnpackSwiAvmsEventReportWAMSParamChange for more information • Bit to check in ParamPresenceMask - **20**

pRegStatus[OUT]	• See UnpackSwiAvmsEventReportRegStatus for more information • Bit to check in ParamPresenceMask - **22**
pDataSessionStatus[OUT]	• See UnpackSwiAvmsEventReportDataSessionStatus for more information • Bit to check in ParamPresenceMask - **23**
pSessionType[OUT]	• See UnpackSwiAvmsEventReportSessionType for more information • Bit to check in ParamPresenceMask - **24**
pHTTPStatus[OUT]	• See UnpackSwiAvmsEventReportHTTPStatus for more information • Bit to check in ParamPresenceMask - **25**
resultcode	- Tlv Result Code.

8.1071.2 Field Documentation

8.1071.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSEventReportInd_t::ParamPresenceMask

8.1071.2.2 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo* unpack_swiavms_SLQSAVMSEventReportInd_t::pBinaryUpdateSessionInfo

8.1071.2.3 UnpackSwiAvmsEventReportConfig* unpack_swiavms_SLQSAVMSEventReportInd_t::pConfig

8.1071.2.4 UnpackSwiAvmsEventReportConnectionRequest* unpack_swiavms_SLQSAVMSEventReportInd_t::pConnectionRequest

8.1071.2.5 UnpackSwiAvmsEventReportDataSessionStatus* unpack_swiavms_SLQSAVMSEventReportInd_t::pDataSessionStatus

8.1071.2.6 UnpackSwiAvmsEventReportHTTPStatus* unpack_swiavms_SLQSAVMSEventReportInd_t::pHTTPStatus

8.1071.2.7 UnpackSwiAvmsEventReportNotification* unpack_swiavms_SLQSAVMSEventReportInd_t::pNotification

8.1071.2.8 UnpackSwiAvmsEventReportPackageID* unpack_swiavms_SLQSAVMSEventReportInd_t::pPackageID

8.1071.2.9 UnpackSwiAvmsEventReportRegStatus* unpack_swiavms_SLQSAVMSEventReportInd_t::pRegStatus

8.1071.2.10 UnpackSwiAvmsEventReportSessionType* unpack_swiavms_SLQSAVMSEventReportInd_t::pSessionType

8.1071.2.11 UnpackSwiAvmsEventReportWAMSParamChange* unpack_swiavms_SLQSAVMSEventReportInd_t::pWAMSParaChanged

8.1071.2.12 uint32_t unpack_swiavms_SLQSAVMSEventReportInd_t::resultcode

8.1072 unpack_swiavms_SLQSAVMSGetSettings_t Struct Reference

Data Fields

- uint32_t OMADMEnabled
- uint8_t AutoConnect
- uint8_t AutoReboot
- uint8_t FwPromptdownload
- uint8_t FwPromptUpdate
- uint8_t FwAutoSDM
- uint32_t * pPollingTimer
- PackSwiAVMSSettingsConnectionRetryTimers * pConnectionRetryTimers
- PackSwiAVMSSettingsAPNInfo * pAPNInfo
- uint8_t * pNotificationStore
- PackSwiAVMSSettingsPeriodsInfo * pPeroidsInfo
- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1072.1 Detailed Description

Structure that contains the session type response for AVMS Set Settings command

Parameters

OMADMEnabled	- OMA DM Enabled. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **1**

Parameters

AutoConnect	- Auto Connect. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **6**

Parameters

AutoReboot	- Auto Reboot. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **21**

Parameters

Fw-Autodownload	- Firmware auto download. • 0 - Disabled. • 1 - Enabled. – Bit to check in ParamPresenceMask - **3**

FwAutoUpdate	- Firmware auto update. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **4**

Parameters

FwAutoSDM	- OMA Automatic UI Alert Response. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **5**

Parameters

pPollingTimer	- Polling timer to connect to AVMS server. • 0-525600 (min) • 0:disabled

- Bit to check in ParamPresenceMask - **16**

Parameters

pConnection-RetryTimers	- Connection Retry timers. • See PackSwiAVMSSettingsConnectionRetryTimers for more information. • Bit to check in ParamPresenceMask - **17**
pAPNInfo	- APN Information. • See PackSwiAVMSSettingsAPNInfo for more information. • Bit to check in ParamPresenceMask - **18**
pNotification-Store	- Notification Storing When Disabled or Offline. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **19**

Parameters

pPeroidsInfo	- Min and Max Period of an Observation. • See PackSwiAVMSSettingsPeriodsInfo for more information. • Bit to check in ParamPresenceMask - **20**
resultcode	- Tlv Result Code.

8.1072.2 Field Documentation

8.1072.2.1 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::AutoConnect

8.1072.2.2 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::AutoReboot

8.1072.2.3 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwAutoSDM

8.1072.2.4 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwPromptdownload

8.1072.2.5 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwPromptUpdate

8.1072.2.6 uint32_t unpack_swiavms_SLQSAVMSGetSettings_t::OMADMEnabled

8.1072.2.7 PackSwiAVMSSettingsAPNInfo* unpack_swiavms_SLQSAVMSGetSettings_t::pAPNInfo

8.1072.2.8 swi_uint256_t unpack_swiavms_SLQSAVMSGetSettings_t::ParamPresenceMask

8.1072.2.9 PackSwiAVMSSettingsConnectionRetryTimers* unpack_swiavms_SLQSAVMSGetSettings_t::pConnectionRetryTimers

8.1072.2.10 uint8_t* unpack_swiavms_SLQSAVMSGetSettings_t::pNotificationStore

8.1072.2.11 PackSwiAVMSSettingsPeriodsInfo* unpack_swiavms_SLQSAVMSGetSettings_t::pPeroidsInfo

8.1072.2.12 uint32_t* unpack_swiavms_SLQSAVMSGetSettings_t::pPollingTimer

8.1072.2.13 uint32_t unpack_swiavms_SLQSAVMSGetSettings_t::resultcode

8.1073 unpack_swiavms_SLQSAVMSGetSettings_v2_t Struct Reference

Data Fields

- uint32_t OMADMEnabled
- uint8_t AutoConnect
- uint8_t FwPromptdownload
- uint8_t FwPromptUpdate
- uint8_t FwAutoSDM
- uint32_t * pPollingTimer
- PackSwiAVMSSettingsConnectionRetryTimers * pConnectionRetryTimers
- PackSwiAVMSSettingsAPNInfo * pAPNInfo
- uint8_t * pNotificationStore
- PackSwiAVMSSettingsPeriodsInfo * pPeroidsInfo
- uint8_t * pAutoReboot
- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1073.1 Detailed Description

Structure that contains the response for AVMS Get Settings command It maps with SLQSAVMSSetSettings_v2 (For AVC2 service)

Parameters

OMADMEnabled	- OMA DM Enabled. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **1**

Parameters

AutoConnect	- Auto Connect. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **6**

Parameters

Fw-Promptdownload	- Firmware Prompt download. • 0 - Disabled. • 1 - Enabled. – Bit to check in ParamPresenceMask - **3**
FwPrompt-Update	- Firmware Prompt update. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **4**

Parameters

FwAutoSDM	- OMA Automatic UI Alert Response. • 0 - Disabled. • 1 - Enabled.

- Bit to check in ParamPresenceMask - **5**

Parameters

pPollingTimer[OPTIONAL]	- Polling timer to connect to AVMS server. • 0-525600 (min) • 0:disabled – Bit to check in ParamPresenceMask - **16**
pConnection-RetryTimers[OPTIONAL]	- Connection Retry timers. • See PackSwiAVMSSettingsConnectionRetryTimers for more information. • Bit to check in ParamPresenceMask - **17**
pAPNInfo[OPTIONAL]	- APN Information. • See PackSwiAVMSSettingsAPNInfo for more information. • Bit to check in ParamPresenceMask - **18**
pNotification-Store[OPTIONAL]	- Notification Storing When Disabled or Offline. • 0 - Disabled. • 1 - Enabled. – Bit to check in ParamPresenceMask - **19**

pPeroidsInfo[OPTIONAL]	- Min and Max Period of an Observation. • See PackSwiAVMSSettingsPeriodsInfo for more information. • Bit to check in ParamPresenceMask - **20**
pAutoReboot[OPTIONAL]	- Auto Reboot. • 0 - Disabled. • 1 - Enabled. – Bit to check in ParamPresenceMask - **21**
resultcode	- Tlv Result Code.

8.1073.2 Field Documentation

8.1073.2.1 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::AutoConnect

8.1073.2.2 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::FwAutoSDM

8.1073.2.3 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::FwPromptdownload

8.1073.2.4 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::FwPromptUpdate

8.1073.2.5 uint32_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::OMADMEnabled

8.1073.2.6 PackSwiAVMSSettingsAPNInfo* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pAPNInfo

8.1073.2.7 swi_uint256_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::ParamPresenceMask

8.1073.2.8 uint8_t* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pAutoReboot

8.1073.2.9 PackSwiAVMSSettingsConnectionRetryTimers* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pConnectionRetryTimers

8.1073.2.10 uint8_t* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pNotificationStore

8.1073.2.11 PackSwiAVMSSettingsPeriodsInfo* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pPeroidsInfo

8.1073.2.12 uint32_t* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pPollingTimer

8.1073.2.13 uint32_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::resultcode

8.1074 unpack_swiavms_SLQSAVMSSendSelection_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1074.1 Detailed Description

Structure that contains the session type response for AVMS Selection command

Parameters

resultcode	- Tlv Result Code. • Bit to check in ParamPresenceMask - **2**

8.1074.2 Field Documentation

8.1074.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSendSelection_t::ParamPresenceMask

8.1074.2.2 uint32_t unpack_swiavms_SLQSAVMSSendSelection_t::resultcode

8.1075 unpack_swiavms_SLQSAVMSSessionGetInfo_t Struct Reference

Data Fields

- UnpackSwiAvmsEventReportBinaryUpdateSessionInfo * pBinaryUpdateSessionInfo
- UnpackSwiAvmsEventReportConfig * pConfig
- UnpackSwiAvmsEventReportNotification * pNotification
- UnpackSwiAvmsEventReportPackageID * pPackageID
- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1075.1 Detailed Description

Structure that contains the session type response for AVMS get session info command

Parameters

pBinaryUpdate-SessionInfo[OU-T]	• See UnpackSwiAvmsEventReportBinaryUpdateSessionInfo for more information • Bit to check in ParamPresenceMask - **16**
pConfig[OUT]	• See UnpackSwiAvmsEventReportConfig for more information • Bit to check in ParamPresenceMask - **17**
pNotification[O-UT]	• See UnpackSwiAvmsEventReportNotification for more information • Bit to check in ParamPresenceMask - **18**
pPackageID[OU-T]	• See UnpackSwiAvmsEventReportPackageID for more information • Bit to check in ParamPresenceMask - **19**
resultcode	- Tlv Result Code.

8.1075.2 Field Documentation

8.1075.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSessionGetInfo_t::ParamPresenceMask

8.1075.2.2 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo* unpack_swiavms_SLQSAVMSSessionGetInfo_t::pBinaryUpdateSessionInfo

8.1075.2.3 UnpackSwiAvmsEventReportConfig∗ unpack_swiavms_SLQSAVMSSessionGetInfo_t::pConfig

8.1075.2.4 UnpackSwiAvmsEventReportNotification∗ unpack_swiavms_SLQSAVMSSessionGetInfo_t::pNotification

8.1075.2.5 UnpackSwiAvmsEventReportPackageID∗ unpack_swiavms_SLQSAVMSSessionGetInfo_t::pPackageID

8.1075.2.6 uint32_t unpack_swiavms_SLQSAVMSSessionGetInfo_t::resultcode

8.1076 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1076.1 Detailed Description

Structure that contains the session type response for AVMS set event report command

Parameters

resultcode	- Tlv Result Code. • Bit to check in ParamPresenceMask - **2**

8.1076.2 Field Documentation

8.1076.2.1 swi_uint256_t unpack_swiavms_SLQSAvmsSetEventReport_t::ParamPresenceMask

8.1076.2.2 uint32_t unpack_swiavms_SLQSAvmsSetEventReport_t::resultcode

8.1077 unpack_swiavms_SLQSAVMSSetSettings_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1077.1 Detailed Description

Structure that contains the session type response for AVMS Set Settings command

Parameters

resultcode	- Tlv Result Code. • Bit to check in ParamPresenceMask - **2**

8.1077.2 Field Documentation

8.1077.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSetSettings_t::ParamPresenceMask

8.1077.2.2 uint32_t unpack_swiavms_SLQSAVMSSetSettings_t::resultcode

8.1078 unpack_swiavms_SLQSAVMSSetSettings_v2_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1078.1 Detailed Description

Structure that contains the session type response for AVMS Set Settings command

Parameters

resultcode	- Tlv Result Code. • Bit to check in ParamPresenceMask - **2**

8.1078.2 Field Documentation

8.1078.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSetSettings_v2_t::ParamPresenceMask

8.1078.2.2 uint32_t unpack_swiavms_SLQSAVMSSetSettings_v2_t::resultcode

8.1079 unpack_swiavms_SLQSAVMSStartSession_t Struct Reference

Data Fields

- uint32_t sessionResponse
- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1079.1 Detailed Description

Structure that contains the session type response for AVMS start session command

Parameters

Session-Response	- OMA-DM availability CHECK for the requested item. • 0x00000001 - Available. • 0x00000002 - Not Available. • 0x00000003 - Check Timed Out. • Bit to check in ParamPresenceMask - **1**
resultcode	- Tlv Result Code.

8.1079.2 Field Documentation

8.1079.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSStartSession_t::ParamPresenceMask

8.1079.2.2 uint32_t unpack_swiavms_SLQSAVMSStartSession_t::resultcode

8.1079.2.3 uint32_t unpack_swiavms_SLQSAVMSStartSession_t::sessionResponse

8.1080 unpack_swiavms_SLQSAVMSStopSession_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.1080.1 Detailed Description

Structure that contains the session type response for AVMS Stop session command

Parameters

resultcode	- Tlv Result Code. • Bit to check in ParamPresenceMask - **2**

8.1080.2 Field Documentation

8.1080.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSStopSession_t::ParamPresenceMask

8.1080.2.2 uint32_t unpack_swiavms_SLQSAVMSStopSession_t::resultcode

8.1081 unpack_swidms_SLQSSwiDmsGetHWWatchdog_t Struct Reference

Data Fields

- swidms_SwiDmsGetHWWatchdog * pHWWatchdog
- swi_uint256_t ParamPresenceMask

8.1081.1 Detailed Description

This structure contains the get hw watchdog response

Parameters

pHWWatchdog	• See swidms_SwiDmsGetHWWatchdog for more information • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1081.2 Field Documentation

8.1081.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsGetHWWatchdog_t::ParamPresenceMask

8.1081.2.2 swidms_SwiDmsGetHWWatchdog* unpack_swidms_SLQSSwiDmsGetHWWatchdog_t::pHWWatchdog

8.1082 unpack_swidms_SLQSSwiDmsGetMTU_t Struct Reference

Data Fields

- swidms_mtuSize3gppTlv * pMTUSize3gpp
- swidms_hrpdMTUSizeTlv * pHrpdMTUSize
- swidms_ehrpdMTUSizeTlv * pEhrpdMTUSize
- swidms_usbMTUSizeTlv * pUsbMTUSize
- swi_uint256_t ParamPresenceMask

8.1082.1 Detailed Description

This structure contains the Get MTU Response parameter.

Parameters

pMTUSize3gpp	[Optional] • See swidms_mtuSize3gppTlv for more information • Bit to check in ParamPresenceMask - **16**
pHrpdMTUSize	[Optional] • See swidms_hrpdMTUSizeTlv for more information • Bit to check in ParamPresenceMask - **17**
pEhrpdMTUSize	[Optional] • See swidms_ehrpdMTUSizeTlv for more information • Bit to check in ParamPresenceMask - **18**
pUsbMTUSize	[Optional] • See swidms_usbMTUSizeTlv for more information • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1082.2 Field Documentation

8.1082.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsGetMTU_t::ParamPresenceMask

8.1082.2.2 swidms_ehrpdMTUSizeTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pEhrpdMTUSize

8.1082.2.3 swidms_hrpdMTUSizeTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pHrpdMTUSize

8.1082.2.4 swidms_mtuSize3gppTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pMTUSize3gpp

8.1082.2.5 swidms_usbMTUSizeTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pUsbMTUSize

8.1083 unpack_swidms_SLQSSwiDmsGetSecureInfo_t Struct Reference

Data Fields

- uint8_t secureBootEnabled
- uint8_t memoryDumpAllowed
- uint8_t jtagAccessAllowed
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.1083.1 Detailed Description

This structure contains secure boot config and other capabilities

Parameters

secureBoot-Enabled	• secure boot enabled or disabled 0- disabled 1- enabled • Bit to check in ParamPresenceMask - **1**
memoryDump-Allowed	• memory dump alloweed or disallowed 0- disallowed 1- allowed • Bit to check in ParamPresenceMask - **3**
jtagAccess-Allowed	• Jtag access allowed or disallowed 0- disallowed 1- allowed • Bit to check in ParamPresenceMask - **4**
Tlvresult	• unpack Tlv result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1083.2 Field Documentation

8.1083.2.1 uint8_t unpack_swidms_SLQSSwiDmsGetSecureInfo_t::jtagAccessAllowed

8.1083.2.2 uint8_t unpack_swidms_SLQSSwiDmsGetSecureInfo_t::memoryDumpAllowed

8.1083.2.3 swi_uint256_t unpack_swidms_SLQSSwiDmsGetSecureInfo_t::ParamPresenceMask

8.1083.2.4 uint8_t unpack_swidms_SLQSSwiDmsGetSecureInfo_t::secureBootEnabled

8.1083.2.5 uint16_t unpack_swidms_SLQSSwiDmsGetSecureInfo_t::TlvResult

8.1084 unpack_swidms_SLQSSwiDmsGetUimSlot_t Struct Reference

Data Fields

- swidms_UimSlotSelectedTlv uimSlotSelected
- swidms_MultiplexedSimModeTlv multiplexedSimMode
- swidms_AutoUimSwitchModeTlv autoUimSwitchMode
- swi_uint256_t ParamPresenceMask

8.1084.1 Detailed Description

This structure contains the Get UIM Slot Response parameter.

Parameters

uimSlotSelected	[Mandatory] • UIM slot selected • See swidms_UimSlotSelectedTlv for more information – Bit to check in ParamPresenceMask - **1**
multiplexedSimMode	[Mandatory] • Multiplexed SIM Mode • See swidms_MultiplexedSimModeTlv for more information – Bit to check in ParamPresenceMask - **3**
autoUimSwitchMode	[Mandatory] • Auto UIM Switch Mode • See swidms_AutoUimSwitchModeTlv for more information – Bit to check in ParamPresenceMask - **4**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1084.2 Field Documentation

8.1084.2.1 swidms_AutoUimSwitchModeTlv unpack_swidms_SLQSSwiDmsGetUimSlot_t::autoUimSwitchMode

8.1084.2.2 swidms_MultiplexedSimModeTlv unpack_swidms_SLQSSwiDmsGetUimSlot_t::multiplexedSimMode

8.1084.2.3 swi_uint256_t unpack_swidms_SLQSSwiDmsGetUimSlot_t::ParamPresenceMask

8.1084.2.4 swidms_UimSlotSelectedTlv unpack_swidms_SLQSSwiDmsGetUimSlot_t::uimSlotSelected

8.1085 unpack_swidms_SLQSSwiDmsGetUsbComp_t Struct Reference

Data Fields

- swidms_intfaceCfgTlv * pInterfaceCfg
- swidms_supportedIntBitmaskTlv * pSupportedBitmasks
- swi_uint256_t ParamPresenceMask

8.1085.1 Detailed Description

This structure contains the get usb composition response

Parameters

pInterfaceCfg	• See swidms_intfaceCfgTlv for more information • Bit to check in ParamPresenceMask - **1**

pSupported-Bitmasks	[Optional] • See swidms_supportedIntBitmaskTlv for more information • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1085.2 Field Documentation

8.1085.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsGetUsbComp_t::ParamPresenceMask

8.1085.2.2 swidms_intfaceCfgTlv* unpack_swidms_SLQSSwiDmsGetUsbComp_t::pInterfaceCfg

8.1085.2.3 swidms_supportedIntBitmaskTlv* unpack_swidms_SLQSSwiDmsGetUsbComp_t::pSupportedBitmasks

8.1086 unpack_swidms_SLQSSwiDmsGetUsbNetNum_t Struct Reference

Data Fields

- uint8_t usbNetNum
- swi_uint256_t ParamPresenceMask

8.1086.1 Detailed Description

This structure contains usb net numbers to get from remote endpoint for QMAP configuration

Parameters

usbNetNum	• value of usb net numbers on the device • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1086.2 Field Documentation

8.1086.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsGetUsbNetNum_t::ParamPresenceMask

8.1086.2.2 uint8_t unpack_swidms_SLQSSwiDmsGetUsbNetNum_t::usbNetNum

8.1087 unpack_swidms_SLQSSwiDmsSetHWWatchdog_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1087.1 Detailed Description

This structure is used to store unpack_swidms_SLQSSwiDmsSetHWWatchdog_t parameters.

Parameters

TlvResult	• unpack Tlv result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1087.2 Field Documentation

8.1087.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsSetHWWatchdog_t::ParamPresenceMask

8.1087.2.2 uint16_t unpack_swidms_SLQSSwiDmsSetHWWatchdog_t::TlvResult

8.1088 unpack_swidms_SLQSSwiDmsSetMTU_t Struct Reference

Data Fields

- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.1088.1 Detailed Description

This structure is used to store unpack_swidms_SLQSSwiDmsSetMTU_t parameters.

Parameters

TlvResult	• unpack Tlv result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1088.2 Field Documentation

8.1088.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsSetMTU_t::ParamPresenceMask

8.1088.2.2 uint16_t unpack_swidms_SLQSSwiDmsSetMTU_t::TlvResult

8.1089 unpack_swidms_SLQSSwiDmsSetUimSlot_t Struct Reference

Data Fields

- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.1089.1 Detailed Description

This structure is used to store unpack_swidms_SLQSSwiDmsSetUimSlot_t parameters.

Parameters

TlvresultParamPresence-Mask	• unpack Tlv result • Bit to check in ParamPresenceMask - **2** • bitmask representation to indicate valid parameters.

8.1089.2 Field Documentation

8.1089.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsSetUimSlot_t::ParamPresenceMask

8.1089.2.2 uint16_t unpack_swidms_SLQSSwiDmsSetUimSlot_t::Tlvresult

8.1090 unpack_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1090.1 Detailed Description

This structure is used to store unpack_swidms_SLQSSwiDmsSetUsbComp_t parameters.

Parameters

TlvresultParamPresence-Mask	• unpack Tlv result • Bit to check in ParamPresenceMask - **2** • bitmask representation to indicate valid parameters.

8.1090.2 Field Documentation

8.1090.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsSetUsbComp_t::ParamPresenceMask

8.1090.2.2 uint16_t unpack_swidms_SLQSSwiDmsSetUsbComp_t::Tlvresult

8.1091 unpack_swiloc_SwiLocGetAutoStart_t Struct Reference

Data Fields

- uint8_t function
- int function_reported

- uint8_t fix_type
- int fix_type_reported
- uint8_t max_time
- int max_time_reported
- uint32_t max_dist
- int max_dist_reported
- uint32_t fix_rate
- int fix_rate_reported
- swi_uint256_t ParamPresenceMask

8.1091.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

function	• Setting to indicate when modem should start an automatic GNSS fix – 0 - disabled – 1 - At bootup – 2 - When NMEA port is opened • Bit to check in ParamPresenceMask - **16**
function_reported	• 0 - not reported by modem • I - reported by modem
fix_type	• Type of GNSS fix: – 1 - Default Engine mode – 2 - MS-Based – 3 - MS-Assisted – 4 - Standalone • Bit to check in ParamPresenceMask - **17**
fix_type_reported	• 0 - not reported by modem • I - reported by modem
max_time	• Maximum time allowed for the receiver to get a fix in seconds • Valid range: 1-255 • Bit to check in ParamPresenceMask - **18**
max_time_reported	• 0 - not reported by modem • I - reported by modem
max_dist	• Maximum uncertainty of a fix measured by distance in meters • Valid range: 1 - 4294967280 • Bit to check in ParamPresenceMask - **19**

max_dist_-reported	• 0 - not reported by modem • l - reported by modem
fix_rate	• Time between fixes in seconds • Valid range: 1–65535 • Bit to check in ParamPresenceMask - **20**
fix_rate_-reported	• 0 - not reported by modem • l - reported by modem

8.1091.2 Field Documentation

8.1091.2.1 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::fix_rate

8.1091.2.2 int unpack_swiloc_SwiLocGetAutoStart_t::fix_rate_reported

8.1091.2.3 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::fix_type

8.1091.2.4 int unpack_swiloc_SwiLocGetAutoStart_t::fix_type_reported

8.1091.2.5 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::function

8.1091.2.6 int unpack_swiloc_SwiLocGetAutoStart_t::function_reported

8.1091.2.7 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::max_dist

8.1091.2.8 int unpack_swiloc_SwiLocGetAutoStart_t::max_dist_reported

8.1091.2.9 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::max_time

8.1091.2.10 int unpack_swiloc_SwiLocGetAutoStart_t::max_time_reported

8.1091.2.11 swi_uint256_t unpack_swiloc_SwiLocGetAutoStart_t::ParamPresenceMask

8.1092 unpack_swioma_SLQSOMADMAlertCallback_ind_t Struct Reference

Data Fields

- uint32_t eventType
- unpack_omaDmFotaTlv_t SessionInfoFota
- unpack_omaDmConfigTlv_t SessionInfoConfig
- unpack_omaDmNotificationsTlv_t SessionInfoNotification
- swi_uint256_t ParamPresenceMask

8.1092.1 Detailed Description

Structure that contains OMA indication information based on eventType Strucutres for which the event is not valid will have values set to 0

Parameters

eventType	• 0x00 - SWIOMA-DM FOTA • 0x01 - SWIOMA-DM Config • 0x02 - SWIOMA-DM Notification • 0xff - indication missing event information.
SessionInfo-Fota[OUT]	• See unpack_omaDmFotaTlv_t for more information • Bit to check in ParamPresenceMask - **16**
SessionInfo-Config[OUT]	• See unpack_omaDmConfigTlv_t for more information • Bit to check in ParamPresenceMask - **17**
SessionInfo-Notification[OUT]	• See unpack_omaDmNotificationsTlv_t for more information • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1092.2 Field Documentation

8.1092.2.1 uint32_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::eventType

8.1092.2.2 swi_uint256_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::ParamPresenceMask

8.1092.2.3 unpack_omaDmConfigTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoConfig

8.1092.2.4 unpack_omaDmFotaTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoFota

8.1092.2.5 unpack_omaDmNotificationsTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfo-Notification

8.1093 unpack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference

Data Fields

- uint8_t Status
- uint16_t UpdateCompleteStatus
- uint8_t Severity
- uint16_t SourceLength
- uint8_t Source [255]
- uint16_t PkgNameLength
- uint8_t PkgName [255]
- uint16_t PkgDescLength
- uint8_t PkgDescription [255]
- uint16_t DateLength
- uint8_t Date [255]
- uint16_t TimeLength
- uint8_t Time [255]

- uint8_t SessionType
- uint8_t SessionState
- uint16_t RetryCount
- swi_uint256_t ParamPresenceMask

8.1093.1 Detailed Description

Structure that contains the session type for OMA get session info unpack command Also used as input parameter to specify the size of variable parameters. (ref. notes)

Parameters

Status	• 1 Byte parameter indicating status – 0x01 - No Firmware available – 0x02 - Query Firmware Download – 0x03 - Firmware Downloading – 0x04 - Firmware Downloaded – 0x05 - Query Firmware Update – 0x06 - Firmware Updating – 0x07 - Firmware Updated • Bit to check in ParamPresenceMask - **16**
Update-CompleteStatus	• 2 byte parameter indicating Update Complete Status – See qaGobiApiTableSwiOMADMUpdateCompleteStatus.h Update Complete Status • Bit to check in ParamPresenceMask - **16**
Severity	• 1 byte parameter indicating severity – 0x01 - Mandatory – 0x02 - Optional • Bit to check in ParamPresenceMask - **16**
SourceLength	• 2 byte parameter indicating Length of Vendor Name String in Bytes. • Bit to check in ParamPresenceMask - **16**
Source	• Variable length parameter indicating Vendor Name in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information • Bit to check in ParamPresenceMask - **16**
PkgNameLength	• 2 byte parameter indicating Length of Package Name String in Bytes. • Bit to check in ParamPresenceMask - **16**
PkgName	• Variable length parameter indicating Package Name in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information • Bit to check in ParamPresenceMask - **16**

PkgDescLength	• 2 byte parameter indicating Length of Package Description String in Bytes. • Bit to check in ParamPresenceMask - **16**
PkgDescription	• Variable length parameter indicating Package Description in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information • Bit to check in ParamPresenceMask - **16**
DateLength	• 2 byte parameter indicating Length of Package Description String in Bytes. • Bit to check in ParamPresenceMask - **16**
Date	• Variable length parameter indicating Package Description in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information • Bit to check in ParamPresenceMask - **16**
TimeLength	• 2 byte parameter indicating Length of Time String in Bytes. • Bit to check in ParamPresenceMask - **16**
Time	• Variable length parameter indicating Time String in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information • Bit to check in ParamPresenceMask - **16**
SessionType	• 1 byte parameter reflects the last session started for Sprint – 0x00 - No session since boot – 0x01 - Sprint CI-DC Session – 0x02 - Sprint CI-PRL Session – 0x03 - Sprint CI-FUMO Session – 0x04 - Sprint HFA-DC Session – 0x05 - Sprint HFA-PRL Session – 0x06 - Sprint HFA-FUMO Session – 0x07 - Sprint NI Session • Bit to check in ParamPresenceMask - **16**
SessionState	• 1 byte parameter indicating session state – 0x01 - idle – 0x02 - active – 0x03 - pending • Bit to check in ParamPresenceMask - **16**
RetryCount	• 1 byte parameter indicating retries left count – valid values 0 to 6 • Bit to check in ParamPresenceMask - **16**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1093.2 Field Documentation

8.1093.2.1 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Date[255]

8.1093.2.2 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::DateLength

8.1093.2.3 swi_uint256_t unpack_swioma_SLQSOMADMGetSessionInfo_t::ParamPresenceMask

8.1093.2.4 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescLength

8.1093.2.5 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescription[255]

8.1093.2.6 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgName[255]

8.1093.2.7 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgNameLength

8.1093.2.8 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::RetryCount

8.1093.2.9 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionState

8.1093.2.10 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.1093.2.11 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Severity

8.1093.2.12 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Source[255]

8.1093.2.13 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SourceLength

8.1093.2.14 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Status

8.1093.2.15 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Time[255]

8.1093.2.16 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::TimeLength

8.1093.2.17 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::UpdateCompleteStatus

8.1094 unpack_swioma_SLQSOMADMGetSessionInfoExt_t Struct Reference

Data Fields

- uint8_t status
- uint8_t sessionState
- uint8_t hfaStatus
- uint16_t hfaMaxRetry
- uint16_t hfaRetryInterval
- uint16_t hfaRetryIndex
- uint8_t fumoState
- uint32_t fumoResultCode
- uint16_t pkgVendorNameLength
- uint8_t pkgVendorName [512]

- uint32_t pkgSize
- uint16_t pkgVersionNameLength
- uint8_t pkgVersionName [512]
- uint16_t pkgNameLength
- uint8_t pkgName [512]
- uint16_t pkgDescLength
- uint8_t pkgDesc [512]
- uint16_t pkgDateLength
- uint8_t pkgDate [512]
- uint16_t pkgInstallTimeLength
- uint8_t pkgInstallTime [512]
- swi_uint256_t ParamPresenceMask

8.1094.1 Detailed Description

Structure containing info for OMADM session Bit to check in ParamPresenceMask - **16**

Parameters

status	• 1 Byte parameter indicating status of OMADM initialization 0 - OMA module initiation not completed. 1 - OMA module initiation completed but no UI is registered. 128 - OMA initiation completed. 255 - OMA initiation error.
sessionState	• 1 byte parameter indicating OMA session state 0 - OMA module in CIDC session. 1 - OMA module in CIFUMO session 2 - OMA module in device initiated FUMO session 3 - OMA module in FUMO resume session 4 - OMA module is sending FUMO report 5 - OMA module in device initiated DM session 6 - OMA module in HFA DC session 7 - OMA module in CI PRL session 8 - OMA module in device initiated PRL session 9 - OMA module in HFA PRL session 10 - OMA module in HFA FUMO session 11 - OMA module in NI PRL session 12 - OMA is storing configuration after DC/PRL session 13 - OMA module is blocked by UI 14 - there is a pending session 15 - OMA module in idle state
hfaStatus	• 1 byte parameter indicating session state 0 - HFA completed 1 - HFA not start 2 - HFA DC completed, but PRL is not completed 3 - HFA PRL completed, but FUMO is not completed
hfaMaxRetry	• 2 byte parameter indicating max retry for HFA 0 - HFA completed
hfaRetryInterval	• 2 byte parameter indicating retry interval in seconds for HFA 0 - HFA completed
hfaRetryIndex	• 2 byte parameter indicating number of retry currently ongoing 0 - no retrying 0xFFFF- Invalid value
fumoState	• 1 byte parameter indicating fumo state 0 - FUMO state READY 1 - FUMO state DISCOVERY 2 - FUMO state DISCOCONF 3 - FUMO state DOWNLOAD 4 - FUMO state RECEIVED 5 - FUMO state CONFIRM 6 - FUMO state CONFIRMED 7 - FUMO state UPDATE 8 - FUMO state UPDATING 9 - FUMO state UNUSABLE 10 - FUMO state REJECTED 11 - FUMO state REJECTCONF 12 - FUMO state FAILED 13 - FUMO state DENIED 14 - FUMO state UPDATED 15 - FUMO state FINCONF 16 - FUMO state FINISHED

fumoResultCode	• 4 byte parameter indicating FUMO update image installation result. 200 - Request succeeded 250 - the start of vendor specified success result code 299 - the end of vendor specified success result code 400 - Management client error 401 - User rejected operation 402 - Corrupted update package 403 - Wrong package for device 404 - Invalid package signature 405 - Update package not acceptable 406 - DL auth failure 407 - DL download timeout 408 - Unsupported operation 409 - Err not defined by other code 410 - Firmware update failed 411 - Malformed or bad DL URL 412 - DL server unavailable 450 - vendor specified client error start 499 - vendor specified client error end 500 - DL server error 501 - DL fails due to out of memory 502 - Update fails, out of memory 503 - DL fails due to net issues 550 - vendor defined DL server error start 599 - vendor defined DL server error end 0xFFFFFFFF - invalid value.
pkgVendor-NameLength	• 2 byte parameter indicating Length of package vendor String in WORDs
pkgVendorName	• Variable length parameter indicating Package Name in UCS2 • size in bytes is 2∗pkgVendorNameLength
pkgSize	• 4 byte parameter indicating pkg size
pkgVersion-NameLength	• 2 byte parameter indicating Length of Package version Name String in WORDs
pkgVersion-Name	• Variable length parameter indicating Package version Name in UCS2 • size in bytes is 2∗pkgVersionNameLength
pkgNameLength	• 2 byte parameter indicating Length of Package Name String in WORDs
pkgName	• Variable length parameter indicating Package Name in UCS2 • size in bytes is 2∗pkgNameLength
pkgDescLength	• 2 byte parameter indicating Length of Package description String in WORDs
pkgDesc	• Variable length parameter indicating Package description in UCS2 • size in bytes is 2∗pkgDescLength
pkgDateLength	• 2 byte parameter indicating Length of Package date String in WORDs
pkgDate	• Variable length parameter indicating Package date in UCS2 • size in bytes is 2∗pkgDateLength
pkgInstallTime-Length	• 2 byte parameter indicating Length of Package install time String in WORDs

pkgInstallTime	• Variable length parameter indicating Package install time in UCS2 • size in bytes is 2∗pkgInstallTimeLength
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1094.2 Field Documentation

8.1094.2.1 uint32_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::fumoResultCode

8.1094.2.2 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::fumoState

8.1094.2.3 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::hfaMaxRetry

8.1094.2.4 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::hfaRetryIndex

8.1094.2.5 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::hfaRetryInterval

8.1094.2.6 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::hfaStatus

8.1094.2.7 swi_uint256_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::ParamPresenceMask

8.1094.2.8 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgDate[512]

8.1094.2.9 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgDateLength

8.1094.2.10 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgDesc[512]

8.1094.2.11 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgDescLength

8.1094.2.12 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgInstallTime[512]

8.1094.2.13 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgInstallTimeLength

8.1094.2.14 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgName[512]

8.1094.2.15 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgNameLength

8.1094.2.16 uint32_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgSize

8.1094.2.17 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgVendorName[512]

8.1094.2.18 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgVendorNameLength

8.1094.2.19 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgVersionName[512]

8.1094.2.20 uint16_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::pkgVersionNameLength

8.1094.2.21 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::sessionState

8.1094.2.22 uint8_t unpack_swioma_SLQSOMADMGetSessionInfoExt_t::status

8.1095 unpack_swioma_SLQSOMADMGetSettings_t Struct Reference

Data Fields

- uint32_t OMADMEnabled
- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t Autosdm
- uint8_t FwAutoCheck
- swi_uint256_t ParamPresenceMask

8.1095.1 Detailed Description

Structure containing the OMA DM settings retrieved from the device

Parameters

OMADMEnabled	[out] • Optional 4 byte parameter indicating OMADM service enabled – 0x00000001 - Client-initiated device configuration – 0x00000002 - Network-initiated device configuration – 0x00000010 - Client-initiated FUMO – 0x00000020 - Network-initiated FUMO • Bit to check in ParamPresenceMask - **16**
FOTAdownload	[out] • Optional 1 Byte parameter indicating support for FOTA Automatic download – 0x00 - Host permission required before downloading – 0x01 - Automatically start downloading, no host permission required – 0x02 - Automatically start downloading, while not roaming – 0x03 - Automatically reject download – 0x04 - Automatically reject download with “Enterprise Reject Policy” • Bit to check in ParamPresenceMask - **17**
FOTAUpdate	[out] • Optional 1 byte parameter indicating FOTA Automatic update – 0x00 - User permission required before updating firmware – 0x01 - No user permission required before updating firmware – 0x02 - User permission required, auto update on power up • Bit to check in ParamPresenceMask - **18**
Autosdm	[out] • Optional 1 byte parameter indicating OMA Automatic UI Alert Response – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject • Bit to check in ParamPresenceMask - **19**

FwAutoCheck	[out] • Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response – 0x00 - Disabled – 0x01 - Enabled • Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1095.2 Field Documentation

8.1095.2.1 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::Autosdm

8.1095.2.2 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAdownload

8.1095.2.3 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAUpdate

8.1095.2.4 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FwAutoCheck

8.1095.2.5 uint32_t unpack_swioma_SLQSOMADMGetSettings_t::OMADMEnabled

8.1095.2.6 swi_uint256_t unpack_swioma_SLQSOMADMGetSettings_t::ParamPresenceMask

8.1096 unpack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t FwAvailability
- swi_uint256_t ParamPresenceMask

8.1096.1 Detailed Description

Structure that contains the responses for OMA start session command

Parameters

FwAvailability	[out] • OMA-DM CHECK FW Available • Values – 0x00000001 - FW Available. For CIDC and CIPRL, this value will be returned by the modem. CIDC and CIPRL are asynchronous OMADM sessions. – 0x00000002 - FW Not Available – 0x00000003 - FW Check Timed Out • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1096.2 Field Documentation

8.1096.2.1 uint32_t unpack_swioma_SLQSOMADMStartSession_t::FwAvailability

8.1096.2.2 swi_uint256_t unpack_swioma_SLQSOMADMStartSession_t::ParamPresenceMask

8.1097 unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1097.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

Tlvresult	• Unpack Result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1097.2 Field Documentation

8.1097.2.1 swi_uint256_t unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t::ParamPresenceMask

8.1097.2.2 uint16_t unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t::Tlvresult

8.1098 unpack_tmd_SLQSTmdGetMitigationDevList_t Struct Reference

Data Fields

- uint8_t MitigationDevListLen
- tmd_mitigationDevList MitigationDevList [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1098.1 Detailed Description

This structure contains mitigation devices list from the remote endpoint

Parameters

MitigationDev-ListLen	• Mitigation Device List Length (Optional) • Number of sets of the following elements – MitigationDevList • Bit to check in ParamPresenceMask - **16**

pMitigationDevList	• Mitigation Device List (Optional) • See tmd_mitigationDevList for more information. • Bit to check in ParamPresenceMask - **16**
Tlvresult	• Unpack Result
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1098.2 Field Documentation

8.1098.2.1 tmd_mitigationDevList unpack_tmd_SLQSTmdGetMitigationDevList_t::MitigationDevList[255]

8.1098.2.2 uint8_t unpack_tmd_SLQSTmdGetMitigationDevList_t::MitigationDevListLen

8.1098.2.3 swi_uint256_t unpack_tmd_SLQSTmdGetMitigationDevList_t::ParamPresenceMask

8.1098.2.4 uint16_t unpack_tmd_SLQSTmdGetMitigationDevList_t::Tlvresult

8.1099 unpack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference

Data Fields

- uint8_t CurrentmitigationLvl
- uint8_t ReqMitigationLvl
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1099.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

CurrentmitigationLvl	• Current thermal mitigation level (Optional) – Bit to check in ParamPresenceMask - **16**
ReqMitigationLvl	• Requested Thermal Mitigation Level (Optional) • The requested thermal mitigation level from the client. The default is zero if the client has not previously set the mitigation level. – Bit to check in ParamPresenceMask - **17**
Tlvresult	• Unpack Result
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1099.2 Field Documentation

8.1099.2.1 uint8_t unpack_tmd_SLQSTmdGetMitigationLvl_t::CurrentmitigationLvl

8.1099.2.2 swi_uint256_t unpack_tmd_SLQSTmdGetMitigationLvl_t::ParamPresenceMask

8.1099.2.3 uint8_t unpack_tmd_SLQSTmdGetMitigationLvl_t::ReqMitigationLvl

8.1099.2.4 uint16_t unpack_tmd_SLQSTmdGetMitigationLvl_t::Tlvresult

8.1100 unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t Struct Reference

Data Fields

- uint8_t deviceIdLen
- char deviceID [255]
- uint8_t lvl
- swi_uint256_t ParamPresenceMask

8.1100.1 Detailed Description

Mitigation Level Report

Parameters

deviceIdLen	• Mitigation Device ID Length • Bit to check in ParamPresenceMask - **1**
deviceID	• Mitigation Device ID • Bit to check in ParamPresenceMask - **1**
lvl	• Current thermal mitigation level • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1100.2 Field Documentation

8.1100.2.1 char unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::deviceID[255]

8.1100.2.2 uint8_t unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::deviceIdLen

8.1100.2.3 uint8_t unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::lvl

8.1100.2.4 swi_uint256_t unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::ParamPresenceMask

8.1101 unpack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1101.1 Detailed Description

This structure contains mitigation devices Level request parameters

Parameters

Tlvresult	• Unpack Result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1101.2 Field Documentation

8.1101.2.1 swi_uint256_t unpack_tmd_SLQSTmdRegNotMitigationLvl_t::ParamPresenceMask

8.1101.2.2 uint16_t unpack_tmd_SLQSTmdRegNotMitigationLvl_t::Tlvresult

8.1102 unpack_uim_ChangePin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1102.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information. • Bit to check in ParamPresenceMask - **16**
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1102.2 Field Documentation

8.1102.2.1 swi_uint256_t unpack_uim_ChangePin_t::ParamPresenceMask

8.1102.2.2 uim_encryptedPIN1∗ unpack_uim_ChangePin_t::pEncryptedPIN1

8.1102.2.3 uint32_t∗ unpack_uim_ChangePin_t::pIndicationToken

8.1102.2.4 uim_remainingRetries∗ unpack_uim_ChangePin_t::pRemainingRetries

8.1102.2.5 uint16_t unpack_uim_ChangePin_t::Tlvresult

8.1103 unpack_uim_GetCardStatus_t Struct Reference

Data Fields

- uim_cardStatus ∗ pCardStatus
- uim_hotSwapStatus ∗ pHotSwapStatus
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1103.1 Detailed Description

This structure contains information of the response parameters associated with a Get Card Status API.

Parameters

pCard-Status(optional)	• See uim_cardStatus for more information. • Bit to check in ParamPresenceMask - **16**
pHotSwap-Status(optional)	• See uim_hotSwapStatus for more information. • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1103.2 Field Documentation

8.1103.2.1 swi_uint256_t unpack_uim_GetCardStatus_t::ParamPresenceMask

8.1103.2.2 uim_cardStatus∗ unpack_uim_GetCardStatus_t::pCardStatus

8.1103.2.3 uim_hotSwapStatus∗ unpack_uim_GetCardStatus_t::pHotSwapStatus

8.1103.2.4 uint16_t unpack_uim_GetCardStatus_t::Tlvresult

8.1104 unpack_uim_GetCardStatusV2_t Struct Reference

Data Fields

- uim_cardStatus * pCardStatus
- uim_hotSwapStatus * pHotSwapStatus
- uim_validCardStatus * pValidCardStatus
- uim_simBusyStatus * pSimBusyStatus
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1104.1 Detailed Description

This structure contains information of the response parameters associated with a Get Card Status API.

Parameters

pCard-Status(optional)	• See uim_cardStatus for more information. • Bit to check in ParamPresenceMask - **16**
pHotSwap-Status(optional)	• See uim_hotSwapStatus for more information. • Bit to check in ParamPresenceMask - **17**
pValidCard-Status(optional)	• See uim_validCardStatus for more information. • Bit to check in ParamPresenceMask - **18**
pSimBusy-Status(optional)	• See uim_simBusyStatus for more information. • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1104.2 Field Documentation

8.1104.2.1 swi_uint256_t unpack_uim_GetCardStatusV2_t::ParamPresenceMask

8.1104.2.2 uim_cardStatus* unpack_uim_GetCardStatusV2_t::pCardStatus

8.1104.2.3 uim_hotSwapStatus* unpack_uim_GetCardStatusV2_t::pHotSwapStatus

8.1104.2.4 uim_simBusyStatus* unpack_uim_GetCardStatusV2_t::pSimBusyStatus

8.1104.2.5 uim_validCardStatus* unpack_uim_GetCardStatusV2_t::pValidCardStatus

8.1104.2.6 uint16_t unpack_uim_GetCardStatusV2_t::Tlvresult

8.1105 unpack_uim_ReadTransparent_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_readResult * pReadResult
- uint32_t * pIndicationToken
- uint8_t * pEncryptedData
- uint16_t TlvresultTlvresult
- swi_uint256_t ParamPresenceMask

8.1105.1 Detailed Description

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

pCardResult	• See uim_cardResult for more information. • Bit to check in ParamPresenceMask - **16**
pReadResult	• See uim_readResult for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**
pEncrypted-Data(optional)	• Encrypted Data. • Indicates whether the data from the card passed in read_result is encrypted. • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1105.2 Field Documentation

8.1105.2.1 swi_uint256_t unpack_uim_ReadTransparent_t::ParamPresenceMask

8.1105.2.2 uim_cardResult* unpack_uim_ReadTransparent_t::pCardResult

8.1105.2.3 uint8_t* unpack_uim_ReadTransparent_t::pEncryptedData

8.1105.2.4 uint32_t* unpack_uim_ReadTransparent_t::pIndicationToken

8.1105.2.5 uim_readResult* unpack_uim_ReadTransparent_t::pReadResult

8.1105.2.6 uint16_t unpack_uim_ReadTransparent_t::Tlvresult

8.1106 unpack_uim_SetPinProtection_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.1106.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information. • Bit to check in ParamPresenceMask - **16**
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1106.2 Field Documentation

8.1106.2.1 swi_uint256_t unpack_uim_SetPinProtection_t::ParamPresenceMask

8.1106.2.2 uim_encryptedPIN1* unpack_uim_SetPinProtection_t::pEncryptedPIN1

8.1106.2.3 uint32_t* unpack_uim_SetPinProtection_t::pIndicationToken

8.1106.2.4 uim_remainingRetries* unpack_uim_SetPinProtection_t::pRemainingRetries

8.1106.2.5 uint16_t unpack_uim_SetPinProtection_t::TlvResult

8.1107 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference

Data Fields

- slots_t slotsstatusChange
- uint8_t bNumberOfPhySlots
- swi_uint256_t ParamPresenceMask

8.1107.1 Detailed Description

Structure consist of card status params

Parameters

slotsstatus-Change	• See slot_t for more information • Bit to check in ParamPresenceMask - **16**
bNumberOfPhy-Slots	• Number of Physical Slot(s) • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1107.2 Field Documentation

8.1107.2.1 uint8_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::bNumberOfPhySlots

8.1107.2.2 swi_uint256_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::ParamPresenceMask

8.1107.2.3 slots_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::slotsstatusChange

8.1108 unpack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_authenticateResult * pAuthenticateResult
- uint32_t * pIndicationToken
- swi_uint256_t ParamPresenceMask

8.1108.1 Detailed Description

This structure contains information of the response parameters associated with a SLQSUIMAuthenticate.

Parameters

pCard-Result(optional)	• See uim_cardResult for more information. • Bit to check in ParamPresenceMask - **16**
pAuthenticate-Result(optional)	• See uim_authenticateResult for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1108.2 Field Documentation

8.1108.2.1 swi_uint256_t unpack_uim_SLQSUIMAuthenticate_t::ParamPresenceMask

8.1108.2.2 uim_authenticateResult* unpack_uim_SLQSUIMAuthenticate_t::pAuthenticateResult

8.1108.2.3 uim_cardResult* unpack_uim_SLQSUIMAuthenticate_t::pCardResult

8.1108.2.4 uint32_t* unpack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.1109 unpack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- swi_uint256_t ParamPresenceMask

8.1109.1 Detailed Description

This structure contains information of the response parameters associated with a Depersonalization.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1109.2 Field Documentation

8.1109.2.1 swi_uint256_t unpack_uim_SLQSUIMDepersonalization_t::ParamPresenceMask

8.1109.2.2 uim_remainingRetries* unpack_uim_SLQSUIMDepersonalization_t::pRemainingRetries

8.1110 unpack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask
- swi_uint256_t ParamPresenceMask

8.1110.1 Detailed Description

This structure contains unpack event register parameter.

Parameters

eventMask	- bit 0 - card status • bit 1 - SAP connection • bit 4 - physical slot status

- Bit to check in ParamPresenceMask - **16**

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1110.2 Field Documentation

8.1110.2.1 uint32_t unpack_uim_SLQSUIMEventRegister_t::eventMask

8.1110.2.2 swi_uint256_t unpack_uim_SLQSUIMEventRegister_t::ParamPresenceMask

8.1111 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint8_t * pAutoSelection
- uim_personalizationStatus * pPersonalizationStatus
- uint8_t * pHaltSubscription
- swi_uint256_t ParamPresenceMask

8.1111.1 Detailed Description

This structure contains information of the response parameters associated with Get Configuration.

Parameters

pAuto-Selection(optional)	• Indicates whether the modem is configured to automatically select the provisioning sessions at powerup. • Valid values – 0 - Automatic provisioning is off – 1 - Automatic provisioning is on • Bit to check in ParamPresenceMask - **16**
p-Personalization-Status(optional)	• See uim_personalizationStatus for more information. • Bit to check in ParamPresenceMask - **17**

pHalt-Subscription(optional)	• Indicates if the modem is configured to publish the subscription after successful initialization. • Valid values – 0 - Modem proceeds with publishing the subscription – 1 - Modem does not publish the subscription • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1111.2 Field Documentation

8.1111.2.1 swi_uint256_t unpack_uim_SLQSUIMGetConfiguration_t::ParamPresenceMask

8.1111.2.2 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pAutoSelection

8.1111.2.3 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pHaltSubscription

8.1111.2.4 uim_personalizationStatus* unpack_uim_SLQSUIMGetConfiguration_t::pPersonalizationStatus

8.1112 unpack_uim_SLQSUIMGetEID_t Struct Reference

Data Fields

- uim_eidInfo * pEidInfo
- swi_uint256_t ParamPresenceMask

8.1112.1 Detailed Description

This structure contains the UIM Get EID Information.

Parameters

pEindInfo	• See uim_eidInfo for more information. • Bit to check in ParamPresenceMask - **16**

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.1112.2 Field Documentation

8.1112.2.1 swi_uint256_t unpack_uim_SLQSUIMGetEID_t::ParamPresenceMask

8.1112.2.2 uim_eidInfo* unpack_uim_SLQSUIMGetEID_t::pEidInfo

8.1113 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_fileAttributes * pFileAttributes
- uint32_t * pIndicationToken
- swi_uint256_t ParamPresenceMask

8.1113.1 Detailed Description

This structure contains information of the response parameters associated with a Get File Attributes

Parameters

pCard-Result(optional)	• See uim_cardResult for more information. • Bit to check in ParamPresenceMask - **16**
pFile-Attributes(optional)	• See uim_fileAttributes for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1113.2 Field Documentation

8.1113.2.1 swi_uint256_t unpack_uim_SLQSUIMGetFileAttributes_t::ParamPresenceMask

8.1113.2.2 uim_cardResult* unpack_uim_SLQSUIMGetFileAttributes_t::pCardResult

8.1113.2.3 uim_fileAttributes* unpack_uim_SLQSUIMGetFileAttributes_t::pFileAttributes

8.1113.2.4 uint32_t* unpack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.1114 unpack_uim_SLQSUIMGetProfileInfo_t Struct Reference

Data Fields

- uim_profileICCIDTlv * pProfileICCID
- uim_profileStateTlv * pProfileState
- uim_profileNicknameTlv * pProfileNickname
- uim_profileSPNTlv * pProfileSPN
- uim_profileNameTlv * pProfileName
- uim_profileIconTypeTlv * pProfileIconType
- uim_profileIconTlv * pProfileIcon
- uim_profileClassTlv * pProfileClass
- uim_profilePolicyRulesTlv * pProfilePolicyRules
- swi_uint256_t ParamPresenceMask

8.1114.1 Detailed Description

This structure contains information of the response parameters associated with profile information API.

Parameters

pProfileICCID	• See uim_profileICCIDTlv for more information. • Bit to check in ParamPresenceMask - **16**
pProfileState	• See uim_profileStateTlv for more information. • Bit to check in ParamPresenceMask - **17**
pProfile-Nickname	• See uim_profileNicknameTlv for more information. • Bit to check in ParamPresenceMask - **16**
pProfileSPN	• See uim_profileSPNTlv for more information. • Bit to check in ParamPresenceMask - **19**
pProfileName	• See uim_profileNameTlv for more information. • Bit to check in ParamPresenceMask - **20**
pProfileIconType	• See uim_profileIconTypeTlv for more information. • Bit to check in ParamPresenceMask - **21**
pProfileIcon	• See uim_profileIconTlv for more information. • Bit to check in ParamPresenceMask - **22**
pProfileClass	• See uim_profileClassTlv for more information. • Bit to check in ParamPresenceMask - **23**
pProfilePolicy-Rules	• See uim_profilePolicyRulesTlv for more information. • Bit to check in ParamPresenceMask - **24**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.1114.2 Field Documentation

8.1114.2.1 swi_uint256_t unpack_uim_SLQSUIMGetProfileInfo_t::ParamPresenceMask

8.1114.2.2 **uim_profileClassTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileClass

8.1114.2.3 **uim_profileICCIDTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileICCID

8.1114.2.4 **uim_profileIconTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileIcon

8.1114.2.5 **uim_profileIconTypeTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileIconType

8.1114.2.6 **uim_profileNameTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileName

8.1114.2.7 **uim_profileNicknameTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileNickname

8.1114.2.8 **uim_profilePolicyRulesTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfilePolicyRules

8.1114.2.9 **uim_profileSPNTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileSPN

8.1114.2.10 **uim_profileStateTlv** ∗ unpack_uim_SLQSUIMGetProfileInfo_t::pProfileState

8.1115 unpack_uim_SLQSUIMGetServiceStatus_t Struct Reference

Data Fields

- uim_UIMGetFDNStatus ∗ pFDNStatus
- uim_UIMGetHiddenKeyStatus ∗ pHiddenKeyStatus
- uim_UIMGetIndex ∗ pIndex
- swi_uint256_t ParamPresenceMask

8.1115.1 Detailed Description

This structure contains information of the response parameters associated with Get Services Status API.

Parameters

pFDN-Status(optional)	• See uim_UIMGetFDNStatus for more information. • Bit to check in ParamPresenceMask - **16**
pHiddenKey-Status(optional)	• See uim_UIMGetHiddenKeyStatus for more information. • Bit to check in ParamPresenceMask - **17**
pIndex(optional)	• See uim_UIMGetIndex for more information. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.1115.2 Field Documentation

8.1115.2.1 swi_uint256_t unpack_uim_SLQSUIMGetServiceStatus_t::ParamPresenceMask

8.1115.2.2 uim_UIMGetFDNStatus* unpack_uim_SLQSUIMGetServiceStatus_t::pFDNStatus

8.1115.2.3 uim_UIMGetHiddenKeyStatus* unpack_uim_SLQSUIMGetServiceStatus_t::pHiddenKeyStatus

8.1115.2.4 uim_UIMGetIndex* unpack_uim_SLQSUIMGetServiceStatus_t::pIndex

8.1116 unpack_uim_SLQSUIMGetSIMProfile_t Struct Reference

Data Fields

- uim_getSimProfileInfo * pProfileInfo
- uim_activeProfileInfo * pActiveProfileInfo
- swi_uint256_t ParamPresenceMask

8.1116.1 Detailed Description

This structure contains information of the response parameters associated with profile information from the card.

Parameters

pProfileInfo	• See uim_getSimProfileInfo for more information. • Bit to check in ParamPresenceMask - **16**
pActiveProfile-Info	• See uim_activeProfileInfo for more information. – Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.1116.2 Field Documentation

8.1116.2.1 uim_activeProfileInfo* unpack_uim_SLQSUIMGetSIMProfile_t::pActiveProfileInfo

8.1116.2.2 swi_uint256_t unpack_uim_SLQSUIMGetSIMProfile_t::ParamPresenceMask

8.1116.2.3 uim_getSimProfileInfo* unpack_uim_SLQSUIMGetSIMProfile_t::pProfileInfo

8.1117 unpack_uim_SLQSUIMGetSlotsStatus_t Struct Reference

Data Fields

- uint8_t * pNumberOfPhySlot
- slots_t * pUimSlotsStatus
- swi_uint256_t ParamPresenceMask

8.1117.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

pNumberOfPhySlot	• Number of sets of the Slot Status. • Bit to check in ParamPresenceMask - **16**
pUimSlotsStatus	• Slots Status See slots_t for more information.. • Bit to check in ParamPresenceMask - **16**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1117.2 Field Documentation

8.1117.2.1 swi_uint256_t unpack_uim_SLQSUIMGetSlotsStatus_t::ParamPresenceMask

8.1117.2.2 uint8_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pNumberOfPhySlot

8.1117.2.3 slots_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pUimSlotsStatus

8.1118 unpack_uim_SLQSUIMGetSlotsStatusV2_t Struct Reference

Data Fields

- uim_GetSlotsStatusTlv * pGetSlotsStatusTlv
- uim_GetSlotsInfoTlv * pGetSlotsInfoTlv
- swi_uint256_t ParamPresenceMask

8.1118.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

pNumberOfPhySlot	• Number of sets of the Slot Status. • Bit to check in ParamPresenceMask - **16**
pUimSlotsStatus	• Slots Status See slots_t for more information.. • Bit to check in ParamPresenceMask - **16**
pNumberOfPhySlotInfo	• Number of Physical slot information. • Bit to check in ParamPresenceMask - **17**

pUimSlotsStatus	• Slots Status See uim_physlotsInfo for more information.. • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1118.2 Field Documentation

8.1118.2.1 swi_uint256_t unpack_uim_SLQSUIMGetSlotsStatusV2_t::ParamPresenceMask

8.1118.2.2 uim_GetSlotsInfoTlv* unpack_uim_SLQSUIMGetSlotsStatusV2_t::pGetSlotsInfoTlv

8.1118.2.3 uim_GetSlotsStatusTlv* unpack_uim_SLQSUIMGetSlotsStatusV2_t::pGetSlotsStatusTlv

8.1119 unpack_uim_SLQSUIMReadRecord_t Struct Reference

Data Fields

- uim_cardResultInfo * pCardResult
- uim_readResultInfo * pReadResult
- uim_additionalReadResult * pAdditionalReadResult
- uim_indToken * pIndicationToken
- swi_uint256_t ParamPresenceMask

8.1119.1 Detailed Description

This structure contains information of the response parameters associated with a Read Record API.

Parameters

pCardResult	• See uim_cardResultInfo for more information. • Bit to check in ParamPresenceMask - **16**
pReadResult	• See uim_readResultInfo for more information. • Bit to check in ParamPresenceMask - **17**
pAdditional-ReadResult	• See uim_additionalReadResult for more information. • Bit to check in ParamPresenceMask - **18**
pIndication-Token(optional)	• See uim_indToken for more information. • Bit to check in ParamPresenceMask - **19**

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.1119.2 Field Documentation

8.1119.2.1 uim_additionalReadResult* unpack_uim_SLQSUIMReadRecord_t::pAdditionalReadResult

8.1119.2.2 swi_uint256_t unpack_uim_SLQSUIMReadRecord_t::ParamPresenceMask

8.1119.2.3 uim_cardResultInfo* unpack_uim_SLQSUIMReadRecord_t::pCardResult

8.1119.2.4 uim_indToken* unpack_uim_SLQSUIMReadRecord_t::pIndicationToken

8.1119.2.5 uim_readResultInfo* unpack_uim_SLQSUIMReadRecord_t::pReadResult

8.1120 unpack_uim_SLQSUIMRefreshCallback_Ind_t Struct Reference

Data Fields

- uint8_t TlvPresent
- uim_refreshevent refreshEvent
- swi_uint256_t ParamPresenceMask

8.1120.1 Detailed Description

This structure hold parameters about UIM refresh event indication.

Parameters

TlvPresent	• value is 1 if refresh event TLV is present in indication, otherwise 0
refreshEvent[OPTIONAL]	• see uim_refreshevent • Bit to check in ParamPresenceMask - **16**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1120.2 Field Documentation

8.1120.2.1 swi_uint256_t unpack_uim_SLQSUIMRefreshCallback_Ind_t::ParamPresenceMask

8.1120.2.2 uim_refreshevent unpack_uim_SLQSUIMRefreshCallback_Ind_t::refreshEvent

8.1120.2.3 uint8_t unpack_uim_SLQSUIMRefreshCallback_Ind_t::TlvPresent

8.1121 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_refreshevent * pRefreshEvent
- swi_uint256_t ParamPresenceMask

8.1121.1 Detailed Description

This structure contains information of the response parameters associated with a Refresh Get Last Event.

Parameters

refreshEvent(-Optional)	• See uim_refreshevent for more information. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1121.2 Field Documentation

8.1121.2.1 swi_uint256_t unpack_uim_SLQSUIMRefreshGetLastEvent_t::ParamPresenceMask

8.1121.2.2 uim_refreshevent* unpack_uim_SLQSUIMRefreshGetLastEvent_t::pRefreshEvent

8.1122 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference

Data Fields

- uim_cardStatus * pCardStatus
- swi_uint256_t ParamPresenceMask

8.1122.1 Detailed Description

This structure contains information about Status change callback.

Parameters

pCardStatus	Card Status • See uim_cardStatus for more information. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1122.2 Field Documentation

8.1122.2.1 swi_uint256_t unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t::ParamPresenceMask

8.1122.2.2 uim_cardStatus* unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t::pCardStatus

8.1123 unpack_uim_SLQSUIMWriteRecord_t Struct Reference

Data Fields

- uim_cardResultInfo * pCardResult
- uim_indToken * pIndicationToken

- swi_uint256_t ParamPresenceMask

8.1123.1 Detailed Description

This structure contains information of the response parameters associated with a Write Record API.

Parameters

pCardResult	• See uim_cardResultInfo for more information. • Bit to check in ParamPresenceMask - **16**
pIndication-Token(optional)	• See uim_indToken for more information. • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.1123.2 Field Documentation

8.1123.2.1 swi_uint256_t unpack_uim_SLQSUIMWriteRecord_t::ParamPresenceMask

8.1123.2.2 uim_cardResultInfo* unpack_uim_SLQSUIMWriteRecord_t::pCardResult

8.1123.2.3 uim_indToken* unpack_uim_SLQSUIMWriteRecord_t::pIndicationToken

8.1124 unpack_uim_SLQSUIMWriteTransparent_t Struct Reference

Data Fields

- uim_cardResultInfo * pCardResult
- uim_indToken * pIndicationToken
- swi_uint256_t ParamPresenceMask

8.1124.1 Detailed Description

This structure contains information of the response parameters associated with a Write Transparent API.

Parameters

pCard-Result(optional)	• See uim_cardResultInfo for more information. • Bit to check in ParamPresenceMask - **16**
pIndication-Token(optional)	• See uim_indToken for more information. • Bit to check in ParamPresenceMask - **17**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.1124.2 Field Documentation

8.1124.2.1 swi_uint256_t unpack_uim_SLQSUIMWriteTransparent_t::ParamPresenceMask

8.1124.2.2 uim_cardResultInfo* unpack_uim_SLQSUIMWriteTransparent_t::pCardResult

8.1124.2.3 uim_indToken* unpack_uim_SLQSUIMWriteTransparent_t::pIndicationToken

8.1125 unpack_uim_UnblockPin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.1125.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information. • Bit to check in ParamPresenceMask - **16**
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1125.2 Field Documentation

8.1125.2.1 swi_uint256_t unpack_uim_UnblockPin_t::ParamPresenceMask

8.1125.2.2 uim_encryptedPIN1* unpack_uim_UnblockPin_t::pEncryptedPIN1

8.1125.2.3 uint32_t* unpack_uim_UnblockPin_t::pIndicationToken

8.1125.2.4 uim_remainingRetries* unpack_uim_UnblockPin_t::pRemainingRetries

8.1125.2.5 uint16_t unpack_uim_UnblockPin_t::Tlvresult

8.1126 unpack_uim_UnblockPinV2_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uim_cardResult * pCardResult
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1126.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information. • Bit to check in ParamPresenceMask - **16**
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**
pCard-Result(optional)	• See uim_cardResult for more information. • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1126.2 Field Documentation

8.1126.2.1 swi_uint256_t unpack_uim_UnblockPinV2_t::ParamPresenceMask

8.1126.2.2 uim_cardResult∗ unpack_uim_UnblockPinV2_t::pCardResult

8.1126.2.3 uim_encryptedPIN1∗ unpack_uim_UnblockPinV2_t::pEncryptedPIN1

8.1126.2.4 uint32_t∗ unpack_uim_UnblockPinV2_t::pIndicationToken

8.1126.2.5 uim_remainingRetries∗ unpack_uim_UnblockPinV2_t::pRemainingRetries

8.1126.2.6 uint16_t unpack_uim_UnblockPinV2_t::TlvresultTlvresult

8.1127 unpack_uim_VerifyPin_t Struct Reference

Data Fields

- uim_remainingRetries ∗ pRemainingRetries
- uim_encryptedPIN1 ∗ pEncryptedPIN1
- uint32_t ∗ pIndicationToken
- uint16_t TlvresultTlvresult
- swi_uint256_t ParamPresenceMask

8.1127.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information. • Bit to check in ParamPresenceMask - **16**
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information. • Bit to check in ParamPresenceMask - **17**
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1127.2 Field Documentation

8.1127.2.1 swi_uint256_t unpack_uim_VerifyPin_t::ParamPresenceMask

8.1127.2.2 uim_encryptedPIN1∗ unpack_uim_VerifyPin_t::pEncryptedPIN1

8.1127.2.3 uint32_t∗ unpack_uim_VerifyPin_t::pIndicationToken

8.1127.2.4 uim_remainingRetries∗ unpack_uim_VerifyPin_t::pRemainingRetries

8.1127.2.5 uint16_t unpack_uim_VerifyPin_t::TlvresultTlvresult

8.1128 unpack_voice_allCallStatusCallback_ind_t Struct Reference

Data Fields

- voice_arrCallInfo arrCallInfomation
- voice_arrRemotePartyNum * pArrRemotePartyNum
- voice_arrRemotePartyName * pArrRemotePartyName
- voice_arrAlertingType * pArrAlertingType
- voice_arrSvcOption * pArrSvcOption
- voice_arrCallEndReason * pArrCallEndReason
- voice_arrAlphaID * pArrAlphaID
- voice_arrConnectPartyNum * pArrConnectPartyNum
- voice_arrDiagInfo * pArrDiagInfo
- voice_arrCalledPartyNum * pArrCalledPartyNum
- voice_arrRedirPartyNum * pArrRedirPartyNum
- voice_arrAlertingPattern * pArrAlertingPattern
- swi_uint256_t ParamPresenceMask

8.1128.1 Detailed Description

This structure contains information about the indication parameters for all the calls originating or terminating from a particular device.

Parameters

arrCallInfo(mandatory)	• See voice_arrCallInfo for more information. • Bit to check in ParamPresenceMask - **1**
pArrRemotePartyNum(optional)	• See voice_arrRemotePartyNum for more information. • Bit to check in ParamPresenceMask - **16**
pArrRemotePartyName(optional)	• See voice_arrRemotePartyName for more information. • Bit to check in ParamPresenceMask - **17**
pArrAlertingType(optional)	• See voice_arrAlertingType for more information. • Bit to check in ParamPresenceMask - **18**
pArrSvcOption(optional)	• See voice_arrSvcOption for more information. • Bit to check in ParamPresenceMask - **19**
pArrCallEndReason(optional)	• See voice_arrCallEndReason for more information. • Bit to check in ParamPresenceMask - **20**
pArrAlphaID(optional)	• See voice_arrAlphaID for more information. • Bit to check in ParamPresenceMask - **21**

pArrConnect-Party-Num(optional)	• See voice_arrConnectPartyNum for more information. • Bit to check in ParamPresenceMask - **22**
pArrDiag-Info(optional)	• See voice_arrDiagInfo for more information. • Bit to check in ParamPresenceMask - **23**
pArrCalledParty-Num(optional)	• See voice_arrCalledPartyNum for more information. • Bit to check in ParamPresenceMask - **24**
pArrRedirParty-Num(optional)	• See voice_arrRedirPartyNum for more information. • Bit to check in ParamPresenceMask - **25**
pArrAlerting-Pattern(optional)	• See voice_arrAlertingPattern for more information. • Bit to check in ParamPresenceMask - **26**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1128.2 Field Documentation

8.1128.2.1 voice_arrCallInfo unpack_voice_allCallStatusCallback_ind_t::arrCallInfomation

8.1128.2.2 swi_uint256_t unpack_voice_allCallStatusCallback_ind_t::ParamPresenceMask

8.1128.2.3 voice_arrAlertingPattern* unpack_voice_allCallStatusCallback_ind_t::pArrAlertingPattern

8.1128.2.4 voice_arrAlertingType* unpack_voice_allCallStatusCallback_ind_t::pArrAlertingType

8.1128.2.5 voice_arrAlphaID* unpack_voice_allCallStatusCallback_ind_t::pArrAlphaID

8.1128.2.6 voice_arrCalledPartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrCalledPartyNum

8.1128.2.7 voice_arrCallEndReason* unpack_voice_allCallStatusCallback_ind_t::pArrCallEndReason

8.1128.2.8 voice_arrConnectPartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrConnectPartyNum

8.1128.2.9 voice_arrDiagInfo* unpack_voice_allCallStatusCallback_ind_t::pArrDiagInfo

8.1128.2.10 voice_arrRedirPartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrRedirPartyNum

8.1128.2.11 voice_arrRemotePartyName* unpack_voice_allCallStatusCallback_ind_t::pArrRemotePartyName

8.1128.2.12 voice_arrRemotePartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrRemotePartyNum

8.1128.2.13 voice_arrSvcOption* unpack_voice_allCallStatusCallback_ind_t::pArrSvcOption

8.1129 unpack_voice_DTMFEventCallback_ind_t Struct Reference

Data Fields

- voice_DTMFInfo DTMFInformation
- uint8_t * pOnLength
- uint8_t * pOffLength
- swi_uint256_t ParamPresenceMask

8.1129.1 Detailed Description

This structure contains the parameters passed for DTMF event indication by the device.

Parameters

DTMF-Information(mandatory)	• See voice_DTMFInfo for more information. • Bit to check in ParamPresenceMask - **1**
pOn-Length(optional)	• DTMF Pulse Width – 0x00 - DTMF_ONLENGTH_95MS - 95 ms – 0x01 - DTMF_ONLENGTH_150MS - 150 ms – 0x02 - DTMF_ONLENGTH_200MS - 200 ms – 0x03 - DTMF_ONLENGTH_250MS - 250 ms – 0x04 - DTMF_ONLENGTH_300MS - 300 ms – 0x05 - DTMF_ONLENGTH_350MS - 350 ms – 0x06 - DTMF_ONLENGTH_SMS - SMS Tx special pulse width • Bit to check in ParamPresenceMask - **16**
pOff-Length(optional)	• DTMF Interdigit Interval – 0x00 - DTMF_OFFLENGTH_60MS - 60 ms – 0x01 - DTMF_OFFLENGTH_100MS - 100 ms – 0x02 - DTMF_OFFLENGTH_150MS - 150 ms – 0x03 - DTMF_OFFLENGTH_200MS - 200 ms • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.1129.2 Field Documentation

8.1129.2.1 voice_DTMFInfo unpack_voice_DTMFEventCallback_ind_t::DTMFInformation

8.1129.2.2 swi_uint256_t unpack_voice_DTMFEventCallback_ind_t::ParamPresenceMask

8.1129.2.3 uint8_t* unpack_voice_DTMFEventCallback_ind_t::pOffLength

8.1129.2.4 uint8_t* unpack_voice_DTMFEventCallback_ind_t::pOnLength

8.1130 unpack_voice_OTASPStatusCallback_ind_t Struct Reference

Data Fields

- uint8_t callID
- uint8_t OTASPStatus
- swi_uint256_t ParamPresenceMask

8.1130.1 Detailed Description

This structure consist of OTASP or OTAPA event params

Parameters

callID	• Call identifier for the call. • Bit to check in ParamPresenceMask - **1**
OTASPStatus	• OTASP status for the OTASP call. Values: – 0x00 - OTASP_STATUS_SPL_UNLOCKED.SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED. SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED.A-key exchanged; only for user-initiated O-TASP – 0x03 - OTASP_STATUS_SSD_UPDATED. SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP(OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP(OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP(OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP(OTAPA) • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1130.2 Field Documentation

8.1130.2.1 uint8_t unpack_voice_OTASPStatusCallback_ind_t::callID

8.1130.2.2 uint8_t unpack_voice_OTASPStatusCallback_ind_t::OTASPStatus

8.1130.2.3 swi_uint256_t unpack_voice_OTASPStatusCallback_ind_t::ParamPresenceMask

8.1131 unpack_voice_SLQSOriginateUSSD_t Struct Reference

Data Fields

- uint16_t * pfailureCause
- voice_alphaIDInfo * pAlphaIDInfo
- struct voice_USSInfo * pUSSDInfo
- uint8_t * pCcResultType
- uint8_t * pCallId
- voice_ccSUPSType * pCCSuppsType
- swi_uint256_t ParamPresenceMask

8.1131.1 Detailed Description

This structure contains the parameters of USS response.

Parameters

pfailureCause	• Failure Cause. • 0 - QMI_FAILURE_CAUSE_OFFLINE - Phone is offline • 20 - QMI_FAILURE_CAUSE_CDMA_LOCK - Phone is CDMA locked until a power cycle; CDMA only • 21 - QMI_FAILURE_CAUSE_NO_SRV - Phone has no service • 22 - QMI_FAILURE_CAUSE_FADE - Call has ended abnormally • 23 - QMI_FAILURE_CAUSE_INTERCEPT - Received intercept from the base station; originating only; CDMA only • 24 - QMI_FAILURE_CAUSE_REORDER - Received reorder from the base station; originating only; CDMA only • 25 - QMI_FAILURE_CAUSE_REL_NORMAL - Received release from the base station; no reason was given • 26 - QMI_FAILURE_CAUSE_REL_SO_REJ - Received release from the base station; SO reject; CDMA only • 27 - QMI_FAILURE_CAUSE_INCOM_CALL - Received incoming call from the base station • 28 - QMI_FAILURE_CAUSE_ALERT_STOP - Received alert stop from the base station; incoming only; CDMA only • 29 - QMI_FAILURE_CAUSE_CLIENT_END - Client ended the call • 30 - QMI_FAILURE_CAUSE_ACTIVATION - Received end activation; OTASP call only; CDMA only • 31 - QMI_FAILURE_CAUSE_MC_ABORT - MC aborted the origination/conversation; CDMA only • 32 - QMI_FAILURE_CAUSE_MAX_ACCESS_PROBE - Maximum access probes were transmitted; CDMA only • 33 - QMI_FAILURE_CAUSE_PSIST_N - Persistence test failure; FEATURE_JCDMA only; CDMA only • 34 - QMI_FAILURE_CAUSE_UIM_NOT_PRESENT - R-UIM is not present • 35 - QMI_FAILURE_CAUSE_ACC_IN_PROG - Access attempt is already in progress • 36 - QMI_FAILURE_CAUSE_ACC_FAIL - Access failure for a reason other than the above • 37 - QMI_FAILURE_CAUSE_RETRY_ORDER - Received retry order; originating only; IS 2000; CDMA only • 38 - QMI_FAILURE_CAUSE_CCS_NOT_SUPPORTED_BYBS - Concurrent service is not supported by the base station • 39 - QMI_FAILURE_CAUSE_NO_RESPONSE_FROM_BS - No response was received from the base station • 40 - QMI_FAILURE_CAUSE_REJECTED_BY_BS - Call was rejected by the base station; CDMA only • 41 - QMI_FAILURE_CAUSE_INCOMPATIBLE - Concurrent services requested were not compatible; CDMA only • 42 - QMI_FAILURE_CAUSE_ACCESS_BLOCK - Access is blocked by the base station; CDMA only • 43 - QMI_FAILURE_CAUSE_ALREADY_IN_TC - Corresponds to CM_CALL_ORIGERR_ALREADY_IN_TC • 44 - QMI_FAILURE_CAUSE_EMERGENCY_FLASHED - Call is ended because an emergency call was flashed over this call; CDMA only • 45 - QMI_FAILURE_CAUSE_USER_CALL_ORIGDURING_GPS - Used if CM is ending a GPS call in preference of a user call • 46 - QMI_FAILURE_CAUSE_USER_CALL_ORIGDURING_SMS - Used if CM is ending an SMS call in preference of a user call • 47 - QMI_FAILURE_CAUSE_USER_CALL_ORIGDURING_DATA - Used if CM is ending a data call in preference of an emergency call • 48 - QMI_FAILURE_CAUSE_REDIR_OR_HANDOFF - Call was rejected because of a redirection or handoff • 49 - QMI_FAILURE_CAUSE_ACCESS_BLOCK_ALL - Access is blocked by the base station for all mobiles; KDDI-specific; CDMA only • 50 - QMI_FAILURE_CAUSE_OTASP_SPC_ERR - To support OTASP SPC Error indication • 51 - QMI_FAILURE_CAUSE_IS707B_MAX_ACC - Maximum access probes for an IS-707B

AlphaIdentifier	• see voice_alphaIDInfo definition • Bit to check in ParamPresenceMask - **17**
pUSSDInfo	• USS Data from Network (See voice_USSInfo) • Bit to check in ParamPresenceMask - **18**
pCcResultType	• CC result code • Bit to check in ParamPresenceMask - **19**
pCallId	• Bit to check in ParamPresenceMask - **20** • NULL pointer - Invalid data.
pCCSuppsType	• See voice_ccSUPSType definition • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1131.2 Field Documentation

8.1131.2.1 voice_alphaIDInfo* unpack_voice_SLQSOriginateUSSD_t::pAlphaIDInfo

8.1131.2.2 swi_uint256_t unpack_voice_SLQSOriginateUSSD_t::ParamPresenceMask

8.1131.2.3 uint8_t* unpack_voice_SLQSOriginateUSSD_t::pCallId

8.1131.2.4 uint8_t* unpack_voice_SLQSOriginateUSSD_t::pCcResultType

8.1131.2.5 voice_ccSUPSType* unpack_voice_SLQSOriginateUSSD_t::pCCSuppsType

8.1131.2.6 uint16_t* unpack_voice_SLQSOriginateUSSD_t::pfailureCause

8.1131.2.7 struct voice_USSInfo* unpack_voice_SLQSOriginateUSSD_t::pUSSDInfo

8.1132 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId
- swi_uint256_t ParamPresenceMask

8.1132.1 Detailed Description

Contains the parameters passed for pack voice Answer Call.

Parameters

pCallId	• Unique call identifier for the call that must be answered. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1132.2 Field Documentation

8.1132.2.1 swi_uint256_t unpack_voice_SLQSVoiceAnswerCall_t::ParamPresenceMask

8.1132.2.2 uint8_t* unpack_voice_SLQSVoiceAnswerCall_t::pCallId

8.1133 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- swi_uint256_t ParamPresenceMask

8.1133.1 Detailed Description

This structure contains Voice Burst DTMF Information for unpack.

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. A burst DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1133.2 Field Documentation

8.1133.2.1 swi_uint256_t unpack_voice_SLQSVoiceBurstDTMF_t::ParamPresenceMask

8.1133.2.2 uint8_t* unpack_voice_SLQSVoiceBurstDTMF_t::pCallID

8.1134 unpack_voice_SLQSVoiceDialCall_t Struct Reference

Data Fields

- uint8_t * pCallID
- voice_alphaIDInfo * pAlphaIDInfo

- uint8_t * pCCResultType
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1134.1 Detailed Description

This structure contains Voice Call Response Parameters

Parameters

pCallID(optional)	• Unique call identifier for the dialed call • Bit to check in ParamPresenceMask - **16**
pAlphaID-Info(optional)	• Pointer to structure of alphaIDInfo – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **17**
pCCResult-Type(optional)	• Call Control Result Type. – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service • Bit to check in ParamPresenceMask - **18**
pCCSUPS-Type(optional)	• Pointer to structure of ccSUPSType • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1134.2 Field Documentation

8.1134.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceDialCall_t::pAlphaIDInfo

8.1134.2.2 swi_uint256_t unpack_voice_SLQSVoiceDialCall_t::ParamPresenceMask

8.1134.2.3 uint8_t* unpack_voice_SLQSVoiceDialCall_t::pCallID

8.1134.2.4 uint8_t* unpack_voice_SLQSVoiceDialCall_t::pCCResultType

8.1134.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceDialCall_t::pCCSUPSType

8.1135 unpack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t * pCallId

- swi_uint256_t ParamPresenceMask

8.1135.1 Detailed Description

This structure contains unpack voice end call parameter.

Parameters

pCallId	• Unique call identifier for the call that must be ended • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1135.2 Field Documentation

8.1135.2.1 swi_uint256_t unpack_voice_SLQSVoiceEndCall_t::ParamPresenceMask

8.1135.2.2 uint8_t* unpack_voice_SLQSVoiceEndCall_t::pCallId

8.1136 unpack_voice_SLQSVoiceGetAllCallInfo_t Struct Reference

Data Fields

- voice_arrCallInfo * pArrCallInfo
- voice_arrRemotePartyNum * pArrRemotePartyNum
- voice_arrRemotePartyName * pArrRemotePartyName
- voice_arrAlertingType * pArrAlertingType
- voice_arrUUSInfo * pArrUUSInfo
- voice_arrSvcOption * pArrSvcOption
- uint8_t * pOTASPStatus
- uint8_t * pVoicePrivacy
- voice_arrCallEndReason * pArrCallEndReason
- voice_arrAlphaID * pArrAlphaID
- voice_arrConnectPartyNum * pArrConnectPartyNum
- voice_arrDiagInfo * pArrDiagInfo
- voice_arrCalledPartyNum * pArrCalledPartyNum
- voice_arrRedirPartyNum * pArrRedirPartyNum
- voice_arrAlertingPattern * pArrAlertingPattern
- swi_uint256_t ParamPresenceMask

8.1136.1 Detailed Description

This structure contains information about the response parameters with all the calls originating or terminating from a particular device.

Parameters

pArrCallInfo(optional)	• See voice_arrCallInfo for more information. • Bit to check in ParamPresenceMask - **16**
pArrRemotePartyNum(optional)	• See voice_arrRemotePartyNum for more information. • Bit to check in ParamPresenceMask - **17**
pArrRemotePartyName(optional)	• See voice_arrRemotePartyName for more information. • Bit to check in ParamPresenceMask - **18**
pArrAlertingType(optional)	• See voice_arrAlertingType for more information. • Bit to check in ParamPresenceMask - **19**
pArrUUSInfo(optional)	• See voice_arrUUSInfo for more information. • Bit to check in ParamPresenceMask - **20**
pArrSvcOption(optional)	• See voice_arrSvcOption for more information. • Bit to check in ParamPresenceMask - **21**

pOTASP-Status(optional)	• OTASP status for the OTASP call. • Applicable only for 3GPP2 devices. – 0x00 - OTASP_STATUS_SPL_UNLOCKED - SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED - SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED - A-key exchanged; only for user-initiated OTASP – 0x03 - OTASP_STATUS_SSD_UPDATED - SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP (OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP (OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP (OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP (OTAPA) – 0xFF - Not Available • Bit to check in ParamPresenceMask - **22**
pVoice-Privacy(optional)	• Voice Privacy. • Values. – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy – 0xFF - Not Available • Bit to check in ParamPresenceMask - **23**
pArrCallEnd-Reason(optional)	• See voice_arrCallEndReason for more information. • Bit to check in ParamPresenceMask - **24**
pArrAlphaI-D(optional)	• See voice_arrAlphaID for more information. • Bit to check in ParamPresenceMask - **25**
pArrConnect-Party-Num(optional)	• See voice_arrConnectPartyNum for more information. • Bit to check in ParamPresenceMask - **26**

pArrDiag-Info(optional)	• See voice_arrDiagInfo for more information. • Bit to check in ParamPresenceMask - **27**
pArrCalledParty-Num(optional)	• See voice_arrCalledPartyNum for more information. • Bit to check in ParamPresenceMask - **28**
pArrRedirParty-Num(optional)	• See voice_arrRedirPartyNum for more information. • Bit to check in ParamPresenceMask - **29**
pArrAlerting-Pattern(optional)	• See voice_arrAlertingPattern for more information. • Bit to check in ParamPresenceMask - **30**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1136.2 Field Documentation

8.1136.2.1 swi_uint256_t unpack_voice_SLQSVoiceGetAllCallInfo_t::ParamPresenceMask

8.1136.2.2 voice_arrAlertingPattern* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingPattern

8.1136.2.3 voice_arrAlertingType* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingType

8.1136.2.4 voice_arrAlphaID* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlphaID

8.1136.2.5 voice_arrCalledPartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCalledPartyNum

8.1136.2.6 voice_arrCallEndReason* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallEndReason

8.1136.2.7 voice_arrCallInfo* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallInfo

8.1136.2.8 voice_arrConnectPartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrConnectPartyNum

8.1136.2.9 voice_arrDiagInfo* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrDiagInfo

8.1136.2.10 voice_arrRedirPartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRedirPartyNum

8.1136.2.11 voice_arrRemotePartyName* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyName

8.1136.2.12 voice_arrRemotePartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyNum

8.1136.2.13 voice_arrSvcOption* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrSvcOption

8.1136.2.14 voice_arrUUSInfo* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrUUSInfo

8.1136.2.15 uint8_t* unpack_voice_SLQSVoiceGetAllCallInfo_t::pOTASPStatus

8.1136.2.16 uint8_t* unpack_voice_SLQSVoiceGetAllCallInfo_t::pVoicePrivacy

8.1137 unpack_voice_SLQSVoiceGetCallBarring_t Struct Reference

Data Fields

- uint8_t * pSvcClass
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1137.1 Detailed Description

This structure contains Voice Get Call Barring Response Parameters

Parameters

pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available • Bit to check in ParamPresenceMask - **16**
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**

pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1137.2 Field Documentation

8.1137.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallBarring_t::pAlphaIDInfo

8.1137.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCallBarring_t::ParamPresenceMask

8.1137.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCallID

8.1137.2.4 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCCResType

8.1137.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallBarring_t::pCCSUPSType

8.1137.2.6 uint16_t* unpack_voice_SLQSVoiceGetCallBarring_t::pFailCause

8.1137.2.7 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.1138 unpack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference

Data Fields

- voice_getCallFWInfo * pGetCallFWInfo
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- voice_getCallFWExtInfo * pGetCallFWExtInfo
- swi_uint256_t ParamPresenceMask

8.1138.1 Detailed Description

This structure contains Voice Get Call Forwarding Status Unpack Parameters

Parameters

pGetCallFWInfo	• Pointer to structure of getCallFWInfo (optional) – See voice_getCallFWInfo for more information • Bit to check in ParamPresenceMask - **16**

pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**
pGetCallFWExt-Info	• Pointer to structure of getCallFWExtInfo (optional) – See voice_getCallFWExtInfo for more information • Bit to check in ParamPresenceMask - **22**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1138.2 Field Documentation

8.1138.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pAlphaIDInfo

8.1138.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCallForwardingStatus_t::ParamPresenceMask

8.1138.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCallID

8.1138.2.4 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCResType

8.1138.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCSUPSType

8.1138.2.6 uint16_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pFailCause

8.1138.2.7 voice_getCallFWExtInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWExtInfo

8.1138.2.8 voice_getCallFWInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWInfo

8.1139 unpack_voice_SLQSVoiceGetCallInfo_t Struct Reference

Data Fields

- voice_callInfo * pCallInfo
- voice_remotePartyNum * pRemotePartyNum
- uint16_t * pSrvOpt
- uint8_t * pVoicePrivacy
- uint8_t * pOTASPStatus
- voice_remotePartyName * pRemotePartyName
- voice_UUSInfo * pUUSInfo
- uint8_t * pAlertType
- voice_alphaIDInfo * pAlphaIDInfo
- voice_connectNumInfo * pConnectNumInfo
- voice_diagInfo * pDiagInfo
- uint32_t * pAlertingPattern
- swi_uint256_t ParamPresenceMask

8.1139.1 Detailed Description

This structure contains information of the unpack parameters associated with a call.

Parameters

pCallInfo(optional)	• See voice_callInfo for more information. • Bit to check in ParamPresenceMask - **16**
pRemotePartyNum(optional)	• See voice_remotePartyNum for more information. • Bit to check in ParamPresenceMask - **17**
pSrvOpt	• Service option(optional) • Applicable only for 3GPP2 devices. • See Table8 qaGobiApiTableServiceOptions.h for standard service option number assignments. • Bit to check in ParamPresenceMask - **18**
pVoicePrivacy	• Voice Privacy.(optional) • Applicable only for 3GPP2 devices. • Values. – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**

pOTASPStatus	• OTASP status for the OTASP call.(optional) • Applicable only for 3GPP2 devices. – 0x00 - OTASP_STATUS_SPL_UNLOCKED - SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED - SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED - A-key exchanged; only for user-initiated OTASP – 0x03 - OTASP_STATUS_SSD_UPDATED - SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP (OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP (OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP (OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP (OTAPA) – 0xFF - Not Available • Bit to check in ParamPresenceMask - **20**
pRemoteParty-Name(optional)	• Applicable only for 3GPP devices. • See voice_remotePartyName for more information. • Bit to check in ParamPresenceMask - **21**
pUUS-Info(optional)	• Applicable only for 3GPP devices. • See voice_UUSInfo for more information. • Bit to check in ParamPresenceMask - **22**
pAlert-Type(optional)	• Alerting type. • Applicable only for 3GPP devices. – 0x00 - ALERTING_LOCAL - Local – 0x01 - ALERTING_REMOTE - Remote – 0xFF - Not Available • Bit to check in ParamPresenceMask - **23**

pAlphaID-Info(optional)	• Applicable only for 3GPP devices. • See voice_alphaIDInfo for more information. • Bit to check in ParamPresenceMask - **24**
pConnectNum-Info(optional)	• See voice_connectNumInfo for more information. • Bit to check in ParamPresenceMask - **25**
pDiag-Info(optional)	• See voice_diagInfo for more information. • Bit to check in ParamPresenceMask - **26**
pAlertingPattern	• Alerting pattern.(optional) – 0x00 - QMI_VOICE_ALERTING_PATTERN_1 - Pattern 1 – 0x01 - QMI_VOICE_ALERTING_PATTERN_2 - Pattern 2 – 0x02 - QMI_VOICE_ALERTING_PATTERN_3 - Pattern 3 – 0x04 - QMI_VOICE_ALERTING_PATTERN_5 - Pattern 5 – 0x05 - QMI_VOICE_ALERTING_PATTERN_6 - Pattern 6 – 0x06 - QMI_VOICE_ALERTING_PATTERN_7 - Pattern 7 – 0x07 - QMI_VOICE_ALERTING_PATTERN_8 - Pattern 8 – 0x08 - QMI_VOICE_ALERTING_PATTERN_9 - Pattern 9 – 0xFF - Not Available • Bit to check in ParamPresenceMask - **27**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1139.2 Field Documentation

8.1139.2.1 uint32_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertingPattern

8.1139.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertType

8.1139.2.3 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pAlphaIDInfo

8.1139.2.4 swi_uint256_t unpack_voice_SLQSVoiceGetCallInfo_t::ParamPresenceMask

8.1139.2.5 voice_callInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pCallInfo

8.1139.2.6 voice_connectNumInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pConnectNumInfo

8.1139.2.7 voice_diagInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pDiagInfo

8.1139.2.8 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pOTASPStatus

8.1139.2.9 voice_remotePartyName* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyName

8.1139.2.10 voice_remotePartyNum* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyNum

8.1139.2.11 uint16_t* unpack_voice_SLQSVoiceGetCallInfo_t::pSrvOpt

8.1139.2.12 voice_UUSInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pUUSInfo

8.1139.2.13 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pVoicePrivacy

8.1140 unpack_voice_SLQSVoiceGetCallWaiting_t Struct Reference

Data Fields

- uint8_t * pSvcClass
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1140.1 Detailed Description

This structure contains Voice Get Call Waiting Unpack Parameters

Parameters

pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available • Bit to check in ParamPresenceMask - **16**
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**

pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1140.2 Field Documentation

8.1140.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallWaiting_t::pAlphaIDInfo

8.1140.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCallWaiting_t::ParamPresenceMask

8.1140.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCallID

8.1140.2.4 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCResType

8.1140.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCSUPSType

8.1140.2.6 uint16_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pFailCause

8.1140.2.7 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.1141 unpack_voice_SLQSVoiceGetCLIP_t Struct Reference

Data Fields

- voice_CLIPResp * pCLIPResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1141.1 Detailed Description

This structure contains Voice Get Calling Line Identification Presentation(CLIP) Response Parameters

Parameters

pCLIPResp	• Pointer to structure of CLIPResp (optional) – See voice_CLIPResp for more information • Bit to check in ParamPresenceMask - **16**
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1141.2 Field Documentation

8.1141.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIP_t::pAlphaIDInfo

8.1141.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCLIP_t::ParamPresenceMask

8.1141.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCallID

8.1141.2.4 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCCResType

8.1141.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIP_t::pCCSUPSType

8.1141.2.6 voice_CLIPResp* unpack_voice_SLQSVoiceGetCLIP_t::pCLIPResp

8.1141.2.7 uint16_t* unpack_voice_SLQSVoiceGetCLIP_t::pFailCause

8.1142 unpack_voice_SLQSVoiceGetCLIR_t Struct Reference

Data Fields

- voice_CLIRResp * pCLIRResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1142.1 Detailed Description

This structure contains Voice Get Calling Line Identification Restriction (CLIR) Response Parameters

Parameters

pCLIRResp	• Pointer to structure of CLIRResp (optional) – See voice_CLIRResp for more information • Bit to check in ParamPresenceMask - **16**
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**

pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1142.2 Field Documentation

8.1142.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIR_t::pAlphaIDInfo

8.1142.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCLIR_t::ParamPresenceMask

8.1142.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCallID

8.1142.2.4 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCCResType

8.1142.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIR_t::pCCSUPSType

8.1142.2.6 voice_CLIRResp* unpack_voice_SLQSVoiceGetCLIR_t::pCLIRResp

8.1142.2.7 uint16_t* unpack_voice_SLQSVoiceGetCLIR_t::pFailCause

8.1143 unpack_voice_SLQSVoiceGetCNAP_t Struct Reference

Data Fields

- voice_CNAPResp * pCNAPResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1143.1 Detailed Description

This structure contains Voice Get Calling Name Presentation(CNAP) Response Parameters

Parameters

pCNAPResp	• Pointer to structure of CNAPResp (optional) – See voice_CNAPResp for more information • Bit to check in ParamPresenceMask - **16**

pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1143.2 Field Documentation

8.1143.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCNAP_t::pAlphaIDInfo

8.1143.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCNAP_t::ParamPresenceMask

8.1143.2.3 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCallID

8.1143.2.4 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCCResType

8.1143.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCNAP_t::pCCSUPSType

8.1143.2.6 voice_CNAPResp* unpack_voice_SLQSVoiceGetCNAP_t::pCNAPResp

8.1143.2.7 uint16_t* unpack_voice_SLQSVoiceGetCNAP_t::pFailCause

8.1144 unpack_voice_SLQSVoiceGetCOLP_t Struct Reference

Data Fields

- voice_COLPResp * pCOLPResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1144.1 Detailed Description

This structure contains Voice Get Connected Line Identification Presentation(COLP) Response Parameters

Parameters

pCOLPResp	• Pointer to structure of COLPResp (optional) – See voice_COLPResp for more information • Bit to check in ParamPresenceMask - **16**
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1144.2 Field Documentation

8.1144.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLP_t::pAlphaIDInfo

8.1144.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCOLP_t::ParamPresenceMask

8.1144.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCallID

8.1144.2.4 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCCResType

8.1144.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLP_t::pCCSUPSType

8.1144.2.6 voice_COLPResp* unpack_voice_SLQSVoiceGetCOLP_t::pCOLPResp

8.1144.2.7 uint16_t* unpack_voice_SLQSVoiceGetCOLP_t::pFailCause

8.1145 unpack_voice_SLQSVoiceGetCOLR_t Struct Reference

Data Fields

- voice_COLRResp * pCOLRResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1145.1 Detailed Description

This structure contains Voice Get Connected Line Identification Restriction(COLR) Response Parameters

Parameters

pCOLRResp	• Pointer to structure of COLRResp (optional) – See voice_COLRResp for more information • Bit to check in ParamPresenceMask - **16**
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **17**

pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **18**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **20**
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1145.2 Field Documentation

8.1145.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLR_t::pAlphaIDInfo

8.1145.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCOLR_t::ParamPresenceMask

8.1145.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCallID

8.1145.2.4 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCCResType

8.1145.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLR_t::pCCSUPSType

8.1145.2.6 voice_COLRResp* unpack_voice_SLQSVoiceGetCOLR_t::pCOLRResp

8.1145.2.7 uint16_t* unpack_voice_SLQSVoiceGetCOLR_t::pFailCause

8.1146 unpack_voice_SLQSVoiceGetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnswerStat
- voice_airTimer * pAirTimerCnt
- voice_roamTimer * pRoamTimerCnt

- uint8_t * pCurrTTYMode
- voice_prefVoiceSO * pCurPrefVoiceSO
- voice_curAMRConfig * pCurAMRConfig
- uint8_t * pCurVoicePrivacyPref
- uint8_t * pCurVoiceDomainPref
- swi_uint256_t ParamPresenceMask

8.1146.1 Detailed Description

This structure contains Voice Get Configuration Response Parameters.

Parameters

pAutoAnswer-Stat(optional)	• Auto Answer Status • Value returned is read from NV_AUTO_ANSWER_I. – 0x00 - Disabled – 0x01 - Enabled – 0xFF - Not Available • Bit to check in ParamPresenceMask - **16**
pAirTimer-Cnt(optional)	• Air Timer Count • Value returned is read from NV_AIR_CNT_I. • See voice_airTimer for more information • Bit to check in ParamPresenceMask - **17**
pRoamTimer-Cnt(optional)	• Roam Timer Count • Value returned is read from NV_ROAM_CNT_I. • See voice_roamTimer for more information • Bit to check in ParamPresenceMask - **18**
pCurrTTY-Mode(optional)	• Current TTY Mode • Value returned is read from NV_TTY_I. – 0x00 - TTY_MODE_FULL - Full – 0x01 - TTY_MODE_VCO - Voice carry over – 0x02 - TTY_MODE_HCO - Hearing carry over – 0x03 - TTY_MODE_OFF - Off – 0xFF - Not Available • Bit to check in ParamPresenceMask - **19**
pCurPrefVoiceS-O(optional)	• Current Preferred Voice SO • Value returned is read from NV_PREF_VOICE_SO_I. • See voice_prefVoiceSO for more information • Bit to check in ParamPresenceMask - **20**

pCurAMR-Config(optional)	• Current Adaptive Multi-Rate Configuration. • Values returned are read from NV_GSM_ARM_CALL_CONFIG_I and NV_UMTS_AMR_CODEC_PREFERENCE_CONFIG_I. • See voice_curAMRConfig for more information • Bit to check in ParamPresenceMask - **21**
pCurVoice-Privacy-Pref(optional)	• Current Voice Privacy Preference • Value returned is read from NV_VOICE_PRIV_I. – 0x00 - Standard privacy – 0x01 - Enhanced privacy – 0xFF - Not Available • Bit to check in ParamPresenceMask - **22**
pCurVoice-Domain-Pref(optional)	• Current Voice Domain Preference. – 0x00 - Circuit-switched (CS) only – 0x01 - Packet-switched (PS) only – 0x02 - CS is preferred; PS is secondary – 0x03 - PS is preferred; CS is secondary – 0xFF - Not Available • Bit to check in ParamPresenceMask - **23**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1146.2 Field Documentation

8.1146.2.1 voice_airTimer* unpack_voice_SLQSVoiceGetConfig_t::pAirTimerCnt

8.1146.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetConfig_t::ParamPresenceMask

8.1146.2.3 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pAutoAnswerStat

8.1146.2.4 voice_curAMRConfig* unpack_voice_SLQSVoiceGetConfig_t::pCurAMRConfig

8.1146.2.5 voice_prefVoiceSO* unpack_voice_SLQSVoiceGetConfig_t::pCurPrefVoiceSO

8.1146.2.6 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurrTTYMode

8.1146.2.7 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoiceDomainPref

8.1146.2.8 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoicePrivacyPref

8.1146.2.9 voice_roamTimer* unpack_voice_SLQSVoiceGetConfig_t::pRoamTimerCnt

8.1147 unpack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint16_t * pFailCause
- swi_uint256_t ParamPresenceMask

8.1147.1 Detailed Description

This structure contains Failure cause Information. Populated when API Fails.

Parameters

pFailCause	• Supplementary service failure causes (optional, supply NULL if not required). • See Table8 qaGobiApiTableVoiceCallEndReasons.h for supplementary services failure cause – 0xFFFF is the value when the information is not received from device • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1147.2 Field Documentation

8.1147.2.1 swi_uint256_t unpack_voice_SLQSVoiceManageCalls_t::ParamPresenceMask

8.1147.2.2 uint16_t* unpack_voice_SLQSVoiceManageCalls_t::pFailCause

8.1148 unpack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID
- swi_uint256_t ParamPresenceMask

8.1148.1 Detailed Description

This structure contains the flash information associated with a call.

Parameters

pCallID	• Unique call identifier associated with the current call. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1148.2 Field Documentation

8.1148.2.1 swi_uint256_t unpack_voice_SLQSVoiceSendFlash_t::ParamPresenceMask

8.1148.2.2 uint8_t* unpack_voice_SLQSVoiceSendFlash_t::pCallID

8.1149 unpack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference

Data Fields

- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1149.1 Detailed Description

This structure contains Voice Set Call Barring Password Unpack Parameters

Parameters

pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available • Bit to check in ParamPresenceMask - **16**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **17**
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available • Bit to check in ParamPresenceMask - **18**
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid. • Bit to check in ParamPresenceMask - **19**
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1149.2 Field Documentation

8.1149.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pAlphaIDInfo

8.1149.2.2 swi_uint256_t unpack_voice_SLQSVoiceSetCallBarringPassword_t::ParamPresenceMask

8.1149.2.3 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCallID

8.1149.2.4 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCResType

8.1149.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCSUPSType

8.1149.2.6 uint16_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pFailCause

8.1150 unpack_voice_SLQSVoiceSetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnsStatus
- uint8_t * pAirTimerStatus
- uint8_t * pRoamTimerStatus
- uint8_t * pTTYConfigStatus
- uint8_t * pPrefVoiceSOStatus
- uint8_t * pVoiceDomainPrefStatus
- swi_uint256_t ParamPresenceMask

8.1150.1 Detailed Description

This structure contains information about the Set Configuration Response Parameters.

Parameters

pAutoAnsStatus	• Auto Answer Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available. • Bit to check in ParamPresenceMask - **16**
pAirTimerStatus	• Air Timer Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available. • Bit to check in ParamPresenceMask - **17**

pRoamTimer-Status	• Roam Timer Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available. • Bit to check in ParamPresenceMask - **18**
pTTYConfig-Status	• TTY Config Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available. • Bit to check in ParamPresenceMask - **19**
pPrefVoiceSO-Status	• Preferred Voice SO Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available. • Bit to check in ParamPresenceMask - **20**
pVoiceDomain-PrefStatus	• Voice-Domain Preference Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available. • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1150.2 Field Documentation

8.1150.2.1 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAirTimerStatus

8.1150.2.2 swi_uint256_t unpack_voice_SLQSVoiceSetConfig_t::ParamPresenceMask

8.1150.2.3 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAutoAnsStatus

8.1150.2.4 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSOStatus

8.1150.2.5 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pRoamTimerStatus

8.1150.2.6 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pTTYConfigStatus

8.1150.2.7 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pVoiceDomainPrefStatus

8.1151 unpack_voice_SLQSVoiceSetSUPSService_t Struct Reference

Data Fields

- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResultType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.1151.1 Detailed Description

This structure contains Supplementary Service response parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

pFailCause	• Supplementary service failure causes (optional, supply NULL if not required). – 0xFFFF is the value when the information is not received from device • Bit to check in ParamPresenceMask - **16**
pAlphaIDInfo	• Pointer to structure of alphaIDInfo. The parameter used to pass the alpha (if any) given by the SIM/R-UIM after call control (optional, supply NULL if not required) – See voice_alphaIDInfo for more information • Bit to check in ParamPresenceMask - **17**
pCCResultType	• Call control result types (optional, supply NULL if not required) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - if the device does not provide this information • Bit to check in ParamPresenceMask - **19**
pCallID	• Unique call identifier for the dialed call (optional, supply NULL if not required) – 0x00 - if the device does not provide this information • Bit to check in ParamPresenceMask - **18**
pCCSUPSType	• Data is present when pCCResultType is present and is other than Voice. (optional, supply NULL if not required) – See voice_ccSUPSType for more information • Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1151.2 Field Documentation

8.1151.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetSUPSService_t::pAlphaIDInfo

8.1151.2.2 swi_uint256_t unpack_voice_SLQSVoiceSetSUPSService_t::ParamPresenceMask

8.1151.2.3 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCallID

8.1151.2.4 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCCResultType

8.1151.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceSetSUPSService_t::pCCSUPSType

8.1151.2.6 uint16_t* unpack_voice_SLQSVoiceSetSUPSService_t::pFailCause

8.1152 unpack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- swi_uint256_t ParamPresenceMask

8.1152.1 Detailed Description

This structure contains parameters of continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Start continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1152.2 Field Documentation

8.1152.2.1 swi_uint256_t unpack_voice_SLQSVoiceStartContDTMF_t::ParamPresenceMask

8.1152.2.2 uint8_t* unpack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.1153 unpack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID
- swi_uint256_t ParamPresenceMask

8.1153.1 Detailed Description

This structure contains parameters of stop continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Stop continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1153.2 Field Documentation

8.1153.2.1 uint8_t unpack_voice_SLQSVoiceStopContDTMF_t::callID

8.1153.2.2 swi_uint256_t unpack_voice_SLQSVoiceStopContDTMF_t::ParamPresenceMask

8.1154 unpack_voice_SLQSVoiceSUPSCallback_ind_t Struct Reference

Data Fields

- voice_SUPSInfo SUPSInformation
- uint8_t * pSvcClass
- uint8_t * pReason
- uint8_t * pCallFWNum
- uint8_t * pCallFWTimerVal
- struct voice_USSInfo * pUSSInfo
- uint8_t * pCallID
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCallBarPasswd
- voice_newPwdData * pNewPwdData
- uint8_t * pDataSrc
- uint16_t * pFailCause
- voice_getCallFWInfo * pCallFwdInfo
- voice_CLIRResp * pCLIRstatus
- voice_CLIPResp * pCLIPstatus
- voice_COLPResp * pCOLPstatus
- voice_COLRResp * pCOLRstatus
- voice_CNAPResp * pCNAPstatus
- swi_uint256_t ParamPresenceMask

8.1154.1 Detailed Description

This structure contains the parameters passed for SUPS info indication by the device.

Parameters

SUPS-Information(mandatory)	See voice_SUPSInfo for more information. • Bit to check in ParamPresenceMask - **1**
pSvc-Class(optional)	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Bit to check in ParamPresenceMask - **16**
p-Reason(optional)	• See qaGobiApiTableCallControlReturnReasons.h for return reasons. • Bit to check in ParamPresenceMask - **17**
pCallFW-Num(optional)	• Call forwarding number to be registered with the network. • ASCII String, NULL terminated. • Bit to check in ParamPresenceMask - **18**
pCallFWTimer-Val(optional)	• Call Forwarding No Reply Timer. – Range: 5 to 30 in steps of 5. • Bit to check in ParamPresenceMask - **19**
pUSS-Info(optional)	• See voice_USSInfo for more information. • Bit to check in ParamPresenceMask - **20**
pCallID(optional)	• Call identifier of the voice call that has been modified to a supplementary service as a result of call control. • Bit to check in ParamPresenceMask - **21**
pAlphaID-Info(optional)	• See voice_alphaIDInfo for more information. • Bit to check in ParamPresenceMask - **22**
pCallBar-Passwd(optional)	• Password is required if call barring is provisioned using a password. – Password consists of 4 ASCII digits. – Range: 0000 to 9999. • This also serves as the old password in the register password scenario. • Bit to check in ParamPresenceMask - **23**
pNewPwd-Data(optional)	• See voice_newPwdData for more information. • Bit to check in ParamPresenceMask - **24**

pDataSrc(optional)	• Sups Data Source. • Used to distinguish between the supplementary service data sent to the network and the response received from the network. • If absent, the supplementary service data in this indication can be assumed as a request sent to the network. • Bit to check in ParamPresenceMask - **25**
pFailCause(optional)	• Supplementary services failure cause. • See qaGobiApiTableVoiceCallEndReasons.h for more information. • Bit to check in ParamPresenceMask - **26**
pCallFwdInfo(optional)	• See voice_getCallFWInfo for more information. • Bit to check in ParamPresenceMask - **27**
pCLIRstatus(optional)	• See voice_CLIRResp for more information. • Bit to check in ParamPresenceMask - **28**
pCLIPstatus(optional)	• See voice_CLIPResp for more information. • Bit to check in ParamPresenceMask - **29**
pCOLPstatus(optional)	• See voice_COLPResp for more information. • Bit to check in ParamPresenceMask - **30**
pCOLRstatus(optional)	• See voice_COLRResp for more information. • Bit to check in ParamPresenceMask - **31**
pCNAPstatus(optional)	• See voice_CNAPResp for more information. • Bit to check in ParamPresenceMask - **32**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

Note

None

8.1154.2 Field Documentation

8.1154.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pAlphaIDInfo

8.1154.2.2 swi_uint256_t unpack_voice_SLQSVoiceSUPSCallback_ind_t::ParamPresenceMask

8.1154.2.3 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallBarPasswd

8.1154.2.4 **voice_getCallFWInfo*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallFwdInfo

8.1154.2.5 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallFWNum

8.1154.2.6 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallFWTimerVal

8.1154.2.7 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallID

8.1154.2.8 **voice_CLIPResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCLIPstatus

8.1154.2.9 **voice_CLIRResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCLIRstatus

8.1154.2.10 **voice_CNAPResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCNAPstatus

8.1154.2.11 **voice_COLPResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCOLPstatus

8.1154.2.12 **voice_COLRResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCOLRstatus

8.1154.2.13 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pDataSrc

8.1154.2.14 uint16_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pFailCause

8.1154.2.15 **voice_newPwdData*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pNewPwdData

8.1154.2.16 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pReason

8.1154.2.17 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pSvcClass

8.1154.2.18 struct **voice_USSInfo*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pUSSInfo

8.1154.2.19 **voice_SUPSInfo** unpack_voice_SLQSVoiceSUPSCallback_ind_t::SUPSInformation

8.1155 unpack_voice_SUPSNotificationCallback_ind_t Struct Reference

Data Fields

- uint8_t callID
- uint8_t notifType
- uint16_t * pCUGIndex
- voice_ECTNum * pECTNum
- swi_uint256_t ParamPresenceMask

8.1155.1 Detailed Description

Contains the parameters passed for SUPS notification indication by the device.

Parameters

callID	• Unique identifier of the call for which the notification is applicable. (mandatory) • Bit to check in ParamPresenceMask - **1**

notifType	• Notification type parameter (mandatory) – 0x01 - NOTIFICATION_TYPE_OUTGOING_CALL_IS_FORWARDED Originated MO call is being forwarded to another user – 0x02 - NOTIFICATION_TYPE_OUTGOING_CALL_IS_WAITING Originated MO call is waiting at the called user – 0x03 - NOTIFICATION_TYPE_OUTGOING_CUG_CALL Outgoing call is a CUG call – 0x04 - NOTIFICATION_TYPE_OUTGOING_CALLS_BARRED Outgoing calls are barred – 0x05 - NOTIFICATION_TYPE_OUTGOING_CALL_IS_DEFLECTED Outgoing call is deflected – 0x06 - NOTIFICATION_TYPE_INCOMING_CUG_CALL Incoming call is a CUG call – 0x07 - NOTIFICATION_TYPE_INCOMING_CALLS_BARRED Incoming calls are barred – 0x08 - NOTIFICATION_TYPE_INCOMING_FORWARDED_CALL Incoming call received is a forwarded call – 0x09 - NOTIFICATION_TYPE_INCOMING_DEFLECTED_CALL Incoming call is a deflected call – 0x0A - NOTIFICATION_TYPE_INCOMING_CALL_IS_FORWARDED Incoming call is forwarded to another user – 0x0B - NOTIFICATION_TYPE_UNCOND_CALL_FORWARD_ACTIVE Unconditional call forwarding is active – 0x0C - NOTIFICATION_TYPE_COND_CALL_FORWARD_ACTIVE Conditional call forwarding is active – 0x0D - NOTIFICATION_TYPE_CLIR_SUPPRESSION_REJECTED CLIR suppression is rejected – 0x0E - NOTIFICATION_TYPE_CALL_IS_ON_HOLD Call is put on hold at the remote party – 0x0F - NOTIFICATION_TYPE_CALL_IS_RETRIEVED Call is retrieved at the remote party from the hold state – 0x10 - NOTIFICATION_TYPE_CALL_IS_IN_MPTY Call is in a conference – 0x11 - NOTIFICATION_TYPE_INCOMING_CALL_IS_ECT Incoming call is an explicit call transfer • Bit to check in ParamPresenceMask - **1**
pCUGIndex	• The CUG Index used to indicate that the incoming/outgoing call is a CUG call. (optional, NULL when not present) Range: 0x00 to 0x7FFF. • Bit to check in ParamPresenceMask - **16**
pECTNum	• The ECT Number is used to indicate that the incoming call is an explicitly transferred call. (optional, NULL when not present) Refer ECTNum for details. • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.1155.2 Field Documentation

8.1155.2.1 uint8_t unpack_voice_SUPSNotificationCallback_ind_t::callID

8.1155.2.2 uint8_t unpack_voice_SUPSNotificationCallback_ind_t::notifType

8.1155.2.3 swi_uint256_t unpack_voice_SUPSNotificationCallback_ind_t::ParamPresenceMask

8.1155.2.4 uint16_t* unpack_voice_SUPSNotificationCallback_ind_t::pCUGIndex

8.1155.2.5 voice_ECTNum* unpack_voice_SUPSNotificationCallback_ind_t::pECTNum

8.1156 unpack_voice_USSDNotificationCallback_ind_t Struct Reference

Data Fields

- uint8_t notification_Type
- voice_USSDNotificationNetworkInfo USSDNotificationNetworkInfo
- swi_uint256_t ParamPresenceMask

8.1156.1 Detailed Description

Structure for storing the USSD notification indication parameters.

Parameters

notification_Type(mandatory)	Values: -0x01-FURTHER_USER_ACTION_NOT_REQUIRED -0x02-FURTHER_USER_ACTION_REQUIRED • Bit to check in ParamPresenceMask - **1**
USSDNotificationNetworkInfo(optional)	• USSD network info (See voice_USSDNotificationNetworkInfo) – Bit to check in ParamPresenceMask - **16**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1156.2 Field Documentation

8.1156.2.1 uint8_t unpack_voice_USSDNotificationCallback_ind_t::notification_Type

8.1156.2.2 swi_uint256_t unpack_voice_USSDNotificationCallback_ind_t::ParamPresenceMask

8.1156.2.3 voice_USSDNotificationNetworkInfo unpack_voice_USSDNotificationCallback_ind_t::USSDNotificationNetworkInfo

8.1157 unpack_voice_USSDNoWaitIndicationCallback_ind_t Struct Reference

Data Fields

- struct voice_VoiceUssdNoWaitErrorTlv NwETlv
- struct voice_VoiceUssdNoWaitFailCauseTlv FCTlv
- struct voice_VoiceUssdNoWaitUSSINFOTlv UInfoTLV
- struct voice_VoiceUssdNoWaitAlphaIDTlv AIDTlv
- swi_uint256_t ParamPresenceMask

8.1157.1 Detailed Description

Contains the parameters passed for SetUSSDNoWaitIndicationCallback by the device.

Parameters

NwETlv	• Network Error • See voice_VoiceUssdNoWaitErrorTlv for more details.
FCTlv	• Supplementary services failure cause • See voice_VoiceUssdNoWaitFailCauseTlv for more details.
UInfoTLV	• USS Data from Network. • See voice_VoiceUssdNoWaitUSSINFOTlv for more details.
AIDTlv	• Alpha identifier • See voice_VoiceUssdNoWaitAlphaIDTlv for more details.

8.1157.2 Field Documentation

8.1157.2.1 struct voice_VoiceUssdNoWaitAlphaIDTlv unpack_voice_USSDNoWaitIndicationCallback_ind_t::AIDTlv

8.1157.2.2 struct voice_VoiceUssdNoWaitFailCauseTlv unpack_voice_USSDNoWaitIndicationCallback_ind_t::FCTlv

8.1157.2.3 struct voice_VoiceUssdNoWaitErrorTlv unpack_voice_USSDNoWaitIndicationCallback_ind_t::NwETlv

8.1157.2.4 swi_uint256_t unpack_voice_USSDNoWaitIndicationCallback_ind_t::ParamPresenceMask

8.1157.2.5 struct voice_VoiceUssdNoWaitUSSINFOTlv unpack_voice_USSDNoWaitIndicationCallback_ind_t::UInfoTLV

8.1158 unpack_voice_VoiceInfoRecCallback_ind_t Struct Reference

Data Fields

- uint8_t callID
- voice_signalInfo * pSignalInfo
- voice_callerIDInfo * pCallerIDInfo

- uint8_t * pDispInfo
- uint8_t * pExtDispInfo
- uint8_t * pCallerNameInfo
- uint8_t * pCallWaitInd
- voice_connectNumInfo * pConnectNumInfo
- voice_connectNumInfo * pCallingPartyInfo
- voice_calledPartyInfo * pCalledPartyInfo
- voice_redirNumInfo * pRedirNumInfo
- uint8_t * pCLIRCause
- voice_NSSAudioCtrl * pNSSAudioCtrl
- uint8_t * pNSSRelease
- voice_lineCtrlInfo * pLineCtrlInfo
- voice_extDispRecInfo * pExtDispRecInfo
- swi_uint256_t ParamPresenceMask

8.1158.1 Detailed Description

This structure contains Voice record Information

Parameters

callID	[Mandatory] • Call identifier for the call. • Bit to check in ParamPresenceMask - **1**
pSignalInfo[-Optional]	• Signal Information • See voice_signalInfo for more information • Bit to check in ParamPresenceMask - **16**
pCallerIDInfo[-Optional]	• Caller ID Information • See voice_callerIDInfo for more information • Bit to check in ParamPresenceMask - **17**
pDispInfo[-Optional]	• Display Information • Bit to check in ParamPresenceMask - **18**
pExtDispInfo[-Optional]	• Extended Display Information • Bit to check in ParamPresenceMask - **19**
pCallerName-Info[Optional]	• Caller Name Information • Bit to check in ParamPresenceMask - **20**
pCallWaitInd[-Optional]	• Call Waiting Indicator • Bit to check in ParamPresenceMask - **21**

pConnectNumInfo[Optional]	• Connected Number Information • see voice_connectNumInfo for more information • Bit to check in ParamPresenceMask - **22**
pCallingPartyInfo[Optional]	• Calling Party Number Information • This structure is having exactly same elements as connectNumInfo • see voice_connectNumInfo for more information • Bit to check in ParamPresenceMask - **23**
pCalledPartyInfo[Optional]	• Called Party Number Information • see voice_calledPartyInfo for more information • Bit to check in ParamPresenceMask - **24**
pRedirNumInfo[Optional]	• Redirecting Number Information • see voice_redirNumInfo for more information • Bit to check in ParamPresenceMask - **25**
pCLIRCause[Optional]	• National Supplementary Services - CLIR • see voice_NSSAudioCtrl for more information • Bit to check in ParamPresenceMask - **26**
pNSSAudioCtrl[Optional]	• National Supplementary Services - Audio Control • Bit to check in ParamPresenceMask - **27**
pNSSRelease[Optional]	• National Supplementary Services - Release • Bit to check in ParamPresenceMask - **28**
pLineCtrlInfo[Optional]	• Line Control Information • see voice_lineCtrlInfo for more information • Bit to check in ParamPresenceMask - **29**
pExtDispRecInfo[Optional]	• Extended Display Record Information • see voice_extDispRecInfo for more information • Bit to check in ParamPresenceMask - **30**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1158.2 Field Documentation

8.1158.2.1 uint8_t unpack_voice_VoiceInfoRecCallback_ind_t::callID

8.1158.2.2 **swi_uint256_t** unpack_voice_VoiceInfoRecCallback_ind_t::ParamPresenceMask

8.1158.2.3 **voice_calledPartyInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pCalledPartyInfo

8.1158.2.4 **voice_callerIDInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pCallerIDInfo

8.1158.2.5 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pCallerNameInfo

8.1158.2.6 **voice_connectNumInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pCallingPartyInfo

8.1158.2.7 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pCallWaitInd

8.1158.2.8 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pCLIRCause

8.1158.2.9 **voice_connectNumInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pConnectNumInfo

8.1158.2.10 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pDispInfo

8.1158.2.11 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pExtDispInfo

8.1158.2.12 **voice_extDispRecInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pExtDispRecInfo

8.1158.2.13 **voice_lineCtrlInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pLineCtrlInfo

8.1158.2.14 **voice_NSSAudioCtrl*** unpack_voice_VoiceInfoRecCallback_ind_t::pNSSAudioCtrl

8.1158.2.15 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pNSSRelease

8.1158.2.16 **voice_redirNumInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pRedirNumInfo

8.1158.2.17 **voice_signalInfo*** unpack_voice_VoiceInfoRecCallback_ind_t::pSignalInfo

8.1159 unpack_voice_voicePrivacyChangeCallback_ind_t Struct Reference

Data Fields

- uint8_t callID
- uint8_t voicePrivacy
- swi_uint256_t ParamPresenceMask

8.1159.1 Detailed Description

Contains the parameters passed for voice privacy change indication

Parameters

callID	• Unique identifier of the call for which the voice privacy is applicable. (mandatory) • Bit to check in ParamPresenceMask - **1**
voicePrivacy	• Voice Privacy (mandatory) – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy • Bit to check in ParamPresenceMask - **1**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

None

8.1159.2 Field Documentation

8.1159.2.1 uint8_t unpack_voice_voicePrivacyChangeCallback_ind_t::callID

8.1159.2.2 swi_uint256_t unpack_voice_voicePrivacyChangeCallback_ind_t::ParamPresenceMask

8.1159.2.3 uint8_t unpack_voice_voicePrivacyChangeCallback_ind_t::voicePrivacy

8.1160 unpack_wda_GetDataFormat_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask
- uint16_t Tlvresult
- uint8_t Qos_format
- uint32_t Qlink_prot
- uint32_t Ul_data_aggregation_protocol
- uint32_t Dl_data_aggregation_protocol
- uint32_t Ndp_signature
- uint32_t Dl_data_aggregation_max_datagrams
- uint32_t Dl_data_aggregation_max_size
- uint32_t Ul_data_aggregation_max_datagrams
- uint32_t Ul_data_aggregation_max_size
- uint32_t Qos_header_format
- uint32_t Dl_minimum_padding
- uint8_t Flow_control
- uint8_t Tcp_coalescing
- uint8_t Udp_coalescing

8.1160.1 Detailed Description

Structure to unpack get data format response

Parameters

ParamPresence-Mask	• Bit mask that indicates presennce of optional parameters
Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

QoSDataFormat	header presence • Inidcates if QoS header is used or not – 0 - Not Used (Default) – 1 - Used (Not supported) • Bit to check in ParamPresenceMask - **16**
Link_Protocol	• 0x01 - 802.3 Ethernet mode (Not supported) – 0x02 - IP mode (Default) • Bit to check in ParamPresenceMask - **17**
UL_Data_-Aggregation_-Protocol	• 0x00 - UL data aggregation is disabled(Default) – 0x05 - UL QMAP is enabled • Bit to check in ParamPresenceMask - **18**
DL_Data_-Aggregation_-Protocol	• 0x00 - DL data aggregation is disabled(Default) – 0x05 - DL QMAP is enabled • Bit to check in ParamPresenceMask - **19**
DL_Data_-Aggregation_-Max_Datagrams	• Number of aggregated QMAP datagrams in a single DL packet • Bit to check in ParamPresenceMask - **21**
DL_Data_-Aggregation_-Max_Size	• Maximum size in bytes of a single aggregated packet allowed on downlink. • Bit to check in ParamPresenceMask - **22**
UL_Data_-Aggregation_-Max_Datagrams	• Number of aggregated QMAP datagrams in a single UL packet • Bit to check in ParamPresenceMask - **23**
UL_Data_-Aggregation_-Max_Size	• Maximum size in bytes of a single aggregated packet allowed on uplink. • Bit to check in ParamPresenceMask - **24**
DL_Padding	• Specifies the minimum padding bytes to be added in between aggregated downlink QMAP packets. • Default: 0

• Bit to check in ParamPresenceMask - **26**

Parameters

Flow_Control	

• Indicates whether flow control will be done by the TE.

– 0 - Flow control will not be done by the TE (Default)

– 1 - Flow control will be done by the TE (Default)

• Bit to check in ParamPresenceMask - **27**

Parameters

pTcp_coalescing	– Specifies whether TCP coalescing is enabled or disabled on the modem. – FALSE - Disable TCP coalecsing (default). – TRUE - Enable TCP coalescing

- Bit to check in ParamPresenceMask - **28**

Parameters

pUdp_coalescing	– Specifies whether UDP coalescing is enabled or disabled on the modem.

  – FALSE - Disable UDP coalecsing (default).
  – TRUE - Enable UDP coalescing

- Bit to check in ParamPresenceMask - **28**

8.1160.2 Field Documentation

8.1160.2.1 uint32_t unpack_wda_GetDataFormat_t::Dl_data_aggregation_max_datagrams

8.1160.2.2 uint32_t unpack_wda_GetDataFormat_t::Dl_data_aggregation_max_size

8.1160.2.3 uint32_t unpack_wda_GetDataFormat_t::Dl_data_aggregation_protocol

8.1160.2.4 uint32_t unpack_wda_GetDataFormat_t::Dl_minimum_padding

8.1160.2.5 uint8_t unpack_wda_GetDataFormat_t::Flow_control

8.1160.2.6 uint32_t unpack_wda_GetDataFormat_t::Ndp_signature

8.1160.2.7 swi_uint256_t unpack_wda_GetDataFormat_t::ParamPresenceMask

8.1160.2.8 uint32_t unpack_wda_GetDataFormat_t::Qlink_prot

8.1160.2.9 uint8_t unpack_wda_GetDataFormat_t::Qos_format

8.1160.2.10 uint32_t unpack_wda_GetDataFormat_t::Qos_header_format

8.1160.2.11 uint8_t unpack_wda_GetDataFormat_t::Tcp_coalescing

8.1160.2.12 uint16_t unpack_wda_GetDataFormat_t::Tlvresult

8.1160.2.13 uint8_t unpack_wda_GetDataFormat_t::Udp_coalescing

8.1160.2.14 uint32_t unpack_wda_GetDataFormat_t::Ul_data_aggregation_max_datagrams

8.1160.2.15 uint32_t unpack_wda_GetDataFormat_t::Ul_data_aggregation_max_size

8.1160.2.16 uint32_t unpack_wda_GetDataFormat_t::Ul_data_aggregation_protocol

8.1161 unpack_wda_GetQmapSettings_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask
- uint16_t Tlvresult
- uint8_t * pQmap_dfc

8.1161.1 Detailed Description

Structure to unpack get qmap settings response

Parameters

ParamPresence-Mask	• Bit mask that indicates presennce of optional parameters
Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
qmap_dfc	• Indicates whether the data flow control via QMAP is enabled/disabled. -Values: -0 - Disabled -1 - Enabled

8.1161.2 Field Documentation

8.1161.2.1 swi_uint256_t unpack_wda_GetQmapSettings_t::ParamPresenceMask

8.1161.2.2 uint8_t* unpack_wda_GetQmapSettings_t::pQmap_dfc

8.1161.2.3 uint16_t unpack_wda_GetQmapSettings_t::Tlvresult

8.1162 unpack_wda_SetDataFormat_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask
- uint16_t Tlvresult
- uint8_t * pQos_format
- uint32_t * pQlink_prot
- uint32_t * pUl_data_aggregation_protocol
- uint32_t * pDl_data_aggregation_protocol
- uint32_t * pNdp_signature
- uint32_t * pDl_data_aggregation_max_datagrams
- uint32_t * pDl_data_aggregation_max_size
- uint32_t * pUl_data_aggregation_max_datagrams
- uint32_t * pUl_data_aggregation_max_size
- uint32_t * pQos_header_format
- uint32_t * pDl_minimum_padding
- uint8_t * pFlow_control
- uint8_t * pTcp_coalescing
- uint8_t * pUdp_coalescing

8.1162.1 Detailed Description

Structure to unpack the set data format response from the modem

Parameters

ParamPresence-Mask	• Bit mask that indicates presennce of optional parameters
Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**
pQos_format	• Configured QOS data format. – 0 - QOS flow header is not present (default) – 1 - QOS flow header is present • Bit to check in ParamPresenceMask - **16**
pLink_prot	• Configured link layer protocol. – 0x01 - 802.3 Ethernet mode (default) – 0x02 - IP mode

• Bit to check in ParamPresenceMask - **17**

Parameters

pUl_data_-aggregation_-protocol	– Configured uplink data aggregation protocol. * 0x00 - UL data aggregation is disabled (default) * 0x01 - UL TLP is enabled * 0x02 - UL QC_NCM is enabled * 0x03 - UL MBIM is enabled * 0x04 - UL RNDIS is enabled * 0x05 - UL QMAP is enabled * 0x06 - UL QMAP V2 is enabled * 0x07 - UL QMAP V3 is enabled * 0x08 - UL QMAP V4 is enabled * 0x09 - UL QMAP V5 is enabled

• Bit to check in ParamPresenceMask - **18**

Parameters

pDL_data_aggregation_protocol	– Configured downlink data aggregation protocol. * 0x00 - DL data aggregation is disabled (default) * 0x01 - DL TLP is enabled * 0x02 - DL QC_NCM is enabled * 0x03 - DL MBIM is enabled * 0x04 - DL RNDIS is enabled * 0x05 - DL QMAP is enabled * 0x06 - DL QMAP V2 is enabled * 0x07 - DL QMAP V3 is enabled * 0x08 - DL QMAP V4 is enabled * 0x09 - DL QMAP V5 is enabled

- Bit to check in ParamPresenceMask - **19**

Parameters

pNdp_signature	– Uses the default value based on the data aggregation protocol.

- Bit to check in ParamPresenceMask - **20**

Parameters

pDl_data_aggregation_max_datagrams	

- Maximum number of datagrams in a single aggregated packet on downlink. The value applies to all downlink data aggregation protocols when downlink data aggregation is enabled. Zero means no limit.
- Bit to check in ParamPresenceMask - **21**

Parameters

pUl_data_aggregation_max_size	– Maximum size in bytes of a single aggregated packet allowed on downlink. The value applies to all downlink data aggregation protocols when downlink data aggregation is enabled.

- Bit to check in ParamPresenceMask - **22**

Parameters

pUl_data_aggregation_max_datagrams	– Maximum number of datagrams supported in the modem in a single aggregated packet on uplink for the currently configured aggregation mode. Zero means there is no limit.

- Bit to check in ParamPresenceMask - **23**

Parameters

pUl_data_aggregation_max_size	

- Maximum size in bytes of a single aggregated packet allowed on uplink.

- Bit to check in ParamPresenceMask - **24**

Parameters

pQos_header_format	– Configured QOS header format used on the uplink, on all the protocols, if QOS is enabled. – 0x00 - Reserved – 0x01 - QOS 6 byte default header – 0x02 - QOS 8 byte header

- Bit to check in ParamPresenceMask - **25**

Parameters

pDl_minimum_padding	– Indicates the minimum padding bytes added in between aggregated downlink QMAP packets.

- Bit to check in ParamPresenceMask - **26**

Parameters

pFlow_control	

- Configured flow control.
  - 0 - Flow control will not be done by the TE (Default)
  - 1 - Flow control will be done by the TE
- Bit to check in ParamPresenceMask - **27**

Parameters

pTcp_coalescing	– Specifies whether TCP coalescing is enabled or disabled on the modem. * FALSE - Disable TCP coalecsing (default). * TRUE - Enable TCP coalescing
pUdp_coalescing	– Specifies whether UDP coalescing is enabled or disabled on the modem. * FALSE - Disable UDP coalecsing (default). * TRUE - Enable UDP coalescing

8.1162.2 Field Documentation

8.1162.2.1 swi_uint256_t unpack_wda_SetDataFormat_t::ParamPresenceMask

8.1162.2.2 uint32_t* unpack_wda_SetDataFormat_t::pDl_data_aggregation_max_datagrams

8.1162.2.3 uint32_t* unpack_wda_SetDataFormat_t::pDl_data_aggregation_max_size

8.1162.2.4 uint32_t* unpack_wda_SetDataFormat_t::pDl_data_aggregation_protocol

8.1162.2.5 uint32_t* unpack_wda_SetDataFormat_t::pDl_minimum_padding

8.1162.2.6 uint8_t* unpack_wda_SetDataFormat_t::pFlow_control

8.1162.2.7 uint32_t* unpack_wda_SetDataFormat_t::pNdp_signature

8.1162.2.8 uint32_t* unpack_wda_SetDataFormat_t::pQlink_prot

8.1162.2.9 uint8_t* unpack_wda_SetDataFormat_t::pQos_format

8.1162.2.10 uint32_t* unpack_wda_SetDataFormat_t::pQos_header_format

8.1162.2.11 uint8_t* unpack_wda_SetDataFormat_t::pTcp_coalescing

8.1162.2.12 uint8_t* unpack_wda_SetDataFormat_t::pUdp_coalescing

8.1162.2.13 uint32_t* unpack_wda_SetDataFormat_t::pUl_data_aggregation_max_datagrams

8.1162.2.14 uint32_t* unpack_wda_SetDataFormat_t::pUl_data_aggregation_max_size

8.1162.2.15 uint32_t* unpack_wda_SetDataFormat_t::pUl_data_aggregation_protocol

8.1162.2.16 uint16_t unpack_wda_SetDataFormat_t::Tlvresult

8.1163 unpack_wda_SetQmapSettings_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask
- uint16_t Tlvresult
- uint8_t * pQmap_dfc

8.1163.1 Detailed Description

Structure to unpack the set qmap settings response from the modem

Parameters

ParamPresence-Mask	• Bit mask that indicates presennce of optional parameters
Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **3**
qmap_dfc	• Indicates whether the data flow control via QMAP is enabled/disabled. Values: 0 - Disabled 1 - Enabled

8.1163.2 Field Documentation

8.1163.2.1 swi_uint256_t unpack_wda_SetQmapSettings_t::ParamPresenceMask

8.1163.2.2 uint8_t* unpack_wda_SetQmapSettings_t::pQmap_dfc

8.1163.2.3 uint16_t unpack_wda_SetQmapSettings_t::Tlvresult

8.1164 unpack_wds_DHCPv4ClientLease_ind_t Struct Reference

Data Fields

- wds_DHCPProfileIdTlv ProfileIdTlv
- wds_DHCPLeaseStateTlv DHCPv4LeaseStateTlv
- wds_IPv4AdTlv IPv4AddrTlv
- wds_DHCPLeaseOptTlv DHCPv4LeaseOptTlv
- swi_uint256_t ParamPresenceMask

8.1164.1 Detailed Description

This structure contains DHCP IPv4 client lease information

Parameters

ProfileIdTlv	profile identifier information, see wds_DHCPProfileIdTlv for more details • Bit to check in ParamPresenceMask - **1**
DHCPv4Lease-StateTlv	lease state, see wds_DHCPLeaseStateTlv for more details • Bit to check in ParamPresenceMask - **2**
IPv4AddrTlv	IPv4 address, see wds_IPv4AdTlv for more details • Bit to check in ParamPresenceMask - **16**
DHCPv4Lease-OptTlv	DHCP lease option, see wds_DHCPLeaseOptTlv for more details • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1164.2 Field Documentation

8.1164.2.1 wds_DHCPLeaseOptTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseOptTlv

8.1164.2.2 wds_DHCPLeaseStateTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseStateTlv

8.1164.2.3 wds_IPv4AdTlv unpack_wds_DHCPv4ClientLease_ind_t::IPv4AddrTlv

8.1164.2.4 swi_uint256_t unpack_wds_DHCPv4ClientLease_ind_t::ParamPresenceMask

8.1164.2.5 wds_DHCPProfileIdTlv unpack_wds_DHCPv4ClientLease_ind_t::ProfileIdTlv

8.1165 unpack_wds_GetAutoconnect_t Struct Reference

Data Fields

- uint32_t * psetting
- swi_uint256_t ParamPresenceMask

8.1165.1 Detailed Description

auto connect data session setting parameter.

Parameters

pSetting	• NDIS auto connect setting – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1165.2 Field Documentation

8.1165.2.1 swi_uint256_t unpack_wds_GetAutoconnect_t::ParamPresenceMask

8.1165.2.2 uint32_t* unpack_wds_GetAutoconnect_t::psetting

8.1166 unpack_wds_GetByteTotals_t Struct Reference

Data Fields

- uint64_t * pTXTotalBytes
- uint64_t * pRXTotalBytes
- swi_uint256_t ParamPresenceMask

8.1166.1 Detailed Description

This structure contains unpack get byte totals information.

Parameters

pTXTotalBytes	• Bytes transmitted without error • Bit to check in ParamPresenceMask - **25**
pRXTotalBytes	• Bytes received without error • Bit to check in ParamPresenceMask - **26**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1166.2 Field Documentation

8.1166.2.1 swi_uint256_t unpack_wds_GetByteTotals_t::ParamPresenceMask

8.1166.2.2 uint64_t* unpack_wds_GetByteTotals_t::pRXTotalBytes

8.1166.2.3 uint64_t* unpack_wds_GetByteTotals_t::pTXTotalBytes

8.1167 unpack_wds_GetConnectionRate_t Struct Reference

Data Fields

- uint32_t currentChannelTXRate
- uint32_t currentChannelRXRate
- uint32_t maxChannelTXRate
- uint32_t maxChannelRXRate
- swi_uint256_t ParamPresenceMask

8.1167.1 Detailed Description

This structure contains unpack get connection rate information.

Parameters

currentChannelTXRate	• Current channel Tx rate (in bps) • Bit to check in ParamPresenceMask - **1**
currentChannelRXRate	• Current channel Rx rate (in bps) • Bit to check in ParamPresenceMask - **1**
maxChannelTXRate	• Maximum Tx rate (bps) that may be assigned to device by serving system. • Bit to check in ParamPresenceMask - **1**
maxChannelRXRate	• Maximum Rx rate (bps) that may be assigned to device by serving system. • Bit to check in ParamPresenceMask - **1**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1167.2 Field Documentation

8.1167.2.1 uint32_t unpack_wds_GetConnectionRate_t::currentChannelRXRate

8.1167.2.2 uint32_t unpack_wds_GetConnectionRate_t::currentChannelTXRate

8.1167.2.3 uint32_t unpack_wds_GetConnectionRate_t::maxChannelRXRate

8.1167.2.4 uint32_t unpack_wds_GetConnectionRate_t::maxChannelTXRate

8.1167.2.5 swi_uint256_t unpack_wds_GetConnectionRate_t::ParamPresenceMask

8.1168 unpack_wds_GetDataBearerTechnology_t Struct Reference

Data Fields

- uint32_t * pDataBearer
- swi_uint256_t ParamPresenceMask

8.1168.1 Detailed Description

This structure contains unpack get data bearer technology information.

Parameters

pDataBearer[OUT]	• Data bearer technology – 0x01 - CDMA2000 1x – 0x02 - CDMA 1xEV-DO Rev 0 – 0x03 - GSM – 0x04 - UMTS – 0x05 - CDMA2000 HRPD (1xEV-DO Rev A) – 0x06 - EDGE – 0x07 - HSDPA AND WCDMA – 0x08 - WCDMA AND HSUPA – 0x09 - HSDPA AND HSUPA – 0x0A - LTE – 0x0B - CDMA2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown • Bit to check in ParamPresenceMask - **1**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1168.2 Field Documentation

8.1168.2.1 swi_uint256_t unpack_wds_GetDataBearerTechnology_t::ParamPresenceMask

8.1168.2.2 uint32_t* unpack_wds_GetDataBearerTechnology_t::pDataBearer

8.1169 unpack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t pdptype
- uint32_t ipaddr
- uint32_t pridns
- uint32_t secdns
- uint16_t ipaddrv6 [8]
- uint16_t pridnsv6 [8]
- uint16_t secdnsv6 [8]
- uint32_t auth
- uint8_t namesize
- int8_t name [255]
- uint8_t apnsize
- int8_t apnname [255]
- uint8_t usersize
- int8_t username [255]
- swi_uint256_t ParamPresenceMask

8.1169.1 Detailed Description

Reads the default profile settings from the device. The default profile is used to establish an auto connect data session.

Parameters

pdptype	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0 - PDP-IP (IPv4) – 1 - PDP-PPP – 2 - PDP-IPv6 – 3 - PDP-IPv4v6 • Bit to check in ParamPresenceMask - **17**
ipaddr	• Preferred IPv4 address to be assigned to device • Bit to check in ParamPresenceMask - **30**
pridns	• Primary DNS IPv4 address preference • Bit to check in ParamPresenceMask - **21**
secdns	• Secondary DNS IPv4 address preference • Bit to check in ParamPresenceMask - **22**
ipaddrv6	• Preferred IPv6 address to be assigned to device • Bit to check in ParamPresenceMask - **143**
pridnsv6	• Primary DNS Pv6 address preference • Bit to check in ParamPresenceMask - **143**

secdnsv6	• Secondary DNS IPv6 address preference • Bit to check in ParamPresenceMask - **143**
auth	• Bitmap that indicates authentication algorithm preference – 0x00000001 - PAP preference * 0 - Never performed * 1 - May be performed – 0x00000002 - CHAP preference * 0 - Never performed * 1 - May be performed – All other bits are reserved and must be set to 0 – If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism. • Bit to check in ParamPresenceMask - **29**
namesize	• Maximum number of characters (including NULL terminator) that profile name array can contain. • Bit to check in ParamPresenceMask - **16**
name	• Profile name • Bit to check in ParamPresenceMask - **16**
apnsize	• Maximum number of characters (including NULL terminator) that APN name array can contain • Bit to check in ParamPresenceMask - **20**
apnname	• Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network. • If value is NULL or omitted, then subscription default value will be requested. • Bit to check in ParamPresenceMask - **20**
usersize	• Maximum number of characters (including NULL terminator) that username array can contain. • Bit to check in ParamPresenceMask - **27**
username	• Username used during network authentication • Bit to check in ParamPresenceMask - **27**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1169.2 Field Documentation

8.1169.2.1 int8_t unpack_wds_GetDefaultProfile_t::apnname[255]

8.1169.2.2 uint8_t unpack_wds_GetDefaultProfile_t::apnsize

8.1169.2.3 uint32_t unpack_wds_GetDefaultProfile_t::auth

8.1169.2.4 uint32_t unpack_wds_GetDefaultProfile_t::ipaddr

8.1169.2.5 uint16_t unpack_wds_GetDefaultProfile_t::ipaddrv6[8]

8.1169.2.6 int8_t unpack_wds_GetDefaultProfile_t::name[255]

8.1169.2.7 uint8_t unpack_wds_GetDefaultProfile_t::namesize

8.1169.2.8 swi_uint256_t unpack_wds_GetDefaultProfile_t::ParamPresenceMask

8.1169.2.9 uint32_t unpack_wds_GetDefaultProfile_t::pdptype

8.1169.2.10 uint32_t unpack_wds_GetDefaultProfile_t::pridns

8.1169.2.11 uint16_t unpack_wds_GetDefaultProfile_t::pridnsv6[8]

8.1169.2.12 uint32_t unpack_wds_GetDefaultProfile_t::secdns

8.1169.2.13 uint16_t unpack_wds_GetDefaultProfile_t::secdnsv6[8]

8.1169.2.14 int8_t unpack_wds_GetDefaultProfile_t::username[255]

8.1169.2.15 uint8_t unpack_wds_GetDefaultProfile_t::usersize

8.1170 unpack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t index
- swi_uint256_t ParamPresenceMask

8.1170.1 Detailed Description

This structure contains unpack Get Default Profile Number information.

Parameters

index	profile index • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1170.2 Field Documentation

8.1170.2.1 uint8_t unpack_wds_GetDefaultProfileNum_t::index

8.1170.2.2 swi_uint256_t unpack_wds_GetDefaultProfileNum_t::ParamPresenceMask

8.1171 unpack_wds_GetDefaultProfileV2_t Struct Reference

Data Fields

- uint32_t pdptype
- uint32_t ipaddr
- uint32_t pridns
- uint32_t secdns
- uint16_t ipaddrv6 [8]
- uint16_t pridnsv6 [8]
- uint16_t secdnsv6 [8]
- uint32_t auth
- uint8_t namesize
- int8_t name [255]
- uint8_t apnsize
- int8_t apnname [255]
- uint8_t usersize
- int8_t username [255]
- uint8_t pwdsize
- int8_t pwd [255]
- swi_uint256_t ParamPresenceMask

8.1171.1 Detailed Description

Reads the default profile settings from the device. The default profile is used to establish an auto connect data session.

Parameters

pdptype	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0 - PDP-IP (IPv4) – 1 - PDP-PPP – 2 - PDP-IPv6 – 3 - PDP-IPv4v6 • Bit to check in ParamPresenceMask - **17**
ipaddr	• Preferred IPv4 address to be assigned to device • Bit to check in ParamPresenceMask - **30**
pridns	• Primary DNS IPv4 address preference • Bit to check in ParamPresenceMask - **21**
secdns	• Secondary DNS IPv4 address preference • Bit to check in ParamPresenceMask - **22**
ipaddrv6	• Preferred IPv6 address to be assigned to device • Bit to check in ParamPresenceMask - **143**

pridnsv6	• Primary DNS Pv6 address preference • Bit to check in ParamPresenceMask - **143**
secdnsv6	• Secondary DNS IPv6 address preference • Bit to check in ParamPresenceMask - **143**
auth	• Bitmap that indicates authentication algorithm preference – 0x00000001 - PAP preference * 0 - Never performed * 1 - May be performed – 0x00000002 - CHAP preference * 0 - Never performed * 1 - May be performed – All other bits are reserved and must be set to 0 – If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism. • Bit to check in ParamPresenceMask - **29**
namesize	• Maximum number of characters (including NULL terminator) that profile name array can contain. • Bit to check in ParamPresenceMask - **16**
name	• Profile name • Bit to check in ParamPresenceMask - **16**
apnsize	• Maximum number of characters (including NULL terminator) that APN name array can contain • Bit to check in ParamPresenceMask - **20**
apnname	• Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network. • If value is NULL or omitted, then subscription default value will be requested. • Bit to check in ParamPresenceMask - **20**
usersize	• Maximum number of characters (including NULL terminator) that username array can contain. • Bit to check in ParamPresenceMask - **27**
username	• Username used during network authentication • Bit to check in ParamPresenceMask - **27**
pwdsize	• Maximum number of characters (including NULL terminator) that pwd name array can contain. • Bit to check in ParamPresenceMask - **28**

pwd	• Password used during network authentication • Bit to check in ParamPresenceMask - **28**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1171.2 Field Documentation

8.1171.2.1 int8_t unpack_wds_GetDefaultProfileV2_t::apnname[255]

8.1171.2.2 uint8_t unpack_wds_GetDefaultProfileV2_t::apnsize

8.1171.2.3 uint32_t unpack_wds_GetDefaultProfileV2_t::auth

8.1171.2.4 uint32_t unpack_wds_GetDefaultProfileV2_t::ipaddr

8.1171.2.5 uint16_t unpack_wds_GetDefaultProfileV2_t::ipaddrv6[8]

8.1171.2.6 int8_t unpack_wds_GetDefaultProfileV2_t::name[255]

8.1171.2.7 uint8_t unpack_wds_GetDefaultProfileV2_t::namesize

8.1171.2.8 swi_uint256_t unpack_wds_GetDefaultProfileV2_t::ParamPresenceMask

8.1171.2.9 uint32_t unpack_wds_GetDefaultProfileV2_t::pdptype

8.1171.2.10 uint32_t unpack_wds_GetDefaultProfileV2_t::pridns

8.1171.2.11 uint16_t unpack_wds_GetDefaultProfileV2_t::pridnsv6[8]

8.1171.2.12 int8_t unpack_wds_GetDefaultProfileV2_t::pwd[255]

8.1171.2.13 uint8_t unpack_wds_GetDefaultProfileV2_t::pwdsize

8.1171.2.14 uint32_t unpack_wds_GetDefaultProfileV2_t::secdns

8.1171.2.15 uint16_t unpack_wds_GetDefaultProfileV2_t::secdnsv6[8]

8.1171.2.16 int8_t unpack_wds_GetDefaultProfileV2_t::username[255]

8.1171.2.17 uint8_t unpack_wds_GetDefaultProfileV2_t::usersize

8.1172 unpack_wds_GetDormancyState_t Struct Reference

Data Fields

- uint32_t dormancyState
- swi_uint256_t ParamPresenceMask

8.1172.1 Detailed Description

This structure contains unpack get dormancy state information.

Parameters

dormancyState	• Dormancy state of current packet data session – 1 - Traffic channel dormant – 2 - Traffic channel active • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1172.2 Field Documentation

8.1172.2.1 uint32_t unpack_wds_GetDormancyState_t::dormancyState

8.1172.2.2 swi_uint256_t unpack_wds_GetDormancyState_t::ParamPresenceMask

8.1173 unpack_wds_GetLastMobileIPError_t Struct Reference

Data Fields

- uint32_t error
- swi_uint256_t ParamPresenceMask

8.1173.1 Detailed Description

structure to store last mobile IP error.

Parameters

error	• Status of last MIP call (or attempt) – Zero - Success – NonZero - Error code See qaGobiApiTableCallEndReasons.h for Mobile IP Error codes • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1173.2 Field Documentation

8.1173.2.1 uint32_t unpack_wds_GetLastMobileIPError_t::error

8.1173.2.2 swi_uint256_t unpack_wds_GetLastMobileIPError_t::ParamPresenceMask

8.1174 unpack_wds_GetMobileIP_t Struct Reference

Data Fields

- uint32_t mipMode
- swi_uint256_t ParamPresenceMask

8.1174.1 Detailed Description

This structure contains unpack get mobile IP information.

Parameters

mipMode	• Mobile IP setting – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1174.2 Field Documentation

8.1174.2.1 uint32_t unpack_wds_GetMobileIP_t::mipMode

8.1174.2.2 swi_uint256_t unpack_wds_GetMobileIP_t::ParamPresenceMask

8.1175 unpack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t enabled
- uint32_t address
- uint32_t primaryHA
- uint32_t secondaryHA
- uint8_t revTunneling
- uint8_t naiSize
- int8_t NAI [255]
- uint32_t HASPI
- uint32_t AAASPI
- uint32_t HAState
- uint32_t AAAState
- swi_uint256_t ParamPresenceMask

8.1175.1 Detailed Description

This structure contains unpack get mobile IP profile information.

Parameters

enabled	• Profile enabled: – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **16**
address	• Home IPv4 address: • Bit to check in ParamPresenceMask - **17**
primaryHA	• Primary home agent IPv4 address • Bit to check in ParamPresenceMask - **18**
secondaryHA	• Secondary home agent IPv4 address • Bit to check in ParamPresenceMask - **19**
revTunneling	• Reverse tunneling enabled – 0 - Disabled – 1 - Enabled • Bit to check in ParamPresenceMask - **20**
naiSize	• The maximum number of characters (including NULL terminator) that the NAI array can contain. • Bit to check in ParamPresenceMask - **21**
NAI	• Network access identifier string • Bit to check in ParamPresenceMask - **21**
HASPI	• Home agent security parameter index • Bit to check in ParamPresenceMask - **22**
AAASPI	• AAA server security parameter index • Bit to check in ParamPresenceMask - **23**
HAState	• Home agent key state – 0 - Unset – 1 - Set, default value – 2 - Set, modified from default – 3 - 0xFFFFFFFF - Unknown • Bit to check in ParamPresenceMask - **26**

AAAState	• AAA key state – 0 - Unset – 1 - Set, default value – 2 - Set, modified from default – 3 - 0xFFFFFFFF - Unknown • Bit to check in ParamPresenceMask - **27**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1175.2 Field Documentation

8.1175.2.1 uint32_t unpack_wds_GetMobileIPProfile_t::AAASPI

8.1175.2.2 uint32_t unpack_wds_GetMobileIPProfile_t::AAAState

8.1175.2.3 uint32_t unpack_wds_GetMobileIPProfile_t::address

8.1175.2.4 uint8_t unpack_wds_GetMobileIPProfile_t::enabled

8.1175.2.5 uint32_t unpack_wds_GetMobileIPProfile_t::HASPI

8.1175.2.6 uint32_t unpack_wds_GetMobileIPProfile_t::HAState

8.1175.2.7 int8_t unpack_wds_GetMobileIPProfile_t::NAI[255]

8.1175.2.8 uint8_t unpack_wds_GetMobileIPProfile_t::naiSize

8.1175.2.9 swi_uint256_t unpack_wds_GetMobileIPProfile_t::ParamPresenceMask

8.1175.2.10 uint32_t unpack_wds_GetMobileIPProfile_t::primaryHA

8.1175.2.11 uint8_t unpack_wds_GetMobileIPProfile_t::revTunneling

8.1175.2.12 uint32_t unpack_wds_GetMobileIPProfile_t::secondaryHA

8.1176 unpack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * pTXPacketSuccesses
- uint32_t * pRXPacketSuccesses
- uint32_t * pTXPacketErrors
- uint32_t * pRXPacketErrors
- uint32_t * pTXPacketOverflows
- uint32_t * pRXPacketOverflows
- uint64_t * pTXOkBytesCount
- uint64_t * pRXOkBytesCount
- uint64_t * pTXOKBytesLastCall
- uint64_t * pRXOKBytesLastCall
- uint32_t * pTXDroppedCount

- uint32_t * pRXDroppedCount
- swi_uint256_t ParamPresenceMask

8.1176.1 Detailed Description

This structure contains unpack get packet statistics information.

Parameters

pTXPacket-Successes	• No of transmitted Packets without error. • Bit to check in ParamPresenceMask - **16**
pRXPacket-Successes	• No of received Packets without error. • Bit to check in ParamPresenceMask - **17**
pTXPacketErrors	• Number of outgoing packets with framing errors. • Bit to check in ParamPresenceMask - **18**
pRXPacket-Errors	• Number of incoming packets with framing errors. • Bit to check in ParamPresenceMask - **19**
pTXPacket-Overflows	• Number of packets dropped because Tx buffer overflowed (out of memory). • Bit to check in ParamPresenceMask - **20**
pRXPacket-Overflows	• Number of packets dropped because Rx buffer overflowed (out of memory). • Bit to check in ParamPresenceMask - **21**
pTXOkBytes-Count	• Number of bytes transmitted without error. • Bit to check in ParamPresenceMask - **25**
pRXOkBytes-Count	• Number of bytes received without error. • Bit to check in ParamPresenceMask - **26**
pTXOKBytes-LastCall	• Number of bytes transmitted without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support • Bit to check in ParamPresenceMask - **27**
pRXOKBytes-LastCall	• Number of bytes received without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support • Bit to check in ParamPresenceMask - **28**

pTXDropped-Count	• Number of outgoing packets dropped. • Bit to check in ParamPresenceMask - **29**
pRXDropped-Count	• Number of incoming packets dropped. • Bit to check in ParamPresenceMask - **30**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1176.2 Field Documentation

8.1176.2.1 swi_uint256_t unpack_wds_GetPacketStatistics_t::ParamPresenceMask

8.1176.2.2 uint32_t* unpack_wds_GetPacketStatistics_t::pRXDroppedCount

8.1176.2.3 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOkBytesCount

8.1176.2.4 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOKBytesLastCall

8.1176.2.5 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketErrors

8.1176.2.6 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketOverflows

8.1176.2.7 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketSuccesses

8.1176.2.8 uint32_t* unpack_wds_GetPacketStatistics_t::pTXDroppedCount

8.1176.2.9 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOkBytesCount

8.1176.2.10 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOKBytesLastCall

8.1176.2.11 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketErrors

8.1176.2.12 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketOverflows

8.1176.2.13 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketSuccesses

8.1177 unpack_wds_GetPacketStatus_t Struct Reference

Data Fields

- uint32_t tXPacketSuccesses
- uint32_t rXPacketSuccesses
- uint32_t tXPacketErrors
- uint32_t rXPacketErrors
- uint32_t tXPacketOverflows
- uint32_t rXPacketOverflows
- uint64_t tXOkBytesCount
- uint64_t rXOkBytesCount
- uint64_t tXOKBytesLastCall

- uint64_t rXOKBytesLastCall
- uint32_t tXDroppedCount
- uint32_t rXDroppedCount
- swi_uint256_t ParamPresenceMask

8.1177.1 Detailed Description

Returns the packet data transfer statistics since the start of the current packet data.

Parameters

tXPacket-Successes	• No. of packets transmitted without error • Bit to check in ParamPresenceMask - **16**
rXPacket-Successes	• No. of packets received without error • Bit to check in ParamPresenceMask - **17**
tXPacketErrors	• No. of outgoing packets with framing errors • Bit to check in ParamPresenceMask - **18**
rXPacketErrors	• No. of incoming packets with framing errors • Bit to check in ParamPresenceMask - **19**
txPacket-Overflows	• Number of packets dropped because Tx buffer overflowed • Bit to check in ParamPresenceMask - **20**
rxPacket-Overflows	• Number of packets dropped because Rx buffer overflowed • Bit to check in ParamPresenceMask - **21**
txOkBytesCount	• No of bytes transmitted without error. • Bit to check in ParamPresenceMask - **25**
rxOkBytesCount	• No of bytes received without error. • Bit to check in ParamPresenceMask - **26**
txOKBytesLast-Call	• No of bytes transmitted without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support • Bit to check in ParamPresenceMask - **27**
rxOKBytesLast-Call	• Number of bytes received without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support • Bit to check in ParamPresenceMask - **28**

txDroppedCount	• Number of outgoing packets dropped. • Bit to check in ParamPresenceMask - **29**
rxDroppedCount	• Number of incoming packets dropped. • Bit to check in ParamPresenceMask - **30**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1177.2 Field Documentation

8.1177.2.1 swi_uint256_t unpack_wds_GetPacketStatus_t::ParamPresenceMask

8.1177.2.2 uint32_t unpack_wds_GetPacketStatus_t::rXDroppedCount

8.1177.2.3 uint64_t unpack_wds_GetPacketStatus_t::rXOkBytesCount

8.1177.2.4 uint64_t unpack_wds_GetPacketStatus_t::rXOKBytesLastCall

8.1177.2.5 uint32_t unpack_wds_GetPacketStatus_t::rXPacketErrors

8.1177.2.6 uint32_t unpack_wds_GetPacketStatus_t::rXPacketOverflows

8.1177.2.7 uint32_t unpack_wds_GetPacketStatus_t::rXPacketSuccesses

8.1177.2.8 uint32_t unpack_wds_GetPacketStatus_t::tXDroppedCount

8.1177.2.9 uint64_t unpack_wds_GetPacketStatus_t::tXOkBytesCount

8.1177.2.10 uint64_t unpack_wds_GetPacketStatus_t::tXOKBytesLastCall

8.1177.2.11 uint32_t unpack_wds_GetPacketStatus_t::tXPacketErrors

8.1177.2.12 uint32_t unpack_wds_GetPacketStatus_t::tXPacketOverflows

8.1177.2.13 uint32_t unpack_wds_GetPacketStatus_t::tXPacketSuccesses

8.1178 unpack_wds_GetSessionDuration_t Struct Reference

Data Fields

- uint64_t callDuration
- swi_uint256_t ParamPresenceMask

8.1178.1 Detailed Description

This structure contains unpack get session duration information.

Parameters

callDuration	• Duration of the current packet session in milliseconds • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1178.2 Field Documentation

8.1178.2.1 uint64_t unpack_wds_GetSessionDuration_t::callDuration

8.1178.2.2 swi_uint256_t unpack_wds_GetSessionDuration_t::ParamPresenceMask

8.1179 unpack_wds_GetSessionDurationV2_t Struct Reference

Data Fields

- uint64_t callDuration
- uint64_t * pLastCallDuration
- uint64_t * pCallActiveDuration
- uint64_t * pLastCallActiveDuration
- swi_uint256_t ParamPresenceMask

8.1179.1 Detailed Description

This structure contains unpack get session duration information -V2.

Parameters

callDuration	• Duration of the current packet session in milliseconds • Bit to check in ParamPresenceMask - **1**
pLastCall-Duration	• Duration of the last data session in milliseconds • Bit to check in ParamPresenceMask - **16**
pCallActive-Duration	• Duration of the active time of current data session in milliseconds • Bit to check in ParamPresenceMask - **17**
pLastCallActive-Duration	• Duration of the active time of last data session in milliseconds • Bit to check in ParamPresenceMask - **18**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1179.2 Field Documentation

8.1179.2.1 uint64_t unpack_wds_GetSessionDurationV2_t::callDuration

8.1179.2.2 swi_uint256_t unpack_wds_GetSessionDurationV2_t::ParamPresenceMask

8.1179.2.3 uint64_t* unpack_wds_GetSessionDurationV2_t::pCallActiveDuration

8.1179.2.4 uint64_t* unpack_wds_GetSessionDurationV2_t::pLastCallActiveDuration

8.1179.2.5 uint64_t* unpack_wds_GetSessionDurationV2_t::pLastCallDuration

8.1180 unpack_wds_GetSessionState_t Struct Reference

Data Fields

- uint32_t connectionStatus
- swi_uint256_t ParamPresenceMask

8.1180.1 Detailed Description

This structure contains unpack get session state information.

Parameters

connection-Status	- state of the current packet data session. • 1 - Disconnected. • 2 - Connected. • 3 - Suspended. • 4 - Authenticating. • Bit to check in ParamPresenceMask - **1**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1180.2 Field Documentation

8.1180.2.1 uint32_t unpack_wds_GetSessionState_t::connectionStatus

8.1180.2.2 swi_uint256_t unpack_wds_GetSessionState_t::ParamPresenceMask

8.1181 unpack_wds_RMSetTransferStatistics_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1181.1 Detailed Description

This structure contains unpack fetch current data system transfer statistics information.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1181.2 Field Documentation

8.1181.2.1 swi_uint256_t unpack_wds_RMSetTransferStatistics_t::ParamPresenceMask

8.1182 unpack_wds_SetMobileIPProfile_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.1182.1 Detailed Description

This structure contains set mobile IP profile information.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1182.2 Field Documentation

8.1182.2.1 swi_uint256_t unpack_wds_SetMobileIPProfile_t::ParamPresenceMask

8.1183 unpack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- PackCreateProfileOut * pCreateProfileOut
- uint8_t * pProfileID
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1183.1 Detailed Description

This structure contains unpack create profile information.

Parameters

pCreateProfile-Out	• SLQS Create profile Information • See PackCreateProfileOut
pProfileID	• SLQS profile identifier information

TlvresultPresent	• TLV present flag, TURE if presented, otherwise FALSE
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1183.2 Field Documentation

8.1183.2.1 swi_uint256_t unpack_wds_SLQSCreateProfile_t::ParamPresenceMask

8.1183.2.2 PackCreateProfileOut* unpack_wds_SLQSCreateProfile_t::pCreateProfileOut

8.1183.2.3 uint8_t* unpack_wds_SLQSCreateProfile_t::pProfileID

8.1183.2.4 uint16_t unpack_wds_SLQSCreateProfile_t::TlvresultPresent

8.1184 unpack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint16_t extendedErrorCode
- swi_uint256_t ParamPresenceMask

8.1184.1 Detailed Description

This structure contains unpack delete profile information.

Parameters

extendedError-Code[OUT]	• The extended error code received from DS Profile subsystem of type eWDS_ERR_PROFILE_-REG_xxx. • Error code will only will be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERN-AL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description. • Bit to check in ParamPresenceMask - **224**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1184.2 Field Documentation

8.1184.2.1 uint16_t unpack_wds_SLQSDeleteProfile_t::extendedErrorCode

8.1184.2.2 swi_uint256_t unpack_wds_SLQSDeleteProfile_t::ParamPresenceMask

8.1185 unpack_wds_SLQSDUNCallInfoCallBack_ind_t Struct Reference

Data Fields

- wds_ConnStatusTlv CSTlv
- wds_LastMdmCallEndRsnTlv LMCERTlv
- wds_TXBytesOKTlv TXBOTlv
- wds_RXBytesOKTlv RXBOTlv
- wds_DormStatTlv DSTlv
- wds_DataBearTechTlv DBTTlv
- wds_ChannelRateTlv CRTlv
- swi_uint256_t ParamPresenceMask

8.1185.1 Detailed Description

Structure used to store Reg Mgr Config Indication Parameters.

Parameters

CSTlv	• Connection status tlv • See wds_ConnStatusTlv for more information • Bit to check in ParamPresenceMask - **16**
LMCERTlv	• last modem call end reason tlv • see wds_LastMdmCallEndRsnTlv for more information • Bit to check in ParamPresenceMask - **17**
TXBOTlv	• Tx bytes OK tlv • see wds_TXBytesOKTlv for more information • Bit to check in ParamPresenceMask - **18**
RXBOTlv	• Rx bytes OK tlv • see wds_RXBytesOKTlv for more information • Bit to check in ParamPresenceMask - **19**
DSTlv	• Dormancy status tlv • see wds_DormStatTlv for more information • Bit to check in ParamPresenceMask - **20**
DBTTlv	• Data bear technology tlv • see wds_DataBearTechTlv for more information • Bit to check in ParamPresenceMask - **21**
CRTlv	• Channel rate Tlv • see wds_ChannelRateTlv for more information • Bit to check in ParamPresenceMask - **22**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

Note

: None

8.1185.2 Field Documentation

8.1185.2.1 wds_ChannelRateTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::CRTlv

8.1185.2.2 wds_ConnStatusTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::CSTlv

8.1185.2.3 wds_DataBearTechTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::DBTTlv

8.1185.2.4 wds_DormStatTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::DSTlv

8.1185.2.5 wds_LastMdmCallEndRsnTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::LMCERTlv

8.1185.2.6 swi_uint256_t unpack_wds_SLQSDUNCallInfoCallBack_ind_t::ParamPresenceMask

8.1185.2.7 wds_RXBytesOKTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::RXBOTlv

8.1185.2.8 wds_TXBytesOKTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::TXBOTlv

8.1186 unpack_wds_SLQSGet3GPPConfigItem_t Struct Reference

Data Fields

- uint16_t LTEAttachProfile
- uint16_t profileList [5]
- uint8_t defaultPDNEnabled
- uint8_t _3gppRelease
- uint16_t LTEAttachProfileList [24]
- uint16_t LTEAttachProfileListLen
- swi_uint256_t ParamPresenceMask

8.1186.1 Detailed Description

This structure contains unpack get 3GPP configure item information.

Parameters

	LTEAttachProfile	• LTE Attach Profile – points to a single WORD Value indicating the attached LTE Profile – Optional parameter with possible values 1-16 (EM/MC73xx or earlier) • This setting is deprecated on MC7430/MC7455/EM7430/EM7455 • Bit to check in ParamPresenceMask - **16**

	profileList	Profile List • an array of 4 profile configurations • Each element points to a single WORD value indicating profile • Optional parameter with possible values – 1 - 16 (MC/EM73xx and before) – 1 - 24 (MC7430/MC7455/EM7430/EM7455 and onwards) • function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present • Bit to check in ParamPresenceMask - **17** • Note: the 5th entry is currently ignored, please set it to zero
out	*defaultPDN-Enabled*	• 0 - disabled • 1 - enabled • Bit to check in ParamPresenceMask - **18**
out	*_3gppRelease*	3GPP release • 0 - Release_99 • 1 - Release_5 • 2 - Release_6 • 3 - Release_7 • 4 - Release_8 • 5 - Release_9 (In 9x30 and toworads) • 6 - Release_10 (In 9x30 and toworads) • 7 - Release_11 (In 9x30 and toworads) • Bit to check in ParamPresenceMask - **19**
out	*LTEAttach-ProfileList*	• pointer to WORD array indiciating LTE Attach Profile List – Optional parameter – possible values: 1-24 – This setting is only supported for MC7430/MC7455/EM7430/EM7455 onwards – Please provide attach profiles in order of decreasing priority in this list. • Bit to check in ParamPresenceMask - **20**
in,out	*LTEAttach-ProfileListLen*	• Number of element in pLTEAttachProfileList – valid range: 1-24 – This setting is only supported for MC7430/MC7455/EM7430/EM7455 onwards • Bit to check in ParamPresenceMask - **20**
	ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1186.2 Field Documentation

8.1186.2.1 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::_3gppRelease

8.1186.2.2 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::defaultPDNEnabled

8.1186.2.3 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfile

8.1186.2.4 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileList[24]

8.1186.2.5 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileListLen

8.1186.2.6 swi_uint256_t unpack_wds_SLQSGet3GPPConfigItem_t::ParamPresenceMask

8.1186.2.7 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::profileList[5]

8.1187 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

Data Fields

- uint8_t prefNetwork
- uint8_t networkInfoLen
- currNetworkInfo currNetworkInfo [255]
- swi_uint256_t ParamPresenceMask

8.1187.1 Detailed Description

Data System Status

Parameters

prefNetwork	• Preferred Network • Values: – 0 - 3GPP – 1 - 3GPP2 • Bit to check in ParamPresenceMask - **16**
networkInfoLen	• As input parameter size assigned to next parameter i.e. network information • As output the actual number of network information elements returned by the device • Bit to check in ParamPresenceMask - **16**
currNetworkInfo	• Network information • See currNetworkInfo for more details • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1187.2 Field Documentation

8.1187.2.1 currNetworkInfo unpack_wds_SLQSGetCurrDataSystemStat_t::currNetworkInfo[255]

8.1187.2.2 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::networkInfoLen

8.1187.2.3 swi_uint256_t unpack_wds_SLQSGetCurrDataSystemStat_t::ParamPresenceMask

8.1187.2.4 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::prefNetwork

8.1188 unpack_wds_SLQSGetCurrentChannelRate_t Struct Reference

Data Fields

- uint32_t current_channel_tx_rate
- uint32_t current_channel_rx_rate
- uint32_t max_channel_tx_rate
- uint32_t max_channel_rx_rate
- swi_uint256_t ParamPresenceMask

8.1188.1 Detailed Description

This structure contains unpack get current channel rate information.

Parameters

current_channel_tx_rate	• Current Channel Tx Rate. • Instantaneous channel Tx rate in bits per second. • In 9x15, this is the total current channel rate for all PDNs combined. • In 9x30 and later, this is the channel rate for a specific PDN. • Bit to check in ParamPresenceMask - **1**
current_channel_rx_rate	• Current Channel Rx Rate. • Instantaneous channel Rx rate in bits per second. • In 9x15, this is the total current channel rate for all PDNs combined. • In 9x30 and later, this is the channel rate for a specific PDN • Bit to check in ParamPresenceMask - **1**
max_channel_tx_rate	• Max Channel Tx Rate. • Maximum total Tx rate that modem is able to support in current serving system in bits per second. • In 9x15, this is a default hard coded value for the current serving system. • Bit to check in ParamPresenceMask - **1**
max_channel_rx_rate	• Max Channel Rx Rate. • Maximum total Rx rate that modem is able to support in current serving system in bits per second. • In 9x15, this is a default hard coded value for the current serving system. • Bit to check in ParamPresenceMask - **1**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1188.2 Field Documentation

8.1188.2.1 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_rx_rate

8.1188.2.2 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_tx_rate

8.1188.2.3 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_rx_rate

8.1188.2.4 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_tx_rate

8.1188.2.5 swi_uint256_t unpack_wds_SLQSGetCurrentChannelRate_t::ParamPresenceMask

8.1189 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference

Data Fields

- uint8_t dataBearerMask
- qmiWDSDataBearerTechnology curDataBearerTechnology
- qmiWDSDataBearerTechnology lastCallDataBearerTechnology
- swi_uint256_t ParamPresenceMask

8.1189.1 Detailed Description

Structure to hold the data bearer technology values

Parameters

dataBearerMask[OUT]	• This bit mask indicates if data bearer information for the current and/or last call has been received from the device. If a bit is set, then the information is available in the corresponding structure i.e. the one provided by the caller. Refer to liteQmiDataBearerMasks for bit-mask positions. • Bit to check in ParamPresenceMask - **1**
curDataBearerTechnology[OUT]	• current data bearer technology value. • See qmiWDSDataBearerTechnology • Bit to check in ParamPresenceMask - **1**
lastCallDataBearerTechnology[OUT]	• last call data bearer technology value. • See qmiWDSDataBearerTechnology – NULL if the parameter is not required • Bit to check in ParamPresenceMask - **16**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1189.2 Field Documentation

8.1189.2.1 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::curDataBearerTechnology

8.1189.2.2 uint8_t unpack_wds_SLQSGetDataBearerTechnology_t::dataBearerMask

8.1189.2.3 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::lastCallDataBearerTechnology

8.1189.2.4 swi_uint256_t unpack_wds_SLQSGetDataBearerTechnology_t::ParamPresenceMask

8.1190 unpack_wds_SLQSGetDUNCallInfo_t Struct Reference

Data Fields

- connectionStatus connectionStatus
- uint16_t callEndReason
- uint64_t txOKBytesCount
- uint64_t rxOKBytesCount
- uint8_t dormancyStatus
- uint8_t dataBearerTech
- dunchannelRate channelRate
- uint64_t lastCallTXOKBytesCnt
- uint64_t lastCallRXOKBytesCnt
- uint64_t mdmCallDurationActive
- uint8_t lastCallDataBearerTech
- swi_uint256_t ParamPresenceMask

8.1190.1 Detailed Description

This structure contains the DUN Call Info response parameters

Parameters

connectionStatus	• See connectionStatus for more information • Bit to check in ParamPresenceMask - **16**
callEndReason	• Last modem call end reason • See qaGobiApiTableCallEndReasons.h for Call End Reason • Only valid if the last call made was DUN, else zero is returned • Bit to check in ParamPresenceMask - **17**
txOKBytesCount	• Number of bytes transmitted without error • Returned only if a data call is up • Bit to check in ParamPresenceMask - **18**
rxOKBytesCount	• Number of bytes received without error • Returned only if a data call is up • Bit to check in ParamPresenceMask - **19**

dormancyStatus	• Current traffic channel status • Returned if a data call is up – 0x01 - Traffic channel dormant – 0x02 - Traffic channel active • Bit to check in ParamPresenceMask - **20**
dataBearerTech	• Current data bearer technology • Returned only if a data call is up – 0x01 - cdma2000 1X – 0x02 - cdma2000 HRPD (1xEV-DO) – 0x03 - GSM – 0x04 - UMTS – 0x05 - cdma200 HRPD (1xEV-DO RevA) – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - cdma2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA • Bit to check in ParamPresenceMask - **21**
channelRate	• See dunchannelRate for more information • Bit to check in ParamPresenceMask - **22**
lastCallTXOK-BytesCnt	• Number of bytes transmitted without error during the last data call (0 if no call was made). • Return only if not in a call and the previous call was made using DUN. • Bit to check in ParamPresenceMask - **23**
lastCallRXOK-BytesCnt	• Number of bytes transmitted without error during the last data call (0 if no call was made). • Return only if not in a call and the previous call was made using DUN. • Bit to check in ParamPresenceMask - **24**

mdmCall-DurationActive	• Duration that the call is active in milliseconds • If the modem connection status is connected, this represents the active duration of the current DUN call • If the modem connection status is disconnected, this represents the duration of the last DUN call since the device was powered up (0 if no call has been made or if the last call was not DUN) • Bit to check in ParamPresenceMask - **25**
lastCallData-BearerTech	• Last Call Data Bearer Technology • Returned only if not in a call and when the previous call was made using DUN – 0x01 - cdma2000 1X – 0x02 - cdma2000 HRPD (1xEV-DO) – 0x03 - GSM – 0x04 - UMTS – 0x05 - cdma200 HRPD (1xEV-DO Rev A) – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - cdma2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA • Bit to check in ParamPresenceMask - **32**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1190.2 Field Documentation

8.1190.2.1 uint16_t unpack_wds_SLQSGetDUNCallInfo_t::callEndReason

8.1190.2.2 dunchannelRate unpack_wds_SLQSGetDUNCallInfo_t::channelRate

8.1190.2.3 connectionStatus unpack_wds_SLQSGetDUNCallInfo_t::connectionStatus

8.1190.2.4 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dataBearerTech

8.1190.2.5 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dormancyStatus

8.1190.2.6 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallDataBearerTech

8.1190.2.7 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallRXOKBytesCnt

8.1190.2.8 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallTXOKBytesCnt

8.1190.2.9 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::mdmCallDurationActive

8.1190.2.10 swi_uint256_t unpack_wds_SLQSGetDUNCallInfo_t::ParamPresenceMask

8.1190.2.11 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::rxOKBytesCount

8.1190.2.12 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::txOKBytesCount

8.1191 unpack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- UnPackGetProfileSettingOut * pProfileSettings
- uint8_t ProfileType
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1191.1 Detailed Description

This structure contains the profile setting information of parameter pOutput for API unpack_wds_SLQSGetProfile-Settings

Parameters

pProfileSettings	- Profile Settings
ProfileType	- Profile Type • 0 - Profile 3GPP • 1 - Profile 3GPP2
Tlvresult	- unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1191.2 Field Documentation

8.1191.2.1 swi_uint256_t unpack_wds_SLQSGetProfileSettings_t::ParamPresenceMask

8.1191.2.2 UnPackGetProfileSettingOut* unpack_wds_SLQSGetProfileSettings_t::pProfileSettings

8.1191.2.3 uint8_t unpack_wds_SLQSGetProfileSettings_t::ProfileType

8.1191.2.4 uint16_t unpack_wds_SLQSGetProfileSettings_t::Tlvresult

8.1192 unpack_wds_SLQSGetProfileSettingsV2_t Struct Reference

Data Fields

- UnPackGetProfileSettingOutV2 * pProfileSettings
- uint8_t ProfileType
- uint16_t Tlvresult

- swi_uint256_t ParamPresenceMask

8.1192.1 Detailed Description

This structure contains the profile setting information of parameter pOutput for API unpack_wds_SLQSGetProfile-SettingsV2

Parameters

pProfileSettings	- Profile Settings
ProfileType	- Profile Type • 0 - Profile 3GPP • 1 - Profile 3GPP2
Tlvresult	- unpack Tlv Result.
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1192.2 Field Documentation

8.1192.2.1 swi_uint256_t unpack_wds_SLQSGetProfileSettingsV2_t::ParamPresenceMask

8.1192.2.2 UnPackGetProfileSettingOutV2* unpack_wds_SLQSGetProfileSettingsV2_t::pProfileSettings

8.1192.2.3 uint8_t unpack_wds_SLQSGetProfileSettingsV2_t::ProfileType

8.1192.2.4 uint16_t unpack_wds_SLQSGetProfileSettingsV2_t::Tlvresult

8.1193 unpack_wds_SLQSGetRuntimeSettings_t Struct Reference

Data Fields

- uint32_t IPv4
- uint8_t ProfileName [128]
- uint32_t PDPType
- uint8_t APNName [128]
- uint32_t PrimaryDNSV4
- uint32_t SecondaryDNSV4
- LibPackUMTSQoS UMTSGrantedQoS
- struct wds_GPRSQoS GPRSGrantedQoS
- uint8_t Username [128]
- uint32_t Authentication
- struct wds_ProfileIdentifier ProfileID
- uint32_t GWAddressV4
- uint32_t SubnetMaskV4
- uint8_t PCSCFAddrPCO
- struct wds_PCSCFIPv4ServerAddressList ServerAddrList
- struct wds_PCSCFFQDNAddressList PCSCFFQDNAddrList
- uint16_t PrimaryDNSV6 [8]
- uint16_t SecondaryDNSV6 [8]
- uint32_t Mtu
- struct wds_DomainNameList DomainList

- uint8_t IPFamilyPreference
- uint8_t IMCNflag
- uint16_t Technology
- struct wds_IPV6AddressInfo IPV6AddrInfo
- struct wds_IPV6GWAddressInfo IPV6GWAddrInfo
- swi_uint256_t ParamPresenceMask

8.1193.1 Detailed Description

This structure contains unpack get runtime settings information.

Parameters

IPv4	ipv4 address • Bit to check in ParamPresenceMask - **30**
ProfileName	profile name • Bit to check in ParamPresenceMask - **16**
PDPType	PDP type • Bit to check in ParamPresenceMask - **17**
APNName	APN name • Bit to check in ParamPresenceMask - **20**
PrimaryDNSV4	primary dns IPV4 • Bit to check in ParamPresenceMask - **21**
SecondaryDNS-V4	secondary dns IPV4 • Bit to check in ParamPresenceMask - **22**
UMTSGranted-QoS	UMTS Granted Qos • Bit to check in ParamPresenceMask - **23**
GPRSGranted-QoS	GPRS Granted QoS • Bit to check in ParamPresenceMask - **25**
Username	username for authentication process • Bit to check in ParamPresenceMask - **27**
Authentication	authentication for authentication process • Bit to check in ParamPresenceMask - **29**
ProfielID	profile ID • Bit to check in ParamPresenceMask - **31**
GWAddressV4	Gateway IPv4 • Bit to check in ParamPresenceMask - **32**
SubnetMaskV4	Subnet mask IPV4 • Bit to check in ParamPresenceMask - **33**
PCSCFAddrPC-O	PCSCF address PCO • Bit to check in ParamPresenceMask - **34**

ServerAddrList	PCSCF server address list IPV4 • Bit to check in ParamPresenceMask - **35**
PCSCFFQDN-AddrList	PCSCF FQDN address list IPV4 • Bit to check in ParamPresenceMask - **36**
PrimaryDNSV6	Primary DNS IPV6 • Bit to check in ParamPresenceMask - **39**
SecondryDNS-V6	Secondry DNS IPV6 • Bit to check in ParamPresenceMask - **40**
Mtu	actual (runtime) Maximum Transfer Unit • Bit to check in ParamPresenceMask - **41**
DomainList	domain list • Bit to check in ParamPresenceMask - **42**
IPFamily-Preference	ip family preference, it could be IPV4 or IPV6 • Bit to check in ParamPresenceMask - **43**
IMCNflag	IM control flag, value: TRUE or FALSE • Bit to check in ParamPresenceMask - **44**
Technology	technology on which current packet data session is in progress • Values: – 32767 - CDMA – 32764 - UMTS – 30592 - EPC – 30584 - modem link local • Bit to check in ParamPresenceMask - **45**
IPV6AddrInfo	address information IPV6 • Bit to check in ParamPresenceMask - **37**
IPV6GWAddr-Info	gateway address information IPV6 • Bit to check in ParamPresenceMask - **38**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1193.2 Field Documentation

8.1193.2.1 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::APNName[128]

8.1193.2.2 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Authentication

8.1193.2.3 struct wds_DomainNameList unpack_wds_SLQSGetRuntimeSettings_t::DomainList

8.1193.2.4 struct wds_GPRSQoS unpack_wds_SLQSGetRuntimeSettings_t::GPRSGrantedQoS

8.1193.2.5 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::GWAddressV4

8.1193.2.6 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IMCNflag

8.1193.2.7 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IPFamilyPreference

8.1193.2.8 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::IPv4

8.1193.2.9 struct wds_IPV6AddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPV6AddrInfo

8.1193.2.10 struct wds_IPV6GWAddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPV6GWAddrInfo

8.1193.2.11 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Mtu

8.1193.2.12 swi_uint256_t unpack_wds_SLQSGetRuntimeSettings_t::ParamPresenceMask

8.1193.2.13 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::PCSCFAddrPCO

8.1193.2.14 struct wds_PCSCFFQDNAddressList unpack_wds_SLQSGetRuntimeSettings_t::PCSCFFQDNAddrList

8.1193.2.15 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PDPType

8.1193.2.16 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV4

8.1193.2.17 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV6[8]

8.1193.2.18 struct wds_ProfileIdentifier unpack_wds_SLQSGetRuntimeSettings_t::ProfileID

8.1193.2.19 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::ProfileName[128]

8.1193.2.20 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV4

8.1193.2.21 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV6[8]

8.1193.2.22 struct wds_PCSCFIPv4ServerAddressList unpack_wds_SLQSGetRuntimeSettings_t::ServerAddrList

8.1193.2.23 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SubnetMaskV4

8.1193.2.24 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::Technology

8.1193.2.25 LibPackUMTSQoS unpack_wds_SLQSGetRuntimeSettings_t::UMTSGrantedQoS

8.1193.2.26 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::Username[128]

8.1194 unpack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint16_t * pExtErrorCode
- swi_uint256_t ParamPresenceMask

8.1194.1 Detailed Description

This structure contains out parameters for unpack_wds_SLQSModifyProfile

Parameters

pExtErrorCode	• The extended error code received from DS Profile subsystem of type eWDS_ERR_PROFILE_-REG_xxx. • Error code will only will be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERN-AL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description. • Bit to check in ParamPresenceMask - **224**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1194.2 Field Documentation

8.1194.2.1 swi_uint256_t unpack_wds_SLQSModifyProfile_t::ParamPresenceMask

8.1194.2.2 uint16_t* unpack_wds_SLQSModifyProfile_t::pExtErrorCode

8.1195 unpack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.1195.1 Detailed Description

This structure contains unpack Set IP family preference information

Parameters

Tlvresult	unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1195.2 Field Documentation

8.1195.2.1 swi_uint256_t unpack_wds_SLQSSetIPFamilyPreference_t::ParamPresenceMask

8.1195.2.2 uint16_t unpack_wds_SLQSSetIPFamilyPreference_t::Tlvresult

8.1196 unpack_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference

Data Fields

- uint8_t conn_status
- uint8_t reconfigReqd
- uint16_t sessionEndReason
- uint16_t verboseSessnEndReasonType
- uint16_t verboseSessnEndReason

- uint8_t ipFamily
- uint16_t techName
- uint8_t bearerID
- swi_uint256_t ParamPresenceMask

8.1196.1 Detailed Description

This structure contains unpack set packet service status callback information.

Parameters

conn_status	connection status. • Current link status. Values: – 1 - DISCONNECTED – 2 - CONNECTED – 3 - SUSPENDED – 4 - AUTHENTICATING • Bit to check in ParamPresenceMask - **1**
reconfigReqd	Indicates whether the network interface on the host needs to be reconfigured. • Values: – 0 - No need to reconfigure. – 1 - Reconfiguration required.

- Bit to check in ParamPresenceMask - **1**

Parameters

sessionEnd-Reason	Call End Reason • See qaGobiApiTableCallEndReasons.h for Call End Reason • Bit to check in ParamPresenceMask - **16**
verboseSessn-EndReasonType	Verbose call end reason type • Call end reason type. Values: – 0 - Unspecified – 1 - Mobile IP – 2 - Internal – 3 - Call Manager defined – 6 - 3GPP Specification defined – 7 - PPP – 8 - EHRPD – 9 - IPv6 • Bit to check in ParamPresenceMask - **17**
verboseSessn-EndReason	Reason the call ended (verbose) • See qaGobiApiTableCallEndReasons.h for Call End Reason • Bit to check in ParamPresenceMask - **17**

ipFamily	IP family of the packet data connection. • Values – 4 - IPv4 – 6 - IPv6 • Bit to check in ParamPresenceMask - **18**
techName	Technology name of the packet data connection. • Values – 32767 - CDMA – 32764 - UMTS – 30592 - EPC – 30590 - EMBMS – 30584 - Modem Link Local EPC is a logical interface to support LTE/eHRPD handoff. It is returned if the device supports IP session continuity. Modem Link Local is an interface for transferring data between entities on the AP and modem. • Bit to check in ParamPresenceMask - **19**
bearerID	• bearer ID (3GPP) or RLP ID (3GPP2) of the packet data connection. • Valid Values - 0 to 16 – Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1196.2 Field Documentation

8.1196.2.1 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::bearerID

8.1196.2.2 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::conn_status

8.1196.2.3 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::ipFamily

8.1196.2.4 swi_uint256_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::ParamPresenceMask

8.1196.2.5 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::reconfigReqd

8.1196.2.6 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::sessionEndReason

8.1196.2.7 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::techName

8.1196.2.8 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReason

8.1196.2.9 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReasonType

8.1197 unpack_wds_SLQSSetWdsEventCallback_ind_t Struct Reference

Data Fields

- uint8_t xferStatAvail
- uint64_t tx_bytes

- uint64_t rx_bytes
- uint64_t tx_pkts
- uint64_t rx_pkts
- uint8_t mipstatAvail
- uint32_t mipStatus
- uint8_t dBTechAvail
- uint32_t dBTechnology
- uint8_t dormancyStatAvail
- uint32_t dormancyStatus
- uint8_t currDBTechAvail
- uint32_t ratMask
- uint32_t soMask
- uint8_t dataSysStatAvail
- uint8_t prefNetwork
- uint8_t netInfoLen
- wds_currNetworkInfo currNWInfo [255]
- uint8_t dBtechExtAvail
- uint32_t dBtechnologyExt
- uint32_t dBTechExtRatValue
- uint64_t dBTechExtSoMask
- swi_uint256_t ParamPresenceMask

8.1197.1 Detailed Description

This structure contains unpack set WDS event callback information.

Parameters

xferStatAvail	transfer statistic available
tx_bytes	• Number of bytes transmitted without error • Bit to check in ParamPresenceMask - **25**
rx_bytes	• Number of bytes received without error • Bit to check in ParamPresenceMask - **26**
tx_pkts	• Number of packets transmitted without error • Bit to check in ParamPresenceMask - **16**
rx_pkts	• Number of packets received without error. • Bit to check in ParamPresenceMask - **17**
mipstatAvail	Mobile IP status available • Bit to check in ParamPresenceMask - **27**
mipStatus	• Status of the last MIP call (or attempt). • Values – 0x00 - Success • Bit to check in ParamPresenceMask - **27**

dBTechAvail	Data Bearer technology available • Bit to check in ParamPresenceMask - **23**
dBTechnology	• Data Bearer technology – Values * 0x01 - cdma2000 ® 1X * 0x02 - cdma2000 ® HRPD (1xEV-DO) * 0x03 - GSM * 0x04 - UMTS * 0x05 - cdma2000 ® HRPD (1xEV-DO RevA) * 0x06 - EDGE * 0x07 - HSDPA and WCDMA * 0x08 - WCDMA and HSUPA * 0x09 - HSDPA and HSUPA * 0x0A - LTE * 0x0B - cdma2000 ® EHRPD * 0x0C - HSDPA+ and WCDMA * 0x0D - HSDPA+ and HSUPA * 0x0E - DC_HSDPA+ and WCDMA * 0x0F - DC_HSDAP+ and HSUPA * 0x10 - HSDPA+ and 64QAM * 0x11 - HSDPA+, 64QAM and HSUPA * 0x12 - TDSCDMA * 0x13 - TDSCDMA and HSDPA * 0x14 - TDSCDMA and HSUPA * -1 - Unknown • Bit to check in ParamPresenceMask - **23**
dormancyStat-Avail	Dormancy status available • Bit to check in ParamPresenceMask - **24**
dormancyStatus	• Dormancy status • Values – 1 - Traffic channel dormant – 2 - Traffic channel active • Bit to check in ParamPresenceMask - **24**
currDBTechAvail	• Current Data Bearer technology available • Bit to check in ParamPresenceMask - **29**

ratMask	• RAT mask to indicate type of technology • Values – 0x00 - DONT_CARE – 0x8000 - NULL_BEARER • CDMA RAT mask – 0x01 - CDMA_1X – 0x02 - EVDO_REV0 – 0x05 - HRPD – 0x0B - EHRPD • UMTS RAT mask – 0x03 - GPRS – 0x04 - WCDMA – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA • Bit to check in ParamPresenceMask - **29**
soMask	• SO Mask • Bit to check in ParamPresenceMask - **29**
dataSysStatAvail	• Data System Status available • Bit to check in ParamPresenceMask - **36**
prefNetwork	• preferred network • Values – 0 - 3GPP – 1 - 3GPP2
currNWInfo	• Current Network Info • see wds_currNetworkInfo for more info
dBtechExtAvail	Data bearer technology extended available • Bit to check in ParamPresenceMask - **42**

dBtechnologyExt	• Data bearer technology Extended. • Values – WDS_BEARER_TECH_NETWORK_3GPP (0) - 3GPP – WDS_BEARER_TECH_NETWORK_3GPP2 (1) - 3GPP2 • Bit to check in ParamPresenceMask - **42**
dBTechExtRat-Value	• Data bearer technology Extended RAT Value • Values – WDS_BEARER_TECH_RAT_EX_NULL_BEARER (0x00) - NULL bearer – WDS_BEARER_TECH_RAT_EX_3GPP_WCDMA (0x01) - 3GPP WCDMA – WDS_BEARER_TECH_RAT_EX_3GPP_GERAN (0x02) - 3GPP GERAN – WDS_BEARER_TECH_RAT_EX_3GPP_LTE (0x03) - 3GPP LTE – WDS_BEARER_TECH_RAT_EX_3GPP_TDSCDMA (0x04) - 3GPP TDSCDMA – WDS_BEARER_TECH_RAT_EX_3GPP_WLAN (0x05) - 3GPP WLAN – WDS_BEARER_TECH_RAT_EX_3GPP_MAX (0x64) - 3GPP maximum – WDS_BEARER_TECH_RAT_EX_3GPP2_1X (0x65) - 3GPP2 1X – WDS_BEARER_TECH_RAT_EX_3GPP2_HRPD (0x66) - 3GPP2 HRPD – WDS_BEARER_TECH_RAT_EX_3GPP2_EHRPD (0x67) - 3GPP2 EHRPD – WDS_BEARER_TECH_RAT_EX_3GPP2_WLAN (0x68) - 3GPP2 WLAN – WDS_BEARER_TECH_RAT_EX_3GPP2_MAX (0xC8) - 3GPP2 maximum • Bit to check in ParamPresenceMask - **42**

dBTechExtSo-Mask	• Data bearer technology Extended SO Mask • Service Option (SO) mask to indicate the service option or type of application.An SO mask value of zero indicates that this field is ignored. • Values – 0x00 - SO mask unspecified – 3GPP SO mask * 0x01 - WCDMA * 0x02 - HSDPA * 0x04 - HSUPA * 0x08 - HSDPAPLUS * 0x10 - DC HSDPAPLUS * 0x20 - 64 QAM * 0x40 - HSPA * 0x80 - GPRS * 0x100 - EDGE * 0x200 - GSM * 0x400 - S2B * 0x800 - LTE limited service * 0x1000 - LTE FDD * 0x2000 - LTE TDD – 3GPP2 SO mask * 0x01000000 - 1X IS95 * 0x02000000 - 1X IS2000 * 0x04000000 - 1X IS2000 REL A * 0x08000000 - HDR REV0 DPA * 0x10000000 - HDR REVA DPA * 0x20000000 - HDR REVB DPA * 0x40000000 - HDR REVA MPA * 0x80000000 - HDR REVB MPA * 0x100000000 - HDR REVA EMPA * 0x200000000 - HDR REVB EMPA * 0x400000000 - HDR REVB MMPA * 0x800000000 - HDR EVDO FMC • Bit to check in ParamPresenceMask - **42**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1197.2 Field Documentation

8.1197.2.1 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::currDBTechAvail

8.1197.2.2 wds_currNetworkInfo unpack_wds_SLQSSetWdsEventCallback_ind_t::currNWInfo[255]

8.1197.2.3 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dataSysStatAvail

8.1197.2.4 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechAvail

8.1197.2.5 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechExtAvail

8.1197.2.6 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtRatValue

8.1197.2.7 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtSoMask

8.1197.2.8 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechnology

8.1197.2.9 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechnologyExt

8.1197.2.10 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatAvail

8.1197.2.11 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatus

8.1197.2.12 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipstatAvail

8.1197.2.13 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipStatus

8.1197.2.14 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::netInfoLen

8.1197.2.15 swi_uint256_t unpack_wds_SLQSSetWdsEventCallback_ind_t::ParamPresenceMask

8.1197.2.16 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::prefNetwork

8.1197.2.17 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::ratMask

8.1197.2.18 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_bytes

8.1197.2.19 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_pkts

8.1197.2.20 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::soMask

8.1197.2.21 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_bytes

8.1197.2.22 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_pkts

8.1197.2.23 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::xferStatAvail

8.1198 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4HwConfig * pHwConfig
- wdsDhcpv4OptionList * pRequestOptionList
- swi_uint256_t ParamPresenceMask

8.1198.1 Detailed Description

WDS SWI DHCPv4 Config Structure

Parameters

pHWConfig	• pointer to HW Config structure • See wdsDhcpv4HwConfig • Bit to check in ParamPresenceMask - **16**
pRequestOption-List	• pointer to Option List structure to be sent in DHCP request • See wdsDhcpv4OptionList • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1198.2 Field Documentation

8.1198.2.1 swi_uint256_t unpack_wds_SLQSSGetDHCPv4ClientConfig_t::ParamPresenceMask

8.1198.2.2 wdsDhcpv4HwConfig* unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pHwConfig

8.1198.2.3 wdsDhcpv4OptionList* unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pRequestOptionList

8.1199 unpack_wds_SLQSSGetLoopback_t Struct Reference

Data Fields

- uint8_t ByteLoopbackMode
- uint8_t ByteLoopbackMultiplier
- swi_uint256_t ParamPresenceMask

8.1199.1 Detailed Description

This structure contains unpack Get loopback information.

Parameters

ByteLoopback-Mode	• Loopback Mode. – 0 - Disable – 1 - Enable • Bit to check in ParamPresenceMask - **16**
ByteLoopback-Multiplier	• Loopback multiplier. Number of downlink bytes to send for each uplink byte. • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1199.2 Field Documentation

8.1199.2.1 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMode

8.1199.2.2 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMultiplier

8.1199.2.3 swi_uint256_t unpack_wds_SLQSSGetLoopback_t::ParamPresenceMask

8.1200 unpack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint32_t * psid
- uint32_t * pFailureReason
- uint32_t * pVerboseFailReasonType
- uint32_t * pVerboseFailureReason
- swi_uint256_t ParamPresenceMask

8.1200.1 Detailed Description

This structure contains unpack Start Data Session Information.

Parameters

psid	• Assigned session ID when starting a data session • Bit to check in ParamPresenceMask - **1**
pFailureReason	• Reason data session failed to be established • See qaGobiApiTableCallEndReasons.h for Call End Reason • Bit to check in ParamPresenceMask - **16**
pVerboseFailReasonType	• Parameter describing type of verbose failure reason • See qaGobiApiTableCallEndReasons.h for Call End Reason Type • Bit to check in ParamPresenceMask - **17**
pVerboseFailureReason	• Verbose reason explaining why call failed. Depends on verbFailReasonType parameter • See qaGobiApiTableCallEndReasons.h for Call End Reason • Bit to check in ParamPresenceMask - **17**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.1200.2 Field Documentation

8.1200.2.1 swi_uint256_t unpack_wds_SLQSStartDataSession_t::ParamPresenceMask

8.1200.2.2 uint32_t* unpack_wds_SLQSStartDataSession_t::pFailureReason

8.1200.2.3 uint32_t* unpack_wds_SLQSStartDataSession_t::psid

8.1200.2.4 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailReasonType

8.1200.2.5 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailureReason

8.1201 unpack_wds_SLQSSwiProfileChangeCallback_Ind_t Struct Reference

Data Fields

- wds_profileChange ProfileTlv
- wds_sourceOfChange srcTlv
- swi_uint256_t ParamPresenceMask

8.1201.1 Detailed Description

Structure for SwiProfileChange unpack.

Parameters

ProfileTlv	• See wds_profileChange for more information. • Bit to check in ParamPresenceMask - **1**
srcTlv	• See wds_sourceOfChange for more information. • Bit to check in ParamPresenceMask - **16**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1201.2 Field Documentation

8.1201.2.1 swi_uint256_t unpack_wds_SLQSSwiProfileChangeCallback_Ind_t::ParamPresenceMask

8.1201.2.2 wds_profileChange unpack_wds_SLQSSwiProfileChangeCallback_Ind_t::ProfileTlv

8.1201.2.3 wds_sourceOfChange unpack_wds_SLQSSwiProfileChangeCallback_Ind_t::srcTlv

8.1202 unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference

Data Fields

- uint8_t contextId
- uint8_t bearerId
- int8_t apnName [100]
- uint32_t ipv4Address
- uint32_t ipv4GWAddress
- uint32_t prDNSIPv4Address
- uint32_t seDNSIPv4Address
- ipv6AddressInfo ipv6Address
- ipv6AddressInfo ipv6GWAddress

- uint16_t prDNSIPv6Address [8]
- uint16_t seDNSIPv6Address [8]
- uint32_t prPCSCFIPv4Address
- uint32_t sePCSCFIPv4Address
- uint16_t prPCSCFIPv6Address [8]
- uint16_t sePCSCFIPv6Address [8]
- swi_uint256_t ParamPresenceMask

8.1202.1 Detailed Description

This structure contains the unpack parameters retrieved by the API unpack_wds_SLQSWdsSwiPDPRuntime-Settings

Parameters

contextId	(optional) • Context Identifier • Bit to check in ParamPresenceMask - **16**
bearerId	(optional) • Bearer Identity • An EPS bearer identity uniquely identifies an EPS bearer for one UE accessing via E-UTRAN. The EPS Bearer Identity is allocated by the MME. • Bit to check in ParamPresenceMask - **17**
apnName	(optional) • APN name associated with the context id • Bit to check in ParamPresenceMask - **18**
ipv4Address	(optional) • IPv4 Address • Bit to check in ParamPresenceMask - **19**
ipv4GWAddress	(optional) • IPv4 Gateway Address • Bit to check in ParamPresenceMask - **20**
prDNSIPv4-Address	(optional) • Primary DNS IPv4 Address • Bit to check in ParamPresenceMask - **21**
seDNSIPv4-Address	(optional) • Secondary DNS IPv4 Address • Bit to check in ParamPresenceMask - **22**
ipv6Address	(optional) • IPv6 Address • Bit to check in ParamPresenceMask - **23**
ipv6GWAddress	(optional) • IPv6 Gateway Address • See ipv6AddressInfo for more information • Bit to check in ParamPresenceMask - **24**

prDNSIPv6-Address	(optional) • Primary IPv6 DNS Address(in network byte order) • This is an 8-element array of 16-bit numbers, each of which is in big-endian format • Bit to check in ParamPresenceMask - **25**
seDNSIPv6-Address	(optional) • Secondary IPv6 DNS Address(in network byte order) • This is an 8-element array of 16-bit numbers, each of which is in big-endian format • Bit to check in ParamPresenceMask - **26**
prPCSCFIPv4-Address	(optional) • Primary PCSCF IPv4 Address • Bit to check in ParamPresenceMask - **27**
sePCSCFIPv4-Address	(optional) • Secondary PCSCF IPv4 Address • Bit to check in ParamPresenceMask - **28**
prPCSCFIPv6-Address	(optional) • Primary PCSCF IPv6 Address • This is an 8-element array of 16-bit numbers, each of which is in big-endian format • Bit to check in ParamPresenceMask - **29**
sePCSCFIPv6-Address	(optional) • Secondary PCSCF IPv6 Address • This is an 8-element array of 16-bit numbers, each of which is in big-endian format • Bit to check in ParamPresenceMask - **30**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.1202.2 Field Documentation

8.1202.2.1 int8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::apnName[100]

8.1202.2.2 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::bearerId

8.1202.2.3 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.1202.2.4 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4Address

8.1202.2.5 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4GWAddress

8.1202.2.6 ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6Address

8.1202.2.7 ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6GWAddress

8.1202.2.8 swi_uint256_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ParamPresenceMask

8.1202.2.9 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv4Address

8.1202.2.10 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv6Address[8]

8.1202.2.11 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv4Address

8.1202.2.12 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv6Address[8]

8.1202.2.13 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv4Address

8.1202.2.14 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv6Address[8]

8.1202.2.15 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv4Address

8.1202.2.16 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv6Address[8]

8.1203 UnPackGetProfileSettingOut Struct Reference

Data Fields

- UnpackQmiProfileInfo curProfile
- uint16_t * pExtErrCode

8.1203.1 Detailed Description

This structure contains the profile settings output

Parameters

curProfile	• Structure containing details of the current profile • See UnpackQmiProfileInfo
pExtErrCode	• pointer to a 2 byte extended error code • Error code will only will be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERNAL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description. • Bit to check in ParamPresenceMask - **224**

8.1203.2 Field Documentation

8.1203.2.1 UnpackQmiProfileInfo UnPackGetProfileSettingOut::curProfile

8.1203.2.2 uint16_t* UnPackGetProfileSettingOut::pExtErrCode

8.1204 UnPackGetProfileSettingOutV2 Struct Reference

Data Fields

- UnpackQmiProfileInfoV2 curProfile
- uint16_t * pExtErrCode

8.1204.1 Detailed Description

This structure contains the profile settings output

Parameters

curProfile	• Structure containing details of the current profile • See UnpackQmiProfileInfoV2
pExtErrCode	• pointer to a 2 byte extended error code • Error code will only will be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERNAL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description. • Bit to check in ParamPresenceMask - **224**

8.1204.2 Field Documentation

8.1204.2.1 UnpackQmiProfileInfoV2 UnPackGetProfileSettingOutV2::curProfile

8.1204.2.2 uint16_t* UnPackGetProfileSettingOutV2::pExtErrCode

8.1205 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo Struct Reference

Data Fields

- uint8_t bBinaryType
- uint8_t bState
- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint32_t ulPkgDownloadSize
- uint32_t ulPkgDownloadComplete
- uint16_t wUpdateCompeteStatus
- uint8_t bSerity
- uint16_t wVersionLength
- uint8_t szVersion [128]
- uint16_t wNameLength
- uint8_t szName [128]
- uint16_t wDescriptionLength
- uint8_t szDescription [1024]
- uint8_t TlvPresent

8.1205.1 Detailed Description

This structure contains unpack event report binary update session information parameters.

Parameters

bBinaryType	: Type • 1 - Firmware. • 2 - User App. • 3 - Legato Framework.
bStat	: State • 0x01 - No binary update available. • 0x02 - Query binary Download. • 0x03 - Binary Downloading. • 0x04 - Binary downloaded. • 0x05 -Query Binary Update. • 0x06 - Binary updating. • 0x07 - Binary updated.
bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
wUserInput-Timeout	Timeout for user input in minutes. A value of 0 means no time-out.
ulPkgDownload-Size	The size (in bytes) of the update package
ulPkgDownload-Complete	The number of bytes being downloaded. For downloading state, this value shall be $>$ 0 and incremented toward the pkg_dload_size. For other states, the value shall be 0 as it is meaningless.
wUpdate-CompeteStatus	Result code.This field should be looked at only when the AVMS session is complete.
bSerity	Serity. • 0x01 - Mandatory. • 0x02 - Optional.
wVersionLength	Length of FW Version string in bytes.
szVersion	FW Version string in ASCII.
wNameLength	Length Package Name string in bytes.
szName	Package Name in UCS2.
wDescription-Length	Length of description in bytes.
szDescription	Description of Update Package in USC2.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1205.2 Field Documentation

8.1205.2.1 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bBinaryType

8.1205.2.2 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bSerity

8.1205.2.3 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bState

8.1205.2.4 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bUserInputRequest

8.1205.2.5 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szDescription[1024]

8.1205.2.6 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szName[128]

8.1205.2.7 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szVersion[128]

8.1205.2.8 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::TlvPresent

8.1205.2.9 uint32_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::ulPkgDownloadComplete

8.1205.2.10 uint32_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::ulPkgDownloadSize

8.1205.2.11 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wDescriptionLength

8.1205.2.12 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wNameLength

8.1205.2.13 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wUpdateCompeteStatus

8.1205.2.14 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wUserInputTimeout

8.1205.2.15 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wVersionLength

8.1206 UnpackSwiAvmsEventReportConfig Struct Reference

Data Fields

- uint8_t bState
- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint16_t wAlertMsgLength
- uint8_t szAlertMsg [200]
- uint8_t TlvPresent

8.1206.1 Detailed Description

This structure contains unpack event report configure parameters.

Parameters

bState	State. • 0x01 - AVMS Read Request. • 0x02 - AVMS Change Request. • 0x03 - AVMS Config Complete.
bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
wUserInput-Timeout	Timeout for user input in seconds. A value of 0 means no time-out
wAlertMsg-Length	Length of Alert message string in bytes.
szAlertMsg	Alert message in UCS2.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1206.2 Field Documentation

8.1206.2.1 uint8_t UnpackSwiAvmsEventReportConfig::bState

8.1206.2.2 uint8_t UnpackSwiAvmsEventReportConfig::bUserInputRequest

8.1206.2.3 uint8_t UnpackSwiAvmsEventReportConfig::szAlertMsg[200]

8.1206.2.4 uint8_t UnpackSwiAvmsEventReportConfig::TlvPresent

8.1206.2.5 uint16_t UnpackSwiAvmsEventReportConfig::wAlertMsgLength

8.1206.2.6 uint16_t UnpackSwiAvmsEventReportConfig::wUserInputTimeout

8.1207 UnpackSwiAvmsEventReportConnectionRequest Struct Reference

Data Fields

- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint8_t TlvPresent

8.1207.1 Detailed Description

This structure contains unpack event report connection request parameters.

Parameters

bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
bUserInput-Request	Timeout for user input in minutes. A value of 0 means no time-out.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1207.2 Field Documentation

8.1207.2.1 uint8_t UnpackSwiAvmsEventReportConnectionRequest::bUserInputRequest

8.1207.2.2 uint8_t UnpackSwiAvmsEventReportConnectionRequest::TlvPresent

8.1207.2.3 uint16_t UnpackSwiAvmsEventReportConnectionRequest::wUserInputTimeout

8.1208 UnpackSwiAvmsEventReportDataSessionStatus Struct Reference

Data Fields

- uint8_t bType
- uint16_t wErrorCode
- uint8_t TlvPresent

8.1208.1 Detailed Description

This structure contains unpack evnet report data session status parameters.

Parameters

bType	Notification type. • 0: Data session closed. • 1: Data session activated. • 2: Register made. -3: Data session error.
wErrorCode	LWM2M Session error code. • 0x0000: none
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1208.2 Field Documentation

8.1208.2.1 uint8_t UnpackSwiAvmsEventReportDataSessionStatus::bType

8.1208.2.2 uint8_t UnpackSwiAvmsEventReportDataSessionStatus::TlvPresent

8.1208.2.3 uint16_t UnpackSwiAvmsEventReportDataSessionStatus::wErrorCode

8.1209 UnpackSwiAvmsEventReportHTTPStatus Struct Reference

Data Fields

- uint16_t wHTTPStatus
- uint8_t TlvPresent

8.1209.1 Detailed Description

This structure contains unpack event report HTTP status parameters.

Parameters

wHTTPStatus	See RFC 7231. • 100 - Continue • 101 - Switching Protocols • 200 - OK • 201 - Created • 202 - Accepted • 203 - Non-Authoritative Information • 204 - No Content • 205 - Reset Content • 206 - Partial Content • 300 - Multiple Choices • 301 - Moved Permanently • 302 - Found • 303 - See Other • 304 - Not Modified • 305 - Use Proxy • 307 - Temporary Redirect • 400 - Bad Request • 401 - Unauthorized • 402 - Payment Required • 403 - Forbidden • 404 - Not Found • 405 - Method Not Allowed • 406 - Not Acceptable • 407 - Proxy Authentication Required • 408 - Request Timeout • 409 - Conflict • 410 - Gone • 411 - Length Required • 412 - Precondition Failed • 413 - Payload Too Large • 414 - URI Too Long • 415 - Unsupported Media Type • 416 - Range Not Satisfiable • 417 - Expectation Failed • 426 - Upgrade Required • 500 - Internal Server Error • 501 - Not Implemented • 502 - Bad Gateway • 503 - Service Unavailable • 504 - Gateway Timeout • 505 - HTTP Version Not Supported
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1209.2 Field Documentation

8.1209.2.1 uint8_t UnpackSwiAvmsEventReportHTTPStatus::TlvPresent

8.1209.2.2 uint16_t UnpackSwiAvmsEventReportHTTPStatus::wHTTPStatus

8.1210 UnpackSwiAvmsEventReportNotification Struct Reference

Data Fields

- uint8_t bNotification
- uint16_t wSessionStatus
- uint8_t TlvPresent

8.1210.1 Detailed Description

This structure contains unpack event report notification parameters.

Parameters

bNotification	Notification. • 0x14 - Module starts sending data to server. • 0x15 - Authentication with the server. • 0x16 - session with the server is ended.
wSessionStatus	This field will set to the session status for notifications that occur at the end of a session, zero for all other notifications. • 0x0000: Successful: Session succeeded • 0x0001: Break: Session succeeded • 0x0002: Large Object Handled: Session succeeded • 0x0003: No status: Session succeeded • 0x0004: No more commands: Session succeeded • 0x0005: User cancel: Session cancelled • 0x0100-0x01FF: General errors • 0x0200-0x02FF: Syncml errors • 0x0300-0x03FF: Authentication errors • 0x0400-0x04FF: Protocol errors • 0x0500-0x05FF: Tree errors (DM Only) • 0x0600-0x06FF: Not applicable • 0x0700-0x07FF: Trigger errors • 0x0800-0x08FF: FUMO errors • 0x0900-0x09FF: Communication errors • 0x0A00-0x0AFF: Parsing errors • 0x0B00-0x0CFF: Not applicable • 0x7F00-0x7F12:Insignia errors • 0x7F13: Illegal text: Text received contains illegal characters • 0x7F14: Download failure: Failed to download FOTA image • 0x7F15: Empty session: Session ran successfully, but no information wasa updated • 0x7F16: Factory reset successful: Factory reset succeeded • 0x7F17: Factory reset fail: Factory reset failed
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1210.2 Field Documentation

8.1210.2.1 uint8_t UnpackSwiAvmsEventReportNotification::bNotification

8.1210.2.2 uint8_t UnpackSwiAvmsEventReportNotification::TlvPresent

8.1210.2.3 uint16_t UnpackSwiAvmsEventReportNotification::wSessionStatus

8.1211 UnpackSwiAvmsEventReportPackageID Struct Reference

Data Fields

- uint8_t bPackageID
- uint8_t TlvPresent

8.1211.1 Detailed Description

This structure contains unpack event report package ID parameters.

Parameters

bPackageID	Package ID of the application binary that this AVMS_EVENT_ID notification is for.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1211.2 Field Documentation

8.1211.2.1 uint8_t UnpackSwiAvmsEventReportPackageID::bPackageID

8.1211.2.2 uint8_t UnpackSwiAvmsEventReportPackageID::TlvPresent

8.1212 UnpackSwiAvmsEventReportRegStatus Struct Reference

Data Fields

- uint8_t bRegStatus
- uint8_t TlvPresent

8.1212.1 Detailed Description

This structure contains unpack event report registration status parameters.

Parameters

bRegStatus	LWM2M Registration status. • 0: Need Bootstrap. • 1: Bootstrap made. • 2: Register made. -3: Update made.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1212.2 Field Documentation

8.1212.2.1 uint8_t UnpackSwiAvmsEventReportRegStatus::bRegStatus

8.1212.2.2 uint8_t UnpackSwiAvmsEventReportRegStatus::TlvPresent

8.1213 UnpackSwiAvmsEventReportSessionType Struct Reference

Data Fields

- uint8_t bType
- uint8_t TlvPresent

8.1213.1 Detailed Description

This structure contains unpack event report session type parameters.

Parameters

bType	Session Type. • 0: Bootstrap session. • 1. DM session
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1213.2 Field Documentation

8.1213.2.1 uint8_t UnpackSwiAvmsEventReportSessionType::bType

8.1213.2.2 uint8_t UnpackSwiAvmsEventReportSessionType::TlvPresent

8.1214 UnpackSwiAvmsEventReportWAMSParamChange Struct Reference

Data Fields

- uint16_t wWamsChangeMask
- uint8_t TlvPresent

8.1214.1 Detailed Description

This structure contains unpack event report WAMS parameter change parameters.

Parameters

wWamsChange-Mask	Mask of WAMS parameters changed. • By default set to 0xFF for all changes. • 0x01 - device_login • 0x02 - device_MD5_key • 0x04 - server_login • 0x08 - server_MD5_key • 0x10 - server_URL • 0x20 - Nonce • 0x40 - Application key
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.1214.2 Field Documentation

8.1214.2.1 uint8_t UnpackSwiAvmsEventReportWAMSParamChange::TlvPresent

8.1214.2.2 uint16_t UnpackSwiAvmsEventReportWAMSParamChange::wWamsChangeMask

8.1215 unpackWdsProfileParam Union Reference

Data Fields

- LibpackProfile3GPP SlqsProfile3GPP
- LibpackProfile3GPP2 SlqsProfile3GPP2

8.1215.1 Detailed Description

This union WdsProfileParam consists of Profile3GPP and Profile3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile • See LibpackProfile3GPP
SlqsProfile3GP-P2	3GPP2 profile • See LibpackProfile3GPP2

8.1215.2 Field Documentation

8.1215.2.1 LibpackProfile3GPP unpackWdsProfileParam::SlqsProfile3GPP

8.1215.2.2 LibpackProfile3GPP2 unpackWdsProfileParam::SlqsProfile3GPP2

8.1216 unpackWdsProfileParamV2 Union Reference

Data Fields

- LibpackProfile3GPPV2 SlqsProfile3GPP
- LibpackProfile3GPP2 SlqsProfile3GPP2

8.1216.1 Detailed Description

This union WdsProfileParam consists of Profile3GPP and Profile3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile • See LibpackProfile3GPP
SlqsProfile3GP-P2	3GPP2 profile • See LibpackProfile3GPP2

8.1216.2 Field Documentation

8.1216.2.1 LibpackProfile3GPPV2 unpackWdsProfileParamV2::SlqsProfile3GPP

8.1216.2.2 LibpackProfile3GPP2 unpackWdsProfileParamV2::SlqsProfile3GPP2

8.1217 voice_airTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t airTimerValue

8.1217.1 Detailed Description

This structure contains information about the Air Timer.

Parameters

namID	• Index of the NAM(Number Assignment Module) to be configured. • Range 0 to 3. • Some modems support only 1 or 2 NAMs. • 0xFF,if not available.
airTimerValue	• Time in minutes. • Cumulative air time is slammed. • 0xFFFFFFFF,if not available.

8.1217.2 Field Documentation

8.1217.2.1 uint32_t voice_airTimer::airTimerValue

8.1217.2.2 uint8_t voice_airTimer::namID

8.1218 voice_allCallsAlphaIDInfo Struct Reference

Data Fields

- uint8_t callID
- voice_alphaIDInfo AlphaIDInfo

8.1218.1 Detailed Description

This structure contains information for Alpha Identifier for All Calls

Parameters

callID	• Unique call identifier for the call.

AlphaIDInfo	• See voice_alphaIDInfo for more information.

8.1218.2 Field Documentation

8.1218.2.1 voice_alphaIDInfo voice_allCallsAlphaIDInfo::AlphaIDInfo

8.1218.2.2 uint8_t voice_allCallsAlphaIDInfo::callID

8.1219 voice_allCallsDiagInfo Struct Reference

Data Fields

- uint8_t callID
- voice_diagInfo DiagInfo

8.1219.1 Detailed Description

This structure contains Diagnostic Information for All Calls

Parameters

callID	• Unique call identifier for the call.
DiagInfo	• See voice_diagInfo for more information.

8.1219.2 Field Documentation

8.1219.2.1 uint8_t voice_allCallsDiagInfo::callID

8.1219.2.2 voice_diagInfo voice_allCallsDiagInfo::DiagInfo

8.1220 voice_allCallsUUSInfo Struct Reference

Data Fields

- uint8_t callID
- voice_UUSInfo uusInfo

8.1220.1 Detailed Description

This structure contains information for User to User Signaling Service for All Calls.

Parameters

callID	• Unique call identifier for the call.

uusInfo	• See voice_UUSInfo for more information.

8.1220.2 Field Documentation

8.1220.2.1 uint8_t voice_allCallsUUSInfo::callID

8.1220.2.2 voice_UUSInfo voice_allCallsUUSInfo::uusInfo

8.1221 voice_alphaIDInfo Struct Reference

Data Fields

- uint8_t alphaDcs
- uint8_t alphaLen
- uint8_t alphaText [255]

8.1221.1 Detailed Description

This structure contains information about the Alpha Identifier.

Parameters

alphaDcs	• Alpha coding scheme – 0x01 - GSM Default_Char – 0x02 - UCS2 – 0xFF - Not Available
alphaLen	• Number of sets of the following elements: – pAlpha_text • If zero(0) then no further information exists.
alphaText[MAX_VOICE_DESCRIPTION_LENGTH]	• Data encoded as per the alpha_dcs

8.1221.2 Field Documentation

8.1221.2.1 uint8_t voice_alphaIDInfo::alphaDcs

8.1221.2.2 uint8_t voice_alphaIDInfo::alphaLen

8.1221.2.3 uint8_t voice_alphaIDInfo::alphaText[255]

8.1222 voice_arrAlertingPattern Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint32_t alertingPattern [20]

8.1222.1 Detailed Description

This structure contains an array of Alerting Pattern.

Parameters

numInstances	• Number of callID, alertingPattern that follow. • If zero(0) then no further information exists.
callID[VOICE_MAX_NO_OF_CALLS]	• Array of Unique call identifier for the call.
alertingPattern[VOICE_MAX_NO_OF_CALLS]	• Array of Alerting pattern. – 0x00 - QMI_VOICE_ALERTING_PATTERN_1 - Pattern 1 – 0x01 - QMI_VOICE_ALERTING_PATTERN_2 - Pattern 2 – 0x02 - QMI_VOICE_ALERTING_PATTERN_3 - Pattern 3 – 0x04 - QMI_VOICE_ALERTING_PATTERN_5 - Pattern 5 – 0x05 - QMI_VOICE_ALERTING_PATTERN_6 - Pattern 6 – 0x06 - QMI_VOICE_ALERTING_PATTERN_7 - Pattern 7 – 0x07 - QMI_VOICE_ALERTING_PATTERN_8 - Pattern 8 – 0x08 - QMI_VOICE_ALERTING_PATTERN_9 - Pattern 9

8.1222.2 Field Documentation

8.1222.2.1 uint32_t voice_arrAlertingPattern::alertingPattern[20]

8.1222.2.2 uint8_t voice_arrAlertingPattern::callID[20]

8.1222.2.3 uint8_t voice_arrAlertingPattern::numInstances

8.1223 voice_arrAlertingType Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint8_t AlertingType [20]

8.1223.1 Detailed Description

This structure contains an array of Alerting Type.

Parameters

numInstances	• Number of callID, AlertingType that follow. • If zero(0) then no further information exists.
callID[VOICE_MAX_NO_OF_CALLS]	• Array of Unique call identifier for the call.
AlertingType[VOICE_MAX_NO_OF_CALLS]	• Array of Alerting type. – 0x00 - ALERTING_LOCAL - Local – 0x01 - ALERTING_REMOTE - Remote

8.1223.2 Field Documentation

8.1223.2.1 uint8_t voice_arrAlertingType::AlertingType[20]

8.1223.2.2 uint8_t voice_arrAlertingType::callID[20]

8.1223.2.3 uint8_t voice_arrAlertingType::numInstances

8.1224 voice_arrAlphaID Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsAlphaIDInfo allCallsAlphaIDInfoArr [20]

8.1224.1 Detailed Description

This structure contains an array of Alpha ID Info

Parameters

numInstances	• Number of allCallsAlphaIDInfo that follow. • If zero(0) then no further information exists.
allCallsAlphaIDInfo[VOICE_MAX_NO_OF_CALLS]	• Array of allCallsAlphaIDInfo. • See voice_allCallsAlphaIDInfo for more information.

8.1224.2 Field Documentation

8.1224.2.1 voice_allCallsAlphaIDInfo voice_arrAlphaID::allCallsAlphaIDInfoArr[20]

8.1224.2.2 uint8_t voice_arrAlphaID::numInstances

8.1225 voice_arrCalledPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo CalledPartyNum [20]

8.1225.1 Detailed Description

This structure contains an array of Called Party Numbers consisting of information of all the numbers which have been called from the device.

Parameters

numInstances	• Number of calledPartyNum that follow. • If zero(0) then no further information exists.
CalledPartyNum[VOICE_MAX_NO_OF_CALLS]	• Array of CalledPartyNum. • See voice_peerNumberInfo for more information.

8.1225.2 Field Documentation

8.1225.2.1 voice_peerNumberInfo voice_arrCalledPartyNum::CalledPartyNum[20]

8.1225.2.2 uint8_t voice_arrCalledPartyNum::numInstances

8.1226 voice_arrCallEndReason Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t callEndReason [20]

8.1226.1 Detailed Description

This structure contains an array of Call End Reasons.

Parameters

numInstances	• Number of callID, callEndReason that follow. • If zero(0) then no further information exists.
callID[VOICE_MAX_NO_OF_CALLS]	• Array of Unique call identifier for the call.
callEndReason[VOICE_MAX_NO_OF_CALLS]	• Array of Call End Reason . • See Table9 qaGobiApiTableVoiceCallEndReasons.h for a list of valid voice-related call end reasons

8.1226.2 Field Documentation

8.1226.2.1 uint16_t voice_arrCallEndReason::callEndReason[20]

8.1226.2.2 uint8_t voice_arrCallEndReason::callID[20]

8.1226.2.3 uint8_t voice_arrCallEndReason::numInstances

8.1227 voice_arrCallInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallInformation getAllCallInfo [20]

8.1227.1 Detailed Description

This structure contains an array of Call Info

Parameters

numInstances	• Number of getAllCallInfo that follow. • If zero(0) then no further information exists.
getAllCallInfo[VOICE_MAX_NO_OF_CALLS]	• Array of CallInfo. • See voice_getAllCallInformation for more information.

8.1227.2 Field Documentation

8.1227.2.1 voice_getAllCallInformation voice_arrCallInfo::getAllCallInfo[20]

8.1227.2.2 uint8_t voice_arrCallInfo::numInstances

8.1228 voice_arrConnectPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo ConnectedPartyNum [20]

8.1228.1 Detailed Description

This structure contains an array of Connected Party Numbers consisting of information regarding all the devices connected.

Parameters

numInstances	• Number of ConnectedPartyNum that follow. • If zero(0) then no further information exists.

ConnectedPartyNum[VOICE_MAX_NO_OF_CALLS]	• Array of ConnectedPartyNum. • See voice_peerNumberInfo for more information.

8.1228.2 Field Documentation

8.1228.2.1 voice_peerNumberInfo voice_arrConnectPartyNum::ConnectedPartyNum[20]

8.1228.2.2 uint8_t voice_arrConnectPartyNum::numInstances

8.1229 voice_arrDiagInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsDiagInfo DiagInfo [20]

8.1229.1 Detailed Description

This structure contains an array of Diagnostic Information.

Parameters

numInstances	• Number of DiagInfo that follow. • If zero(0) then no further information exists.
DiagInfo[VOICE_MAX_NO_OF_CALLS]	• Array of DiagInfo. • See voice_allCallsDiagInfo for more information.

8.1229.2 Field Documentation

8.1229.2.1 voice_allCallsDiagInfo voice_arrDiagInfo::DiagInfo[20]

8.1229.2.2 uint8_t voice_arrDiagInfo::numInstances

8.1230 voice_arrRedirPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo RedirPartyNum [20]

8.1230.1 Detailed Description

This structure contains an array of Redirecting Party Numbers consisting of information of all the numbers which have been redirected from the device.

Parameters

numInstances	• Number of redirPartyNum that follow. • If zero(0) then no further information exists.
RedirPartyNum[-VOICE_MAX_NO_OF_CALLS]	• Array of RedirPartyNum. • See voice_peerNumberInfo for more information.

8.1230.2 Field Documentation

8.1230.2.1 uint8_t voice_arrRedirPartyNum::numInstances

8.1230.2.2 voice_peerNumberInfo voice_arrRedirPartyNum::RedirPartyNum[20]

8.1231 voice_arrRemotePartyName Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyName GetAllCallRmtPtyName [20]

8.1231.1 Detailed Description

This structure contains an array of Remote Party Names

Parameters

numInstances	• Number of remotePartyName that follow. • If zero(0) then no further information exists.
GetAllCallRmtPtyName[MAX_NO_OF_CALLS]	• Array of remotePartyName. • See voice_getAllCallRmtPtyName for more information.

8.1231.2 Field Documentation

8.1231.2.1 voice_getAllCallRmtPtyName voice_arrRemotePartyName::GetAllCallRmtPtyName[20]

8.1231.2.2 uint8_t voice_arrRemotePartyName::numInstances

8.1232 voice_arrRemotePartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyNum RmtPtyNum [20]

8.1232.1 Detailed Description

This structure contains an array of Remote Party Numbers

Parameters

numInstances	• Number of remotePartyNum that follow. • If zero(0) then no further information exists.
RmtPtyNum[VOICE_MAX_NO_OF_CALLS]	• Array of remotePartyNum. • See voice_getAllCallRmtPtyNum for more information.

8.1232.2 Field Documentation

8.1232.2.1 uint8_t voice_arrRemotePartyNum::numInstances

8.1232.2.2 voice_getAllCallRmtPtyNum voice_arrRemotePartyNum::RmtPtyNum[20]

8.1233 voice_arrSvcOption Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t srvOption [20]

8.1233.1 Detailed Description

This structure contains array an of Servicing option.

Parameters

numInstances	• Number of callID, srvOption that follow. • If zero(0) then no further information exists.
callID[VOICE_MAX_NO_OF_CALLS]	• Array of Unique call identifier for the call.
srvOption[VOICE_MAX_NO_OF_CALLS]	• Array of Service option. • See Table9 qaGobiApiTableServiceOptions.h for standard service option number assignments.

8.1233.2 Field Documentation

8.1233.2.1 uint8_t voice_arrSvcOption::callID[20]

8.1233.2.2 uint8_t voice_arrSvcOption::numInstances

8.1233.2.3 uint16_t voice_arrSvcOption::srvOption[20]

8.1234 voice_arrUUSInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsUUSInfo AllCallsUUSInfo [20]

8.1234.1 Detailed Description

This structure contains an array of User to User Signaling Service Information

Parameters

numInstances	• Number of allCallsUUSInfo that follow. • If zero(0) then no further information exists.
AllCallsUUSInfo[VOICE_MAX_NO_OF_CALLS]	• Array of allCallsUUSInfo. • See voice_allCallsUUSInfo for more information.

8.1234.2 Field Documentation

8.1234.2.1 voice_allCallsUUSInfo voice_arrUUSInfo::AllCallsUUSInfo[20]

8.1234.2.2 uint8_t voice_arrUUSInfo::numInstances

8.1235 voice_burstDTMFInfo Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t digitCnt
- uint8_t pDigitBuff [255]

8.1235.1 Detailed Description

This structure contains Voice Burst DTMF Information

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. A burst DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device • NULL pointer - Invalid data.

digitCnt	• Length of DTMF digit buffer which follows
pDigitBuff[MAX_VOICE_DESCRIPTION_LENGTH]	• DTMF digit buffer in ASCII, NULL terminated

8.1235.2 Field Documentation

8.1235.2.1 uint8_t voice_burstDTMFInfo::digitCnt

8.1235.2.2 uint8_t* voice_burstDTMFInfo::pCallID

8.1235.2.3 uint8_t voice_burstDTMFInfo::pDigitBuff[255]

8.1236 voice_calledPartyInfo Struct Reference

Data Fields

- uint8_t PI
- uint8_t SI
- uint8_t numType
- uint8_t numPlan
- uint8_t numLen
- uint8_t number [255]

8.1236.1 Detailed Description

This structure contains Called party Number Information

Parameters

PI	• Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
SI	• Number of sets of following elements – Caller Id
SI	• Number screening indicator. • Values: – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network

numType	• Number type. • Values: – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
numPlan	• Number plan. • Values: – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
numLen	• Provides the length of number which follow.
number[255]	• number of numLen length, NULL terminated.

8.1236.2 Field Documentation

8.1236.2.1 uint8_t voice_calledPartyInfo::number[255]

8.1236.2.2 uint8_t voice_calledPartyInfo::numLen

8.1236.2.3 uint8_t voice_calledPartyInfo::numPlan

8.1236.2.4 uint8_t voice_calledPartyInfo::numType

8.1236.2.5 uint8_t voice_calledPartyInfo::PI

8.1236.2.6 uint8_t voice_calledPartyInfo::SI

8.1237 voice_calledPartySubAdd Struct Reference

Data Fields

- uint8_t extBit

- uint8_t subAddrType
- uint8_t oddEvenInd
- uint8_t subAddrLen
- uint8_t subAddr [255]

8.1237.1 Detailed Description

This structure contains information about the Called Sub Party Addresses.

Parameters

extBit	• Extension bit.
subAddrType	• Subaddress type. – 0x00 - NSAP – 0x01 - USER
oddEvenInd	• Even/odd indicator. – 0x00 - Even number of address signals – 0x01 - Odd number of address signals
subAddrLen	• Number of sets of the following elements: – SubAddress
subAddr[MAX_VOICE_DESCRIPTION_LENGTH]	• Array of the SubAddress in BCD number format.

8.1237.2 Field Documentation

8.1237.2.1 uint8_t voice_calledPartySubAdd::extBit

8.1237.2.2 uint8_t voice_calledPartySubAdd::oddEvenInd

8.1237.2.3 uint8_t voice_calledPartySubAdd::subAddr[255]

8.1237.2.4 uint8_t voice_calledPartySubAdd::subAddrLen

8.1237.2.5 uint8_t voice_calledPartySubAdd::subAddrType

8.1238 voice_callerIDInfo Struct Reference

Data Fields

- uint8_t PI
- uint8_t callerIDLen
- uint8_t callerID [255]

8.1238.1 Detailed Description

This structure contains Caller ID Information

Parameters

PI	• Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
callerIDLen	• Number of sets of following elements – Caller Id
pCallerID	• Caller ID in ASCII string.

8.1238.2 Field Documentation

8.1238.2.1 uint8_t voice_callerIDInfo::callerID[255]

8.1238.2.2 uint8_t voice_callerIDInfo::callerIDLen

8.1238.2.3 uint8_t voice_callerIDInfo::PI

8.1239 voice_callFwdTypeAndPlan Struct Reference

Data Fields

- uint8_t numberType
- uint8_t numberPlan

8.1239.1 Detailed Description

This structure contains Supplementary Service request parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

numberType	• Call forwarding number type – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION Reserved extension

numberPlan	• Call forwarding number plan – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION Reserved extension

8.1239.2 Field Documentation

8.1239.2.1 uint8_t voice_callFwdTypeAndPlan::numberPlan

8.1239.2.2 uint8_t voice_callFwdTypeAndPlan::numberType

8.1240 voice_callFWExtInfo Struct Reference

Data Fields

- uint8_t SvcStatus
- uint8_t SvcClass
- uint8_t noReplyTimer
- uint8_t PI
- uint8_t SI
- uint8_t numType
- uint8_t numPlan
- uint8_t numLen
- uint8_t number [255]

8.1240.1 Detailed Description

This structure contains information for Get Call Forwarding Extended Information.

Parameters

SvcStatus	• Service status. Values: – 0x00 - SERVICE_STATUS_INACTIVE - Inactive – 0x01 - SERVICE_STATUS_ACTIVE - Active
SvcClass	• Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.
noReplyTimer	• No reply timer value in seconds • A value of 0 indicates that no_reply_timer is ignored.

PI	• Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
SI	• Number screening indicator. • Values: – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
numType	• Number type. • Values: – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
numPlan	• Number plan. • Values: – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
numLen	• Provides the length of number which follow.
number[255]	• number of numLen length, NULL terminated.

8.1240.2 Field Documentation

8.1240.2.1 uint8_t voice_callFWExtInfo::noReplyTimer

8.1240.2.2 uint8_t voice_callFWExtInfo::number[255]

8.1240.2.3 uint8_t voice_callFWExtInfo::numLen

8.1240.2.4 uint8_t voice_callFWExtInfo::numPlan

8.1240.2.5 uint8_t voice_callFWExtInfo::numType

8.1240.2.6 uint8_t voice_callFWExtInfo::PI

8.1240.2.7 uint8_t voice_callFWExtInfo::SI

8.1240.2.8 uint8_t voice_callFWExtInfo::SvcClass

8.1240.2.9 uint8_t voice_callFWExtInfo::SvcStatus

8.1241 voice_callFWInfo Struct Reference

Data Fields

- uint8_t SvcStatus
- uint8_t SvcClass
- uint8_t numLen
- uint8_t number [255]
- uint8_t noReplyTimer

8.1241.1 Detailed Description

This structure contains information for Get Call Forwarding Information.

Parameters

SvcStatus	• Service status. Values: – 0x00 - SERVICE_STATUS_INACTIVE - Inactive – 0x01 - SERVICE_STATUS_ACTIVE - Active
SvcClass	• Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.
numLen	• Provides the length of number which follow.
number[255]	• number of numLen length, NULL terminated.
noReplyTimer	• No reply timer value in seconds • A value of 0 indicates that noReplyTimer is ignored.

8.1241.2 Field Documentation

8.1241.2.1 uint8_t voice_callFWInfo::noReplyTimer

8.1241.2.2 uint8_t voice_callFWInfo::number[255]

8.1241.2.3 uint8_t voice_callFWInfo::numLen

8.1241.2.4 uint8_t voice_callFWInfo::SvcClass

8.1241.2.5 uint8_t voice_callFWInfo::SvcStatus

8.1242 voice_callInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t callState
- uint8_t callType
- uint8_t direction
- uint8_t mode

8.1242.1 Detailed Description

This structure contains Information about call state changes. For example, when an incoming call is received, this structure is populated and indicate the incoming call information. When this incoming call is answered, the call status changes from INCOMING to CONVERSATION, which means a change in the call information and this structure is populated again with the changes and notified to/retrived by the user.

Parameters

callID	• Call identifier for the call queried for information. • If zero(0) then invalid.
callState	• Call state. – 0x01 - CALL_STATE_ORIGINATION - Origination – 0x02 - CALL_STATE_INCOMING - Incoming – 0x03 - CALL_STATE_CONVERSATION - Conversation – 0x04 - CALL_STATE_CC_IN_PROGRESS - Call is originating but waiting for call control to complete – 0x05 - CALL_STATE_ALERTING - Alerting – 0x06 - CALL_STATE_HOLD - Hold – 0x07 - CALL_STATE_WAITING - Waiting – 0x08 - CALL_STATE_DISCONNECTING - Disconnecting – 0x09 - CALL_STATE_END - End – 0x0A - CALL_STATE_SETUP - MT call is in Setup state in 3GPP – 0xFF - Not Available

callType	• Call type. – 0x00 - CALL_TYPE_VOICE - Voice – 0x02 - CALL_TYPE_VOICE_IP - Voice over IP – 0x06 - CALL_TYPE_OTAPA - OTAPA – 0x07 - CALL_TYPE_STD_OTASP - Standard OTASP – 0x08 - CALL_TYPE_NON_STD_OTASP - Nonstandard OTASP – 0x09 - CALL_TYPE_EMERGENCY - Emergency – 0xFF - Not Available
direction	• Direction. – 0x01 - CALL_DIRECTION_MO - MO call – 0x02 - CALL_DIRECTION_MT - MT call – 0xFF - Not Available
mode	• Mode. • If the mode field is “0x01 - CDMA”,the optional Service Option, Voice Privacy, and OTASP Status (only for OTASP calls) TLVs are included in the response. – 0x01 - CALL_MODE_CDMA - CDMA – 0x02 - CALL_MODE_GSM - GSM – 0x03 - CALL_MODE_UMTS - UMTS – 0x04 - CALL_MODE_LTE - LTE – 0x05 - CALL_MODE_TDS - TD-SCDMA – 0xFF - Not Available

8.1242.2 Field Documentation

8.1242.2.1 uint8_t voice_callInfo::callID

8.1242.2.2 uint8_t voice_callInfo::callState

8.1242.2.3 uint8_t voice_callInfo::callType

8.1242.2.4 uint8_t voice_callInfo::direction

8.1242.2.5 uint8_t voice_callInfo::mode

8.1243 voice_callingPartyInfo Struct Reference

Data Fields

- uint8_t PI
- uint8_t SI
- uint8_t numType
- uint8_t numPlan
- uint8_t numLen
- uint8_t number [255]

8.1243.1 Detailed Description

This structure contains Calling party Number Information

Parameters

PI	• Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
SI	• Number of sets of following elements – Caller Id
SI	• Number screening indicator. • Values: – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
numType	• Number type. • Values: – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
numPlan	• Number plan. • Values: – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
numLen	• Provides the length of number which follow.

number[255]	• number of numLen length, NULL terminated.

8.1243.2 Field Documentation

8.1243.2.1 uint8_t voice_callingPartyInfo::number[255]

8.1243.2.2 uint8_t voice_callingPartyInfo::numLen

8.1243.2.3 uint8_t voice_callingPartyInfo::numPlan

8.1243.2.4 uint8_t voice_callingPartyInfo::numType

8.1243.2.5 uint8_t voice_callingPartyInfo::PI

8.1243.2.6 uint8_t voice_callingPartyInfo::SI

8.1244 voice_ccSUPSType Struct Reference

Data Fields

- uint8_t svcType
- uint8_t reason

8.1244.1 Detailed Description

This structure contains information about the Call Control Supplementary Service Types

Parameters

svcType	• Service type. – 0x01 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_ACTIVATE - Activate – 0x02 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_DEACTIVATE - Deactivate – 0x03 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_REGISTER - Register – 0x04 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_ERASE - Erase – 0x05 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_INTERROGATE - Interrogate – 0x06 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_REGISTER_PASSWORD - Register password – 0x07 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_USSD - USSD – 0xFF - Not Available
reason	• Call control supplementary service result reason • Values: – See qaGobiApiTableCallControlReturnReasons.h for return reasons.

8.1244.2 Field Documentation

8.1244.2.1 uint8_t voice_ccSUPSType::reason

8.1244.2.2 uint8_t voice_ccSUPSType::svcType

8.1245 voice_CLIPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1245.1 Detailed Description

This structure contains information about the Calling Line Identification Presentation (CLIP) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.1245.2 Field Documentation

8.1245.2.1 uint8_t voice_CLIPResp::ActiveStatus

8.1245.2.2 uint8_t voice_CLIPResp::ProvisionStatus

8.1246 voice_CLIRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1246.1 Detailed Description

This structure contains information about the Calling Line Identification Restriction (CLIR) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED_PERMANENT - Permanently provisioned – 0x02 - PROVISION_STATUS_PRESENTATION_RESTRICTED - Restricted presentation – 0x03 - PROVISION_STATUS_PRESENTATION_ALLOWED - Allowed presentation – 0xFF - Not Available

8.1246.2 Field Documentation

8.1246.2.1 uint8_t voice_CLIRResp::ActiveStatus

8.1246.2.2 uint8_t voice_CLIRResp::ProvisionStatus

8.1247 voice_CNAPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1247.1 Detailed Description

This structure contains information about the Calling Name Presentation (CNAP) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.1247.2 Field Documentation

8.1247.2.1 uint8_t voice_CNAPResp::ActiveStatus

8.1247.2.2 uint8_t voice_CNAPResp::ProvisionStatus

8.1248 voice_COLPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1248.1 Detailed Description

This structure contains information about the Connected Line Identification Presentation (COLP) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.1248.2 Field Documentation

8.1248.2.1 uint8_t voice_COLPResp::ActiveStatus

8.1248.2.2 uint8_t voice_COLPResp::ProvisionStatus

8.1249 voice_COLRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1249.1 Detailed Description

This structure contains information about the Connected Line Identification Restriction (COLR) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.1249.2 Field Documentation

8.1249.2.1 uint8_t voice_COLRResp::ActiveStatus

8.1249.2.2 uint8_t voice_COLRResp::ProvisionStatus

8.1250 voice_connectNumInfo Struct Reference

Data Fields

- uint8_t numPresInd
- uint8_t screeningInd
- uint8_t numType
- uint8_t numPlan
- uint8_t callerIDLen
- uint8_t callerID [81]

8.1250.1 Detailed Description

This structure contains information about the numbers connected to a device. It contains information such as number type, eg International or Local.

Parameters

numPresInd	• Presentation indicator – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available

screeningInd	• Screening indicator. – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network – 0xFF - Not Available
numType	• Number type. – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension – 0xFF - Not Available
numPlan	• Number plan. – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension – 0xFF - Not Available
callerIDLen	• Provides the length of caller ID which follow. • If zero(0) then no further information exists.
callerID[MAX_VOICE_CALL_NO_LEN]	• callerID of numLen length, NULL terminated.

8.1250.2 Field Documentation

8.1250.2.1 uint8_t voice_connectNumInfo::callerID[81]

8.1250.2.2 uint8_t voice_connectNumInfo::callerIDLen

8.1250.2.3 uint8_t voice_connectNumInfo::numPlan

8.1250.2.4 uint8_t voice_connectNumInfo::numPresInd

8.1250.2.5 uint8_t voice_connectNumInfo::numType

8.1250.2.6 uint8_t voice_connectNumInfo::screeningInd

8.1251 voice_CUGInfo Struct Reference

Data Fields

- uint16_t CUGIndex
- uint8_t SuppPrefCUG
- uint8_t SuppOA

8.1251.1 Detailed Description

This structure contains Closed User Group Information

Parameters

CUGIndex	• Range 0x00... 0x7FFF
SuppPrefCUG	• Suppress preferential CUG – 0x00 - FALSE – 0x01 - TRUE
SuppOA	• Suppress OA subscription option – 0x00 - FALSE – 0x01 - TRUE

8.1251.2 Field Documentation

8.1251.2.1 uint16_t voice_CUGInfo::CUGIndex

8.1251.2.2 uint8_t voice_CUGInfo::SuppOA

8.1251.2.3 uint8_t voice_CUGInfo::SuppPrefCUG

8.1252 voice_curAMRConfig Struct Reference

Data Fields

- uint8_t gsmAmrStat
- uint8_t wcdmaAmrStat

8.1252.1 Detailed Description

This structure contains the Current Adaptive Multi Rate Configuration Information.

Parameters

gsmAmrStat	• GSM AMR Status – 0x00 - Disable – 0x01 - Enable – 0xFF - Not Available
wcdmaAmrStat	• WCDMA AMR Status • One or a combination of the following bitmask values: – Bit 0 - AMR codec advertised is not supported – Bit 1 - Controls WCDMA AMR wideband – Bit 2 - Controls GSM half rate AMR – Bit 3 - Controls GSM AMR wideband – Bit 4 - Controls GSM AMR narrowband • 0xFF, if not available

8.1252.2 Field Documentation

8.1252.2.1 uint8_t voice_curAMRConfig::gsmAmrStat

8.1252.2.2 uint8_t voice_curAMRConfig::wcdmaAmrStat

8.1253 voice_diagInfo Struct Reference

Data Fields

- uint8_t diagInfoLen
- uint8_t diagnosticInfo [255]

8.1253.1 Detailed Description

This structure contains Diagnostic Information

Parameters

diagInfoLen	• Provides the length of information which follow. • If zero(0) then no further information exists.
diagnosticInfo[MAX_VOICE_DESCRIPTION_LENGTH]	• Diagnostic information.

8.1253.2 Field Documentation

8.1253.2.1 uint8_t voice_diagInfo::diagInfoLen

8.1253.2.2 uint8_t voice_diagInfo::diagnosticInfo[255]

8.1254 voice_DTMFInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t DTMFEvent
- uint8_t digitCnt
- uint8_t digitBuff [255]

8.1254.1 Detailed Description

This structure contains information about the DTMF (Dual Tone Multi-Frequency).

Parameters

callID	• Call identifier for the current call.
DTMFEvent	• DTMF event – 0x00 - DTMF_EVENT_REV_BURST - Sends a CDMA-burst DTMF – 0x01 - DTMF_EVENT_REV_START_CONT - Starts a continuous DTMF tone – 0x03 - DTMF_EVENT_REV_STOP_CONT - Stops a continuous DTMF tone – 0x05 - DTMF_EVENT_FWD_BURST - Received a CDMA-burst DTMF message – 0x06 - DTMF_EVENT_FWD_START_CONT - Received a start-continuous DTMF tone order – 0x07 - DTMF_EVENT_FWD_STOP_CONT - Received a stop-continuous DTMF tone order
digitCnt	• Number of set of following element i.e. digitBuff.
digitBuff[MAX_VOICE_DESCRIPTION_LENGTH]	• DTMF digit buffer in ASCII string which is NULL terminated

8.1254.2 Field Documentation

8.1254.2.1 uint8_t voice_DTMFInfo::callID

8.1254.2.2 uint8_t voice_DTMFInfo::digitBuff[255]

8.1254.2.3 uint8_t voice_DTMFInfo::digitCnt

8.1254.2.4 uint8_t voice_DTMFInfo::DTMFEvent

8.1255 voice_DTMFLengths Struct Reference

Data Fields

- uint8_t DTMFPulseWidth
- uint8_t DTMFInterdigitInterval

8.1255.1 Detailed Description

This structure contains Voice Burst DTMF pulse length information

Parameters

DTMFPulse-Width	• DTMF pulse width. Values: – 0x00 - DTMF_ONLENGTH_95MS - 95 ms – 0x01 - DTMF_ONLENGTH_150MS - 150 ms – 0x02 - DTMF_ONLENGTH_200MS - 200 ms – 0x03 - DTMF_ONLENGTH_250MS - 250 ms – 0x04 - DTMF_ONLENGTH_300MS - 300 ms – 0x05 - DTMF_ONLENGTH_350MS - 350 ms – 0x06 - DTMF_ONLENGTH_SMS SMS Tx special pulse width
DTMFInterdigit-Interval	• DTMF interdigit interval Values: – 0x00 - DTMF_OFFLENGTH_60MS - 60 ms – 0x01 - DTMF_OFFLENGTH_100MS - 100 ms – 0x02 - DTMF_OFFLENGTH_150MS - 150 ms – 0x03 - DTMF_OFFLENGTH_200MS - 200 ms

8.1255.2 Field Documentation

8.1255.2.1 uint8_t voice_DTMFLengths::DTMFInterdigitInterval

8.1255.2.2 uint8_t voice_DTMFLengths::DTMFPulseWidth

8.1256 voice_ECTNum Struct Reference

Data Fields

- uint8_t ECTCallState
- uint8_t presentationInd
- uint8_t number [81]

8.1256.1 Detailed Description

Contains the parameters passed for Explicit Communication Transfer by the device.

Parameters

ECTCallState	• ECT call state: – 0x00 - ECT_CALL_STATE_NONE - None – 0x01 - ECT_CALL_STATE_ALERTING - Alerting – 0x02 - ECT_CALL_STATE_ACTIVE - Active

presentationInd	• Presentation indicator – 0x00 - presentationAllowedAddress – 0x01 - presentationRestricted – 0x02 - numberNotAvailable – 0x04 - presentationRestrictedAddress
number	• Number in ASCII characters terminated by NULL

8.1256.2 Field Documentation

8.1256.2.1 uint8_t voice_ECTNum::ECTCallState

8.1256.2.2 uint8_t voice_ECTNum::number[81]

8.1256.2.3 uint8_t voice_ECTNum::presentationInd

8.1257 voice_extDispRecInfo Struct Reference

Data Fields

- uint8_t dispType
- uint8_t extDispInfoLen
- uint8_t extDispInfo [255]

8.1257.1 Detailed Description

This structure contains Line Control Information

Parameters

dispType	• Values are per [S1, Table 3.7.5.16-1].
extDispInfoLen	• Number of sets of the following elements: – ext_display_info
extDispInfo	• Extended display information buffer containing the display record; refer to [S1, Section 3.7.5.16] for the format information of the buffer contents.

8.1257.2 Field Documentation

8.1257.2.1 uint8_t voice_extDispRecInfo::dispType

8.1257.2.2 uint8_t voice_extDispRecInfo::extDispInfo[255]

8.1257.2.3 uint8_t voice_extDispRecInfo::extDispInfoLen

8.1258 voice_getAllCallInformation Struct Reference

Data Fields

- voice_callInfo CallInfo
- uint8_t isEmpty
- uint8_t ALS

8.1258.1 Detailed Description

This structure contains information related to call state change.

Parameters

CallInfo	• See voice_callInfo for more information.
isEmpty	• Multiparty indicator. – 0x00 - False – 0x01 - True
ALS	• Alternate Line Service line indicator. • Feature for supporting two different phone numbers on the same mobile device. – 0x00 - ALS_LINE1 - Line 1 (default) – 0x01 - ALS_LINE2 - Line 2

8.1258.2 Field Documentation

8.1258.2.1 uint8_t voice_getAllCallInformation::ALS

8.1258.2.2 voice_callInfo voice_getAllCallInformation::CallInfo

8.1258.2.3 uint8_t voice_getAllCallInformation::isEmpty

8.1259 voice_getAllCallRmtPtyName Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyName RemotePartyName

8.1259.1 Detailed Description

This structure contains information for All Call Remote Party Names

Parameters

callID	• Unique call identifier for the call.
RemoteParty-Name	• See voice_remotePartyName for more information.

8.1259.2 Field Documentation

8.1259.2.1 uint8_t voice_getAllCallRmtPtyName::callID

8.1259.2.2 voice_remotePartyName voice_getAllCallRmtPtyName::RemotePartyName

8.1260 voice_getAllCallRmtPtyNum Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyNum RemotePartyNum

8.1260.1 Detailed Description

This structure contains information for All Call Remote Party Numbers

Parameters

callID	• Unique call identifier for the call.
RemoteParty-Num	• See voice_remotePartyNum for more information.

8.1260.2 Field Documentation

8.1260.2.1 uint8_t voice_getAllCallRmtPtyNum::callID

8.1260.2.2 voice_remotePartyNum voice_getAllCallRmtPtyNum::RemotePartyNum

8.1261 voice_getCallFWExtInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWExtInfo CallFWExtInfo [20]

8.1261.1 Detailed Description

This structure contains an array of Call Forwarded Extended Information.

Parameters

numInstances	• Number of callFWExtInfo that follow. • If zero(0) then no further information exists.
CallFWExtInfo[VOICE_MAX_NO_OF_CALLS]	• Array of CallFWExtInfo. – See voice_callFWExtInfo for more information.

8.1261.2 Field Documentation

8.1261.2.1 voice_callFWExtInfo voice_getCallFWExtInfo::CallFWExtInfo[20]

8.1261.2.2 uint8_t voice_getCallFWExtInfo::numInstances

8.1262 voice_getCallFWInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWInfo CallFWInfo [20]

8.1262.1 Detailed Description

This structure contains an array of Call Forwarded Information.

Parameters

numInstances	• Number of callFWInfo that follow. • If zero(0) then no further information exists.
CallFWInfo[VOICE_MAX_NO_OF_CALLS]	• Array of callFWInfo. – See voice_callFWInfo for more information.

8.1262.2 Field Documentation

8.1262.2.1 voice_callFWInfo voice_getCallFWInfo::CallFWInfo[20]

8.1262.2.2 uint8_t voice_getCallFWInfo::numInstances

8.1263 voice_lineCtrlInfo Struct Reference

Data Fields

- uint8_t polarityIncluded
- uint8_t toggleMode
- uint8_t revPolarity
- uint8_t pwrDenialTime

8.1263.1 Detailed Description

This structure contains Line Control Information

Parameters

polarityIncluded	• Included Polarity; Boolean Value
toggleMode	• Toggle mode; Boolean Value
revPolarity	• Reverse Polarity; Boolean Value
pwrDenialTime	• Power denial time; refer to [S1, Section 3.7.5.15 Line Control] for valid values

8.1263.2 Field Documentation

8.1263.2.1 uint8_t voice_lineCtrlInfo::polarityIncluded

8.1263.2.2 uint8_t voice_lineCtrlInfo::pwrDenialTime

8.1263.2.3 uint8_t voice_lineCtrlInfo::revPolarity

8.1263.2.4 uint8_t voice_lineCtrlInfo::toggleMode

8.1264 voice_newPwdData Struct Reference

Data Fields

- uint8_t newPwd [4]
- uint8_t newPwdAgain [4]

8.1264.1 Detailed Description

This structure contains New Password Data.

Parameters

newPwd[BARRING_PASSWORD_LENGTH]	• New password. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.
newPwdAgain[BARRING_PASSWORD_LENGTH]	• New password again. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.

8.1264.2 Field Documentation

8.1264.2.1 uint8_t voice_newPwdData::newPwd[4]

8.1264.2.2 uint8_t voice_newPwdData::newPwdAgain[4]

8.1265 voice_NSSAudioCtrl Struct Reference

Data Fields

- uint8_t upLink
- uint8_t downLink

8.1265.1 Detailed Description

This structure contains National Supplementary Services - Audio Control Information

Parameters

upLink	• Values as per[S24, 4.10 Reservation response].
downLink	• Values as per[S24, 4.10 Reservation response].

8.1265.2 Field Documentation

8.1265.2.1 uint8_t voice_NSSAudioCtrl::downLink

8.1265.2.2 uint8_t voice_NSSAudioCtrl::upLink

8.1266 voice_peerNumberInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t numPI
- uint8_t numSI
- uint8_t numType
- uint8_t numPlan
- uint8_t numLen
- uint8_t number [81]

8.1266.1 Detailed Description

This structure contains information for Connected Peer Numbers.

Parameters

callID	• Unique call identifier for the call.

numPI	• Number presentation indicator. – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available
numSI	• Number screening indicator. – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
numType	• Number type. – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
numPlan	• Number plan. – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
numLen	• Provides the length of number which follow.
number[MAX_VOICE_CALL_NO_LEN]	• number of numLen length, NULL terminated.

8.1266.2 Field Documentation

8.1266.2.1 uint8_t voice_peerNumberInfo::callID

8.1266.2.2 uint8_t voice_peerNumberInfo::number[81]

8.1266.2.3 uint8_t voice_peerNumberInfo::numLen

8.1266.2.4 uint8_t voice_peerNumberInfo::numPI

8.1266.2.5 uint8_t voice_peerNumberInfo::numPlan

8.1266.2.6 uint8_t voice_peerNumberInfo::numSI

8.1266.2.7 uint8_t voice_peerNumberInfo::numType

8.1267 voice_prefVoiceSO Struct Reference

Data Fields

- uint8_t namID
- uint8_t evrcCapability
- uint16_t homePageVoiceSO
- uint16_t homeOrigVoiceSO
- uint16_t roamOrigVoiceSO

8.1267.1 Detailed Description

This structure contains information about the Preferred Voice Service Options.

Parameters

namID	• Index of the NAM(Number Assignment Module) to be configured. • Range 0 to 3. • Some modems support only 1 or 2 NAMs. • 0xFF,if not available.
evrcCapability	• EVRC capability. • Values: – 0x00 - Disable – 0x01 - Enable – 0xFF - Not Available

homePageVoiceSO	• Home page voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARR0W_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available
homeOrigVoiceSO	• Home origination voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARR0W_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available
roamOrigVoiceSO	• Roaming origination voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARR0W_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available

8.1267.2 Field Documentation

8.1267.2.1 uint8_t voice_prefVoiceSO::evrcCapability

8.1267.2.2 uint16_t voice_prefVoiceSO::homeOrigVoiceSO

8.1267.2.3 uint16_t voice_prefVoiceSO::homePageVoiceSO

8.1267.2.4 uint8_t voice_prefVoiceSO::namID

8.1267.2.5 uint16_t voice_prefVoiceSO::roamOrigVoiceSO

8.1268 voice_redirNumInfo Struct Reference

Data Fields

- uint8_t PI
- uint8_t SI
- uint8_t numType
- uint8_t numPlan
- uint8_t reason
- uint8_t numLen
- uint8_t number [255]

8.1268.1 Detailed Description

This structure contains Redirecting Number Information

Parameters

PI	• Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
SI	• Number of sets of following elements – Caller Id
SI	• Number screening indicator. • Values: – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network

numType	• Number type. • Values: – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
numPlan	• Number plan. • Values: – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
reason	-Redirecting reason; refer to [S1, Table 3.7.5.11-1] for valid values
numLen	• Provides the length of number which follow.
number[255]	• number of numLen length, NULL terminated.

8.1268.2 Field Documentation

8.1268.2.1 uint8_t voice_redirNumInfo::number[255]

8.1268.2.2 uint8_t voice_redirNumInfo::numLen

8.1268.2.3 uint8_t voice_redirNumInfo::numPlan

8.1268.2.4 uint8_t voice_redirNumInfo::numType

8.1268.2.5 uint8_t voice_redirNumInfo::PI

8.1268.2.6 uint8_t voice_redirNumInfo::reason

8.1268.2.7 uint8_t voice_redirNumInfo::SI

8.1269 voice_remotePartyName Struct Reference

Data Fields

- uint8_t namePI
- uint8_t codingScheme
- uint8_t nameLen
- uint8_t callerName [255]

8.1269.1 Detailed Description

This structure contains information about the names that are dialed from the device or from which a call is received on the device.

Parameters

namePI	• Name presentation indicator. – 0x00 - PRESENTATION_NAME_PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_NAME_PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NAME_UNAVAILABLE - Unavailable presentation – 0x03 - PRESENTATION_NAME_NAME_PRESENTATION_RESTRICTED - Restricted name presentation – 0xFF - Not Available
codingScheme	• Refer to Table10 qaGobiApiTableCodingScheme.h for coding schemes • 0xFF - Not Available
nameLen	• Provides the length of name which follow. • If zero(0) then no further information exists.
callerName[MAX_VOICE_DESCRIPTION_LENGTH]	• Name in ASCII, NULL ending.

8.1269.2 Field Documentation

8.1269.2.1 uint8_t voice_remotePartyName::callerName[255]

8.1269.2.2 uint8_t voice_remotePartyName::codingScheme

8.1269.2.3 uint8_t voice_remotePartyName::nameLen

8.1269.2.4 uint8_t voice_remotePartyName::namePI

8.1270 voice_remotePartyNum Struct Reference

Data Fields

- uint8_t presentationInd
- uint8_t numLen
- uint8_t remPartyNumber [81]

8.1270.1 Detailed Description

This structure contains information about the numbers that are dialed from the device or from which a call is received on the device.

Parameters

presentationInd	• Presentation indicator. – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available
numLen	• Provides the length of number which follow. • If zero(0) then no further information exists.
remParty-Number[MAX_-CALL_NO_LEN]	• Array of numbers in ASCII, NULL ending.

8.1270.2 Field Documentation

8.1270.2.1 uint8_t voice_remotePartyNum::numLen

8.1270.2.2 uint8_t voice_remotePartyNum::presentationInd

8.1270.2.3 uint8_t voice_remotePartyNum::remPartyNumber[81]

8.1271 voice_roamTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t roamTimerValue

8.1271.1 Detailed Description

This structure contains information about the Roam Timer.

Parameters

namID	• Index of the NAM(Number Assignment Module) to be configured. • Range 0 to 3. • Some modems support only 1 or 2 NAMs. • 0xFF,if not available.
roamTimerValue	• Time in minutes. • Cumulative air time is slammed. • 0xFFFFFFFF,if not available.

8.1271.2 Field Documentation

8.1271.2.1 uint8_t voice_roamTimer::namID

8.1271.2.2 uint32_t voice_roamTimer::roamTimerValue

8.1272 voice_signalInfo Struct Reference

Data Fields

- uint8_t signalType
- uint8_t alertPitch
- uint8_t signal

8.1272.1 Detailed Description

This structure contains Signal Information

Parameters

signalType	• Call identifier for the call.
alertPitch	• Signal Information
signal	• Caller ID Information

8.1272.2 Field Documentation

8.1272.2.1 uint8_t voice_signalInfo::alertPitch

8.1272.2.2 uint8_t voice_signalInfo::signal

8.1272.2.3 uint8_t voice_signalInfo::signalType

8.1273 voice_SUPSInfo Struct Reference

Data Fields

- uint8_t svcType
- uint8_t isModByCC

8.1273.1 Detailed Description

This structure contains information about the Supplementary Services.

Parameters

svcType	• Service type. – 0x01 - SERVICE_TYPE_ACTIVATE - Activate – 0x02 - SERVICE_TYPE_DEACTIVATE - Deactivate – 0x03 - SERVICE_TYPE_REGISTER - Register – 0x04 - SERVICE_TYPE_ERASE - Erase – 0x05 - SERVICE_TYPE_INTERROGATE - Interrogate – 0x06 - SERVICE_TYPE_REGISTER_PASSWORD - Register password – 0x07 - SERVICE_TYPE_USSD - USSD
isModByCC	• Indicates whether the supplementary service data is modified by the card (SIM/USIM) as part of the call control: – 0 - False – 1 - True

8.1273.2 Field Documentation

8.1273.2.1 uint8_t voice_SUPSInfo::isModByCC

8.1273.2.2 uint8_t voice_SUPSInfo::svcType

8.1274 voice_USSDNotificationNetworkInfo Struct Reference

Data Fields

- uint8_t tlvPresent
- struct voice_USSInfo networkInfo

8.1274.1 Detailed Description

Structure for storing the USS info present in USSDNotification callback.

Parameters

tlvPresent	Values: • 0 - Not present • 1 - Tlv is present

NetworkInfo	• USS Data from Network (See structure voice_USSInfo) • NULL pointer - Invalid data.

8.1274.2 Field Documentation

8.1274.2.1 struct voice_USSInfo voice_USSDNotificationNetworkInfo::networkInfo

8.1274.2.2 uint8_t voice_USSDNotificationNetworkInfo::tlvPresent

8.1275 voice_USSInfo Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.1275.1 Detailed Description

This structure contains USS Information

Parameters

ussDCS	• 1 - ASCII coding scheme • 2 - 8-BIT coding scheme • 3 - UCS2
ussLen	• Range 1 to 182
ussData	• Data encoded as per the DCS

8.1275.2 Field Documentation

8.1275.2.1 uint8_t voice_USSInfo::ussData[182]

8.1275.2.2 uint8_t voice_USSInfo::ussDCS

8.1275.2.3 uint8_t voice_USSInfo::ussLen

8.1276 voice_UUSInfo Struct Reference

Data Fields

- uint8_t UUSType
- uint8_t UUSDcs

- uint8_t UUSDatalen
- uint8_t UUSData [255]

8.1276.1 Detailed Description

This structure contains User to User Signaling Service Information.

Parameters

UUSType	• UUS type values are: – 0x00 - UUS_DATA – 0x01 - UUS_TYPE1_IMPLICIT – 0x02 - UUS_TYPE1_REQUIRED – 0x03 - UUS_TYPE1_NOT_REQUIRED – 0x04 - UUS_TYPE2_REQUIRED – 0x05 - UUS_TYPE2_NOT_REQUIRED – 0x06 - UUS_TYPE3_REQUIRED – 0x07 - UUS_TYPE3_NOT_REQUIRED – 0xFF - Not Available
UUSDcs	• UUS data coding scheme values are: – 0x01 - UUS_DCS_USP – 0x02 - UUS_DCS_OHLP – 0x03 - UUS_DCS_X244 – 0x04 - UUS_DCS_SMCF – 0x05 - UUS_DCS_IA5 – 0x06 - UUS_DCS_RV12RD – 0x07 - UUS_DCS_Q931UNCCM – 0xFF - Not Available
UUSDatalen	• Number of sets of the following elements. – UUSData • If zero(0) then no further information exists.
UUSData[MAX_VOICE_DESCRIPTION_LENGTH]	• UUS data encoded as per coding scheme

8.1276.2 Field Documentation

8.1276.2.1 uint8_t voice_UUSInfo::UUSData[255]

8.1276.2.2 uint8_t voice_UUSInfo::UUSDatalen

8.1276.2.3 uint8_t voice_UUSInfo::UUSDcs

8.1276.2.4 uint8_t voice_UUSInfo::UUSType

8.1277 voice_VoiceUssdNoWaitAlphaIDTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- voice_alphaIDInfo Alpha_Identifier

8.1277.1 Detailed Description

Contains the parameters passed for VoiceUssdNoWaitAlphaIDTlv by the device.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
Alpha_Identifier	• Alpha coding scheme. • Values: – 0x01 - ALPHA_DCS_GSM - GSM Default_Char – 0x02 - ALPHA_DCS_UCS2 - UCS2 • See voice_alphaIDInfo for more details.

8.1277.2 Field Documentation

8.1277.2.1 voice_alphaIDInfo voice_VoiceUssdNoWaitAlphaIDTlv::Alpha_Identifier

8.1277.2.2 uint8_t voice_VoiceUssdNoWaitAlphaIDTlv::TlvPresent

8.1278 voice_VoiceUssdNoWaitErrorTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t Error

8.1278.1 Detailed Description

Contains the parameters passed for VoiceUssdNoWaitErrorTlv by the device.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
Error	• Type of error

8.1278.2 Field Documentation

8.1278.2.1 uint16_t voice_VoiceUssdNoWaitErrorTlv::Error

8.1278.2.2 uint8_t voice_VoiceUssdNoWaitErrorTlv::TlvPresent

8.1279 voice_VoiceUssdNoWaitFailCauseTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t Failure_Cause

8.1279.1 Detailed Description

Contains the parameters passed for VoiceUssdNoWaitFailCauseTlv by the device.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
Failure_Cause	• Supplementary services failure cause

8.1279.2 Field Documentation

8.1279.2.1 uint16_t voice_VoiceUssdNoWaitFailCauseTlv::Failure_Cause

8.1279.2.2 uint8_t voice_VoiceUssdNoWaitFailCauseTlv::TlvPresent

8.1280 voice_VoiceUssdNoWaitUSSINFOTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- struct voice_USSInfo NetworkInfo

8.1280.1 Detailed Description

Contains the parameters passed for VoiceUssdNoWaitUSSINFOTlv by the device.

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
NetworkInfo	• USSD information from the network • See voice_USSInfo for more details.

8.1280.2 Field Documentation

8.1280.2.1 struct voice_USSInfo voice_VoiceUssdNoWaitUSSINFOTlv::NetworkInfo

8.1280.2.2 uint8_t voice_VoiceUssdNoWaitUSSINFOTlv::TlvPresent

8.1281 wds_channelRate Struct Reference

Data Fields

- uint32_t CurrChanTxRate
- uint32_t CurrChanRxRate

8.1281.1 Detailed Description

This structure contains Channel Rate

Parameters

CurrChanTxRate	• Max channel Tx rate in bits per second
CurrChanRx-Rate	• Max channel Rx rate in bits per second

8.1281.2 Field Documentation

8.1281.2.1 uint32_t wds_channelRate::CurrChanRxRate

8.1281.2.2 uint32_t wds_channelRate::CurrChanTxRate

8.1282 wds_ChannelRateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- wds_channelRate ChannelRate

8.1282.1 Detailed Description

This structure stores information about channel rate TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
ChannelRate	• see wds_channelRate for more info

8.1282.2 Field Documentation

8.1282.2.1 wds_channelRate wds_ChannelRateTlv::ChannelRate

8.1282.2.2 uint8_t wds_ChannelRateTlv::TlvPresent

8.1283 wds_ConnStatusTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t MDMConnStatus

8.1283.1 Detailed Description

This structure stores information about modem connection status TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
MDMConn-Status	-Modem connecton status • 0x01 - disconnected • 0x02 - connected

8.1283.2 Field Documentation

8.1283.2.1 uint8_t wds_ConnStatusTlv::MDMConnStatus

8.1283.2.2 uint8_t wds_ConnStatusTlv::TlvPresent

8.1284 wds_currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.1284.1 Detailed Description

Network information structure

Parameters

NetworkType	• Values: – 0 - 3GPP – 1 - 3GPP2

RATMask	• RAT mask to indicate type of technology • Values – 0x00 - DONT_CARE – 0x8000 - NULL_BEARER • CDMA RAT mask – 0x01 - CDMA_1X – 0x02 - EVDO_REV0 – 0x05 - HRPD – 0x0B - EHRPD • UMTS RAT mask – 0x03 - GPRS – 0x04 - WCDMA – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA
SOMask	• SO Mask

8.1284.2 Field Documentation

8.1284.2.1 uint8_t wds_currNetworkInfo::NetworkType

8.1284.2.2 uint32_t wds_currNetworkInfo::RATMask

8.1284.2.3 uint32_t wds_currNetworkInfo::SOMask

8.1285 wds_DataBearTechTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t DataBearerTech

8.1285.1 Detailed Description

This structure hold parameters about data bearer technology TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0

DataBearerTech	• Data bear technology – 0x01 - cdma2000 1X – 0x02 - cdma2000 HRPD (1xEV-DO) – 0x03 - GSM – 0x04 - UMTS – 0x05 - cdma200 HRPD (1xEV-DO RevA) – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - cdma2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown

8.1285.2 Field Documentation

8.1285.2.1 uint8_t wds_DataBearTechTlv::DataBearerTech

8.1285.2.2 uint8_t wds_DataBearTechTlv::TlvPresent

8.1286 wds_DataULongLongTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t ullData

8.1286.1 Detailed Description

This structure contains Data unsigned long long TLV data.

Parameters

TlvPresent	a flag indicates whether TLV presented or not
ullData	unsigend long long data in 64bit

8.1286.2 Field Documentation

8.1286.2.1 uint8_t wds_DataULongLongTlv::TlvPresent

8.1286.2.2 uint64_t wds_DataULongLongTlv::ullData

8.1287 wds_DataULongTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ulData

8.1287.1 Detailed Description

This structure contains Data unsigned long TLV data.

Parameters

TlvPresent	a flag indicates whether TLV presented or not
ulData	unsigend long data in 32bit

8.1287.2 Field Documentation

8.1287.2.1 uint8_t wds_DataULongTlv::TlvPresent

8.1287.2.2 uint32_t wds_DataULongTlv::ulData

8.1288 wds_DHCPLeaseOptTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t numOpt
- wds_DHCPOpt optList [30]
- uint8_t optListData [2048]

8.1288.1 Detailed Description

DHCP lease option information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
numOpt	number of sets of wds_DHCPOpt
optList	option list
optListData	option list data

8.1288.2 Field Documentation

8.1288.2.1 uint8_t wds_DHCPLeaseOptTlv::numOpt

8.1288.2.2 wds_DHCPOpt wds_DHCPLeaseOptTlv::optList[30]

8.1288.2.3 uint8_t wds_DHCPLeaseOptTlv::optListData[2048]

8.1288.2.4 uint8_t wds_DHCPLeaseOptTlv::TlvPresent

8.1289 wds_DHCPLeaseStateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t leaseState

8.1289.1 Detailed Description

DHCP lease state information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
leaseState	lease state • 0 - active, newly acquired • 1 - active, renewed • 2 - active, renewing • 3 - active, rebinding • 4 - inactive, expired • 5 - inactive, renew refused • 6 - inactive, rebind refused • 7 - inactive, other

8.1289.2 Field Documentation

8.1289.2.1 uint8_t wds_DHCPLeaseStateTlv::leaseState

8.1289.2.2 uint8_t wds_DHCPLeaseStateTlv::TlvPresent

8.1290 wds_DHCPOpt Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t * pOptVal

8.1290.1 Detailed Description

DHCP option code information

Parameters

optCode	option code • values: – 0 - 255

optValLen	length of option code • values: – 0 - 255
pOptVal	option value

8.1290.2 Field Documentation

8.1290.2.1 uint8_t wds_DHCPOpt::optCode

8.1290.2.2 uint8_t wds_DHCPOpt::optValLen

8.1290.2.3 uint8_t* wds_DHCPOpt::pOptVal

8.1291 wds_DHCPProfileIdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t profileType
- uint8_t profileId

8.1291.1 Detailed Description

DHCP profile id TLV information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
profileType	identifying the type of the profile • 0 - 3GPP
profileId	index identifying the profile • 1-24 valid for 3GPP profile type (9x30 and onwards)

8.1291.2 Field Documentation

8.1291.2.1 uint8_t wds_DHCPProfileIdTlv::profileId

8.1291.2.2 uint8_t wds_DHCPProfileIdTlv::profileType

8.1291.2.3 uint8_t wds_DHCPProfileIdTlv::TlvPresent

8.1292 wds_DHCPv4HWConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.1292.1 Detailed Description

WDS SWI DHCPv4 HW Config Structure.

Parameters

hwType	• HW Type 1 - Ethernet 20 - Serial
chaddrlen	• chaddrlen
chaddr	• chaddr. Max size 16 bytes

8.1292.2 Field Documentation

8.1292.2.1 uint8_t wds_DHCPv4HWConfig::chaddr[16]

8.1292.2.2 uint8_t wds_DHCPv4HWConfig::chaddrLen

8.1292.2.3 uint8_t wds_DHCPv4HWConfig::hwType

8.1293 wds_DHCPv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.1293.1 Detailed Description

WDS SWI DHCPv4 Option Structure

Parameters

optCode	• Option code – 0 - 255
optValLen	• Option value length – 0 - 255
optVal	• Option value

8.1293.2 Field Documentation

8.1293.2.1 uint8_t wds_DHCPv4Option::optCode

8.1293.2.2 uint8_t wds_DHCPv4Option::optVal[255]

8.1293.2.3 uint8_t wds_DHCPv4Option::optValLen

8.1294 wds_DHCPv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wds_DHCPv4Option * pOptList

8.1294.1 Detailed Description

WDS SWI DHCPv4 Option List Structure

Parameters

numOpt	• number of options – 0 - 255
pOptList	• pointer to list of DHCP Options

8.1294.2 Field Documentation

8.1294.2.1 uint8_t wds_DHCPv4OptionList::numOpt

8.1294.2.2 wds_DHCPv4Option* wds_DHCPv4OptionList::pOptList

8.1295 wds_DHCPv4ProfileId Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileId

8.1295.1 Detailed Description

WDS SWI DHCPv4 Profile Identifier Structure

Parameters

profileType	• 0 for 3GPP
profileId	• 1 to 24 for 3GPP profile

8.1295.2 Field Documentation

8.1295.2.1 uint8_t wds_DHCPv4ProfileId::profileId

8.1295.2.2 uint8_t wds_DHCPv4ProfileId::profileType

8.1296 wds_Domain Struct Reference

Data Fields

- uint16_t domainLen
- uint8_t domainName [256]

8.1296.1 Detailed Description

This structure contains the DomainName Information

Parameters

domainLen	• length of the recieved Domain name
domainName	• Domain name(Max 256 characters)

8.1296.2 Field Documentation

8.1296.2.1 uint16_t wds_Domain::domainLen

8.1296.2.2 uint8_t wds_Domain::domainName[256]

8.1297 wds_DomainNameList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_Domain domain [10]

8.1297.1 Detailed Description

This structure contains the DomainNameList Information

Parameters

numInstances	• Number of Domain name receieved
domain	• Domain name information(Max 10 Domain names)

Note

Data invalid when numInstances equal 0xff.

8.1297.2 Field Documentation

8.1297.2.1 struct wds_Domain wds_DomainNameList::domain[10]

8.1297.2.2 uint8_t wds_DomainNameList::numInstances

8.1298 wds_DormStatTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t DormancyStat

8.1298.1 Detailed Description

This structure stores information about dormancy status TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
DormancyStat	-Dormancy status • 0x01 - Traffic channel dormant • 0x02 - Traffic channel active

8.1298.2 Field Documentation

8.1298.2.1 uint8_t wds_DormStatTlv::DormancyStat

8.1298.2.2 uint8_t wds_DormStatTlv::TlvPresent

8.1299 wds_GPRSQoS Struct Reference

Data Fields

- uint32_t precedenceClass
- uint32_t delayClass
- uint32_t reliabilityClass
- uint32_t peakThroughputClass
- uint32_t meanThroughputClass

8.1299.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

Parameters

precedence-Class	• Precedence class

delayClass	• Delay class
reliabilityClass	• Reliability class
peak-Throughput-Class	• Peak throughput class
mean-Throughput-Class	• Mean throughput class

Note

Data invalid when all parameters are equal to 0xffffffff.

8.1299.2 Field Documentation

8.1299.2.1 uint32_t wds_GPRSQoS::delayClass

8.1299.2.2 uint32_t wds_GPRSQoS::meanThroughputClass

8.1299.2.3 uint32_t wds_GPRSQoS::peakThroughputClass

8.1299.2.4 uint32_t wds_GPRSQoS::precedenceClass

8.1299.2.5 uint32_t wds_GPRSQoS::reliabilityClass

8.1300 wds_IPv4AdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t IPv4Addr

8.1300.1 Detailed Description

IPv4 address TLV information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
IPv4Addr	IPv4 address

8.1300.2 Field Documentation

8.1300.2.1 uint32_t wds_IPv4AdTlv::IPv4Addr

8.1300.2.2 uint8_t wds_IPv4AdTlv::TlvPresent

8.1301 wds_IPV6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.1301.1 Detailed Description

This structure contains the IPV6 Address Information

Parameters

IPV6PrefixLen	• Length of the received IPv6 address in no. of bits; can take value between 0 and 128 – 0xFF - Not Available
IPAddressV6	• IPv6 address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

8.1301.2 Field Documentation

8.1301.2.1 uint16_t wds_IPV6AddressInfo::IPAddressV6[8]

8.1301.2.2 uint8_t wds_IPV6AddressInfo::IPV6PrefixLen

8.1302 wds_IPV6GWAddressInfo Struct Reference

Data Fields

- uint8_t gwV6PrefixLen
- uint16_t gwAddressV6 [8]

8.1302.1 Detailed Description

This structure contains the IPV6 Gateway Address Information

Parameters

gwV6PrefixLen	• Length of the received IPv6 Gateway address in no. of bits; can take value between 0 and 128
IPAddressV6	• IPv6 Gateway address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

Note

Data invalid when IPV6PrefixLen equal 0xff.

8.1302.2 Field Documentation

8.1302.2.1 uint16_t wds_IPV6GWAddressInfo::gwAddressV6[8]

8.1302.2.2 uint8_t wds_IPV6GWAddressInfo::gwV6PrefixLen

8.1303 wds_LastMdmCallEndRsnTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t CallEndReason

8.1303.1 Detailed Description

This structure stores information about modem last call end reason TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
CallEndReason	-Modem call end reason • See qaGobiApiTableCallEndReasons.h for Call End Reason

8.1303.2 Field Documentation

8.1303.2.1 uint16_t wds_LastMdmCallEndRsnTlv::CallEndReason

8.1303.2.2 uint8_t wds_LastMdmCallEndRsnTlv::TlvPresent

8.1304 wds_PCSCFFQDNAddress Struct Reference

Data Fields

- uint16_t fqdnLen
- uint8_t fqdnAddr [256]

8.1304.1 Detailed Description

This structure contains the PCSCFFQDNAddress Information

Parameters

fqdnLen	• length of the received FQDN address
fqdnAddr	• FQDN address(Max 256 characters)

8.1304.2 Field Documentation

8.1304.2.1 uint8_t wds_PCSCFFQDNAddress::fqdnAddr[256]

8.1304.2.2 uint16_t wds_PCSCFFQDNAddress::fqdnLen

8.1305 wds_PCSCFFQDNAddressList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_PCSCFFQDNAddress pcsfFQDNAddress [10]

8.1305.1 Detailed Description

This structure contains the PCSCFFQDNAddressList Information

Parameters

numInstances	• Number of FQDN addresses received
pcsfFQDN-Address	• FQDN address information(Max 10 addresses)

Note

Data invalid when numInstances equal 0xff.

8.1305.2 Field Documentation

8.1305.2.1 uint8_t wds_PCSCFFQDNAddressList::numInstances

8.1305.2.2 struct wds_PCSCFFQDNAddress wds_PCSCFFQDNAddressList::pcsfFQDNAddress[10]

8.1306 wds_PCSCFIPv4ServerAddressList Struct Reference

Data Fields

- uint8_t numInstances
- uint32_t pscsfIPv4Addr [64]

8.1306.1 Detailed Description

This structure contains the PCSCFIPv4ServerAddressList Information

Parameters

numInstances	• number of address following
pscsfIPv4Addr	• P-CSCF IPv4 server addresses(Max 16 address, 4 bytes each)

Note

Data invalid when numInstances equal 0xff and pscsfIPv4Addr equal to 0xffffffff.

8.1306.2 Field Documentation

8.1306.2.1 uint8_t wds_PCSCFIPv4ServerAddressList::numInstances

8.1306.2.2 uint32_t wds_PCSCFIPv4ServerAddressList::pscsfIPv4Addr[64]

8.1307 wds_profileChange Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileInx

8.1307.1 Detailed Description

This structure contains Profile Change Info

Parameters

profileType	• Identifies the type of the profile
profileInx	• Index identifying the profile

8.1307.2 Field Documentation

8.1307.2.1 uint8_t wds_profileChange::profileInx

8.1307.2.2 uint8_t wds_profileChange::profileType

8.1308 wds_ProfileIdentifier Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.1308.1 Detailed Description

This structure contains the Profile Identifier Information

Parameters

profileType	• Identifies the type of profile 0x00 = 3GPP

profileIndex	• Index of profile whose settings were loaded prior to session parameter negotiation for the current call. If this TLV is not present, data call parameters are based on device default settings for each parameter

Note

Data invalid when all parameters are equal to 0xff.

8.1308.2 Field Documentation

8.1308.2.1 uint8_t wds_ProfileIdentifier::profileIndex

8.1308.2.2 uint8_t wds_ProfileIdentifier::profileType

8.1309 wds_profileInfo Union Reference

Data Fields

- LibPackprofile_3GPP SlqsProfile3GPP
- LibPackprofile_3GPP2 SlqsProfile3GPP2

8.1309.1 Detailed Description

This union consist of profile_3GPP and profile_3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile See LibPackprofile_3GPP
SlqsProfile3GP-P2	3GPP2 profile See LibPackprofile_3GPP2

8.1309.2 Field Documentation

8.1309.2.1 LibPackprofile_3GPP wds_profileInfo::SlqsProfile3GPP

8.1309.2.2 LibPackprofile_3GPP2 wds_profileInfo::SlqsProfile3GPP2

8.1310 wds_RXBytesOKTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t RxByteOKCnt

8.1310.1 Detailed Description

This structure stores information about Rx Bytes OK TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
RxByteOKCnt	-Number of bytes received without error

8.1310.2 Field Documentation

8.1310.2.1 uint64_t wds_RXBytesOKTlv::RxByteOKCnt

8.1310.2.2 uint8_t wds_RXBytesOKTlv::TlvPresent

8.1311 wds_sourceOfChange Struct Reference

Data Fields

- uint8_t source

8.1311.1 Detailed Description

This structure contains Source of change Info

Parameters

source	• source of change

8.1311.2 Field Documentation

8.1311.2.1 uint8_t wds_sourceOfChange::source

8.1312 wds_transferStatInd Struct Reference

Data Fields

- uint8_t StatsPeriod
- uint32_t StatsMask

8.1312.1 Detailed Description

This structure contains the transfer statistic indication information

Parameters

StatsPeriod	Field Period between transfer statistic reports. • 0 - Do not report. • 1 - Other - Period between reports (in seconds).
StatsMask	requested statistic bit mask. • 0x00000040 - Tx bytes OK • 0x00000080 - Rx bytes OK

8.1312.2 Field Documentation

8.1312.2.1 uint32_t wds_transferStatInd::StatsMask

8.1312.2.2 uint8_t wds_transferStatInd::StatsPeriod

8.1313 wds_TrStatInd Struct Reference

Data Fields

- uint8_t statsPeriod
- uint32_t statsMask

8.1313.1 Detailed Description

This structure contains the information about the Transfer Statistics Indicator parameters.

Parameters

statsPeriod	• Period between transfer statistics reports. – 0 - Do not report – Other - Period between reports (seconds)
statsMask	• Requested statistic bit mask. – 0x00000001 - Tx packets OK – 0x00000002 - Rx packets OK – 0x00000004 - Tx packet errors – 0x00000008 - Rx packet errors – 0x00000010 - Tx overflows – 0x00000020 - Rx overflows – 0x00000040 - Tx bytes OK – 0x00000080 - Rx bytes OK • Each bit set causes the corresponding optional information to be sent in SLQSWdsEventReport-CallBack. • All unlisted bits are reserved for future use and must be set to zero.

8.1313.2 Field Documentation

8.1313.2.1 uint32_t wds_TrStatInd::statsMask

8.1313.2.2 uint8_t wds_TrStatInd::statsPeriod

8.1314 wds_TXBytesOKTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t TxByteOKCnt

8.1314.1 Detailed Description

This structure stores information about Tx Bytes OK TLV.

Parameters

TlvPresent	• value is 1 if this TLV is present in indication, otherwise 0
TxByteOKCnt	• Number of bytes transmitted without error

8.1314.2 Field Documentation

8.1314.2.1 uint8_t wds_TXBytesOKTlv::TlvPresent

8.1314.2.2 uint64_t wds_TXBytesOKTlv::TxByteOKCnt

8.1315 wds_UMTSMinQoS Struct Reference

Data Fields

- uint8_t trafficClass
- uint32_t maxUplinkBitrate
- uint32_t maxDownlinkBitrate
- uint32_t grntUplinkBitrate
- uint32_t grntDownlinkBitrate
- uint8_t qosDeliveryOrder
- uint32_t maxSDUSize
- uint8_t sduErrorRatio
- uint8_t resBerRatio
- uint8_t deliveryErrSDU
- uint32_t transferDelay
- uint32_t trafficPriority

8.1315.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

Parameters

trafficClass	• 0x00 - Subscribed • 0x01 - Conversational • 0x02 - Streaming • 0x03 - Interactive • 0x04 - Background
maxUplinkBitrate	• Maximum uplink bit rate in bits/sec
maxDownlink-Bitrate	• Maximum downlink bit rate in bits/sec

grntUplinkBitrate	• Guaranteed uplink bit rate in bits/sec
grntDownlink-Bitrate	• Guaranteed downlink bit rate in bits/sec
qosDelivery-Order	- Qos delivery order • 0x00 - Subscribe • 0x01 - Delivery order on • 0x02 - Delivery order off
maxSDUSize	• Maximum SDU size
sduErrorRatio	- SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - 1∗10^(-2) • 0x02 - 7∗10^(-3) • 0x03 - 1∗10^(-3) • 0x04 - 1∗10^(-4) • 0x05 - 1∗10^(-5) • 0x06 - 1∗10^(-6) • 0x07 - 1∗10^(-1)
resBerRatio	- Residual bit error ratio • Target value for undetected bit error ratio in in the delivered SDUs. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)
deliveryErrSDU	- delivery of erroneous SDUs • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)

transferDelay	- Transfer delay (ms) • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
trafficPriority	- Transfer handling priority • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

8.1315.2 Field Documentation

8.1315.2.1 uint8_t wds_UMTSMinQoS::deliveryErrSDU

8.1315.2.2 uint32_t wds_UMTSMinQoS::grntDownlinkBitrate

8.1315.2.3 uint32_t wds_UMTSMinQoS::grntUplinkBitrate

8.1315.2.4 uint32_t wds_UMTSMinQoS::maxDownlinkBitrate

8.1315.2.5 uint32_t wds_UMTSMinQoS::maxSDUSize

8.1315.2.6 uint32_t wds_UMTSMinQoS::maxUplinkBitrate

8.1315.2.7 uint8_t wds_UMTSMinQoS::qosDeliveryOrder

8.1315.2.8 uint8_t wds_UMTSMinQoS::resBerRatio

8.1315.2.9 uint8_t wds_UMTSMinQoS::sduErrorRatio

8.1315.2.10 uint8_t wds_UMTSMinQoS::trafficClass

8.1315.2.11 uint32_t wds_UMTSMinQoS::trafficPriority

8.1315.2.12 uint32_t wds_UMTSMinQoS::transferDelay

8.1316 wdsDhcpv4HwConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.1316.1 Detailed Description

Structure contain DHCP V4 Hardware Configuration.

Parameters

hwType	DHCP HW Type, examples: • 0 - Ethernet • 20 - Serial

chaddrLen	Length of chaddr field, examples: • 6 for Ethernet MAC address
chaddr	Client hardware address

8.1316.2 Field Documentation

8.1316.2.1 uint8_t wdsDhcpv4HwConfig::chaddr[16]

8.1316.2.2 uint8_t wdsDhcpv4HwConfig::chaddrLen

8.1316.2.3 uint8_t wdsDhcpv4HwConfig::hwType

8.1317 wdsDhcpv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.1317.1 Detailed Description

This structure contains DHCP V4 Option information.

Parameters

optCode	Option code • 0 - 255
optValLen	Option value length • 0 - 255
optVal	Option Value

8.1317.2 Field Documentation

8.1317.2.1 uint8_t wdsDhcpv4Option::optCode

8.1317.2.2 uint8_t wdsDhcpv4Option::optVal[255]

8.1317.2.3 uint8_t wdsDhcpv4Option::optValLen

8.1318 wdsDhcpv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wdsDhcpv4Option * pOptList

8.1318.1 Detailed Description

Structure contain DHCP V4 Option List

Parameters

numOpt	number of options • 0 - 255
pOptList	pointer to list of DHCP Options • See wdsDhcpv4Option

8.1318.2 Field Documentation

8.1318.2.1 uint8_t wdsDhcpv4OptionList::numOpt

8.1318.2.2 wdsDhcpv4Option* wdsDhcpv4OptionList::pOptList

8.1319 wdsDhcpv4ProfileId Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileId

8.1319.1 Detailed Description

This structure contains DHCP v4 ProfielID Information.

Parameters

profileType	profile type • 0 - 3GPP
profileId	profile index • index identifiying the profile 1-24 valid for 3GPP profile type (EM7430/EM7455 and onwards)

8.1319.2 Field Documentation

8.1319.2.1 uint8_t wdsDhcpv4ProfileId::profileId

8.1319.2.2 uint8_t wdsDhcpv4ProfileId::profileType

Chapter 9

File Documentation

9.1 apdoxypages.c File Reference

Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages.

Namespaces

- Tables

9.1.1 Detailed Description

Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages.

9.2 audio.h File Reference

Data Structures

- struct pack_audio_SLQSGetAudioProfile_t
- struct unpack_audio_SLQSGetAudioProfile_t
- struct pack_audio_SLQSSetAudioProfile_t
- struct pack_audio_SLQSGetAudioPathConfig_t
- struct audio_TXPCMIIRFltr
- struct audio_RXPCMIIRFltr
- struct audio_RXAGCList
- struct audio_RXAVCList
- struct audio_TXAGCList
- struct unpack_audio_SLQSGetAudioPathConfig_t
- struct pack_audio_SLQSSetAudioPathConfig_t
- struct pack_audio_SLQSGetAudioVolTLBConfig_t
- struct unpack_audio_SLQSGetAudioVolTLBConfig_t
- struct pack_audio_SLQSSetAudioVolTLBConfig_t
- struct unpack_audio_SLQSSetAudioVolTLBConfig_t

Functions

- int pack_audio_SLQSGetAudioProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_audio_SLQSGetAudioProfile_t *pReqParam)
- int unpack_audio_SLQSGetAudioProfile (uint8_t *pResp, uint16_t respLen, unpack_audio_SLQSGetAudioProfile_t *pOutput)
- int pack_audio_SLQSSetAudioProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_audio_SLQSSetAudioProfile_t *pReqParam)
- int unpack_audio_SLQSSetAudioProfile (uint8_t *pResp, uint16_t respLen)
- int pack_audio_SLQSGetAudioPathConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_audio_SLQSGetAudioPathConfig_t *pReqParam)
- int unpack_audio_SLQSGetAudioPathConfig (uint8_t *pResp, uint16_t respLen, unpack_audio_SLQSGetAudioPathConfig_t *pOutput)
- int pack_audio_SLQSSetAudioPathConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_audio_SLQSSetAudioPathConfig_t *pReqParam)
- int unpack_audio_SLQSSetAudioPathConfig (uint8_t *pResp, uint16_t respLen)
- int pack_audio_SLQSGetAudioVolTLBConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_audio_SLQSGetAudioVolTLBConfig_t *pReqParam)
- int unpack_audio_SLQSGetAudioVolTLBConfig (uint8_t *pResp, uint16_t respLen, unpack_audio_SLQSGetAudioVolTLBConfig_t *pOutput)
- int pack_audio_SLQSSetAudioVolTLBConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_audio_SLQSSetAudioVolTLBConfig_t *pReqParam)
- int unpack_audio_SLQSSetAudioVolTLBConfig (uint8_t *pResp, uint16_t respLen, unpack_audio_SLQSSetAudioVolTLBConfig_t *pOutput)

9.2.1 Function Documentation

9.2.1.1 int pack_audio_SLQSGetAudioPathConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_audio_SLQSGetAudioPathConfig_t * *pReqParam*)

Gets the audio path configuration parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: SL9090

9.2.1.2 int pack_audio_SLQSGetAudioProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_audio_SLQSGetAudioProfile_t * *pReqParam*)

Gets get the profile content of the requested audio generator. pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: SL9090

9.2.1.3 int pack_audio_SLQSGetAudioVolTLBConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_audio_SLQSGetAudioVolTLBConfig_t * *pReqParam*)

Gets the audio VolTLB configuration parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: SL9090

9.2.1.4 int pack_audio_SLQSSetAudioPathConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_audio_SLQSSetAudioPathConfig_t * *pReqParam*)

sets the audio path configuration parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: SL9090

9.2.1.5 int pack_audio_SLQSSetAudioProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_audio_SLQSSetAudioProfile_t * *pReqParam*)

sets an audio profile pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: SL9090

9.2.1.6 int pack_audio_SLQSSetAudioVoITLBConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_audio_SLQSSetAudioVoITLBConfig_t * *pReqParam*)

Sets the audio VoITLB configuration parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: SL9090

9.2.1.7 int unpack_audio_SLQSGetAudioPathConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_audio_SLQSGetAudioPathConfig_t * *pOutput*)

Gets the audio path configuration parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.1.8 int unpack_audio_SLQSGetAudioProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_audio_SLQSGetAudioProfile_t * *pOutput*)

Gets get the profile content of the requested audio generator unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.1.9 int unpack_audio_SLQSGetAudioVoITLBConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_audio_SLQSGetAudioVoITLBConfig_t * *pOutput*)

Gets the audio VoITLB configuration parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.1.10 int unpack_audio_SLQSSetAudioPathConfig (uint8_t * *pResp*, uint16_t *respLen*)

sets the audio path configuration parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.1.11 int unpack_audio_SLQSSetAudioProfile (uint8_t * *pResp*, uint16_t *respLen*)

sets an audio profile unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.1.12 int unpack_audio_SLQSSetAudioVoITLBConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_audio_SLQSSetAudioVoITLBConfig_t * *pOutput*)

Sets the audio VoITLB configuration parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3 BasicConnectDeviceService.h File Reference

Data Structures

- struct MbimEventEntry

Typedefs

- typedef struct MbimEventEntry MbimEventEntry

Functions

- int BasicConnectDeviceService_DeviceServiceSubscribeSetSend (MbimTransaction *pTransaction, MbimTransport *pTransport, uint32_t elementCount, MbimEventEntry *elements)
- int BasicConnectDeviceService_DeviceServiceSubscribeParse (uint8_t *informationBuffer, uint32_t informationBufferLength, uint32_t *pElementCount, MbimEventEntry *elements)
- const uint8_t * BasicConnectDeviceService_Uuid ()

9.3.1 Typedef Documentation

9.3.1.1 typedef struct MbimEventEntry MbimEventEntry

This structure represents the command IDs (CIDs) of indications related to a specific MBIM device service for which a user will be informed.

Parameters

DeviceServiceId	• The MBIM device service's UUID which relates to the indications.
CidCount	• Number of CIDs for which indications are to be generated. 0 means all CIDs of that MBIM device service.
Cids	• Fixed size array of CIDs. The first 'CidCount' elements of the array will be passed to the device, upto a maximum of MBIM_MAX_CIDS.

9.3.2 Function Documentation

9.3.2.1 int BasicConnectDeviceService_DeviceServiceSubscribeParse (uint8_t * *informationBuffer*, uint32_t *informationBufferLength*, uint32_t * *pElementCount*, MbimEventEntry * *elements*)

Parse a MBIM_CID_DEVICE_SERVICE_SUBSCRIBE_LIST response to an array of MbimEventEntry elements.

Parameters

in	*informationBuffer*	Response's information data.
in	*information-BufferLength*	Number of bytes of information in response.
in,out	*pElementCount*	On input, max number of elements which will be filled. On output, actual number of elements filled by parsing response.
in	*elements*	Array of elements to be written to by parsing response.

Returns

0 on success, $<$ 0 on failure.

9.3.2.2 int BasicConnectDeviceService_DeviceServiceSubscribeSetSend (MbimTransaction * *pTransaction*, MbimTransport * *pTransport*, uint32_t *elementCount*, MbimEventEntry * *elements*)

Send a MBIM_CID_DEVICE_SERVICE_SUBSCRIBE_LIST command to the device service and return immediately.

Parameters

in	*pTransaction*	Transaction object which will track the command/response.
in	*pTransport*	MBIM transport object which will carry out the command.
in	*elementCount*	Number of MbimEventEntry elements in the command.
in	*elements*	Array of elements.

Returns

0 on success, $<$ 0 on failure.

Note

pTransaction must have been initialized with MbimTransaction_Initialize prior to this call.

9.3.2.3 const uint8_t* BasicConnectDeviceService_Uuid ( )

Returns the UUID for the MBIM Basic Connect device service.

Returns

UUID

9.4 cat.h File Reference

Data Structures

- struct pack_cat_SetCATEventCallback_t
- struct unpack_cat_SetCATEventCallback_t
- struct cat_EventIDDataTlv
- struct cat_AlPhaIdentifierTlv
- struct cat_EventListTlv
- struct cat_RefreshTlv
- struct cat_EndProactiveSessionTlv
- union cat_currentCatEvent

- struct cat_commonEventTlv
- struct unpack_cat_SetCatEventCallback_ind_t
- struct pack_cat_CATSendEnvelopeCommand_t
- struct pack_cat_CATSendTerminalResponse_t

Macros

- #define CAT_EVENT_DATA_MAX_LENGTH 255
- #define CAN_COMMON_EVENT_TLV_NUMBER 11

Typedefs

- typedef unpack_result_t unpack_cat_CATSendEnvelopeCommand_t
- typedef unpack_result_t unpack_cat_CATSendTerminalResponse_t

Functions

- int pack_cat_SetCATEventCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_cat_SetCATEventCallback_t *reqArg)
- int unpack_cat_SetCATEventCallback (uint8_t *pResp, uint16_t respLen, unpack_cat_SetCATEventCallback_t *pOutput)
- int unpack_cat_SetCatEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_cat_SetCatEventCallback_ind_t *pOutput)
- int pack_cat_CATSendEnvelopeCommand (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_cat_CATSendEnvelopeCommand_t *reqArg)
- int unpack_cat_CATSendEnvelopeCommand (uint8_t *pResp, uint16_t respLen, unpack_cat_CATSendEnvelopeCommand_t *pOutput)
- int pack_cat_CATSendTerminalResponse (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_cat_CATSendTerminalResponse_t *reqArg)
- int unpack_cat_CATSendTerminalResponse (uint8_t *pResp, uint16_t respLen, unpack_cat_CATSendTerminalResponse_t *pOutput)

9.4.1 Macro Definition Documentation

9.4.1.1 #define CAN_COMMON_EVENT_TLV_NUMBER 11

9.4.1.2 #define CAT_EVENT_DATA_MAX_LENGTH 255

9.4.2 Typedef Documentation

9.4.2.1 typedef unpack_result_t unpack_cat_CATSendEnvelopeCommand_t

9.4.2.2 typedef unpack_result_t unpack_cat_CATSendTerminalResponse_t

9.4.3 Function Documentation

9.4.3.1 int pack_cat_CATSendEnvelopeCommand (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_cat_CATSendEnvelopeCommand_t * *reqArg*)

Sends the envelope command to the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.3.2 int pack_cat_CATSendTerminalResponse (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_cat_CATSendTerminalResponse_t * *reqArg*)

Sends the terminal response to the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter •

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.3.3 int pack_cat_SetCATEventCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_cat_SetCATEventCallback_t * *reqArg*)

Enables/disables the CAT event callback pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.3.4 int unpack_cat_CATSendEnvelopeCommand (uint8_t * *pResp*, uint16_t *respLen*, unpack_cat_CATSendEnvelopeCommand_t * *pOutput*)

Sends the envelope command to the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.3.5 int unpack_cat_CATSendTerminalResponse (uint8_t * *pResp*, uint16_t *respLen*, unpack_cat_CATSendTerminalResponse_t * *pOutput*)

Sends the terminal response to the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.3.6 int unpack_cat_SetCATEventCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_cat_SetCATEventCallback_t * *pOutput*)

Enables/disables the CAT event callback unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.3.7 int unpack_cat_SetCatEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_cat_SetCatEventCallback_ind_t * *pOutput*)

CAT event indication unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.5 common.h File Reference

Data Structures

- struct pack_qmi_t
- struct unpack_qmi_t
- struct unpack_result_t

Macros

- #define SDU_HDR_LEN (3)
- #define MINREQBKLEN (2048)
- #define MSGID_AND_LEN (4)
- #define MSGID_DONT_CARE (0xffff)
- #define UNUSEDPARAM(x) (void)x
- #define DEAULT_LOC_TIMEOUT_IN_SEC 2
- #define SDK_VALIDATE_INPUT_PACK_PARAM(pCtx, pBuf, pLen)
- #define SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID(pCtx, pBuf, pLen)
- #define SDK_VALIDATE_INPUT_UNPACK_PARAM(pResp, respLen, pOutput)
- #define SWI_INIT_UNPACK_RESULT_VALUE {0, SWI_UINT256_INT_VALUE}

Enumerations

- enum eLOG_LEVEL {
  eLOG_INFO,
  eLOG_DEBUG,
  eLOG_WARN,
  eLOG_FATAL }

- enum eTimeout {
  eTIMEOUT_2_S = 2000,
  eTIMEOUT_5_S = 5000,
  eTIMEOUT_8_S = 8000,
  eTIMEOUT_10_S = 10000,
  eTIMEOUT_20_S = 20000,
  eTIMEOUT_30_S = 30000,
  eTIMEOUT_60_S = 60000,
  eTIMEOUT_300_S = 300000,
  eTIMEOUT_DEFAULT = eTIMEOUT_8_S }
- enum eQMI_SVC {
  eCTL =0,
  eWDS,
  eDMS,
  eNAS =3,
  eQOS,
  eSMS =5,
  ePDS =0x06,
  eVOICE =0x09,
  eCAT =0x0A,
  eUIM =0x0B,
  eLOC =0x10,
  eSAR =0x11,
  eIMS =0x12,
  eTMD =0x18,
  eWDA =0x1A,
  eIMSA =0x21,
  eDSD =0x2A,
  eRMS =225,
  eSWIOMA =240,
  eAUDIO =241,
  eSWIM2MCMD =243,
  eSWIM2MCMD_AVC2 =0xfd,
  eSWIOMAEXT =244,
  eSWIAUDIO = 245,
  eSWILOC =246,
  eSWIDMS =254 }
- enum msgtype {
  eREQ =0,
  eRSP =2,
  eIND =4 }

Functions

- uint16_t helper_get_xid (uint8_t *qmi_resp)
- uint16_t helper_get_error_code (uint8_t *qmi_resp)
- char * helper_get_error_reason (uint16_t retVal)
- const char * helper_get_resp_ctx (uint8_t svc, uint8_t *pbuf, uint32_t len, unpack_qmi_t *pCtx)
- const char * helper_get_req_str (uint8_t svc, uint8_t *req, uint32_t len)
- unsigned unpack_result_code_only (uint8_t *pMdmResp)
- int helper_set_log_func (logger *func)
- void liteqmi_log (uint8_t lvl, const char *fmt,...)
- int helper_set_log_lvl (uint8_t lvl)
- void fill_sdu_hdr (pack_qmi_t *pCtx, uint8_t *pReqBuf)
- void fill_pack_ctx (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t svc, int timeout)
- char * get_version ()
- char * liteqmi_GetVersion ()

- void liteqmi_helper_decode7bitAsciiEncString (uint8_t *encoded_str, uint8_t encoded_len, uint8_t *decoded_str)
- int helper_isBootLoader_DebugEnabled (const char *szPath, const char *pQsn)

Variables

- logger * glog
- uint8_t gloglvl

9.5.1 Macro Definition Documentation

9.5.1.1 #define DEAULT_LOC_TIMEOUT_IN_SEC 2

9.5.1.2 #define MINREQBKLEN (2048)

9.5.1.3 #define MSGID_AND_LEN (4)

9.5.1.4 #define MSGID_DONT_CARE (0xffff)

9.5.1.5 #define SDK_VALIDATE_INPUT_PACK_PARAM(*pCtx, pBuf, pLen*)

Value:

```
if ((pCtx == NULL) || (pBuf == NULL) || (pLen == NULL) ) \
        {\
        liteqmi_log(eLOG_DEBUG,"[  pack] %s parameter NULL\n",__func__); \
        return eQCWWAN_ERR_INVALID_ARG; \
        }
```

9.5.1.6 #define SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID(*pCtx, pBuf, pLen*)

Value:

```
if ((pCtx == NULL) || (pBuf == NULL) || (pLen == NULL) ) \
        {\
        liteqmi_log(eLOG_DEBUG,"[  pack] %s parameter NULL\n",__func__); \
        return eQCWWAN_ERR_INVALID_ARG; \
        } \
    if (pCtx->xid == 0) \
       return eQCWWAN_ERR_INVALID_XID; \
    pBuf[0] = eREQ; \
    pBuf[1] = pCtx->xid & 0xff; \
    pBuf[2] = pCtx->xid >> 8;
```

9.5.1.7 #define SDK_VALIDATE_INPUT_UNPACK_PARAM(*pResp, respLen, pOutput*)

Value:

```
if ((pResp == NULL) || (respLen == 0) || (pOutput == NULL) ) \
        {\
        liteqmi_log(eLOG_DEBUG,"[ unpack] %s parameter NULL or invalid\n",
      __func__); \
        return eQCWWAN_ERR_INVALID_ARG; \
        }
```

9.5.1.8 #define SDU_HDR_LEN (3)

9.5.1.9 #define SWI_INIT_UNPACK_RESULT_VALUE {0, SWI_UINT256_INT_VALUE}

9.5.1.10 #define UNUSEDPARAM(*x*) (void)x

9.5.2 Enumeration Type Documentation

9.5.2.1 enum eLOG_LEVEL

log levels

Enumerator

eLOG_INFO
eLOG_DEBUG
eLOG_WARN
eLOG_FATAL

9.5.2.2 enum eQMI_SVC

qmi service

Enumerator

eCTL
eWDS
eDMS
eNAS
eQOS
eSMS
ePDS
eVOICE
eCAT
eUIM
eLOC
eSAR
eIMS
eTMD
eWDA
eIMSA
eDSD
eRMS
eSWIOMA
eAUDIO
eSWIM2MCMD
eSWIM2MCMD_AVC2
eSWIOMAEXT
eSWIAUDIO
eSWILOC
eSWIDMS

9.5.2.3 enum eTimeout

eTimeout

Enumerator

eTIMEOUT_2_S

eTIMEOUT_5_S

eTIMEOUT_8_S

eTIMEOUT_10_S

eTIMEOUT_20_S

eTIMEOUT_30_S

eTIMEOUT_60_S

eTIMEOUT_300_S

eTIMEOUT_DEFAULT

9.5.2.4 enum msgtype

qmi message type

Enumerator

eREQ

eRSP

eIND

9.5.3 Function Documentation

9.5.3.1 void fill_pack_ctx (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* uint8_t *svc,* int *timeout*)

9.5.3.2 void fill_sdu_hdr (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf*)

9.5.3.3 char* get_version ( )

Returns

version string

9.5.3.4 uint16_t helper_get_error_code (uint8_t * *qmi_resp*)

9.5.3.5 char* helper_get_error_reason (uint16_t *retVal*)

9.5.3.6 const char* helper_get_req_str (uint8_t *svc,* uint8_t * *req,* uint32_t *len*)

extract msgid string from modem req

Parameters

in	*svc*	qmi service
in	*req*	qmi request
in	*len*	request length

Returns

qmi message string

9.5.3.7 const char* helper_get_resp_ctx (uint8_t *svc*, uint8_t * *pbuf*, uint32_t *len*, unpack_qmi_t * *pCtx*)

extract msgid/xid/type from modem reply

Parameters

in	*svc*	qmi service
in	*pbuf*	qmi response/indication
in	*len*	response/indication length
out	*pCtx*	unpacked context

Returns

qmi message string

9.5.3.8 uint16_t helper_get_xid (uint8_t * *qmi_resp*)

9.5.3.9 int helper_isBootLoader_DebugEnabled (const char * *szPath*, const char * *pQsn*)

Parameters

in	*szPath*	the full path of the provided bootloader or firmware file
in	*pQsn*	the QSN of a specific module user wants to check, it is a 4 bytes hex value without the prefix 0X. It can be retrieved by AT command AT!ENTERCND="<password>", AT!SECBOOTCFG?, the first parameter of the response of AT!SECBOOTCFG?.

Returns

values listed below

- -1 bootloader is not debug enabled
- 0 unbale to retrieve the bootloader debug info
- 1 bootloader is debug enabled for the provided QSN
- 4 either szPath or pQsn is invalid (NULL)
- 18 invalid file

9.5.3.10 int helper_set_log_func (logger * *func*)

set log function

9.5.3.11 int helper_set_log_lvl (uint8_t *lvl*)

set log level

9.5.3.12 char* liteqmi_GetVersion ( )

Returns

version string

9.5.3.13 void liteqmi_helper_decode7bitAsciiEncString (uint8_t * *encoded_str,* uint8_t *encoded_len,* uint8_t * *decoded_str*)

decode 7bit encoded string

Parameters

in	*encoded_str*	7 bit encoded ASCII
in	*encoded_len*	encoded length
out	*decoded_str*	decoded ASCII string

9.5.3.14 void liteqmi_log (uint8_t *lvl,* const char * *fmt,* ...)

9.5.3.15 unsigned unpack_result_code_only (uint8_t * *pMdmResp*)

common handler for unpacking response with TLV type 0x02 only

9.5.4 Variable Documentation

9.5.4.1 logger* glog

9.5.4.2 uint8_t gloglvl

9.6 ctl.h File Reference

Data Structures

- struct pack_ctl_GetClientId_t
- struct unpack_ctl_GetClientId_t
- struct pack_ctl_ReleaseClientId_t
- struct unpack_ctl_ReleaseClientId_t

Functions

- void helper_get_ctl_resp_ctx (uint8_t *rsp, uint32_t len, unpack_qmi_t *pCtx)
- int pack_ctl_GetClientId (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ctl_GetClientId_t *reqArg)
- int unpack_ctl_GetClientId (uint8_t *pResp, uint16_t respLen, unpack_ctl_GetClientId_t *pOutput)
- int pack_ctl_ReleaseClientId (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ctl_ReleaseClientId_t *reqArg)
- int unpack_ctl_ReleaseClientId (uint8_t *pResp, uint16_t respLen, unpack_ctl_ReleaseClientId_t *pOutput)

9.6.1 Function Documentation

9.6.1.1 void helper_get_ctl_resp_ctx (uint8_t * *rsp,* uint32_t *len,* unpack_qmi_t * *pCtx*)

9.6.1.2 int pack_ctl_GetClientId (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* pack_ctl_GetClientId_t * *reqArg*)

Get Client ID pack Gets a client ID for a given QMI service.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.6.1.3 int pack_ctl_ReleaseClientId (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_ctl_ReleaseClientId_t * *reqArg*)

Release Client ID pack. Releases a client ID for a given QMI service.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.6.1.4 int unpack_ctl_GetClientId (uint8_t * *pResp*, uint16_t *respLen*, unpack_ctl_GetClientId_t * *pOutput*)

Get Client ID unpack. Gets a client ID for a given QMI service.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.6.1.5 int unpack_ctl_ReleaseClientId (uint8_t * *pResp,* uint16_t *respLen,* unpack_ctl_ReleaseClientId_t * *pOutput*)

Release Client ID unpack. Releases a client ID for a given QMI service.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7 CtlService.h File Reference

Data Structures

- struct CtlService

Typedefs

- typedef struct CtlService CtlService

Functions

- void CtlService_Initialize (CtlService *pThis, struct QmuxTransport *pTransport)
- void CtlService_ShutDown (CtlService *pThis)
- int CtlService_GetClientId (CtlService *pThis, uint8_t serviceType, uint8_t *clientId)
- int CtlService_ReleaseClientId (CtlService *pThis, uint8_t serviceType, uint8_t clientId)
- int CtlService_InitializeRegularService (CtlService *pThis, QmiService *pRegularService, uint8_t serviceType, QMI_INDICATION_CALLBACK pIndicationCallback, void *pIndicationCallbackContext)
- int CtlService_ShutDownRegularService (CtlService *pThis, QmiService *pRegularService)

9.7.1 Typedef Documentation

9.7.1.1 typedef struct CtlService CtlService

This structure abstracts a CTL QMI service. It wraps a QMI service which is initialized to type CTL. Synchronous methods are provided for getting and releasing client IDs on behalf of other (regular) QMI services.

Parameters

service	• QMI service object, initialized to be of type CTL.

9.7.2 Function Documentation

9.7.2.1 int CtlService_GetClientId (CtlService * pThis, uint8_t serviceType, uint8_t * clientId)

Get a client ID for a given QMI service type.

Parameters

in	*pThis*	The primary object of this call.
in	*serviceType*	QMI service type.
out	*clientId*	Returned client ID.

Returns

0 on success, $<$ 0 on failure.

9.7.2.2 void CtlService_Initialize (CtlService * pThis, struct QmuxTransport * pTransport)

Initialize CTL service.

Parameters

in	*pThis*	The primary object of this call.
in	*pTransport*	Pointer to underlying transport layer which is responsible for sending and receiving QMUX packets

9.7.2.3 int CtlService_InitializeRegularService (CtlService * pThis, QmiService * pRegularService, uint8_t serviceType, QMI_INDICATION_CALLBACK pIndicationCallback, void * pIndicationCallbackContext)

Convenience function to initialize a QMI service object. Under the hood, it gets a client ID for a given QMI service type.

Parameters

in	*pThis*	The primary object of this call.
in	*pRegularService*	Uninitialized QMI service object.
in	*serviceType*	QMI service type.
in	*pIndication-Callback*	User function to be be called when QMI indications are received by the service.
in	*pIndication-CallbackContext*	A user context for callback.

Returns

0 on success, $<$ 0 on failure.

9.7.2.4 int CtlService_ReleaseClientId (CtlService * pThis, uint8_t serviceType, uint8_t clientId)

Release a client ID for a given QMI service type. Client ID will have beeb previously obtained by invoking CtlService-_GetClientId.

Parameters

in	*pThis*	The primary object of this call.
in	*serviceType*	QMI service type.
in	*clientId*	Client ID to be released.

Returns

0 on success, < 0 on failure.

9.7.2.5 void CtlService_ShutDown (CtlService * *pThis*)

Shut down CTL service.

Parameters

in	*pThis*	The primary object of this call.

9.7.2.6 int CtlService_ShutDownRegularService (CtlService * *pThis*, QmiService * *pRegularService*)

Convenience function to shuit down a QMI service object. Under the hood, it releases a client ID for a given QMI service type.

Parameters

in	*pThis*	The primary object of this call.
in	*pRegularService*	Previously initialized QMI service object.

Returns

0 on success, < 0 on failure.

9.8 DirectLayer.h File Reference

Functions

- void * DirectLayer_Create (QmuxTransport *pOwner, char *devicePath, void *pInterfaceSpecific)
- void DirectLayer_ShutDown (void *pLayer)
- int DirectLayer_SendQmuxPacket (void *pLayer, uint8_t qmiServiceType, uint8_t qmiClientId, uint16_t qmiTransactionId, uint8_t *qmuxPacket, uint16_t qmuxPacketSize)
- int DirectLayer_GetClientID (QmuxTransport *pOwner, uint8_t serviceType)
- int DirectLayer_GetDeviceFD (QmuxTransport *pOwner)
- uint8_t DriectLay_GetDummyServiceClientID (QmuxTransport *pOwner, uint8_t serviceType)

9.8.1 Function Documentation

9.8.1.1 void* DirectLayer_Create (QmuxTransport * *pOwner*, char * *devicePath*, void * *pInterfaceSpecific*)

9.8.1.2 int DirectLayer_GetClientID (QmuxTransport * *pOwner*, uint8_t *serviceType*)

9.8.1.3 int DirectLayer_GetDeviceFD (QmuxTransport * *pOwner*)

9.8.1.4 int DirectLayer_SendQmuxPacket (void * *pLayer*, uint8_t *qmiServiceType*, uint8_t *qmiClientId*, uint16_t *qmiTransactionId*, uint8_t * *qmuxPacket*, uint16_t *qmuxPacketSize*)

9.8.1.5 void DirectLayer_ShutDown (void * *pLayer*)

9.8.1.6 uint8_t DriectLay_GetDummyServiceClientID (QmuxTransport * *pOwner*, uint8_t *serviceType*)

9.9 dms.h File Reference

Data Structures

- struct unpack_dms_GetModelID_t
- struct unpack_dms_GetIMSI_t
- struct unpack_dms_GetFirmwareInfo_t
- struct unpack_dms_GetPower_t
- struct unpack_dms_GetSerialNumbers_t
- struct unpack_dms_GetHardwareRevision_t
- struct unpack_dms_SLQSGetBandCapability_t
- struct dms_LteBandsSupport
- struct unpack_dms_SLQSGetBandCapabilityExt_t
- struct unpack_dms_GetDeviceCapabilities_t
- struct dms_devCaps
- struct dms_devSubsCfgList
- struct dms_devMultiSimCaps
- struct dms_devMultiSimVoiceDataCaps
- struct dms_devCurSubsCaps
- struct dms_devSubsVoiceDataList
- struct dms_devSubsVoiceDataCaps
- struct dms_devSubsFeatureModeCaps
- struct dms_devMaxSubsCaps
- struct dms_devSubsList
- struct dms_devMaxCfgListCaps
- struct unpack_dms_GetDeviceCapabilitiesV2_t
- struct unpack_dms_GetFirmwareRevisions_t
- struct unpack_dms_GetFirmwareRevision_t
- struct unpack_dms_GetDeviceSerialNumbers_t
- struct unpack_dms_GetPRLVersion_t
- struct unpack_dms_GetNetworkTime_t
- struct unpack_dms_GetNetworkTimeV2_t
- struct unpack_dms_GetVoiceNumber_t
- struct unpack_dms_GetDeviceHardwareRev_t
- struct unpack_dms_GetFSN_t
- struct unpack_dms_GetDeviceCap_t
- struct pack_dms_SetPower_t
- struct unpack_dms_SetPower_t
- struct unpack_dms_GetBandCapability_t
- struct unpack_dms_GetUSBComp_t
- struct pack_dms_SetUSBComp_t
- struct unpack_dms_SetUSBComp_t
- struct pack_dms_SetCustFeature_t
- struct unpack_dms_SetCustFeature_t
- struct unpack_dms_GetCustFeature_t
- struct unpack_dms_SetFirmwarePreference_t
- struct unpack_dms_GetCrashAction_t
- struct pack_dms_SetCrashAction_t
- struct unpack_dms_SetCrashAction_t
- struct unpack_dms_GetDeviceMfr_t
- struct pack_dms_SetEventReport_t
- struct unpack_dms_SetEventReport_t
- struct dms_OperatingModeTlv
- struct dms_ActivationStatusTlv

- struct unpack_dms_SetEventReport_ind_t
- struct pack_dms_UIMGetICCID_t
- struct unpack_dms_UIMGetICCID_t
- struct pack_dms_SetCustFeaturesV2_t
- struct unpack_dms_SetCustFeaturesV2_t
- struct pack_dms_GetCustFeaturesV2_t
- struct DMSgetCustomInput
- struct DMScustSettingInfo
- struct DMScustSettingList
- struct DMSgetCustomFeatureV2
- struct unpack_dms_GetCustFeaturesV2_t
- struct unpack_dms_GetActivationState_t
- struct image_info_t
- struct unpack_dms_SLQSSwiGetFirmwareCurr_t
- struct pack_dms_SLQSSwiSetDyingGaspCfg_t
- struct unpack_dms_SLQSSwiSetDyingGaspCfg_t
- struct unpack_dms_SLQSSwiClearDyingGaspStatistics_t
- struct packgetDyingGaspStatistics
- struct unpack_dms_SLQSSwiGetDyingGaspStatistics_t
- struct packgetDyingGaspCfg
- struct unpack_dms_SLQSSwiGetDyingGaspCfg_t
- struct unpack_dms_SLQSDmsSwiGetResetInfo_t
- struct unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t
- struct pack_dms_SLQSDmsSwiIndicationRegister_t
- struct unpack_dms_SLQSDmsSwiIndicationRegister_t
- struct unpack_dms_SLQSSwiGetFwUpdateStatus_t
- struct unpack_dms_GetManufacturer_t
- struct unpack_dms_GetOfflineReason_t
- struct pack_dms_SetActivationStatusCallback_t
- struct unpack_dms_SetActivationStatusCallback_t
- struct pack_dms_UIMSetPINProtection_t
- struct unpack_dms_UIMSetPINProtection_t
- struct pack_dms_UIMUnblockPIN_t
- struct pack_dms_UIMVerifyPIN_t
- struct pack_dms_UIMChangePIN_t
- struct pack_dms_UIMGetControlKeyStatus_t
- struct unpack_dms_UIMGetControlKeyStatus_t
- struct unpack_dms_UIMGetPINStatus_t
- struct pack_dms_UIMSetControlKeyProtection_t
- struct unpack_dms_UIMSetControlKeyProtection_t
- struct pack_dms_UIMUnblockControlKey_t
- struct unpack_dms_UIMUnblockControlKey_t
- struct pack_dms_ResetToFactoryDefaults_t
- struct unpack_dms_ResetToFactoryDefaults_t
- struct pack_dms_ActivateAutomatic_t
- struct eriDataparams
- struct unpack_dms_SLQSGetERIFile_t
- struct unpack_dms_SLQSUIMGetState_t
- struct pack_dms_SLQSSwiGetCrashInfo_t
- struct crashInformation
- struct crashInfoParams
- struct unpack_dms_SLQSSwiGetCrashInfo_t
- struct unpack_dms_SLQSSwiGetHostDevInfo_t
- struct pack_dms_SLQSSwiSetHostDevInfo_t
- struct unpack_dms_SLQSSwiSetHostDevInfo_t

- struct unpack_dms_SLQSSwiGetOSInfo_t
- struct pack_dms_SLQSSwiSetOSInfo_t
- struct unpack_dms_SLQSSwiSetOSInfo_t
- struct unpack_dms_SLQSSwiGetSerialNoExt_t
- struct unpack_dms_SLQSDmsSwiGetPCInfo_t
- struct dms_PSMEnableStateTlv
- struct dms_PSMDurationThresholdTlv
- struct dms_PSMDurationDueToOOSTlv
- struct dms_PSMRandomizationWindowTlv
- struct dms_PSMActiveTimerTlv
- struct dms_PSMPeriodicUpdateTimerTlv
- struct dms_PSMEarlyWakeupTimeTlv
- struct unpack_dms_SLQSGetPowerSaveModeConfig_t
- struct pack_dms_SLQSSetPowerSaveModeConfig_t
- struct unpack_dms_SLQSSetPowerSaveModeConfig_t
- struct pack_dms_SetIndicationRegister_t
- struct unpack_dms_SetIndicationRegister_t
- struct dms_PSMEnableStateIndTlv
- struct dms_PSMActiveTimerIndTlv
- struct dms_PSMPeriodicUpdateTimerIndTlv
- struct unpack_dms_PSMCfgChange_ind_t
- struct pack_dms_SwiSetEventReport_t
- struct unpack_dms_SwiSetEventReport_t
- struct dms_TemperatureTlv
- struct dms_VoltageTlv
- struct dms_UimStatusTlv
- struct unpack_dms_SwiEventReportCallBack_ind_t
- struct pack_dms_SwiUimSelect_t
- struct unpack_dms_SwiUimSelect_t
- struct dms_UimAutoSwitchActSlotTlv
- struct unpack_dms_SLQSDmsSwiGetUimSelection_t

Macros

- #define DMS_UINT8_MAX_STRING_SZ 255
- #define DMS_MAX_CUST_ID_LEN 64
- #define DMS_MAX_CUST_VALUE_LEN 8
- #define DMS_IMGDETAILS_LEN 16
- #define DMS_MAX_FWUPDATE_LOG_STR_SZ 255
- #define DMS_MAX_FWUPDATE_REF_STR_SZ 15
- #define DMS_VALID_FSN_LEN 14
- #define DMS_MAX_RADIO_IFCS_SIZE 255
- #define DMS_MAX_SUBS_CFG_LIST_SIZE 32
- #define DMS_MAX_SUBS_LIST_SIZE 32
- #define DMS_MAX_SUPPORTED_LTE_BANDS 255
- #define DMS_SLQSFWINFO_MODELID_SZ 20
- #define DMS_SLQSFWINFO_BOOTVERSION_SZ 85
- #define DMS_SLQSFWINFO_APPVERSION_SZ 85
- #define DMS_SLQSFWINFO_SKU_SZ 15
- #define DMS_SLQSFWINFO_PACKAGEID_SZ 85
- #define DMS_SLQSFWINFO_CARRIER_SZ 20
- #define DMS_SLQSFWINFO_PRIVERSION_SZ 16
- #define DMS_SLQSFWINFO_CUR_CARR_NAME 17
- #define DMS_SLQSFWINFO_CUR_CARR_REV 13

- #define MAX_BUILD_ID_LEN 255
- #define UNIQUE_ID_LEN 16
- #define SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160
- #define SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20
- #define SPC_SIZE 6
- #define CK_MAX_SIZE 8
- #define ACT_CODE_MAX_SIZE 81
- #define ERI_DATA_MAX_SIZE 1024
- #define MEID_MAX_SIZE 8
- #define DMS_PM_ONLINE 0x00 /∗ Online ∗/
- #define DMS_PM_LOW 0x01 /∗ Low Power ∗/
- #define DMS_PM_FACTORY 0x02 /∗ Factory Test Mode ∗/
- #define DMS_PM_OFFLINE 0x03 /∗ Offline ∗/
- #define DMS_PM_RESET 0x04 /∗ Reset ∗/
- #define DMS_PM_SHUT_DOWN 0x05 /∗ Shut Down ∗/
- #define DMS_PM_PERSISTENT_LOW 0x06 /∗ Persistent Low Power ∗/
- #define DMS_SET_REPORT_ENABLE 1
- #define DMS_SET_REPORT_DISABLE 0
- #define DMS_SWI_SET_IND_ENABLE 1
- #define DMS_SWI_SET_IND_DISABLE 0
- #define DMS_SET_REG_IND_ENABLE 1
- #define DMS_SET_REG_IND_DISABLE 0
- #define DMS_SET_REG_IND_NO_CHANGE 2

Functions

- int pack_dms_GetIMSI (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetIMSI (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetIMSI_t ∗pOutput)
- int pack_dms_GetModelID (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetModelID (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetModelID_t ∗pOutput)
- int pack_dms_GetFirmwareInfo (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetFirmwareInfo (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetFirmwareInfo_t ∗pOutput)
- int pack_dms_GetPower (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetPower (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetPower_t ∗pOutput)
- int pack_dms_GetSerialNumbers (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetSerialNumbers (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetSerialNumbers_t ∗pOutput)
- int pack_dms_GetHardwareRevision (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetHardwareRevision (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetHardwareRevision_t ∗pOutput)
- int pack_dms_SLQSGetBandCapability (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_SLQSGetBandCapability (uint8_t ∗pResp, uint16_t respLen, unpack_dms_SLQSGetBandCapability_t ∗pOutput)
- int unpack_dms_SLQSGetBandCapabilityExt (uint8_t ∗pResp, uint16_t respLen, unpack_dms_SLQSGetBandCapabilityExt_t ∗pOutput)
- int pack_dms_GetDeviceCapabilities (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetDeviceCapabilities (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetDeviceCapabilities_t ∗pOutput)
- int pack_dms_GetDeviceCapabilitiesV2 (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetDeviceCapabilitiesV2 (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetDeviceCapabilitiesV2_t ∗pOutput)
- int pack_dms_GetFirmwareRevisions (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetFirmwareRevisions (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetFirmwareRevisions_t ∗pOutput)

- int pack_dms_GetFirmwareRevision (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetFirmwareRevision (uint8_t *pResp, uint16_t respLen, unpack_dms_GetFirmwareRevision_t *pOutput)
- int pack_dms_GetDeviceSerialNumbers (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetDeviceSerialNumbers (uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceSerialNumbers_t *pOutput)
- int pack_dms_GetPRLVersion (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetPRLVersion (uint8_t *pResp, uint16_t respLen, unpack_dms_GetPRLVersion_t *pOutput)
- int pack_dms_GetNetworkTime (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetNetworkTime (uint8_t *pResp, uint16_t respLen, unpack_dms_GetNetworkTime_t *pOutput)
- int pack_dms_GetNetworkTimeV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetNetworkTimeV2 (uint8_t *pResp, uint16_t respLen, unpack_dms_GetNetworkTimeV2_t *pOutput)
- int pack_dms_GetVoiceNumber (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetVoiceNumber (uint8_t *pResp, uint16_t respLen, unpack_dms_GetVoiceNumber_t *pOutput)
- int pack_dms_GetDeviceHardwareRev (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetDeviceHardwareRev (uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceHardwareRev_t *pOutput)
- int pack_dms_GetFSN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetFSN (uint8_t *pResp, uint16_t respLen, unpack_dms_GetFSN_t *pOutput)
- int pack_dms_GetDeviceCap (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetDeviceCap (uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceCap_t *pOutput)
- int pack_dms_SetPower (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetPower_t *reqArg)
- int unpack_dms_SetPower (uint8_t *pResp, uint16_t respLen, unpack_dms_SetPower_t *pOutput)
- int pack_dms_GetBandCapability (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetBandCapability (uint8_t *pResp, uint16_t respLen, unpack_dms_GetBandCapability_t *pOutput)
- int pack_dms_GetUSBComp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetUSBComp (uint8_t *pResp, uint16_t respLen, unpack_dms_GetUSBComp_t *pOutput)
- int pack_dms_SetUSBComp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetUSBComp_t *reqArg)
- int unpack_dms_SetUSBComp (uint8_t *pResp, uint16_t respLen, unpack_dms_SetUSBComp_t *pOutput)
- int pack_dms_SetCustFeature (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCustFeature_t *reqArg)
- int unpack_dms_SetCustFeature (uint8_t *pResp, uint16_t respLen, unpack_dms_SetCustFeature_t *pOutput)
- int pack_dms_GetCustFeature (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetCustFeature (uint8_t *pResp, uint16_t respLen, unpack_dms_GetCustFeature_t *pOutput)
- int pack_dms_SetFirmwarePreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_SetFirmwarePreference (uint8_t *pResp, uint16_t respLen, unpack_dms_SetFirmwarePreference_t *pOutput)
- int pack_dms_GetCrashAction (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetCrashAction (uint8_t *pResp, uint16_t respLen, unpack_dms_GetCrashAction_t *pOutput)
- int pack_dms_SetCrashAction (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCrashAction_t reqArg)
- int unpack_dms_SetCrashAction (uint8_t *pResp, uint16_t respLen, unpack_dms_SetCrashAction_t *pOutput)
- int pack_dms_GetDeviceMfr (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetDeviceMfr (uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceMfr_t *pOutput)

- int pack_dms_SetEventReport (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetEventReport_t *reqArg)
- int unpack_dms_SetEventReport (uint8_t *pResp, uint16_t respLen, unpack_dms_SetEventReport_t *pOutput)
- int unpack_dms_SetEventReport_ind (uint8_t *pResp, uint16_t respLen, unpack_dms_SetEventReport_ind_t *pOutput)
- int pack_dms_UIMGetICCID (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMGetICCID_t *reqArg)
- int unpack_dms_UIMGetICCID (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetICCID_t *pOutput)
- int pack_dms_SetCustFeaturesV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCustFeaturesV2_t *reqArg)
- int unpack_dms_SetCustFeaturesV2 (uint8_t *pResp, uint16_t respLen, unpack_dms_SetCustFeaturesV2_t *pOutput)
- int pack_dms_GetCustFeaturesV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_GetCustFeaturesV2_t *reqArg)
- int unpack_dms_GetCustFeaturesV2 (uint8_t *pResp, uint16_t respLen, unpack_dms_GetCustFeaturesV2_t *pOutput)
- int pack_dms_GetActivationState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_GetActivationState (uint8_t *pResp, uint16_t respLen, unpack_dms_GetActivationState_t *pOutput)
- int pack_dms_SLQSSwiGetFirmwareCurr (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetFirmwareCurr (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetFirmwareCurr_t *pOutput)
- int pack_dms_SLQSSwiSetDyingGaspCfg (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetDyingGaspCfg_t *reqArg)
- int unpack_dms_SLQSSwiSetDyingGaspCfg (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetDyingGaspCfg_t *pOutput)
- int pack_dms_SLQSSwiClearDyingGaspStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiClearDyingGaspStatistics (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiClearDyingGaspStatistics_t *pOutput)
- int pack_dms_SLQSSwiGetDyingGaspStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetDyingGaspStatistics (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspStatistics_t *pOutput)
- int pack_dms_SLQSSwiGetDyingGaspCfg (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetDyingGaspCfg (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspCfg_t *pOutput)
- int pack_dms_SLQSDmsSwiGetResetInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSDmsSwiGetResetInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_t *pOutput)
- int unpack_dms_SLQSDmsSwiGetResetInfo_Ind (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t *pOutput)
- int pack_dms_SLQSDmsSwiIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSDmsSwiIndicationRegister_t *reqArg)
- int unpack_dms_SLQSDmsSwiIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiIndicationRegister_t *pOutput)
- int pack_dms_SLQSSwiGetFwUpdateStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetFwUpdateStatus (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetFwUpdateStatus_t *pOutput)
- int pack_dms_GetManufacturer (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_GetManufacturer (uint8_t *pResp, uint16_t respLen, unpack_dms_GetManufacturer_t *pOutput)
- int pack_dms_GetOfflineReason (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_GetOfflineReason (uint8_t *pResp, uint16_t respLen, unpack_dms_GetOfflineReason_t *pOutput)
- int pack_dms_SetActivationStatusCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetActivationStatusCallback_t *reqArg)

- int unpack_dms_SetActivationStatusCallback (uint8_t *pResp, uint16_t respLen, unpack_dms_SetActivationStatusCallback_t *pOutput)
- int pack_dms_UIMSetPINProtection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMSetPINProtection_t *pReq)
- int unpack_dms_UIMSetPINProtection (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)
- int pack_dms_UIMUnblockPIN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMUnblockPIN_t *pReq)
- int unpack_dms_UIMUnblockPIN (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)
- int pack_dms_UIMVerifyPIN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMVerifyPIN_t *pReq)
- int unpack_dms_UIMVerifyPIN (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)
- int pack_dms_UIMChangePIN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMChangePIN_t *pReq)
- int unpack_dms_UIMChangePIN (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)
- int pack_dms_UIMGetControlKeyStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMGetControlKeyStatus_t *pReq)
- int unpack_dms_UIMGetControlKeyStatus (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetControlKeyStatus_t *pOutput)
- int pack_dms_UIMGetPINStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_UIMGetPINStatus (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetPINStatus_t *pOutput)
- int pack_dms_UIMSetControlKeyProtection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMSetControlKeyProtection_t *pReq)
- int unpack_dms_UIMSetControlKeyProtection (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetControlKeyProtection_t *pOutput)
- int pack_dms_UIMUnblockControlKey (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMUnblockControlKey_t *pReq)
- int unpack_dms_UIMUnblockControlKey (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMUnblockControlKey_t *pOutput)
- int pack_dms_ResetToFactoryDefaults (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ResetToFactoryDefaults_t *pReq)
- int unpack_dms_ResetToFactoryDefaults (uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)
- int pack_dms_ValidateSPC (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ResetToFactoryDefaults_t *pReq)
- int unpack_dms_ValidateSPC (uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)
- int pack_dms_ActivateAutomatic (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ActivateAutomatic_t *pReq)
- int unpack_dms_ActivateAutomatic (uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)
- int pack_dms_SLQSGetERIFile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSGetERIFile (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSGetERIFile_t *pOutput)
- int pack_dms_SLQSUIMGetState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSUIMGetState (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSUIMGetState_t *pOutput)
- int pack_dms_SLQSSwiGetCrashInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiGetCrashInfo_t *pReq)
- int unpack_dms_SLQSSwiGetCrashInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetCrashInfo_t *pOutput)
- int pack_dms_SLQSSwiGetHostDevInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)

- int unpack_dms_SLQSSwiGetHostDevInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetHostDevInfo_t *pOutput)
- int pack_dms_SLQSSwiSetHostDevInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetHostDevInfo_t *pReq)
- int unpack_dms_SLQSSwiSetHostDevInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetHostDevInfo_t *pOutput)
- int pack_dms_SLQSSwiGetOSInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetOSInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetOSInfo_t *pOutput)
- int pack_dms_SLQSSwiSetOSInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetOSInfo_t *pReq)
- int unpack_dms_SLQSSwiSetOSInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetOSInfo_t *pOutput)
- int pack_dms_SLQSSwiGetSerialNoExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetSerialNoExt (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetSerialNoExt_t *pOutput)
- int pack_dms_SLQSDmsSwiGetPCInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_SLQSDmsSwiGetPCInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetPCInfo_t *pOutput)
- int pack_dms_SLQSGetPowerSaveModeConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_SLQSGetPowerSaveModeConfig (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSGetPowerSaveModeConfig_t *pOutput)
- int pack_dms_SLQSSetPowerSaveModeConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSetPowerSaveModeConfig_t *reqArg)
- int unpack_dms_SLQSSetPowerSaveModeConfig (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSetPowerSaveModeConfig_t *pOutput)
- int pack_dms_SetIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetIndicationRegister_t *reqArg)
- int unpack_dms_SetIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_dms_SetIndicationRegister_t *pOutput)
- int unpack_dms_PSMCfgChange_ind (uint8_t *pResp, uint16_t respLen, unpack_dms_PSMCfgChange_ind_t *pOutput)
- int pack_dms_SwiSetEventReport (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SwiSetEventReport_t *reqArg)
- int unpack_dms_SwiSetEventReport (uint8_t *pResp, uint16_t respLen, unpack_dms_SwiSetEventReport_t *pOutput)
- int unpack_dms_SwiEventReportCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_dms_SwiEventReportCallBack_ind_t *pOutput)
- int pack_dms_SwiUimSelect (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SwiUimSelect_t *reqArg)
- int unpack_dms_SwiUimSelect (uint8_t *pResp, uint16_t respLen, unpack_dms_SwiUimSelect_t *pOutput)
- int pack_dms_SLQSDmsSwiGetUimSelection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_SLQSDmsSwiGetUimSelection (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetUimSelection_t *pOutput)

9.9.1 Macro Definition Documentation

9.9.1.1 #define ACT_CODE_MAX_SIZE 81

9.9.1.2 #define CK_MAX_SIZE 8

9.9.1.3 #define DMS_IMGDETAILS_LEN 16

9.9.1.4 #define DMS_MAX_CUST_ID_LEN 64

9.9.1.5 #define DMS_MAX_CUST_VALUE_LEN 8

9.9.1.6 #define DMS_MAX_FWUPDATE_LOG_STR_SZ 255

9.9.1.7 #define DMS_MAX_FWUPDATE_REF_STR_SZ 15

9.9.1.8 #define DMS_MAX_RADIO_IFCS_SIZE 255

9.9.1.9 #define DMS_MAX_SUBS_CFG_LIST_SIZE 32

9.9.1.10 #define DMS_MAX_SUBS_LIST_SIZE 32

9.9.1.11 #define DMS_MAX_SUPPORTED_LTE_BANDS 255

9.9.1.12 #define DMS_PM_FACTORY 0x02 /∗ Factory Test Mode ∗/

9.9.1.13 #define DMS_PM_LOW 0x01 /∗ Low Power ∗/

9.9.1.14 #define DMS_PM_OFFLINE 0x03 /∗ Offline ∗/

9.9.1.15 #define DMS_PM_ONLINE 0x00 /∗ Online ∗/

9.9.1.16 #define DMS_PM_PERSISTENT_LOW 0x06 /∗ Persistent Low Power ∗/

9.9.1.17 #define DMS_PM_RESET 0x04 /∗ Reset ∗/

9.9.1.18 #define DMS_PM_SHUT_DOWN 0x05 /∗ Shut Down ∗/

9.9.1.19 #define DMS_SET_REG_IND_DISABLE 0

9.9.1.20 #define DMS_SET_REG_IND_ENABLE 1

9.9.1.21 #define DMS_SET_REG_IND_NO_CHANGE 2

9.9.1.22 #define DMS_SET_REPORT_DISABLE 0

9.9.1.23 #define DMS_SET_REPORT_ENABLE 1

9.9.1.24 #define DMS_SLQSFWINFO_APPVERSION_SZ 85

9.9.1.25 #define DMS_SLQSFWINFO_BOOTVERSION_SZ 85

9.9.1.26 #define DMS_SLQSFWINFO_CARRIER_SZ 20

9.9.1.27 #define DMS_SLQSFWINFO_CUR_CARR_NAME 17

9.9.1.28 #define DMS_SLQSFWINFO_CUR_CARR_REV 13

9.9.1.29 #define DMS_SLQSFWINFO_MODELID_SZ 20

9.9.1.30 #define DMS_SLQSFWINFO_PACKAGEID_SZ 85

9.9.1.31 #define DMS_SLQSFWINFO_PRIVERSION_SZ 16

9.9.1.32 #define DMS_SLQSFWINFO_SKU_SZ 15

9.9.1.33 #define DMS_SWI_SET_IND_DISABLE 0

9.9.1.34 #define DMS_SWI_SET_IND_ENABLE 1

9.9.1.35 #define DMS_UINT8_MAX_STRING_SZ 255

9.9.1.36 #define DMS_VALID_FSN_LEN 14

9.9.1.37 #define ERI_DATA_MAX_SIZE 1024

9.9.1.38 #define MAX_BUILD_ID_LEN 255

9.9.1.39 #define MEID_MAX_SIZE 8

9.9.1.40 #define SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160

9.9.1.41 #define SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20

9.9.1.42 #define SPC_SIZE 6

9.9.1.43 #define UNIQUE_ID_LEN 16

9.9.2 Function Documentation

9.9.2.1 int pack_dms_ActivateAutomatic (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ActivateAutomatic_t * *pReq*)

Requests the device to perform automatic service activation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.2 int pack_dms_GetActivationState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Activation State pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

This is only supported for 3GPP2 devices.

9.9.2.3 int pack_dms_GetBandCapability (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Band Capability pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.4 int pack_dms_GetCrashAction (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Crash Action pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.5 int pack_dms_GetCustFeature (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Custom Feature pack. This API is deprecated for EM7455/EM7430/MC7455/MC7430, please use pack_dms_-GetCustFeaturesV2() instead for EM7455/EM7430/MC7455/MC7430.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.6 int pack_dms_GetCustFeaturesV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_GetCustFeaturesV2_t * *reqArg*)

Get Custom Feature pack This function queries the modem for a list of supported features. This function is for firmware version 2.0 and newer. Currently supported Customization features:

- GPIOSARENABLE
- GPSSEL
- IMSWITCHHIDE
- IPV6ENABLE
- WAKEHOSTEN

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.7 int pack_dms_GetDeviceCap (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Device Capabilities pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.8 int pack_dms_GetDeviceCapabilities (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get device capability pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.9 int pack_dms_GetDeviceCapabilitiesV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get device capability pack v2

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.10 int pack_dms_GetDeviceHardwareRev (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Hardware Revision of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.11 int pack_dms_GetDeviceMfr (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Device Manufacture Name pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.12 int pack_dms_GetDeviceSerialNumbers (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Device Serial Number pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.13 int pack_dms_GetFirmwareInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get firmware info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.14 int pack_dms_GetFirmwareRevision (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Firmware Revision pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.15 int pack_dms_GetFirmwareRevisions (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Firmware Revisions pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.16 int pack_dms_GetFSN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get FSN pack. This API get the Factory Sequence Number of the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.17 int pack_dms_GetHardwareRevision (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get hardware revision of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.18 int pack_dms_GetIMSI (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get IMSI pack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_-xx_xx and EM7455/EM7430 firmware versions. Please use pack_uim_ReadTransparent()(EF ID: 3F00 7F20 6F07 for 2G card and 3F00 7FFF 6F07 for 3G card) instead for new firmware versions and new modules.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.19 int pack_dms_GetManufacturer (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get device manufacturer information pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.20 int pack_dms_GetModelID (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get model id pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.21 int pack_dms_GetNetworkTime (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Network Time pack Returns the current time of the device based on the value supported by the network.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.22 int pack_dms_GetNetworkTimeV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Network Time pack Returns the current time of the device based on the value supported by the network.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.23 int pack_dms_GetOfflineReason (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get operating mode offline reason pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.24 int pack_dms_GetPower (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get operating mode of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.25 int pack_dms_GetPRLVersion (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get PRL Versions pack. Return version of the active Preferred Roaming List (PRL) in use by the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.26 int pack_dms_GetSerialNumbers (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get serial numbers pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.27 int pack_dms_GetUSBComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get USB Comp pack This API queries the modem's USB interface configuration and supported configuration parameters.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.28 int pack_dms_GetVoiceNumber (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Voice Number pack. Returns the voice number in use by the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.29 int pack_dms_ResetToFactoryDefaults (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ResetToFactoryDefaults_t * *pReq*)

Resets to default factory settings of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.30 int pack_dms_SetActivationStatusCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetActivationStatusCallback_t * *reqArg*)

Set activation status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.31 int pack_dms_SetCrashAction (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetCrashAction_t *reqArg*)

Set Crash Action pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.32 int pack_dms_SetCustFeature (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetCustFeature_t * *reqArg*)

Set Custom Feature pack. This API is deprecated for EM7455/EM7430/MC7455/MC7430, please use pack_dms_SetCustFeaturesV2() instead for EM7455/EM7430/MC7455/MC7430.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.33 int pack_dms_SetCustFeaturesV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetCustFeaturesV2_t * *reqArg*)

Set Cust Features pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.34 int pack_dms_SetEventReport (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetEventReport_t * *reqArg*)

Set Event Report pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.35 int pack_dms_SetFirmwarePreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Set Firmware Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.36 int pack_dms_SetIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetIndicationRegister_t * *reqArg*)

Set Indication register pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.37 int pack_dms_SetPower (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetPower_t * *reqArg*)

Set Power pack Sets the operating mode of the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.38 int pack_dms_SetUSBComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetUSBComp_t * *reqArg*)

Set USB Comp pack This API is used to change the modem's USB interface configuration thus allowing a device to have multiple USB compositions. Devices will, by default, be configured to support a minimal set of interfaces to reduce end user modem installation time. Developers and some customers, however, require access to a custom set of interfaces. A reset is required for any change in the USB composition to take effect.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.39 int pack_dms_SLQSDmsSwiGetPCInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get device power control status information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.40 int pack_dms_SLQSDmsSwiGetResetInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get reset info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.41 int pack_dms_SLQSDmsSwiGetUimSelection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get device UIM Selection information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.42 int pack_dms_SLQSDmsSwiIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSDmsSwiIndicationRegister_t * *reqArg*)

Set the registration state for different indication pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM7430/EM7455/MC7430/MC7455 onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.43 int pack_dms_SLQSGetBandCapability (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get band capability of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.44 int pack_dms_SLQSGetERIFile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set UIM Get ERI file pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.45 int pack_dms_SLQSGetPowerSaveModeConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Power Save Mode (PSM) configuration parameter pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.46 int pack_dms_SLQSSetPowerSaveModeConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSSetPowerSaveModeConfig_t * *reqArg*)

Sets the Power Save Mode (PSM) configuration parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.47 int pack_dms_SLQSSwiClearDyingGaspStatistics (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Clear Dying GASP Statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.48 int pack_dms_SLQSSwiGetCrashInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSSwiGetCrashInfo_t * pReq)

To Get the Crash Information from the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.49 int pack_dms_SLQSSwiGetDyingGaspCfg (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Get Dying GASP Config pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.50 int pack_dms_SLQSSwiGetDyingGaspStatistics (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Get Dying GASP Statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.51 int pack_dms_SLQSSwiGetFirmwareCurr (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Get currently active image pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.52 int pack_dms_SLQSSwiGetFwUpdateStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

To get Firmware Update status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.53 int pack_dms_SLQSSwiGetHostDevInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

To get host dev information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.54 int pack_dms_SLQSSwiGetOSInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

To Get Host OS Info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.55 int pack_dms_SLQSSwiGetSerialNoExt (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

To get serial number extension pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.56 int pack_dms_SLQSSwiSetDyingGaspCfg (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSSwiSetDyingGaspCfg_t * reqArg)

Set Dying GASP Config pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.57 int pack_dms_SLQSSwiSetHostDevInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSSwiSetHostDevInfo_t * pReq)

To set Host Dev Info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.58 int pack_dms_SLQSSwiSetOSInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_dms_SLQSSwiSetOSInfo_t * pReq)

To set Host OS Info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.59 int pack_dms_SLQSUIMGetState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Returns the UIM state pack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7455/EM7430 firmware versions. Please use API pack_uim_GetCardStatus unpack_uim_GetCardStatus for new firmware versions and new modules

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.60 int pack_dms_SwiSetEventReport (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SwiSetEventReport_t * *reqArg*)

Sets the DMS swi event report parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.61 int pack_dms_SwiUimSelect (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SwiUimSelect_t * *reqArg*)

Sets the DMS swi UIM Select parameters pack. A reset is required for any change in the UIM selection to take effect.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.62 int pack_dms_UIMChangePIN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMChangePIN_t * *pReq*)

To set UIM change PIN pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.63 int pack_dms_UIMGetControlKeyStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMGetControlKeyStatus_t * *pReq*)

To set UIM Get Control Key Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.64 int pack_dms_UIMGetICCID (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMGetICCID_t * *reqArg*)

Packs the UIMGetICCID response message to a user-provided response structure. This API is deprecated on MC73xx/EM73xx modules. Since firmware version SWI9X15C_05_xx_xx_xx and EM7455/EM7430 firmware versions. Please use pack_uim_ReadTransparent()(EF ID: 3F00 2FE2) instead for new firmware versions and new modules.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.65 int pack_dms_UIMGetPINStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set UIM Get PIN Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.66 int pack_dms_UIMSetControlKeyProtection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMSetControlKeyProtection_t * *pReq*)

To set UIM Set Control Key Protection pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.67 int pack_dms_UIMSetPINProtection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMSetPINProtection_t * *pReq*)

To set UIM PIN protection pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.68 int pack_dms_UIMUnblockControlKey (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMUnblockControlKey_t * *pReq*)

To set UIM Unblock Control Key pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.69 int pack_dms_UIMUnblockPIN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMUnblockPIN_t * *pReq*)

To set UIM unblock PIN pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.70 int pack_dms_UIMVerifyPIN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMVerifyPIN_t * *pReq*)

To set UIM verify PIN pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.71 int pack_dms_ValidateSPC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ResetToFactoryDefaults_t * *pReq*)

Requests the device to perform automatic service activation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.72 int unpack_dms_ActivateAutomatic (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactory-Defaults_t * *pOutput*)

Requests the device to perform automatic service activation unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.73 int unpack_dms_GetActivationState (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetActivationState_t * *pOutput*)

Get Activation State unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

This is only supported for 3GPP2 devices.

9.9.2.74 int unpack_dms_GetBandCapability (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetBandCapability_t * *pOutput*)

Get Band Capabilities unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.75 int unpack_dms_GetCrashAction (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCrashAction_t * *pOutput*)

Get Crash Action unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.76 int unpack_dms_GetCustFeature (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCustFeature_t * *pOutput*)

Get Custom Feature unpack. This API is deprecated for EM7455/EM7430/MC7455/MC7430, please use unpack_dms_GetCustFeaturesV2() instead for EM7455/EM7430/MC7455/MC7430.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.77 int unpack_dms_GetCustFeaturesV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCustFeaturesV2_t * *pOutput*)

Get Custom Feature unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.78 int unpack_dms_GetDeviceCap (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceCap_t * *pOutput*)

Get Device Capabilities unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.79 int unpack_dms_GetDeviceCapabilities (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceCapabilities_t * *pOutput*)

Get device capability unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.80 int unpack_dms_GetDeviceCapabilitiesV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceCapabilitiesV2_t * *pOutput*)

Get device capability unpack v2

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.81 int unpack_dms_GetDeviceHardwareRev (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceHardwareRev_t * *pOutput*)

Get Hardware Revision of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.82 int unpack_dms_GetDeviceMfr (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceMfr_t * *pOutput*)

Get Device Manufacture Name unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.83 int unpack_dms_GetDeviceSerialNumbers (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceSerialNumbers_t * *pOutput*)

Get Device Serial Number unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.84 int unpack_dms_GetFirmwareInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmwareInfo_t * *pOutput*)

Get firmware info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.85 int unpack_dms_GetFirmwareRevision (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmware-Revision_t * *pOutput*)

Get Firmware Revision unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.86 int unpack_dms_GetFirmwareRevisions (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmwareRevisions_t * *pOutput*)

Get Firmware Revisions unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.87 int unpack_dms_GetFSN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFSN_t * *pOutput*)

Get FSN unpack This API get the Factory Sequence Number of the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.88 int unpack_dms_GetHardwareRevision (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetHardwareRevision_t * *pOutput*)

Get hardware revision of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.89 int unpack_dms_GetIMSI (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetIMSI_t * *pOutput*)

Get IMSI unpack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7455/EM7430 firmware versions. Please use unpack_uim_ReadTransparent()(EF ID: 3F00 7F20 6F07 for 2G card and 3F00 7FFF 6F07 for 3G card) instead for new firmware versions and new modules.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.90 int unpack_dms_GetManufacturer (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetManufacturer_t * *pOutput*)

To get device manufacturer information unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.91 int unpack_dms_GetModelID (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetModelID_t * *pOutput*)

Get model id unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.92 int unpack_dms_GetNetworkTime (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetNetworkTime_t * *pOutput*)

Get Network Time unpack. Returns the current time of the device based on the value supported by the network.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.93 int unpack_dms_GetNetworkTimeV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetNetworkTimeV2_t * *pOutput*)

Get Network Time unpack. Returns the current time of the device based on the value supported by the network.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.94 int unpack_dms_GetOfflineReason (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetOfflineReason_t * *pOutput*)

To get operating mode offline reason unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.95 int unpack_dms_GetPower (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetPower_t * *pOutput*)

Get operating mode of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.96 int unpack_dms_GetPRLVersion (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetPRLVersion_t * *pOutput*)

Get PRL Versions unpack Return version of the active Preferred Roaming List (PRL) in use by the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.97 int unpack_dms_GetSerialNumbers (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetSerialNumbers_t * *pOutput*)

Get serial numbers unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.98 int unpack_dms_GetUSBComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetUSBComp_t * *pOutput*)

Get USB Comp unpack This API queries the modem's USB interface configuration and supported configuration parameters.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.99 int unpack_dms_GetVoiceNumber (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetVoiceNumber_t * *pOutput*)

Get Voice Number unpack Returns the voice number in use by the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.100 int unpack_dms_PSMCfgChange_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_PSMCfgChange_ind_t * *pOutput*)

PSM Configuration change Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.101 int unpack_dms_ResetToFactoryDefaults (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactoryDefaults_t * *pOutput*)

Resets to default factory settings of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.102 int unpack_dms_SetActivationStatusCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetActivationStatusCallback_t * *pOutput*)

Set Activation status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.103 int unpack_dms_SetCrashAction (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetCrashAction_t * *pOutput*)

Set Crash Action unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response. Not used

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.104 int unpack_dms_SetCustFeature (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetCustFeature_t * *pOutput*)

Set Custom Feature unpack. This API is deprecated for EM7455/EM7430/MC7455/MC7430, please use unpack_dms_SetCustFeaturesV2() instead for EM7455/EM7430/MC7455/MC7430.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.105 int unpack_dms_SetCustFeaturesV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetCustFeaturesV2_t * *pOutput*)

Set Cust features unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.106 int unpack_dms_SetEventReport (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetEventReport_t * *pOutput*)

Set Event Report unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.107 int unpack_dms_SetEventReport_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetEventReport_ind_t * *pOutput*)

Event Report Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.108 int unpack_dms_SetFirmwarePreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetFirmwarePreference_t * *pOutput*)

Set Firmware Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.109 int unpack_dms_SetIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetIndicationRegister_t * *pOutput*)

Set Indication register unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.110 int unpack_dms_SetPower (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetPower_t * *pOutput*)

Set Power unpack. Sets the operating mode of the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.111 int unpack_dms_SetUSBComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetUSBComp_t * *pOutput*)

Set USB Comp unpack This API is used to change the modem's USB interface configuration thus allowing a device to have multiple USB compositions. Devices will, by default, be configured to support a minimal set of interfaces to reduce end user modem installation time. Developers and some customers, however, require access to a custom set of interfaces. A reset is required for any change in the USB composition to take effect.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.112 int unpack_dms_SLQSDmsSwiGetPCInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiGetPCInfo_t * *pOutput*)

Get device power control status information unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.113 int unpack_dms_SLQSDmsSwiGetResetInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiGetResetInfo_t * *pOutput*)

To get reset info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.114 int unpack_dms_SLQSDmsSwiGetResetInfo_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t * *pOutput*)

DMS reset info Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM7430/EM7455/MC7430/MC7455 onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.115 int unpack_dms_SLQSDmsSwiGetUimSelection (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiGetUimSelection_t * *pOutput*)

Get device UIM selection information unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.116 int unpack_dms_SLQSDmsSwiIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiIndicationRegister_t * *pOutput*)

Set the registration state for different indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM7430/EM7455/MC7430/MC7455 onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.117 int unpack_dms_SLQSGetBandCapability (uint8_t * *pResp*, uint16_t *respLen*, **unpack_dms_SLQSGetBandCapability_t** * *pOutput*)

Get band capability of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.118 int unpack_dms_SLQSGetBandCapabilityExt (uint8_t * *pResp*, uint16_t *respLen*, **unpack_dms_SLQSGetBandCapabilityExt_t** * *pOutput*)

Get band capability of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.119 int unpack_dms_SLQSGetERIFile (uint8_t * *pResp*, uint16_t *respLen*, **unpack_dms_SLQSGetERIFile_t** * *pOutput*)

To get UIM Get ERI file unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.120 int unpack_dms_SLQSGetPowerSaveModeConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSGetPowerSaveModeConfig_t * *pOutput*)

Get Power Save Mode (PSM) configuration parameter unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.121 int unpack_dms_SLQSSetPowerSaveModeConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSetPowerSaveModeConfig_t * *pOutput*)

Sets the Power Save Mode (PSM) configuration parameters unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.122 int unpack_dms_SLQSSwiClearDyingGaspStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiClearDyingGaspStatistics_t * *pOutput*)

Clear Dying GASP Statistics unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.123 int unpack_dms_SLQSSwiGetCrashInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetCrashInfo_t * *pOutput*)

To Get the Crash Information from the device unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.124 int unpack_dms_SLQSSwiGetDyingGaspCfg (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetDyingGaspCfg_t * *pOutput*)

Get Dying GASP Config unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.125 int unpack_dms_SLQSSwiGetDyingGaspStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetDyingGaspStatistics_t * *pOutput*)

Get Dying GASP Statistics unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.126 int unpack_dms_SLQSSwiGetFirmwareCurr (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetFirmwareCurr_t * *pOutput*)

Get currently active image unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.127 int unpack_dms_SLQSSwiGetFwUpdateStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetFwUpdateStatus_t * *pOutput*)

To get Firmware Update status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.128 int unpack_dms_SLQSSwiGetHostDevInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetHostDevInfo_t * *pOutput*)

To get host dev information unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.129 int unpack_dms_SLQSSwiGetOSInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetOSInfo_t * *pOutput*)

To get Host OS Info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.130 int unpack_dms_SLQSSwiGetSerialNoExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetSerialNoExt_t * *pOutput*)

To get serial number extension unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.131 int unpack_dms_SLQSSwiSetDyingGaspCfg (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetDyingGaspCfg_t * *pOutput*)

Set Dying GASP Config unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.132 int unpack_dms_SLQSSwiSetHostDevInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetHostDevInfo_t * *pOutput*)

To set Host Dev Info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.133 int unpack_dms_SLQSSwiSetOSInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetOSInfo_t * *pOutput*)

To set Host OS Info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.134 int unpack_dms_SLQSUIMGetState (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSUIMGetState_t * *pOutput*)

Returns the UIM state unpack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7455/EM7430 firmware versions. Please use API pack_uim_GetCardStatus unpack_uim_GetCardStatus for new firmware versions and new modules

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.135 int unpack_dms_SwiEventReportCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SwiEventReportCallBack_ind_t * *pOutput*)

unpack for DMS swi event report indication

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.136 int unpack_dms_SwiSetEventReport (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SwiSetEventReport_t * *pOutput*)

Sets the swi event report parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.137 int unpack_dms_SwiUimSelect (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SwiUimSelect_t * *pOutput*)

Sets the DMS swi UIM Select parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.138 int unpack_dms_UIMChangePIN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM change PIN unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.139 int unpack_dms_UIMGetControlKeyStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMGetControlKeyStatus_t * *pOutput*)

To get UIM Get ControlKeyStatus unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.140 int unpack_dms_UIMGetICCID (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMGetICCID_t * *pOutput*)

Unpacks the UIMGetICCID response message to a user-provided response structure.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.141 int unpack_dms_UIMGetPINStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMGetPINStatus_t * *pOutput*)

To get UIM Get PIN Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.142 int unpack_dms_UIMSetControlKeyProtection (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetControlKeyProtection_t * *pOutput*)

To get UIM Set Control Key Protection unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.143 int unpack_dms_UIMSetPINProtection (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM Set PIN protection unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.144 int unpack_dms_UIMUnblockControlKey (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMUnblockControlKey_t * *pOutput*)

To get UIM Unblock Control Key unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.145 int unpack_dms_UIMUnblockPIN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM unblock PIN unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.146 int unpack_dms_UIMVerifyPIN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM verify PIN unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.2.147 int unpack_dms_ValidateSPC (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactoryDefaults_t * *pOutput*)

Requests the device to perform automatic service activation unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.10 dsd.h File Reference

Data Structures

- struct DsdSystemInfo_t
- struct DsdApnAvailableSystemInfo_t
- struct DsdApnCurRecAvailableSystemInfo_t
- struct DsdApnCurRecPreferredSystemInfo_t
- struct DsdNullBearerReason_t
- struct unpack_dsd_SLQSGetSystemStatus_t
- struct DsdGlobalUIInformation_t

- struct DsdUIApnAvailableSystemInfo_t
- struct unpack_dsd_SLQSGetDsdUIInfo_t
- struct pack_dsd_system_status_changes_t
- struct pack_dsd_register_ui_info_changes_t
- struct unpack_dsd_SLQSSystemStatusChange_Ind_t
- struct unpack_dsd_SLQSUIInfo_Ind_t

Macros

- #define MAX_APN_STRING 100
- #define MAX_AVAIL_SYSTEM_INFO 255

Typedefs

- typedef unpack_result_t unpack_dsd_SystemStatusChnage_t
- typedef unpack_result_t unpack_dsd_RegisterUIInfoChange_t

Functions

- int pack_dsd_SLQSGetSystemStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dsd_SLQSGetSystemStatus (uint8_t *pResp, uint16_t respLen, unpack_dsd_SLQSGetSystemStatus_t *pOutput)
- int pack_dsd_SLQSGetUIInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dsd_SLQSGetUIInfo (uint8_t *pResp, uint16_t respLen, unpack_dsd_SLQSGetDsdUIInfo_t *pOutput)
- int pack_dsd_SystemStatusChange (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dsd_system_status_changes_t *pSystemStatusChange)
- int unpack_dsd_SystemStatusChange (uint8_t *pResp, uint16_t respLen, unpack_dsd_SystemStatusChnage_t *pOutput)
- int pack_dsd_RegisterUIInfoChange (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dsd_register_ui_info_changes_t *pRegisterUIInfoChange)
- int unpack_dsd_RegisterUIInfoChange (uint8_t *pResp, uint16_t respLen, unpack_dsd_RegisterUIInfoChange_t *pOutput)
- int unpack_dsd_SLQSSystemStatusChangeInd (uint8_t *pResp, uint16_t respLen, unpack_dsd_SLQSSystemStatusChange_Ind_t *pOutput)
- int unpack_dsd_SLQSUIInfoChangeInd (uint8_t *pResp, uint16_t respLen, unpack_dsd_SLQSUIInfo_Ind_t *pOutput)

9.10.1 Detailed Description

9.10.2 DSDSOMask SO Mask

- SO mask to indicate the service option or type of application.
  - An SO mask value of zero indicates that this field is ignored.
  - Values:
    - 0x00 - DONT_CARE
  - CDMA 1X SO mask:
    - 0x01 - CDMA_1X_IS95
    - 0x02 - CDMA_1X_IS2000
    - 0x04 - CDMA_1X_IS2000_REL_A
  - CDMA EV-DO Rev 0 SO mask:

    - 0x01 - DPA
  - CDMA EV-DO Rev A SO mask:
    - 0x01 - DPA
    - 0x02 - MFPA
    - 0x04 - EMPA
    - 0x08 - EMPA_EHRPD
  - CDMA EV-DO Rev B SO mask:
    - 0x01 - DPA
    - 0x02 - MFPA
    - 0x04 - EMPA
    - 0x08 - EMPA_EHRPD
    - 0x10 - MMPA
    - 0x20 - MMPA_EHRPD

9.10.3 DSDRatMask RAT Mask

- RAT mask to indicate the type of technology. A RAT mask value of zero indicates that this field is ignored.

  Values:

  - 0x00 - DONT_CARE
  - 0x8000 - NULL_BEARER

- CDMA RAT mask:

  - 0x01 - CDMA_1X
  - 0x02 - EVDO_REV0
  - 0x04 - EVDO_REVA
  - 0x08 - EVDO_REVB
  - 0x10 - EHRPD
  - 0x20 - FMC

- UMTS RAT mask:

  - 0x01 - WCDMA
  - 0x02 - GPRS
  - 0x04 - HSDPA
  - 0x08 - HSUPA
  - 0x10 - EDGE
  - 0x20 - LTE
  - 0x40 - HSDPA+
  - 0x80 - DC_HSDPA+
  - 0x100 - 64_QAM
  - 0x200 - TD-SCDMA

9.10.4 Macro Definition Documentation

9.10.4.1 #define MAX_APN_STRING 100

9.10.4.2 #define MAX_AVAIL_SYSTEM_INFO 255

9.10.5 Typedef Documentation

9.10.5.1 typedef unpack_result_t unpack_dsd_RegisterUIInfoChange_t

9.10.5.2 typedef unpack_result_t unpack_dsd_SystemStatusChnage_t

9.10.6 Function Documentation

9.10.6.1 int pack_dsd_RegisterUIInfoChange (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dsd_register_ui_info_changes_t * *pRegisterUIInfoChange*)

System Status Changes flag pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pRegisterUIInfo-Change*	indications to register for.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

For_5G

9.10.6.2 int pack_dsd_SLQSGetSystemStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

set event callback pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.10.6.3 int pack_dsd_SLQSGetUIInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get UI Information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.10.6.4 int pack_dsd_SystemStatusChange (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dsd_system_status_changes_t * *pSystemStatusChange*)

System Status Changes flag pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pSystemStatus-Change*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

For_5G

9.10.6.5 int unpack_dsd_RegisterUIInfoChange (uint8_t * *pResp*, uint16_t *respLen*, unpack_dsd_RegisterUIInfo-Change_t * *pOutput*)

Unpack System Status Change response.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

For_5G

9.10.6.6 int unpack_dsd_SLQSGetSystemStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_dsd_SLQSGetSystemStatus_t * *pOutput*)

Get System Status unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.10.6.7 int unpack_dsd_SLQSGetUIInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dsd_SLQSGetDsdUIInfo_t * *pOutput*)

Get UI Information unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.10.6.8 int unpack_dsd_SLQSSystemStatusChangeInd (uint8_t * *pResp*, uint16_t *respLen*, unpack_dsd_SLQSSystemStatusChange_Ind_t * *pOutput*)

Unpack System Status Change Indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked indication

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

For_5G

9.10.6.9 int unpack_dsd_SLQSUIInfoChangeInd (uint8_t * *pResp*, uint16_t *respLen*, unpack_dsd_SLQSUIInfo_Ind_t * *pOutput*)

Unpack UI information change Indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked indication

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

For_5G

9.10.6.10 int unpack_dsd_SystemStatusChange (uint8_t * *pResp*, uint16_t *respLen*, unpack_dsd_SystemStatusChnage_t * *pOutput*)

Unpack System Status Change response.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

For_5G

9.11 fms.h File Reference

Data Structures

- struct CarrierImage_t
- struct pack_fms_GetImagesPreference_t
- struct FMSImageElement
- struct FMSPrefImageList
- struct unpack_fms_GetImagesPreference_t
- struct pack_fms_GetStoredImages_t
- struct FMSImageIdElement
- struct FMSImageIDEntries
- struct FMSImageList
- struct unpack_fms_GetStoredImages_t
- struct pack_fms_SetImagesPreference_t
- struct unpack_fms_SetImagesPreference_t

Macros

- #define FMS_GOBI_MBN_IMG_ID_SIZE 16
- #define LITE_TOTAL_IMAGE_ID_STRING_SIZE 100
- #define FMS_GOBI_MBN_IMG_ID_STR_LEN LITE_TOTAL_IMAGE_ID_STRING_SIZE
- #define FMS_GOBI_MBN_BUILD_ID_STR_LEN LITE_TOTAL_IMAGE_ID_STRING_SIZE
- #define FMS_GOBI_LISTENTRIES_MAX 2
- #define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255
- #define FMS_MAX_IMAGE_ID_ELEMENT 50
- #define FMS_IMAGE_ID_MAX_ENTRIES 2
- #define FMS_FW_PRI_BUILD_MATCH_LEN 11
- #define FMS_IMAGE_ID_IMG_ID_LEN 16
- #define FMS_IMAGE_ID_BUILD_ID_LEN 32
- #define FMS_IMAGE_ID_PRI_IMGTYPE 0x01

Functions

- int pack_fms_GetImagesPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetImagesPreference_t *reqArg)
- int unpack_fms_GetImagesPreference (uint8_t *pResp, uint16_t respLen, unpack_fms_GetImagesPreference_t *pOutput)
- int pack_fms_GetStoredImages (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetStoredImages_t *reqArg)
- int unpack_fms_GetStoredImages (uint8_t *pResp, uint16_t respLen, unpack_fms_GetStoredImages_t *pOutput)

- int pack_fms_SetImagesPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_SetImagesPreference_t *reqArg)
- int unpack_fms_SetImagesPreference (uint8_t *pResp, uint16_t respLen, unpack_fms_SetImagesPreference_t *pOutput)
- uint32_t GetValidFwPriCombinations (FMSImageList *pStoredImageList, uint32_t *pValidCombinationSize, CarrierImage_t *pValidCombinations)

9.11.1 Macro Definition Documentation

9.11.1.1 #define FMS_FW_PRI_BUILD_MATCH_LEN 11

9.11.1.2 #define FMS_GOBI_LISTENTRIES_MAX 2

9.11.1.3 #define FMS_GOBI_MBN_BUILD_ID_STR_LEN LITE_TOTAL_IMAGE_ID_STRING_SIZE

9.11.1.4 #define FMS_GOBI_MBN_IMG_ID_SIZE 16

9.11.1.5 #define FMS_GOBI_MBN_IMG_ID_STR_LEN LITE_TOTAL_IMAGE_ID_STRING_SIZE

9.11.1.6 #define FMS_IMAGE_ID_BUILD_ID_LEN 32

9.11.1.7 #define FMS_IMAGE_ID_IMG_ID_LEN 16

9.11.1.8 #define FMS_IMAGE_ID_MAX_ENTRIES 2

9.11.1.9 #define FMS_IMAGE_ID_PRI_IMGTYPE 0x01

9.11.1.10 #define FMS_MAX_IMAGE_ID_ELEMENT 50

9.11.1.11 #define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255

9.11.1.12 #define LITE_TOTAL_IMAGE_ID_STRING_SIZE 100

9.11.2 Function Documentation

9.11.2.1 uint32_t GetValidFwPriCombinations (FMSImageList * *pStoredImageList*, uint32_t * *pValidCombinationSize*, CarrierImage_t * *pValidCombinations*)

This API distills valid Firmware/PRI combinations from GetStoredImages result

Parameters

in	*pStoredImageList*	• image list returned from GetStoredImages • See FMSImageList
in,out	*pValidCombinationSize*	• number of combination passed in and returned
out	*pValidCombinations*	• valid combinations returned • See CarrierImage_t

Returns

- eQCWWAN_ERR_INVALID_ARG - Invalid parameters
- eQCWWAN_ERR_BUFFER_SZ - No enough element to store combinatons returned

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.11.2.2 int pack_fms_GetImagesPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_fms_GetImagesPreference_t * *reqArg*)

Get Images Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.11.2.3 int pack_fms_GetStoredImages (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_fms_GetStoredImages_t * *reqArg*)

Get Images Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.11.2.4 int pack_fms_SetImagesPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_fms_SetImagesPreference_t * *reqArg*)

Set Images Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.11.2.5 int unpack_fms_GetImagesPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_fms_GetImagesPreference_t * *pOutput*)

Get Images Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

out	*pOutput*	unpacked response

9.11.2.6 int unpack_fms_GetStoredImages (uint8_t * *pResp*, uint16_t *respLen*, unpack_fms_GetStoredImages_t * *pOutput*)

Get Images Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

9.11.2.7 int unpack_fms_SetImagesPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_fms_SetImagesPreference_t * *pOutput*)

Set Images Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

9.12 ims.h File Reference

Data Structures

- struct pack_ims_SLQSSetSIPConfig_t
- struct unpack_ims_SLQSSetSIPConfig_t
- struct pack_ims_SLQSSetRegMgrConfig_t
- struct unpack_ims_SLQSSetRegMgrConfig_t
- struct pack_ims_SLQSSetIMSSMSConfig_t
- struct unpack_ims_SLQSSetIMSSMSConfig_t
- struct pack_ims_SLQSSetIMSUserConfig_t
- struct unpack_ims_SLQSSetIMSUserConfig_t
- struct pack_ims_SLQSSetIMSVoIPConfig_t
- struct unpack_ims_SLQSSetIMSVoIPConfig_t
- struct unpack_ims_SLQSGetSIPConfig_t
- struct unpack_ims_SLQSGetRegMgrConfig_t
- struct unpack_ims_SLQSGetIMSSMSConfig_t
- struct unpack_ims_SLQSGetIMSUserConfig_t
- struct unpack_ims_SLQSGetIMSVoIPConfig_t
- struct pack_ims_SLQSImsConfigIndicationRegister_t
- struct ims_PCSCFPortInfo
- struct ims_CSCFPortNameInfo
- struct ims_IMSTestModeInfo
- struct unpack_ims_SLQSRegMgrCfgCallBack_ind_t
- struct ims_SIPPortInfo
- struct ims_SIPRegnTmrInfo
- struct ims_SubscrTmrInfo

- struct ims_TmrT1Info
- struct ims_TmrT2Info
- struct ims_TmrTfInfo
- struct ims_SigCompEnInfo
- struct unpack_ims_SLQSSIPCfgCallBack_ind_t
- struct ims_SMSFmtInfo
- struct ims_SMSoIPNwInfo
- struct ims_PhCtxtURIInfo
- struct unpack_ims_SLQSSMSCfgCallBack_ind_t
- struct ims_IMSDomainInfo
- struct unpack_ims_SLQSUserCfgCallBack_ind_t
- struct ims_SessDurInfo
- struct ims_MinSessExpInfo
- struct ims_EnabAMRWBInfo
- struct ims_EnabSCRAMRInfo
- struct ims_EnabSCRAMRWBInfo
- struct ims_AMRModeInfo
- struct ims_AMRWBModeInfo
- struct ims_AMROctAlgnInfo
- struct ims_AMRWBOctAlgnInfo
- struct ims_RngTmrInfo
- struct ims_RngBkTmrInfo
- struct ims_RTPRTCPInactTmrDurInfo
- struct unpack_ims_SLQSVoIPCfgCallBack_ind_t

Macros

- #define MAX_NAME_LEN 255

Typedefs

- typedef unpack_result_t unpack_ims_SLQSImsConfigIndicationRegister_t

Functions

- int pack_ims_SLQSSetSIPConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ims_SLQSSetSIPConfig_t *pReqParam)
- int unpack_ims_SLQSSetSIPConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSSetSIPConfig_t *pOutput)
- int pack_ims_SLQSSetRegMgrConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ims_SLQSSetRegMgrConfig_t *pReqParam)
- int unpack_ims_SLQSSetRegMgrConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSSetRegMgrConfig_t *pOutput)
- int pack_ims_SLQSSetIMSSMSConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ims_SLQSSetIMSSMSConfig_t *pReqParam)
- int unpack_ims_SLQSSetIMSSMSConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSSetIMSSMSConfig_t *pOutput)
- int pack_ims_SLQSSetIMSUserConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ims_SLQSSetIMSUserConfig_t *pReqParam)
- int unpack_ims_SLQSSetIMSUserConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSSetIMSUserConfig_t *pOutput)
- int pack_ims_SLQSSetIMSVoIPConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ims_SLQSSetIMSVoIPConfig_t *pReqParam)

- int unpack_ims_SLQSSetIMSVoIPConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSSetIMSVoIPConfig_t *pOutput)
- int pack_ims_SLQSGetSIPConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_ims_SLQSGetSIPConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSGetSIPConfig_t *pOutput)
- int pack_ims_SLQSGetRegMgrConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_ims_SLQSGetRegMgrConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSGetRegMgrConfig_t *pOutput)
- int pack_ims_SLQSGetIMSSMSConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_ims_SLQSGetIMSSMSConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSGetIMSSMSConfig_t *pOutput)
- int pack_ims_SLQSGetIMSUserConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_ims_SLQSGetIMSUserConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSGetIMSUserConfig_t *pOutput)
- int pack_ims_SLQSGetIMSVoIPConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_ims_SLQSGetIMSVoIPConfig (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSGetIMSVoIPConfig_t *pOutput)
- int pack_ims_SLQSImsConfigIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_ims_SLQSImsConfigIndicationRegister_t *pReqParam)
- int unpack_ims_SLQSImsConfigIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSImsConfigIndicationRegister_t *pOutput)
- int unpack_ims_SLQSRegMgrCfgCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSRegMgrCfgCallBack_ind_t *pOutput)
- int unpack_ims_SLQSSIPCfgCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSSIPCfgCallBack_ind_t *pOutput)
- int unpack_ims_SLQSSMSCfgCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSSMSCfgCallBack_ind_t *pOutput)
- int unpack_ims_SLQSUserCfgCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSUserCfgCallBack_ind_t *pOutput)
- int unpack_ims_SLQSVoIPCfgCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_ims_SLQSVoIPCfgCallBack_ind_t *pOutput)

9.12.1 Macro Definition Documentation

9.12.1.1 #define MAX_NAME_LEN 255

9.12.2 Typedef Documentation

9.12.2.1 typedef unpack_result_t unpack_ims_SLQSImsConfigIndicationRegister_t

9.12.3 Function Documentation

9.12.3.1 int pack_ims_SLQSGetIMSSMSConfig (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen*)

Retrieves the SMS configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.2 int pack_ims_SLQSGetIMSUserConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Retrieves the IMS User configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.3 int pack_ims_SLQSGetIMSVoIPConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Retrieves the IMS VoIP configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.4 int pack_ims_SLQSGetRegMgrConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Retrieves the Session Initiation Protocol(SIP) configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.5 int pack_ims_SLQSGetSIPConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Retrieves the Session Initiation Protocol(SIP) configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.6 int pack_ims_SLQSImsConfigIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_ims_SLQSImsConfigIndicationRegister_t * *pReqParam*)

Sets the registration state for different QMI_IMS indications for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

```
- This function is used by a device to register/deregister for different
  QMI IMS indications.
- The device&apos;s registration state variables that control registration
  for indications will be modified to reflect the settings indicated in the
  request message.
- At least one optional parameter must be present in the request.
```

9.12.3.7 int pack_ims_SLQSSetIMSSMSConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_ims_SLQSSetIMSSMSConfig_t * *pReqParam*)

Sets the IMS SMS configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.8 int pack_ims_SLQSSetIMSUserConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_ims_SLQSSetIMSUserConfig_t * *pReqParam*)

Sets the IMS user configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.9 int pack_ims_SLQSSetIMSVoIPConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_ims_SLQSSetIMSVoIPConfig_t * *pReqParam*)

Sets the IMS Voice over Internet Protocol (VoIP) configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.10 int pack_ims_SLQSSetRegMgrConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_ims_SLQSSetRegMgrConfig_t * *pReqParam*)

Sets the IMS registration manager configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7430/MC7455 and EM7430/EM7455

9.12.3.11 int pack_ims_SLQSSetSIPConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_ims_SLQSSetSIPConfig_t * *pReqParam*)

Sets the IMS Session Initiation Protocol(SIP) configuration parameters for the requesting control point pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC73xx, MC7455/MC7430 and EM7455/EM7430

9.12.3.12 int unpack_ims_SLQSGetIMSSMSConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSGetIMSSMSConfig_t * *pOutput*)

Retrieves the SMS configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.13 int unpack_ims_SLQSGetIMSUserConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSGetIMSUserConfig_t * *pOutput*)

Retrieves the IMS User configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.14 int unpack_ims_SLQSGetIMSVoIPConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSGetIMSVoIPConfig_t * *pOutput*)

Retrieves the IMS VoIP configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.15 int unpack_ims_SLQSGetRegMgrConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSGetRegMgrConfig_t * *pOutput*)

Retrieves the Session Initiation Protocol(SIP) configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.16 int unpack_ims_SLQSGetSIPConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSGetSIPConfig_t * *pOutput*)

Retrieves the Session Initiation Protocol(SIP) configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.17 int unpack_ims_SLQSImsConfigIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSImsConfigIndicationRegister_t * *pOutput*)

Sets the registration state for different QMI_IMS indications for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.18 int unpack_ims_SLQSRegMgrCfgCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSRegMgrCfgCallBack_ind_t * *pOutput*)

Unpack indication about IMS registration manager configuration

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.19 int unpack_ims_SLQSSetIMSSMSConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSSetIMSSMSConfig_t * *pOutput*)

Sets the IMS SMS configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.20 int unpack_ims_SLQSSetIMSUserConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSSetIMSUserConfig_t * *pOutput*)

Sets the IMS user configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.21 int unpack_ims_SLQSSetIMSVoIPConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSSetIMSVoIPConfig_t * *pOutput*)

Sets the IMS Voice over Internet Protocol (VoIP) configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.22 int unpack_ims_SLQSSetRegMgrConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSSetRegMgrConfig_t * *pOutput*)

Sets the IMS registration manager configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.23 int unpack_ims_SLQSSetSIPConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSSetSIPConfig_t * *pOutput*)

Sets the IMS Session Initiation Protocol(SIP) configuration parameters for the requesting control point unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.24 int unpack_ims_SLQSSIPCfgCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSSIPCfgCallBack_ind_t * *pOutput*)

Unpack indication about SIP configuration info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.25 int unpack_ims_SLQSSMSCfgCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSSMSCfgCallBack_ind_t * *pOutput*)

Unpack indication about SMS configuration info for IMS

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.26 int unpack_ims_SLQSUserCfgCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSUserCfgCallBack_ind_t * *pOutput*)

Unpack indication about User configuration info for IMS

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.12.3.27 int unpack_ims_SLQSVoIPCfgCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSVoIPCfgCallBack_ind_t * *pOutput*)

Unpack indication about VOIP configuration info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13 imsa.h File Reference

Data Structures

- struct pack_imsa_SLQSRegisterIMSAIndication_t
- struct unpack_imsa_SLQSGetIMSARegStatus_t
- struct unpack_imsa_SLQSGetIMSAServiceStatus_t
- struct imsa_SmsSvcStatusInfo
- struct imsa_SmsRatInfo
- struct imsa_VoipSvcStatusInfo
- struct imsa_VoipRatInfo
- struct imsa_VtSvcStatusInfo
- struct imsa_VtRatInfo
- struct imsa_UtSvcStatusInfo
- struct imsa_UtRatInfo
- struct unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t
- struct imsa_IMSRegStatusInfo
- struct imsa_IMSRegStatusErrorCodeInfo
- struct imsa_NewIMSRegStatusInfo
- struct unpack_imsa_SLQSImsaRegStatusCallBack_ind_t
- struct imsa_RatHandoverStatusInfo
- struct unpack_imsa_SLQSImsaRatStatusCallBack_ind_t
- struct imsa_IMSFailErrCodeTlv
- struct unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t

Macros

- #define MAX_ERROR_CODE_LEN 256

Typedefs

- typedef unpack_result_t unpack_imsa_SLQSRegisterIMSAIndication_t

Functions

- int pack_imsa_SLQSRegisterIMSAIndication (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_imsa_SLQSRegisterIMSAIndication_t *reqParam)
- int unpack_imsa_SLQSRegisterIMSAIndication (uint8_t *pResp, uint16_t respLen, unpack_imsa_SLQSRegisterIMSAIndication_t *pOutput)
- int pack_imsa_SLQSGetIMSARegStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_imsa_SLQSGetIMSARegStatus (uint8_t *pResp, uint16_t respLen, unpack_imsa_SLQSGetIMSARegStatus_t *pOutput)
- int pack_imsa_SLQSGetIMSAServiceStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_imsa_SLQSGetIMSAServiceStatus (uint8_t *pResp, uint16_t respLen, unpack_imsa_SLQSGetIMSAServiceStatus_t *pOutput)

- int unpack_imsa_SLQSImsaSvcStatusCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t *pOutput)
- int unpack_imsa_SLQSImsaRegStatusCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_imsa_SLQSImsaRegStatusCallBack_ind_t *pOutput)
- int unpack_imsa_SLQSImsaRatStatusCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_imsa_SLQSImsaRatStatusCallBack_ind_t *pOutput)
- int unpack_imsa_SLQSImsaPdpStatusCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t *pOutput)

9.13.1 Macro Definition Documentation

9.13.1.1 #define MAX_ERROR_CODE_LEN 256

9.13.2 Typedef Documentation

9.13.2.1 typedef unpack_result_t unpack_imsa_SLQSRegisterIMSAIndication_t

9.13.3 Function Documentation

9.13.3.1 int pack_imsa_SLQSGetIMSARegStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack Get IMSA Registration Status command.

Parameters

pCtx	[OUT] • See pack_qmi_t for more information
pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.2 int pack_imsa_SLQSGetIMSAServiceStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack Get IMSA Service Status command

Parameters

pCtx	[OUT] • See pack_qmi_t for more information

pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.3 int pack_imsa_SLQSRegisterIMSAIndication (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_imsa_SLQSRegisterIMSAIndication_t * *reqParam*)

Function to pack Register IMSA Indication command

Parameters

pCtx	[OUT] • See pack_qmi_t for more information
pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[IN] • See pack_imsa_SLQSRegisterIMSAIndication_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.4 int unpack_imsa_SLQSGetIMSARegStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_imsa_SLQSGetIMSARegStatus_t * *pOutput*)

Function to unpack Get IMSA Registration Status response from modem. *

Parameters

pResp	[IN] • Response from modem
respLen	[IN] • Length of pResp from modem
pOutput	[OUT] • See unpack_imsa_SLQSGetIMSARegStatus_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.5 int unpack_imsa_SLQSGetIMSAServiceStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_imsa_SLQSGetIMSAServiceStatus_t * *pOutput*)

Function to unpack Get IMSA Service Status response from modem

Parameters

pResp	[IN] • Response from modem
respLen	[IN] • Length of pResp from modem
pOutput	[OUT] • See unpack_imsa_SLQSGetIMSAServiceStatus_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.6 int unpack_imsa_SLQSImsaPdpStatusCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t * *pOutput*)

Unpack indication about IMSA PDP status info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.7 int unpack_imsa_SLQSImsaRatStatusCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_imsa_SLQSImsaRatStatusCallBack_ind_t * *pOutput*)

Unpack indication about IMSA RAT status info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.8 int unpack_imsa_SLQSImsaRegStatusCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_imsa_SLQSImsaRegStatusCallBack_ind_t * *pOutput*)

Unpack indication about IMSA registration status info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.9 int unpack_imsa_SLQSImsaSvcStatusCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t * *pOutput*)

Unpack indication about IMSA service status info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.13.3.10 int unpack_imsa_SLQSRegisterIMSAIndication (uint8_t * *pResp*, uint16_t *respLen*, unpack_imsa_SLQSRegisterIMSAIndication_t * *pOutput*)

Function to unpack Register IMSA Indication response from modem

Parameters

pResp	[IN] • Response from modem
respLen	[IN] • Length of pResp from modem
pOutput	[OUT] • See unpack_imsa_SLQSRegisterIMSAIndication_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.14 lite-fw.h File Reference

Data Structures

- struct _litefw_FirmwareInfo_
- struct _litefw_FirmwarePartNo_
- struct _litefw_FirmwareFileInfo

Macros

- #define IMG_MASK_MDM (1<<0)
- #define IMG_MASK_PRI (1<<1)
- #define IMG_MASK_CLEAR (0x000)
- #define IMG_MASK_GENERIC (0xFFFF)
- #define LITEFW_CARRIER_PACKAGE_SKU "9999999"
- #define LITEFW_SKU_STRING_LENGTH 7
- #define FIRMWARE_INFO_STRING_SIZE 20
- #define NULL_TERMINATOR_CHAR_SIZE 1
- #define FIRMWARE_IMAGE_SIZE_MAX 512
- #define FIRMWARE_BCHVERSTRSIZE 84

- #define FIRMWARE_BCHDATESIZE 8
- #define MAX_IMAGE_PRODUCT_LENGTH 10
- #define libSDP_GetVersion litefw_GetVersion
- #define libSDP_CalculateImageMask litefw_CalculateImageMask
- #define libSDP_getFileType litefw_getFileType
- #define libSDP_ExtractFirmwareParametersByPath litefw_ExtractFirmwareParametersByPath
- #define libSDP_GetModelFamily litefw_GetModelFamily
- #define libSDP_CheckValidFirmwareInfo litefw_CheckValidFirmwareInfo
- #define libSDP_BuildImagesPreferenceRequest litefw_BuildImagesPreferenceRequest
- #define libSDP_DownloadFW litefw_DownloadFW
- #define libsdp_set_log_func litefw_set_log_func
- #define libsdp_SetReadBlockSize litefw_SetReadBlockSize
- #define LIBSDP_CARRIER_PACKAGE_SKU LITEFW_CARRIER_PACKAGE_SKU
- #define LIBSDP_SKU_STRING_LENGTH LITEFW_SKU_STRING_LENGTH

Typedefs

- typedef struct _litefw_FirmwareInfo_ litefw_FirmwareInfo
- typedef struct _litefw_FirmwarePartNo_ litefw_FirmwarePartNo
- typedef struct _litefw_FirmwareFileInfo litefw_FirmwareFileInfo
- typedef void(∗ litefwlogger)(uint8_t lvl, const char ∗buff)
- typedef litefw_FirmwareInfo libSDP_FirmwareInfo
- typedef litefwlogger libsdplogger

Enumerations

- enum litefw_fwdwl_error_codes {
  eSDP_FWDWL_SUCCESS = 0,
  eSDP_FWDWL_ERR_GENERAL = 100,
  eSDP_FWDWL_ERR_SDK,
  eSDP_FWDWL_ERR_SET_CBK,
  eSDP_FWDWL_ERR_PATH_TOO_LONG,
  eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED,
  eSDP_FWDWL_ERR_FW_UPGRADE,
  eSDP_FWDWL_ERR_INVALID_DEV,
  eSDP_FWDWL_ERR_INVALID_PATH,
  eSDP_FWDWL_ERR_TIMEOUT,
  eSDP_FWDWL_ERR_FAIL,
  eSDP_FWDWL_ERR_PRI_FAIL,
  eSDP_FWDWL_ERR_FW_VERSION_FAIL,
  eSDP_FWDWL_ERR_SDP_TIMEOUT,
  eFIREHOSE_ERR_SECBOOT_INVALID_CERT_CHAIN,
  eFIREHOSE_ERR_RC7611_RETRY_TEST,
  eLITE_FWDWL_APP_ERR_RC7611_ERROR_AFTER_RETRY,
  eSDP_FWDWL_ERR_END }
- enum litefw_Fw_Type {
  eFW_TYPE_MBN_GOBI,
  eFW_TYPE_MBN,
  eFW_TYPE_CWE,
  eFW_TYPE_NVU,
  eFW_TYPE_SPK,
  eFW_TYPE_INVALID,
  eFW_TYPE_CWE_NVU }

- enum litefw_fileimgtype {
  eFILE_TYPE_NONE = 0,
  eFILE_TYPE_CAR_PRI,
  eFILE_TYPE_OEM_PRI,
  eFILE_TYPE_COMPO_PRI }
- enum litefw_imagetype {
  eIMAGE_TYPE_MIN = 0,
  eIMAGE_TYPE_BOOT = eIMAGE_TYPE_MIN,
  eIMAGE_TYPE_APPL,
  eIMAGE_TYPE_APPS,
  eIMAGE_TYPE_FILE,
  eIMAGE_TYPE_SPKG,
  eIMAGE_TYPE_MODM,
  eIMAGE_TYPE_USER,
  eIMAGE_TYPE_KEYS,
  eIMAGE_TYPE_MAX,
  eIMAGE_TYPE_INVALID = 0xFF,
  eIMAGE_TYPE_ANY = eIMAGE_TYPE_INVALID }
- enum litefw_Models {
  eModel_Unknown =-1,
  eModel_9X15 =0,
  eModel_WP9X15 =1,
  eModel_9X30 =2,
  eModel_9x07 =3,
  eModel_9x06 =4,
  eModel_9x05 =5 }
- enum litefw_QDL_MODEs {
  eQDL_MODE_Unknown =-1,
  eQDL_MODE_INIT =0,
  eQDL_MODE_TTYUSB =1,
  eQDL_MODE_UART0 =2 }
- enum litefw_QDL_FLOW_CONTROLs {
  eQDL_HW_FLOW_Unknown =-1,
  eQDL_HW_FLOW_INIT =0,
  eQDL_HW_FLOW_ENABLE =1,
  eQDL_HW_FLOW_DISABLE =2 }

Functions

- void litefw_logsenable (int log_en)
- int litefw_ExtractFirmwareParametersByPath (char *pImagePath, litefw_FirmwareInfo *info)
- int litefw_ExtractFirmwarePartNoByPath (char *pImagePath, litefw_FirmwarePartNo *partno)
- int litefw_BuildImagesPreferenceRequest (litefw_FirmwareInfo info, pack_fms_SetImagesPreference_t *pack)
- int litefw_CalculateImageMask (unpack_fms_SetImagesPreference_t SetPrefRspFromModem)
- int litefw_getFileType (char *szPath)
- unsigned int litefw_DownloadFW (char *pImagePath, char *szTTYPath, int iFWImageType, int image_mask, int iModelFamily)
- int litefw_GetModelFamily (char *pModelString)
- int litefw_CheckValidFirmwareInfo (litefw_FirmwareInfo info)
- char * litefw_GetVersion ()
- int litefw_set_log_func (litefwlogger *func)
- void litefw_SetReadBlockSize (unsigned long lBlockSize)
- int litefw_switch_9x07_to_downloadmode (char *szTTYPath)
- int litefw_switch_to_BootHoldMode (char *szTTYPath)
- int litefw_SetQTLDownloadMode (int iMode)

- int litefw_GetQTLDownloadMode ()
- int litefw_SetQTLHWFlowControl (int iMode)
- int litefw_GetQTLHWFlowControl ()
- enum eQCWWANError litefw_SLQSGetFirmwareFileInfo (char *fullFilePath, litefw_FirmwareFileInfo *info, uint8_t *pNumOfItems)
- void litefw_SetRetryFirehoseSwitch (int mode)
- int litefw_GetRetryFirehoseSwitch ()
- unsigned long GetFlashSectorSize ()
- void SetFlashSectorSize (unsigned long ulSize)

9.14.1 Detailed Description

Filename: lite-fw.h

Purpose: Global definitions used inside the SDK

Copyright: © 2016 Sierra Wireless Inc., all rights reserved

9.14.2 Macro Definition Documentation

9.14.2.1 #define FIRMWARE_BCHDATESIZE 8

9.14.2.2 #define FIRMWARE_BCHVERSTRSIZE 84

9.14.2.3 #define FIRMWARE_IMAGE_SIZE_MAX 512

9.14.2.4 #define FIRMWARE_INFO_STRING_SIZE 20

9.14.2.5 #define IMG_MASK_CLEAR (0x000)

9.14.2.6 #define IMG_MASK_GENERIC (0xFFFF)

9.14.2.7 #define IMG_MASK_MDM (1<<0)

9.14.2.8 #define IMG_MASK_PRI (1<<1)

9.14.2.9 #define libSDP_BuildImagesPreferenceRequest litefw_BuildImagesPreferenceRequest

9.14.2.10 #define libSDP_CalculateImageMask litefw_CalculateImageMask

9.14.2.11 #define LIBSDP_CARRIER_PACKAGE_SKU LITEFW_CARRIER_PACKAGE_SKU

9.14.2.12 #define libSDP_CheckValidFirmwareInfo litefw_CheckValidFirmwareInfo

9.14.2.13 #define libSDP_DownloadFW litefw_DownloadFW

9.14.2.14 #define libSDP_ExtractFirmwareParametersByPath litefw_ExtractFirmwareParametersByPath

9.14.2.15 #define libSDP_getFileType litefw_getFileType

9.14.2.16 #define libSDP_GetModelFamily litefw_GetModelFamily

9.14.2.17 #define libSDP_GetVersion litefw_GetVersion

9.14.2.18 #define libsdp_set_log_func litefw_set_log_func

9.14.2.19 #define libsdp_SetReadBlockSize litefw_SetReadBlockSize

9.14.2.20 #define LIBSDP_SKU_STRING_LENGTH LITEFW_SKU_STRING_LENGTH

9.14.2.21 #define LITEFW_CARRIER_PACKAGE_SKU "9999999"

9.14.2.22 #define LITEFW_SKU_STRING_LENGTH 7

9.14.2.23 #define MAX_IMAGE_PRODUCT_LENGTH 10

9.14.2.24 #define NULL_TERMINATOR_CHAR_SIZE 1

9.14.3 Typedef Documentation

9.14.3.1 typedef litefw_FirmwareInfo libSDP_FirmwareInfo

9.14.3.2 typedef litefwlogger libsdplogger

9.14.3.3 typedef struct _litefw_FirmwareFileInfo litefw_FirmwareFileInfo

This structure provided more detailed information of the provided firmware file

Parameters

fullPath	• full path of the file
imagemask	• Bitmask provides type of file – bit0 - cwe – bit1 - nvu – bit2 - spk
headerType	• see litefw_imagetype
imageType	• see litefw_fileimgtype
modelIdStr	• device model id
partNoStr	• part number id
skuStr	• sku id
packageIdStr	• cwe sierra package id
carrierStr	• carrier id

priVersionStr	• pri version
versionStr	• firmware version
releaseDate	• release date of the file

9.14.3.4 typedef struct _litefw_FirmwareInfo_ litefw_FirmwareInfo

This structure contains information of the provided firmware file

Parameters

szModelid_str	• Model Name String
szFwversion_str	• Firmware Version String.
szSku_str	• SKU String.
szPackageid_str	• Package ID String.
szCarrier_str	• Carrier String.
szCarrier-Priversion_str	• Carrier PRI Version String.

9.14.3.5 typedef struct _litefw_FirmwarePartNo_ litefw_FirmwarePartNo

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

szPartno_str	• Part Number of the provided firmware image

9.14.3.6 typedef void(∗ litefwlogger)(uint8_t lvl, const char ∗buff)

This Custom Log prototype.

Parameters

in	*lvl*	• Log level.
in	*buff*	• Log String.

Returns

none

9.14.4 Enumeration Type Documentation

9.14.4.1 enum litefw_fileimgtype

Enumerator

eFILE_TYPE_NONE

eFILE_TYPE_CAR_PRI

eFILE_TYPE_OEM_PRI

eFILE_TYPE_COMPO_PRI

9.14.4.2 enum litefw_Fw_Type

Enumerator

eFW_TYPE_MBN_GOBI

eFW_TYPE_MBN

eFW_TYPE_CWE

eFW_TYPE_NVU

eFW_TYPE_SPK

eFW_TYPE_INVALID

eFW_TYPE_CWE_NVU

9.14.4.3 enum litefw_fwdwl_error_codes

Enumerator

eSDP_FWDWL_SUCCESS

eSDP_FWDWL_ERR_GENERAL

eSDP_FWDWL_ERR_SDK

eSDP_FWDWL_ERR_SET_CBK

eSDP_FWDWL_ERR_PATH_TOO_LONG

eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED

eSDP_FWDWL_ERR_FW_UPGRADE

eSDP_FWDWL_ERR_INVALID_DEV

eSDP_FWDWL_ERR_INVALID_PATH

eSDP_FWDWL_ERR_TIMEOUT

eSDP_FWDWL_ERR_FAIL

eSDP_FWDWL_ERR_PRI_FAIL

eSDP_FWDWL_ERR_FW_VERSION_FAIL

eSDP_FWDWL_ERR_SDP_TIMEOUT

eFIREHOSE_ERR_SECBOOT_INVALID_CERT_CHAIN

eFIREHOSE_ERR_RC7611_RETRY_TEST

eLITE_FWDWL_APP_ERR_RC7611_ERROR_AFTER_RETRY

eSDP_FWDWL_ERR_END

9.14.4.4 enum litefw_imagetype

Enumerator

eIMAGE_TYPE_MIN

eIMAGE_TYPE_BOOT

eIMAGE_TYPE_APPL

eIMAGE_TYPE_APPS

eIMAGE_TYPE_FILE

eIMAGE_TYPE_SPKG

eIMAGE_TYPE_MODM

eIMAGE_TYPE_USER

eIMAGE_TYPE_KEYS

eIMAGE_TYPE_MAX

eIMAGE_TYPE_INVALID

eIMAGE_TYPE_ANY

9.14.4.5 enum litefw_Models

Enumerator

eModel_Unknown

eModel_9X15

eModel_WP9X15

eModel_9X30

eModel_9x07

eModel_9x06

eModel_9x05

9.14.4.6 enum litefw_QDL_FLOW_CONTROLs

Enumerator

eQDL_HW_FLOW_Unknown

eQDL_HW_FLOW_INIT

eQDL_HW_FLOW_ENABLE

eQDL_HW_FLOW_DISABLE

9.14.4.7 enum litefw_QDL_MODEs

Enumerator

eQDL_MODE_Unknown

eQDL_MODE_INIT

eQDL_MODE_TTYUSB

eQDL_MODE_UART0

9.14.5 Function Documentation

9.14.5.1 unsigned long GetFlashSectorSize ( )

This API get flash sector size.

Returns

0 when flah sector is not set, other flash sector size

9.14.5.2 int litefw_BuildImagesPreferenceRequest (litefw_FirmwareInfo *info*, pack_fms_SetImagesPreference_t * *pack*)

This API Build Image Preference Request Using Firmware Information.

Parameters

in	*info*	• See litefw_FirmwareInfo for more information.
out	*pack*	• See pack_fms_SetImagesPreference_t for more information.

Returns

0 on success, litefw_fwdwl_error_codes error value otherwise

See Also

See litefw_fwdwl_error_codes for error values

9.14.5.3 int litefw_CalculateImageMask (unpack_fms_SetImagesPreference_t *SetPrefRspFromModem*)

This API Calculate Image Mask for Firmware Download.

Parameters

in	*SetPrefRsp-FromModem*	• See unpack_fms_SetImagesPreference_t for more information.

Returns

Image Mask

- IMG_MASK_MDM | IMG_MASK_PRI

9.14.5.4 int litefw_CheckValidFirmwareInfo (litefw_FirmwareInfo *info*)

This API Check Valid Firmware Information to build Image Prefernce Request.

Parameters

in	*info*	• See litefw_FirmwareInfo for more information.

Returns

0 on success, -1 error value otherwise

9.14.5.5 unsigned int litefw_DownloadFW (char * *pImagePath*, char * *szTTYPath*, int *iFWImageType*, int *image_mask*, int *iModelFamily*)

This API Download Firmware.

Parameters

in	*pImagePath*	• Firmware Folder Path.
in	*szTTYPath*	• QDL Device Path.
in	*iFWImageType*	• Firmware Type. • See litefw_Fw_Type
in	*image_mask*	• Image Mask. – IMG_MASK_MDM | IMG_MASK_PRI
in	*iModelFamily*	• Modem Family. • See litefw_Models

Returns

0 on success, litefw_fwdwl_error_codes error value otherwise

See Also

See litefw_fwdwl_error_codes for error values

9.14.5.6 int litefw_ExtractFirmwareParametersByPath (char * *pImagePath,* litefw_FirmwareInfo * *info*)

This API Extrace Firmware Parameters From Path.

Parameters

in	*pImagePath*	• Firmware Folder Path.
out	*info*	• See litefw_FirmwareInfo for more information.

Returns

0 on success, litefw_fwdwl_error_codes error value otherwise

See Also

See litefw_fwdwl_error_codes for error values

9.14.5.7 int litefw_ExtractFirmwarePartNoByPath (char * *pImagePath,* litefw_FirmwarePartNo * *partno*)

This API Extrace Firmware Part Number From Path.

Parameters

in	*pImagePath*	• Firmware Folder Path.
out	*partno*	• See litefw_FirmwarePartNo for more information.

Returns

0 on success, litefw_fwdwl_error_codes error value otherwise

See Also

See litefw_fwdwl_error_codes for error values

9.14.5.8 int litefw_getFileType (char * *szPath*)

This API Get File Type By Path

Parameters

in	*szPath*	• See litefw_FirmwareInfo for more information.

Returns

eFW_TYPE_INVALID on error, litefw_Fw_Type value otherwise

See Also

See litefw_Fw_Type for values

9.14.5.9 int litefw_GetModelFamily (char * *pModelString*)

This API Get Model Famliy from a model string.

Parameters

in	*pModelString*	• Model String.

Returns

eModel_Unknown on Error, litefw_Models value otherwise

See Also

See litefw_Models for values

9.14.5.10 int litefw_GetQTLDownloadMode ( )

This API Get QDL port donwload mode.

Returns

eQDL_MODE_Unknown on Error, litefw_QDL_MODEs value otherwise

9.14.5.11 int litefw_GetQTLHWFlowControl ( )

This API Get QDL port Hardware flow control.

Returns

eQDL_HW_FLOW_Unknown on Error, litefw_QDL_FLOW_CONTROLs value otherwise

9.14.5.12 int litefw_GetRetryFirehoseSwitch ( )

9.14.5.13 char* litefw_GetVersion ( )

This API Get Lib SDP Version.

Returns

Version String

9.14.5.14 void litefw_logsenable (int *log_en*)

This function enable/disable lite firmware debug logs.

Parameters

in	*log_en*	• Logs enable/disable flag.

9.14.5.15 int litefw_set_log_func (litefwlogger * *func*)

This API Set Custom Log function.

Parameters

in	*func*	• See litefwlogger for more information.

Returns

none

9.14.5.16 int litefw_SetQTLDownloadMode (int *iMode*)

This API set QDL port donwload mode.

Parameters

in	*iMode*	• QDL Download Mode. – litefw_QDL_MODEs

Returns

0 on success, -1 error value otherwise

9.14.5.17 int litefw_SetQTLHWFlowControl (int *iMode*)

This API Set QDL port Hardware flow control.

Parameters

in	*iMode*	• litefw_QDL_FLOW_CONTROLs

Returns

0 on success, -1 error value otherwise

9.14.5.18 void litefw_SetReadBlockSize (unsigned long *lBlockSize*)

This API Set Read BlockSize.

Parameters

in	*lBlockSize*	• Firmware Read Block Size.

Returns

none

9.14.5.19 void litefw_SetRetryFirehoseSwitch (int *mode*)

9.14.5.20 enum eQCWWANError litefw_SLQSGetFirmwareFileInfo (char * *fullFilePath*, litefw_FirmwareFileInfo * *info*, uint8_t * *pNumOfItems*)

This API Get the firmware information of the provided image

Parameters

in	*fullFilePath*	• file path or directory of the firmware image
out	*info*	• See litefw_FirmwareFileInfo, application should allocate the memory • for the struct array, this API will fill in the details in the provided storage
out	*pNumOfItems*	• number of litefw_FirmwareFileInfo

Returns

enum eQCWWANError

9.14.5.21 int litefw_switch_9x07_to_downloadmode (char * *szTTYPath*)

This API switch 9x07 modem to download mode.

Parameters

in	*szTTYPath*	• QDL Device Path.

Returns

0 on success, -1 error value otherwise

9.14.5.22 int litefw_switch_to_BootHoldMode (char * *szTTYPath*)

This API switch modem to boot hold modem via QDL port.

Parameters

in	*szTTYPath*	• QDL Device Path.

Returns

0 on success, -1 error value otherwise

9.14.5.23 void SetFlashSectorSize (unsigned long *ulSize*)

This API set flash sector size.

Parameters

in	*ulSize*	• Flash sector size.

Returns

NIL

9.15 loc.h File Reference

Data Structures

- struct loc_LocApplicationInfo
- struct loc_SV
- struct loc_SVInfo
- struct loc_GnssData
- struct loc_CellDb
- struct loc_ClkInfo
- struct loc_BdsSV
- struct loc_BdsSVInfo
- struct pack_loc_EventRegister_t
- struct unpack_loc_EventRegister_t
- struct pack_loc_SetExtPowerState_t
- struct unpack_loc_SetExtPowerState_t
- struct pack_loc_Start_t
- struct unpack_loc_Start_t
- struct pack_loc_Stop_t
- struct unpack_loc_Stop_t
- struct pack_loc_SetOperationMode_t
- struct unpack_loc_SetOperationMode_t
- struct pack_loc_Delete_Assist_Data_t
- struct unpack_loc_Delete_Assist_Data_t
- struct loc_precisionDilution
- struct loc_sensorDataUsage

- struct loc_svUsedforFix
- struct loc_gpsTime
- struct unpack_loc_PositionRpt_Ind_t
- struct unpack_loc_EngineState_Ind_t
- struct unpack_loc_SetExtPowerConfig_Ind_t
- struct unpack_loc_SLQSLOCGetBestAvailPos_t
- struct pack_loc_SLQSLOCGetBestAvailPos_t
- struct unpack_loc_BestAvailPos_Ind_t
- struct unpack_loc_SetOperationMode_Ind_t
- struct unpack_loc_DeleteAssistData_Ind_t
- struct loc_satelliteInfo
- struct unpack_loc_GnssSvInfo_Ind_t
- struct pack_loc_SLQSLOCInjectUTCTime_t
- struct altSrcInfo_t
- struct pack_loc_SLQSLOCInjectPosition_t
- struct pack_loc_SLQSLOCSetCradleMountConfig_t
- struct sensorData_t
- struct tempData_t
- struct pack_loc_SLQSLOCInjectSensorData_t
- struct unpack_loc_EventNMEA_Ind_t
- struct pack_loc_SLQSLOCGetServer_t
- struct loc_IPv4Info
- struct loc_IPv6Info
- struct loc_urlAddr
- struct unpack_loc_GetServer_Ind_t
- struct unpack_loc_CradleMountCallback_Ind_t
- struct unpack_loc_EventTimeSyncCallback_Ind_t
- struct unpack_loc_InjectTimeSyncDataCallback_Ind_t
- struct unpack_loc_InjectSensorDataCallback_Ind_t
- struct loc_accelAcceptReady
- struct loc_gyroAcceptReady
- struct loc_accelTempAcceptReady
- struct loc_gyroTempAcceptReady
- struct unpack_loc_SensorStreamingCallback_Ind_t
- struct unpack_loc_InjectUTCTimeCallback_Ind_t
- struct unpack_loc_InjectPositionCallback_Ind_t
- struct loc_IPv4Config
- struct loc_IPv6Config
- struct loc_URLAddrInfo
- struct pack_loc_SLQSLOCSetServer_t
- struct unpack_loc_SetServer_Ind_t
- struct unpack_loc_SLQSLOCGetOpMode_t
- struct unpack_loc_GetOpMode_Ind_t
- struct loc_FixCriteriaStatusTlv
- struct loc_HorAccuracyLvlTlv
- struct loc_IntermediateRptStateTlv
- struct loc_MinIntervalTlv
- struct loc_AppProviderInfoTlv
- struct unpack_loc_FixCriteria_Ind_t

Macros

- #define LOC_UINT8_MAX_STRING_SZ 255
- #define LOCEVENTMASKPOSITIONREPORT 0x00000001
- #define LOCEVENTMASKGNSSSVINFO 0x00000002
- #define LOCEVENTMASKNMEA 0x00000004
- #define LOCEVENTMASKNINOTIFYVERIFYREQ 0x00000008
- #define LOCEVENTMASKINJECTTIMEREQ 0x00000010
- #define LOCEVENTMASKINJECTPREDICTEDORBITSREQ 0x00000020
- #define LOCEVENTMASKINJECTPOSITIONREQ 0x00000040
- #define LOCEVENTMASKENGINESTATE 0x00000080
- #define LOCEVENTMASKFIXSESSIONSTATE 0x00000100
- #define LOCEVENTMASKWIFIREQ 0x00000200
- #define LOCEVENTMASKSENSORSTREAMINGREADYSTATUS 0x00000400
- #define LOCEVENTMASKTIMESYNCREQ 0x00000800
- #define LOCEVENTMASKSETSPISTREAMINGREPORT 0x00001000
- #define LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ 0x00002000
- #define LOCEVENTMASKNIGEOFENCENOTIFICATION 0x00004000
- #define LOCEVENTMASKGEOFENCEGENALERT 0x00008000
- #define LOCEVENTMASKGEOFENCEBREACHNOTIFICATION 0x00010000
- #define LOCEVENTMASKPEDOMETERCONTROL 0x00020000
- #define LOCEVENTMASKMOTIONDATACONTROL 0x00040000
- #define LOCEVENTMASKBATCHFULLNOTIFICATION 0x00080000
- #define LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT 0x00100000
- #define LOCEVENTMASKINJECTWIFIAPDATAREQ 0x00200000
- #define LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION 0x00400000
- #define LOCEVENTMASKVEHICLEDATAREADYSTATUS 0x00800000
- #define LOCEVENTMASKGNSSMEASUREMENTREPORT 0x01000000
- #define LOCEVENTMASKINVALIDVALUE 0xFFFFFFFF
- #define MAX_SENSOR_DATA_LEN 64
- #define MAX_TEMP_DATA_LEN 64
- #define MAX_LOC_NMEA_STR_LEN 201

Typedefs

- typedef unpack_loc_Delete_Assist_Data_t unpack_loc_DeleteAssistData_t
- typedef unpack_result_t unpack_loc_SLQSLOCInjectUTCTime_t
- typedef unpack_result_t unpack_loc_SLQSLOCInjectPosition_t
- typedef unpack_result_t unpack_loc_SLQSLOCSetCradleMountConfig_t
- typedef unpack_result_t unpack_loc_SLQSLOCInjectSensorData_t
- typedef unpack_result_t unpack_loc_SLQSLOCGetServer_t
- typedef unpack_result_t unpack_loc_GetFixCriteria_t

Enumerations

- enum {
  eQMI_LOC_SESS_STATUS_SUCCESS =0,
  eQMI_LOC_SESS_STATUS_IN_PROGRESS =1,
  eQMI_LOC_SESS_STATUS_FAILURE =2,
  eQMI_LOC_SESS_STATUS_TIMEOUT =3 }

Functions

- int pack_loc_EventRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_EventRegister_t *reqArg)
- int unpack_loc_EventRegister (uint8_t *pResp, uint16_t respLen, unpack_loc_EventRegister_t *pOutput)
- int pack_loc_SetExtPowerState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SetExtPowerState_t *reqArg)
- int unpack_loc_SetExtPowerState (uint8_t *pResp, uint16_t respLen, unpack_loc_SetExtPowerState_t *pOutput)
- int pack_loc_Start (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Start_t *reqArg)
- int unpack_loc_Start (uint8_t *pResp, uint16_t respLen, unpack_loc_Start_t *pOutput)
- int pack_loc_Stop (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Stop_t *reqArg)
- int unpack_loc_Stop (uint8_t *pResp, uint16_t respLen, unpack_loc_Stop_t *pOutput)
- int pack_loc_SetOperationMode (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SetOperationMode_t *reqArg)
- int unpack_loc_SetOperationMode (uint8_t *pResp, uint16_t respLen, unpack_loc_SetOperationMode_t *pOutput)
- int pack_loc_DeleteAssistData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Delete_Assist_Data_t *reqArg)
- int unpack_loc_DeleteAssistData (uint8_t *pResp, uint16_t respLen, unpack_loc_Delete_Assist_Data_t *pOutput)
- int unpack_loc_PositionRpt_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_PositionRpt_Ind_t *pOutput)
- int unpack_loc_EngineState_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_EngineState_Ind_t *pOutput)
- int unpack_loc_SetExtPowerConfig_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_SetExtPowerConfig_Ind_t *pOutput)
- int pack_loc_SLQSLOCGetBestAvailPos (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCGetBestAvailPos_t *reqArg)
- int unpack_loc_SLQSLOCGetBestAvailPos (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCGetBestAvailPos_t *pOutput)
- int unpack_loc_BestAvailPos_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_BestAvailPos_Ind_t *pOutput)
- int unpack_loc_SetOperationMode_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_SetOperationMode_Ind_t *pOutput)
- int unpack_loc_DeleteAssistData_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_DeleteAssistData_Ind_t *pOutput)
- int unpack_loc_GnssSvInfo_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_GnssSvInfo_Ind_t *pOutput)
- int pack_loc_SLQSLOCInjectUTCTime (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectUTCTime_t *reqArg)
- int unpack_loc_SLQSLOCInjectUTCTime (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCInjectUTCTime_t *pOutput)
- int pack_loc_SLQSLOCInjectPosition (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectPosition_t *reqArg)
- int unpack_loc_SLQSLOCInjectPosition (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCInjectPosition_t *pOutput)
- int pack_loc_SLQSLOCSetCradleMountConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCSetCradleMountConfig_t *reqArg)
- int unpack_loc_SLQSLOCSetCradleMountConfig (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCSetCradleMountConfig_t *pOutput)
- int pack_loc_SLQSLOCInjectSensorData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectSensorData_t *reqArg)
- int unpack_loc_SLQSLOCInjectSensorData (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCInjectSensorData_t *pOutput)
- int unpack_loc_EventNMEA_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_EventNMEA_Ind_t *pOutput)
- int pack_loc_SLQSLOCGetServer (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCGetServer_t *reqArg)

- int unpack_loc_SLQSLOCGetServer (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCGetServer_t *pOutput)
- int unpack_loc_GetServer_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_GetServer_Ind_t *pOutput)
- int unpack_loc_CradleMountCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_CradleMountCallback_Ind_t *pOutput)
- int unpack_loc_EventTimeSyncCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_EventTimeSyncCallback_Ind_t *pOutput)
- int unpack_loc_InjectTimeSyncDataCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_InjectTimeSyncDataCallback_Ind_t *pOutput)
- int unpack_loc_InjectSensorDataCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_InjectSensorDataCallback_Ind_t *pOutput)
- int unpack_loc_SensorStreamingCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_SensorStreamingCallback_Ind_t *pOutput)
- int unpack_loc_InjectUTCTimeCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_InjectUTCTimeCallback_Ind_t *pOutput)
- int unpack_loc_InjectPositionCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_InjectPositionCallback_Ind_t *pOutput)
- int pack_loc_SLQSLOCSetServer (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCSetServer_t *reqArg)
- int unpack_loc_SLQSLOCSetServer (uint8_t *pResp, uint16_t respLen)
- int unpack_loc_SetServer_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_SetServer_Ind_t *pOutput)
- int pack_loc_SLQSLOCGetOpMode (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_loc_SLQSLOCGetOpMode (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCGetOpMode_t *pOutput)
- int unpack_loc_GetOpMode_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_GetOpMode_Ind_t *pOutput)
- int pack_loc_GetFixCriteria (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_loc_GetFixCriteria (uint8_t *pResp, uint16_t respLen, unpack_loc_GetFixCriteria_t *pOutput)
- int unpack_loc_FixCriteria_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_FixCriteria_Ind_t *pOutput)

9.15.1 Macro Definition Documentation

9.15.1.1 #define LOC_UINT8_MAX_STRING_SZ 255

9.15.1.2 #define LOCEVENTMASKBATCHFULLNOTIFICATION 0x00080000

The control point must enable this mask to receive notification when a batch is full. The location engine sends this event to notify of Batch Full for ongoing batching session.

9.15.1.3 #define LOCEVENTMASKENGINESTATE 0x00000080

The control point must enable this mask to receive engine state report event indications.

9.15.1.4 #define LOCEVENTMASKFIXSESSIONSTATE 0x00000100

The control point must enable this mask to receive fix session status report event indications.

9.15.1.5 #define LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION 0x00400000

The control point must enable this mask to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach notification is for multiple Geofences. Breaches from multiple Geofences are all batched and sent in the same notification.

9.15.1.6 #define LOCEVENTMASKGEOFENCEBREACHNOTIFICATION 0x00010000

The control point must enable this mask to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach report is for a single Geofence.

9.15.1.7 #define LOCEVENTMASKGEOFENCEGENALERT 0x00008000

The control point must enable this mask to receive Geofence alerts. These alerts are generated to inform the client of the changes that may affect a Geofence, for example, if GPS is turned off or if the network is unavailable.

9.15.1.8 #define LOCEVENTMASKGNSSMEASUREMENTREPORT 0x01000000

The control point must enable this mask to receive system clock and satellite measurement report events (system clock, SV time, Doppler, etc.). Reports are generated only for the GNSS satellite constellations that are enabled using QMI_LOC_SET_GNSS_ CONSTELL_REPORT_CONFIG(Not yet supported).

9.15.1.9 #define LOCEVENTMASKGNSSSVINFO 0x00000002

The control point must enable this mask to receive satellite report event indications. These reports are sent at a 1 Hz rate.

9.15.1.10 #define LOCEVENTMASKINJECTPOSITIONREQ 0x00000040

The control point must enable this mask to receive position injection request event indications.

9.15.1.11 #define LOCEVENTMASKINJECTPREDICTEDORBITSREQ 0x00000020

The control point must enable this mask to receive predicted orbits request event indications.

9.15.1.12 #define LOCEVENTMASKINJECTTIMEREQ 0x00000010

The control point must enable this mask to receive time injection request event indications.

9.15.1.13 #define LOCEVENTMASKINJECTWIFIAPDATAREQ 0x00200000

The control point must enable this mask to receive Wi-Fi Access Point (AP) data inject request event indications.

9.15.1.14 #define LOCEVENTMASKINVALIDVALUE 0xFFFFFFFF

Invalid Event Mask

9.15.1.15 #define LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT 0x00100000

The control point must enable this mask to receive position report indications along with an ongoing batching session. The location engine sends this event to notify the batched position report while a batching session is ongoing.

9.15.1.16 #define LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ 0x00002000

The control point must enable this mask to receive location server requests. These requests are generated when the service wishes to establish a connection with a location server.

9.15.1.17 #define LOCEVENTMASKMOTIONDATACONTROL 0x00040000

The control point must enable this mask to register for motion data control requests from the location engine. The location engine sends this event to control the injection of motion data.

9.15.1.18 #define LOCEVENTMASKNIGEOFENCENOTIFICATION 0x00004000

The control point must enable this mask to receive notifications related to network-initiated Geofences. These events notify the client when a network-initiated Geofence is added, deleted, or edited.

9.15.1.19 #define LOCEVENTMASKNINOTIFYVERIFYREQ 0x00000008

The control point must enable this mask to receive NI Notify/Verify request event indications.

9.15.1.20 #define LOCEVENTMASKNMEA 0x00000004

The control point must enable this mask to receive NMEA reports for position and satellites in view. The report is at a 1 Hz rate.

9.15.1.21 #define LOCEVENTMASKPEDOMETERCONTROL 0x00020000

The control point must enable this mask to register for pedometer control requests from the location engine. The location engine sends this event to control the injection of pedometer reports.

9.15.1.22 #define LOCEVENTMASKPOSITIONREPORT 0x00000001

The control point must enable this mask to receive position report event indications.

9.15.1.23 #define LOCEVENTMASKSENSORSTREAMINGREADYSTATUS 0x00000400

The control point must enable this mask to receive notifications from the location engine indicating its readiness to accept data from the sensors (accelerometer, gyroscope, etc.).

9.15.1.24 #define LOCEVENTMASKSETSPISTREAMINGREPORT 0x00001000

The control point must enable this mask to receive Stationary Position Indicator (SPI) streaming report indications.

9.15.1.25 #define LOCEVENTMASKTIMESYNCREQ 0x00000800

The control point must enable this mask to receive time sync requests from the GPS engine. Time sync enables the GPS engine to synchronize its clock with the sensor processor's clock.

9.15.1.26 #define LOCEVENTMASKVEHICLEDATAREADYSTATUS 0x00800000

The control point must enable this mask to receive notifications from the location engine indicating its readiness to accept vehicle data (vehicle accelerometer, vehicle angular rate, vehicle odometry, etc.).

9.15.1.27 #define LOCEVENTMASKWIFIREQ 0x00000200

The control point must enable this mask to receive Wi-Fi position request event indications.

9.15.1.28 #define MAX_LOC_NMEA_STR_LEN 201

9.15.1.29 #define MAX_SENSOR_DATA_LEN 64

9.15.1.30 #define MAX_TEMP_DATA_LEN 64

9.15.2 Typedef Documentation

9.15.2.1 typedef unpack_loc_Delete_Assist_Data_t unpack_loc_DeleteAssistData_t

9.15.2.2 typedef unpack_result_t unpack_loc_GetFixCriteria_t

9.15.2.3 typedef unpack_result_t unpack_loc_SLQSLOCGetServer_t

9.15.2.4 typedef unpack_result_t unpack_loc_SLQSLOCInjectPosition_t

9.15.2.5 typedef unpack_result_t unpack_loc_SLQSLOCInjectSensorData_t

9.15.2.6 typedef unpack_result_t unpack_loc_SLQSLOCInjectUTCTime_t

9.15.2.7 typedef unpack_result_t unpack_loc_SLQSLOCSetCradleMountConfig_t

9.15.3 Enumeration Type Documentation

9.15.3.1 anonymous enum

Enumerator

eQMI_LOC_SESS_STATUS_SUCCESS

eQMI_LOC_SESS_STATUS_IN_PROGRESS

eQMI_LOC_SESS_STATUS_FAILURE

eQMI_LOC_SESS_STATUS_TIMEOUT

9.15.4 Function Documentation

9.15.4.1 int pack_loc_DeleteAssistData (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_Delete_Assist_Data_t * *reqArg*)

Delete Assistant Data pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.2 int pack_loc_EventRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_EventRegister_t * *reqArg*)

Event Register pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.3 int pack_loc_GetFixCriteria (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Pack Get Fix Criteria.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.4 int pack_loc_SetExtPowerState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SetExtPowerState_t * *reqArg*)

Set Ext Power State pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.5 int pack_loc_SetOperationMode (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SetOperationMode_t * *reqArg*)

Set Operation Mode pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.6 int pack_loc_SLQSLOCGetBestAvailPos (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCGetBestAvailPos_t * *reqArg*)

Get Best Avail position pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.7 int pack_loc_SLQSLOCGetOpMode (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Pack Get Operation mode.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.8 int pack_loc_SLQSLOCGetServer (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCGetServer_t * *reqArg*)

Pack get server.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

The request is acknowledged through the response, but the SUCCESS/FAILURE status is sent through unpack_loc_GetServer_Ind callback If successful, the callback also contains the A-GPS server address.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.9 int pack_loc_SLQSLOCInjectPosition (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCInjectPosition_t * *reqArg*)

Pack inject position.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.10 int pack_loc_SLQSLOCInjectSensorData (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SLQSLOCInjectSensorData_t * reqArg)

Pack inject sensor data.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.11 int pack_loc_SLQSLOCInjectUTCTime (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SLQSLOCInjectUTCTime_t * reqArg)

Pack inject UTC time.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.12 int pack_loc_SLQSLOCSetCradleMountConfig (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_SLQSLOCSetCradleMountConfig_t * reqArg)

Pack set cradle mount configure.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.13 int pack_loc_SLQSLOCSetServer (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCSetServer_t * *reqArg*)

Pack Set server.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

If multiple types of addresses are specified in the request, the IPv4 address takes precedence over the IPv6 address and the IPv6 address takes precedence over the URL address.The request is acknowledged through the response, but the SUCCESS/FAILURE status is sent through unpack_loc_SetServer_Ind callback

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.14 int pack_loc_Start (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_Start_t * *reqArg*)

LOC Start pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.15 int pack_loc_Stop (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_loc_Stop_t * reqArg)

Loc Stop pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.16 int unpack_loc_BestAvailPos_Ind (uint8_t * pResp, uint16_t respLen, unpack_loc_BestAvailPos_Ind_t * pOutput)

Loc Best Avial position Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.17 int unpack_loc_CradleMountCallback_Ind (uint8_t * pResp, uint16_t respLen, unpack_loc_CradleMountCallback_Ind_t * pOutput)

Unpack the Cradle Mount Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.18 int unpack_loc_DeleteAssistData (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Delete_Assist_Data_t * *pOutput*)

Delete Assistant Data unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.19 int unpack_loc_DeleteAssistData_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_DeleteAssistData_Ind_t * *pOutput*)

Unpack the status of delete the location engine assistance data

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.20 int unpack_loc_EngineState_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EngineState_Ind_t * *pOutput*)

Loc Engine State Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.21 int unpack_loc_EventNMEA_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EventNMEA_Ind_t * *pOutput*)

Unpack the NMEA Event Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.22 int unpack_loc_EventRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EventRegister_t * *pOutput*)

Event Register unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.23 int unpack_loc_EventTimeSyncCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EventTimeSyncCallback_Ind_t * *pOutput*)

Unpack the Event Time Sync indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.24 int unpack_loc_FixCriteria_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_FixCriteria_Ind_t * *pOutput*)

Unpack the Get Fix Criteria Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.25 int unpack_loc_GetFixCriteria (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_GetFixCriteria_t * *pOutput*)

Get Fix Criteria unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.26 int unpack_loc_GetOpMode_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_GetOpMode_Ind_t * *pOutput*)

Unpack the Set Server Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.27 int unpack_loc_GetServer_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_GetServer_Ind_t * *pOutput*)

Unpack the Get Server Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.28 int unpack_loc_GnssSvInfo_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_GnssSvInfo_Ind_t * *pOutput*)

Unpack the GNSS SV Info Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.29 int unpack_loc_InjectPositionCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_InjectPositionCallback_Ind_t * *pOutput*)

Unpack the Inject position indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.30 int unpack_loc_InjectSensorDataCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_InjectSensorDataCallback_Ind_t * *pOutput*)

Unpack the inject sensor data indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.31 int unpack_loc_InjectTimeSyncDataCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_InjectTimeSyncDataCallback_Ind_t * *pOutput*)

Unpack the inject time sync data indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.32 int unpack_loc_InjectUTCTimeCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_InjectUTCTimeCallback_Ind_t * *pOutput*)

Unpack the Inject UTC time indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.33 int unpack_loc_PositionRpt_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_PositionRpt_Ind_t * *pOutput*)

Loc Position Report Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.34 int unpack_loc_SensorStreamingCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SensorStreamingCallback_Ind_t * *pOutput*)

Unpack the Sensor streaming status indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.35 int unpack_loc_SetExtPowerConfig_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetExtPowerConfig_Ind_t * *pOutput*)

Loc Set External Power Configure Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.36 int unpack_loc_SetExtPowerState (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetExtPowerState_t * *pOutput*)

Set Ext Power State unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.37 int unpack_loc_SetOperationMode (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetOperationMode_t * *pOutput*)

Set Operation Mode unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.38 int unpack_loc_SetOperationMode_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetOperationMode_Ind_t * *pOutput*)

Unpack the engine to use the specified operation mode.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.39 int unpack_loc_SetServer_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetServer_Ind_t * *pOutput*)

Unpack the Set Server Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.40 int unpack_loc_SLQSLOCGetBestAvailPos (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCGetBestAvailPos_t * *pOutput*)

Get Best Avail position unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.41 int unpack_loc_SLQSLOCGetOpMode (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCGetOpMode_t * *pOutput*)

Get Operation mode unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.42 int unpack_loc_SLQSLOCGetServer (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCGetServer_t * *pOutput*)

Unpack get server.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.43 int unpack_loc_SLQSLOCInjectPosition (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCInject-Position_t * *pOutput*)

Unpack inject position.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.44 int unpack_loc_SLQSLOCInjectSensorData (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCInjectSensorData_t * *pOutput*)

Unpack inject sensor data.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.45 int unpack_loc_SLQSLOCInjectUTCTime (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCInjectUTCTime_t * *pOutput*)

Unpack inject UTC time.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.46 int unpack_loc_SLQSLOCSetCradleMountConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCSetCradleMountConfig_t * *pOutput*)

Unpack set cradle mount configure.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.47 int unpack_loc_SLQSLOCSetServer (uint8_t * *pResp*, uint16_t *respLen*)

Unpack set server.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.48 int unpack_loc_Start (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Start_t * *pOutput*)

Loc Start unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.15.4.49 int unpack_loc_Stop (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Stop_t * *pOutput*)

Loc Stop unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.16 MbimIndicator.h File Reference

Data Structures

- struct MbimIndicator

Typedefs

- typedef void(* MBIM_INDICATION_CALLBACK)(uint8_t *deviceServiceId, uint32_t cid, uint8_t *informationBuffer, uint32_t informationBufferLength, void *pIndicationCallbackContext)

- typedef struct MbimIndicator MbimIndicator

Functions

- void MbimIndicator_Initialize (MbimIndicator *pThis, MBIM_INDICATION_CALLBACK pIndicationCallback, void *pIndicationCallbackContext)

9.16.1 Typedef Documentation

9.16.1.1 typedef void(* MBIM_INDICATION_CALLBACK)(uint8_t *deviceServiceId, uint32_t cid, uint8_t *informationBuffer, uint32_t informationBufferLength, void *pIndicationCallbackContext)

This funcion prototype represents the callback which is called by the MBIM transport when a MBIM indication is received.

Parameters

in	*informationBuffer*	Indication's informationBuffer.
in	*information-BufferLength*	Number of bytes in indication's informationBuffer.
in	*pIndication-CallbackContext*	User context supplied when indicator was initialized.

9.16.1.2 typedef struct MbimIndicator MbimIndicator

This structure abstracts a MBIM indication callback. Users who wish to be notified of MBIM indications can register one of these objects with the MbimTransport object using MbimTransport_AttachIndicator/MbimTransport_Detach-Indicator calls.

Parameters

pPrev	• A link so that this object can be stored in a linked list by the transport layer. This points to the previous object or NULL if this is the first object in the list.
pNext	• A link so that this object can be stored in a linked list by the transport layer. This points to the next object or NULL if this is the last object in the list.
pIndication-Callback	• User callback for MBIM indication.
pIndication-CallbackContext	• User context for user callback.

9.16.2 Function Documentation

9.16.2.1 void MbimIndicator_Initialize (MbimIndicator * *pThis*, MBIM_INDICATION_CALLBACK *pIndicationCallback*, void * *pIndicationCallbackContext*)

Initialize the indicator object.

Parameters

in	*pThis*	The primary object of this call.
in	*pIndication-Callback*	User function to be be called when MBIM indication is received.
in	*pIndication-CallbackContext*	A user context for callback.

9.17 MbimInternal.h File Reference

Data Structures

- struct MBIM_MESSAGE_HEADER
- struct MBIM_FRAGMENT_HEADER
- struct MBIM_OPEN_MSG
- struct MBIM_OPEN_DONE_MSG
- struct MBIM_CLOSE_MSG
- struct MBIM_CLOSE_DONE_MSG
- struct MBIM_COMMAND_MSG
- struct MBIM_COMMAND_DONE_MSG
- struct MBIM_INDICATE_STATUS_MSG
- struct MBIM_FRAGMENT_MSG

Macros

- #define MBIM_OPEN 0x1
- #define MBIM_OPEN_DONE 0x80000001
- #define MBIM_CLOSE 0x2
- #define MBIM_CLOSE_DONE 0x80000002
- #define MBIM_COMMAND 0x3
- #define MBIM_COMMAND_DONE 0x80000003
- #define MBIM_HOST_ERROR 0x4
- #define MBIM_FUNCTION_ERROR 0x80000004
- #define MBIM_INDICATE_STATUS 0x80000007

Typedefs

- typedef struct MBIM_MESSAGE_HEADER MBIM_MESSAGE_HEADER
- typedef struct MBIM_FRAGMENT_HEADER MBIM_FRAGMENT_HEADER
- typedef struct MBIM_OPEN_MSG MBIM_OPEN_MSG
- typedef struct MBIM_OPEN_DONE_MSG MBIM_OPEN_DONE_MSG
- typedef struct MBIM_CLOSE_MSG MBIM_CLOSE_MSG
- typedef struct MBIM_CLOSE_DONE_MSG MBIM_CLOSE_DONE_MSG
- typedef struct MBIM_COMMAND_MSG MBIM_COMMAND_MSG
- typedef struct MBIM_COMMAND_DONE_MSG MBIM_COMMAND_DONE_MSG
- typedef struct MBIM_INDICATE_STATUS_MSG MBIM_INDICATE_STATUS_MSG
- typedef struct MBIM_FRAGMENT_MSG MBIM_FRAGMENT_MSG

9.17.1 Macro Definition Documentation

9.17.1.1 #define MBIM_CLOSE 0x2

9.17.1.2 #define MBIM_CLOSE_DONE 0x80000002

9.17.1.3 #define MBIM_COMMAND 0x3

9.17.1.4 #define MBIM_COMMAND_DONE 0x80000003

9.17.1.5 #define MBIM_FUNCTION_ERROR 0x80000004

9.17.1.6 #define MBIM_HOST_ERROR 0x4

9.17.1.7 #define MBIM_INDICATE_STATUS 0x80000007

9.17.1.8 #define MBIM_OPEN 0x1

9.17.1.9 #define MBIM_OPEN_DONE 0x80000001

9.17.2 Typedef Documentation

9.17.2.1 typedef struct MBIM_CLOSE_DONE_MSG MBIM_CLOSE_DONE_MSG

9.17.2.2 typedef struct MBIM_CLOSE_MSG MBIM_CLOSE_MSG

9.17.2.3 typedef struct MBIM_COMMAND_DONE_MSG MBIM_COMMAND_DONE_MSG

9.17.2.4 typedef struct MBIM_COMMAND_MSG MBIM_COMMAND_MSG

9.17.2.5 typedef struct MBIM_FRAGMENT_HEADER MBIM_FRAGMENT_HEADER

9.17.2.6 typedef struct MBIM_FRAGMENT_MSG MBIM_FRAGMENT_MSG

9.17.2.7 typedef struct MBIM_INDICATE_STATUS_MSG MBIM_INDICATE_STATUS_MSG

9.17.2.8 typedef struct MBIM_MESSAGE_HEADER MBIM_MESSAGE_HEADER

9.17.2.9 typedef struct MBIM_OPEN_DONE_MSG MBIM_OPEN_DONE_MSG

9.17.2.10 typedef struct MBIM_OPEN_MSG MBIM_OPEN_MSG

9.18 MbimLayer.h File Reference

Functions

- void * MbimLayer_Create (QmuxTransport *pOwner, char *devicePath, void *pInterfaceSpecific)
- void MbimLayer_ShutDown (void *pLayer)
- int MbimLayer_SendQmuxPacket (void *pLayer, uint8_t qmiServiceType, uint8_t qmiClientId, uint16_t qmiTransactionId, uint8_t *qmuxPacket, uint16_t qmuxPacketSize)

9.18.1 Function Documentation

9.18.1.1 void* MbimLayer_Create (QmuxTransport * *pOwner,* char * *devicePath,* void * *pInterfaceSpecific*)

9.18.1.2 int MbimLayer_SendQmuxPacket (void * *pLayer,* uint8_t *qmiServiceType,* uint8_t *qmiClientId,* uint16_t *qmiTransactionId,* uint8_t * *qmuxPacket,* uint16_t *qmuxPacketSize*)

9.18.1.3 void MbimLayer_ShutDown (void * *pLayer*)

9.19 MbimSyncObject.h File Reference

Data Structures

- struct MbimSyncObject

Typedefs

- typedef struct MbimSyncObject MbimSyncObject

Functions

- int MbimSyncObject_Initialize (MbimSyncObject *pThis, uint8_t *informationBuffer, uint32_t informationBufferLength)
- int MbimSyncObject_Destroy (MbimSyncObject *pThis)
- int MbimSyncObject_Lock (MbimSyncObject *pThis)
- int MbimSyncObject_Unlock (MbimSyncObject *pThis)
- int MbimSyncObject_Wait (MbimSyncObject *pThis)
- int MbimSyncObject_TimedWait (MbimSyncObject *pThis, time_t timeoutSeconds)
- int MbimSyncObject_Signal (MbimSyncObject *pThis)
- void MbimSyncObject_DoneCallback (uint32_t status, uint32_t transactionId, uint8_t *informationBuffer, uint32_t informationBufferLength, void *pDoneCallbackContext)

9.19.1 Typedef Documentation

9.19.1.1 typedef struct MbimSyncObject MbimSyncObject

This structure is a wrapper around a mutex and condition object and an external buffer. It is a convenience object which may be used to create synchronous MBIM APIs.

Parameters

mutex	• For thread safety.
condition	• Triggered on completion.
status	• The returned MBIM transaction status.
informationBuffer	• Buffer to which the transaction response's informationBuffer will be copied.
informationBufferLength	• Set to maximum expected size in bytes of transaction response's informationBuffer. On request completion, it is set to size of transaction response's informationBuffer.

9.19.2 Function Documentation

9.19.2.1 int MbimSyncObject_Destroy (MbimSyncObject * *pThis*)

Destroy sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, < 0 on failure.

9.19.2.2 void MbimSyncObject_DoneCallback (uint32_t *status,* uint32_t *transactionId,* uint8_t * *informationBuffer,* uint32_t *informationBufferLength,* void * *pDoneCallbackContext*)

This funcion copies MBIM command response's informationBuffer to this object's informationBuffer and signals the waiting thread.

Parameters

in	*status*	MBIM status of the MBIM transaction.
in	*transactionId*	Transaction ID of associated MBIM transaction.
in	*informationBuffer*	Transaction response's informationBuffer.
in	*information-BufferLength*	Number of bytes in transaction response's informationBuffer.
in	*pDoneCallback-Context*	Pointer to this sync object.

9.19.2.3 int MbimSyncObject_Initialize (MbimSyncObject * *pThis,* uint8_t * *informationBuffer,* uint32_t *informationBufferLength*)

Initialize sync object.

Parameters

in	*pThis*	• The primary object of this call.
in	*informationBuffer*	• Buffer to which the transaction response's informationBuffer will be copied.
in	*information-BufferLength*	• Maximum expected size in bytes of transaction response's informationBuffer.

Returns

0 on success, < 0 on failure.

9.19.2.4 int MbimSyncObject_Lock (MbimSyncObject * *pThis*)

Lock sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, $<$ 0 on failure.

9.19.2.5 int MbimSyncObject_Signal (MbimSyncObject * *pThis*)

Signal sync object. Waiting thread should wake up.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, ETIMEDOUT on timeout, or other negative on failure.

9.19.2.6 int MbimSyncObject_TimedWait (MbimSyncObject * *pThis,* time_t *timeoutSeconds*)

Block, waiting on sync object for a number of seconds.

Parameters

in	*pThis*	The primary object of this call.
in	*timeoutSeconds*	Timeout in seconds before abondoning wait.

Returns

0 on success, $<$ 0 on failure.

9.19.2.7 int MbimSyncObject_Unlock (MbimSyncObject * *pThis*)

Unlock sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, $<$ 0 on failure.

9.19.2.8 int MbimSyncObject_Wait (MbimSyncObject * *pThis*)

Block, waiting on sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, $<$ 0 on failure.

9.20 MbimTransaction.h File Reference

Data Structures

- struct MbimTransaction

Typedefs

- typedef void(∗ MBIM_DONE_CALLBACK)(uint32_t status, uint32_t transactionId, uint8_t ∗informationBuffer, uint32_t informationBufferLength, void ∗pDoneCallbackContext)
- typedef struct MbimTransaction MbimTransaction

Functions

- void MbimTransaction_Initialize (MbimTransaction ∗pThis, uint32_t transactionId, MBIM_DONE_CALLBACK pDoneCallback, void ∗pDoneCallbackContext)

9.20.1 Typedef Documentation

9.20.1.1 typedef void(∗ MBIM_DONE_CALLBACK)(uint32_t status, uint32_t transactionId, uint8_t ∗informationBuffer, uint32_t informationBufferLength, void ∗pDoneCallbackContext)

This function prototype represents the callback which is called by the MBIM transport when a MBIM transaction completes.

Parameters

in	*status*	MBIM status of the MBIM transaction.
in	*transactionId*	Transaction ID of associated MBIM transaction.
in	*informationBuffer*	Transaction response's informationBuffer.
in	*information-BufferLength*	Number of bytes in transaction response's informationBuffer.
in	*pDoneCallback-Context*	User context supplied when transaction was initialized.

9.20.1.2 typedef struct MbimTransaction MbimTransaction

This structure abstracts a MBIM transaction. It is responsible for tracking a MBIM request to its response.

Parameters

pPrev	• A link so that this object can be stored in a linked list by the transport layer. This points to the previous object or NULL if this is the first object in the list.
pNext	• A link so that this object can be stored in a linked list by the transport layer. This points to the next object or NULL if this is the last object in the list.

transactionId	• MBIM transaction ID. Can never be 0.
pDoneCallback	• User callback for MBIM response.
pDoneCallback-Context	• User context for user callback.
status	• MBIM status of response, typically contained in first response fragment.

9.20.2 Function Documentation

9.20.2.1 void MbimTransaction_Initialize (MbimTransaction * *pThis*, uint32_t *transactionId*, MBIM_DONE_CALLBACK *pDoneCallback*, void * *pDoneCallbackContext*)

Initialize MBIM transaction.

Parameters

in	*pThis*	The primary object of this call.
in	*transactionId*	Transaction ID obtained from a call to MbimTransport_GetNextTransactionId.
in	*pDoneCallback*	User function to be be called when MBIM response is received.
in	*pDoneCallback-Context*	A user context for callback.

9.21 MbimTransport.h File Reference

Data Structures

- struct MbimTransport

Macros

- #define MBIM_UUID_SIZE 16
- #define MBIM_MAX_CIDS 64
- #define MBIM_MAX_CTRL_TRANSFER 4096
- #define MBIM_COMMAND_TYPE_QUERY 0
- #define MBIM_COMMAND_TYPE_SET 1
- #define MBIM_STATUS_SUCCESS 0
- #define MBIM_STATUS_FAILURE 2

Typedefs

- typedef struct MbimTransport MbimTransport

Functions

- int MbimTransport_Initialize (MbimTransport *pThis, char *devicePath, uint32_t maxExpectedInformation-Length)

- void MbimTransport_ShutDown (MbimTransport *pThis)
- int MbimTransport_SendCommand (MbimTransport *pThis, const uint8_t *deviceServiceId, uint32_t cid, uint32_t commandType, uint8_t *informationBuffer, uint16_t informationBufferLength, MbimTransaction *pTransaction)
- uint32_t MbimTransport_GetNextTransactionId (MbimTransport *pThis)
- void MbimTransport_CancelTransaction (MbimTransport *pThis, uint32_t transactionId)
- void MbimTransport_AttachIndicator (MbimTransport *pThis, MbimIndicator *pIndicator)
- void MbimTransport_DetachIndicator (MbimTransport *pThis, MbimIndicator *pIndicator)

9.21.1 Macro Definition Documentation

9.21.1.1 #define MBIM_COMMAND_TYPE_QUERY 0

MBIM query command

9.21.1.2 #define MBIM_COMMAND_TYPE_SET 1

MBIM set command

9.21.1.3 #define MBIM_MAX_CIDS 64

Max CIDs per device service. Chosen with some head-room. Current max is 25.

9.21.1.4 #define MBIM_MAX_CTRL_TRANSFER 4096

This value should be determined at run-time.

9.21.1.5 #define MBIM_STATUS_FAILURE 2

Standard MBIM return code failure.

9.21.1.6 #define MBIM_STATUS_SUCCESS 0

Standard MBIM return code success.

9.21.1.7 #define MBIM_UUID_SIZE 16

Number of bytes in a MBIM device service's UUID

9.21.2 Typedef Documentation

9.21.2.1 typedef struct MbimTransport MbimTransport

This structure abstracts a MBIM transport layer. It is responsible for applying MBIM headers to packets destined for MBIM device services and forwarding them to the device. It is responsible for receiving MBIM packets from the device. On the basis of MBIM headers, it assembles multi-fragment responses and indications and delivers them back to user.

Parameters

deviceFd	• File descriptor of device.
shutdownFd	• Event descriptor. Used to signal read thread to terminate.
writeLock	• Provides mutual exclusion for write operations.
readThread	• Thread responsible for reading from device.
transactionId	• MBIM transaction ID. Upper limit 0xffffffff. Never 0. Increments for each transaction.
transactionId-Lock	• Mutex to make transactionId access thread-safe.
transactionList	• Head of linked list of outstanding transactions.
transactionList-Lock	• Provides thread safety for the transaction list.
pIncoming-Message	• Private workspace to assemble incoming information fragments.
indicatorList	• Head of linked list of indicators which are called when MBIM indications are received.
indicatorListLock	• Provides thread safety for the indicators list.

9.21.3 Function Documentation

9.21.3.1 void MbimTransport_AttachIndicator (MbimTransport * *pThis,* MbimIndicator * *pIndicator*)

Attaches a MBIM indicator to this MBIM transport object.

Parameters

in	*pThis*	The primary object of this call.
in	*pIndicator*	Indicator to be attached.

9.21.3.2 void MbimTransport_CancelTransaction (MbimTransport * *pThis,* uint32_t *transactionId*)

Cancel an outstanding MBIM transaction.

Parameters

in	*pThis*	The primary object of this call.
in	*transactionId*	ID of transaction to be cancelled.

9.21.3.3 void MbimTransport_DetachIndicator (MbimTransport * *pThis,* MbimIndicator * *pIndicator*)

Detaches a MBIM indicator to this MBIM transport object.

Parameters

in	*pThis*	The primary object of this call.
in	*pIndicator*	Indicator to be detached.

9.21.3.4 uint32_t MbimTransport_GetNextTransactionId (MbimTransport * *pThis*)

Returns next transaction ID for MBIM transport.

Parameters

in	*pThis*	The primary object of this call.

Returns

transaction ID

9.21.3.5 int MbimTransport_Initialize (MbimTransport * *pThis,* char * *devicePath,* uint32_t *maxExpectedInformationLength*)

Initialize this MBIM transport object.

Parameters

in	*pThis*	The primary object of this call.
in	*devicePath*	Absolute path to device.
in	*maxExpected-Information-Length*	Maximum expected length of information content.

Returns

0 on success, $<$ 0 on failure.

Note

maxExpectedInformationLength is used to allocate a buffer for assembling fragments for incoming command responses and indications.

9.21.3.6 int MbimTransport_SendCommand (MbimTransport * *pThis,* const uint8_t * *deviceServiceId,* uint32_t *cid,* uint32_t *commandType,* uint8_t * *informationBuffer,* uint16_t *informationBufferLength,* MbimTransaction * *pTransaction*)

Send a MBIM command to the device and return immediately.

Parameters

in	*pThis*	The primary object of this call.
in	*deviceServiceId*	Device service UUID which is recipient of command.
in	*cid*	Command ID (CID) which is specific to that device service.

in	*commandType*	1 = set, 0 = get.
in	*informationBuffer*	Buffer of information content.
in	*information-BufferLength*	Size in bytes of information content.
in	*pTransaction*	Transaction object which will track the command/response.

Returns

0 on success, $<$ 0 on failure.

Note

pTransaction must have been initialized with MbimTransaction_Initialize prior to this call.

9.21.3.7 void MbimTransport_ShutDown (MbimTransport * *pThis*)

Shut down this MBIM transport object.

Parameters

in	*pThis*	The primary object of this call.

9.22 nas.h File Reference

Data Structures

- struct unpack_nas_GetSignalStrengths_t
- struct unpack_nas_SLQSGetSysSelectionPref_t
- struct nas_netSelectionPref
- struct nas_acqOrderPref
- struct nas_CSGID
- struct pack_nas_SLQSSetSysSelectionPref_t
- struct nas_lteBandPrefExt
- struct nas_ciotAcqOrderPref
- struct nas_nr5gBandPref
- struct pack_nas_SLQSSetSysSelectionPrefExt_t
- struct pack_nas_SLQSSetBandPreference_t
- struct pack_nas_SLQSNasIndicationRegisterExt_t
- struct pack_nas_SLQSNasIndicationRegisterV2_t
- struct RFBandInfoElements
- struct unpack_nas_GetRFInfo_t
- struct cdmaSSInfo
- struct hdrSSInfo
- struct lteSSInfo
- struct tdscdmaSigInfoExt
- struct unpack_nas_SLQSNasGetSigInfo_t
- struct unpack_nas_SLQSNasSigInfoCallback_ind_t
- struct unpack_nas_GetHomeNetwork_t
- struct nas_homeNwMNC3GppTlv
- struct nas_nwNameSrc3GppTlv
- struct unpack_nas_SLQSGetHomeNetwork_t
- struct nas_SrvStatusInfo

- struct nas_GSMSrvStatusInfo
- struct nas_sysInfoCommon
- struct nas_CDMASysInfo
- struct nas_HDRSysInfo
- struct nas_GSMSysInfo
- struct nas_WCDMASysInfo
- struct nas_LTESysInfo
- struct nas_AddCDMASysInfo
- struct nas_AddSysInfo
- struct nas_CallBarringSysInfo
- struct nas_LteCiotOpModeTlv
- struct nas_NR5GSerStatTlv
- struct nas_NR5GSystemInfoTlv
- struct nas_NR5GCellStatusTlv
- struct unpack_nas_SLQSGetSysInfo_t
- struct nas_LteEmbmsCoverageTlv
- struct nas_SimRejInfoTlv
- struct nas_ImsVoiceSupportLteTlv
- struct nas_LteVoiceDomainTlv
- struct nas_SrvRegRestrictionTlv
- struct nas_LteRegDomainTlv
- struct nas_LteEmbmsTraceIdTlv
- struct nas_NR5GSrvStatusTlv
- struct nas_NR5GSysInfoTlv
- struct nas_NR5GCellStatusInfoTlv
- struct unpack_nas_SLQSGetSysInfoV2_t
- struct unpack_nas_SLQSSysInfoCallback_ind_t
- struct unpack_nas_GetServingNetwork_t
- struct unpack_nas_GetServingNetworkCapabilities_t
- struct nas_QmiNas3GppNetworkInfo
- struct nas_QmiNas3GppNetworkRAT
- struct nas_QmisNasPcsDigit
- struct nas_QmisNasSlqsNasPCICellInfo
- struct nas_QmisNasSlqsNasPCIInfo
- struct nas_lteOpModeTlv
- struct unpack_nas_PerformNetworkScan_t
- struct nas_networkNameSrcTlv
- struct unpack_nas_SLQSPerformNetworkScanV2_t
- struct unpack_nas_SLQSSwiGetLteCQI_t
- struct nas_CommInfo
- struct nas_LTEInfo
- struct unpack_nas_SLQSNasSwiModemStatus_t
- struct nas_servSystem
- struct nas_dataSrvCapabilities
- struct nas_currentPLMN
- struct nas_roamIndList
- struct nas_qaQmi3Gpp2TimeZone
- struct nas_detailSvcInfo
- struct nas_CDMASysInfoExt
- struct nas_callBarStatus
- struct unpack_nas_SLQSGetServingSystem_t
- struct nas_MNCPCSDigitStatus
- struct unpack_nas_SLQSGetServingSystemV2_t
- struct nas_rxSignalStrengthListElement
- struct nas_ecioListElement

- struct nas_errorRateListElement
- struct nas_rsrqInformation
- struct nas_lteSnrinformation
- struct nas_lteRsrpinformation
- struct unpack_nas_SLQSGetSignalStrength_t
- struct nas_SLQSSignalStrengthsIndReq
- struct pack_nas_SLQSSetSignalStrengthsCallback_t
- struct nas_SLQSSignalStrengthsInformation
- struct nas_RejectReasonTlv
- struct nas_SignalStrengthTlv
- struct nas_RFInfoTlv
- struct nas_SLQSSignalStrengthsTlv
- struct unpack_nas_SetEventReportInd_t
- struct unpack_nas_GetCDMANetworkParameters_t
- struct unpack_nas_GetANAAAAuthenticationStatus_t
- struct unpack_nas_GetACCOLC_t
- struct pack_nas_SetACCOLC_t
- struct nas_CDMARSSIThresh
- struct nas_CDMAECIOThresh
- struct nas_HDRRSSIThresh
- struct nas_HDRECIOThresh
- struct nas_HDRSINRThreshold
- struct nas_HDRIOThresh
- struct nas_GSMRSSIThresh
- struct nas_WCDMARSSIThresh
- struct nas_WCDMAECIOThresh
- struct nas_LTERSSIThresh
- struct nas_LTESNRThreshold
- struct nas_LTERSRQThresh
- struct nas_LTERSRPThresh
- struct nas_LTESigRptConfig
- struct nas_TDSCDMARSCPThresh
- struct nas_TDSCDMARSSIThresh
- struct nas_TDSCDMAECIOThresh
- struct nas_TDSCDMASINRThresh
- struct pack_nas_SLQSNasConfigSigInfo2_t
- struct unpack_nas_SetDataCapabilitiesCallback_ind_t
- struct unpack_nas_GetNetworkPreference_t
- struct pack_nas_SetNetworkPreference_t
- struct unpack_nas_SetNetworkPreference_t
- struct unpack_nas_SetRoamingIndicatorCallback_ind_t
- struct NASServingSystemInfo
- struct unpack_nas_SetServingSystemCallback_ind_t
- struct NASPhyCaAggScellIndType
- struct NASPhyCaAggScellDlBw
- struct NASPhyCaAggScellInfo
- struct NASPhyCaAggPcellInfo
- struct NASPhyCaAggScellIndex
- struct NASPhyCaAggScellArray
- struct NasGetLTECphyCaInfo
- struct unpack_nas_SlqsGetLTECphyCAInfo_t
- struct NASEmergencyModeTlv
- struct NASModePreferenceTlv
- struct NASBandPreferenceTlv
- struct NASPRLPreferenceTlv

- struct NASRoamPreferenceTlv
- struct NASLTEBandPreferenceTlv
- struct NASNetSelPreferenceTlv
- struct NASServDomainPrefTlv
- struct NASGWAcqOrderPrefTlv
- struct NASAcqOrderPrefTlv
- struct NASRatDisabledMaskTlv
- struct NASCiotLteOpModePrefTlv
- struct NASLteM1BandPrefTlv
- struct NASLteNB1BandPrefTlv
- struct NASCiotAcqOrderPrefTlv
- struct NASNr5gBandPrefTlv
- struct NASQmiCbkNasSystemSelPrefInd
- struct unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t
- struct NASOTAMessageTlv
- struct NASLteNasReleaseInfoTlv
- struct NASTimeInfoTlv
- struct NASQmiCbkNasSwiOTAMessageInd
- struct unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t
- struct nas_MNRInfo
- struct pack_nas_SLQSInitiateNetworkRegistration_t
- struct pack_nas_SLQSNasSwiIndicationRegister_t
- struct pack_nas_SLQSGetPLMNName_t
- struct unpack_nas_SLQSGetPLMNName_t
- struct nas_nmrCellInfo
- struct nas_GERANInfo
- struct nas_geranInstInfo
- struct nas_UMTSinstInfo
- struct nas_UMTSInfo
- struct nas_CDMAInfo
- struct nas_cellParams
- struct nas_LTEInfoIntrafreq
- struct nas_infoInterFreq
- struct nas_LTEInfoInterfreq
- struct nas_gsmCellInfo
- struct nas_lteGsmCellInfo
- struct nas_LTEInfoNeighboringGSM
- struct nas_wcdmaCellInfo
- struct nas_lteWcdmaCellInfo
- struct nas_LTEInfoNeighboringWCDMA
- struct nas_umtsLTENbrCell
- struct nas_WCDMAInfoLTENeighborCell
- struct unpack_nas_SLQSNasGetCellLocationInfo_t
- struct nas_WCDMACellInfoExt
- struct nas_umtsInstArr
- struct nas_geranInstArr
- struct nas_UMTSExtInfo
- struct nas_LteEarfcnInfo
- struct unpack_nas_SLQSNasGetCellLocationInfoV2_t
- struct nas_timeInfo
- struct unpack_nas_SLQSGetNetworkTime_t
- struct nas_UniversalTime
- struct unpack_nas_SLQSNasNetworkTimeCallBack_ind_t
- struct nas_PhyCaAggScellIndType
- struct nas_PhyCaAggScellDlBw

- struct nas_PhyCaAggScellInfo
- struct nas_PhyCaAggPcellInfo
- struct nas_PhyCaAggScellIndex
- struct nas_PhyCaAggDlBW
- struct nas_NumScellsConfig
- struct unpack_nas_SetNasLTECphyCaIndCallback_ind_t
- struct nas_RxSigInfo
- struct nas_SccRxInfo
- struct unpack_nas_SLQSSwiGetLteSccRxInfo_t
- struct unpack_nas_SLQSNasTimerCallback_ind_t
- struct pack_nas_InitiateDomainAttach_t
- struct pack_nas_SetCDMANetworkParameters_t
- struct unpack_nas_SLQSNasGetHDRColorCode_t
- struct pack_nas_SLQSNasGetTxRxInfo_t
- struct nas_rxInfo
- struct nas_txInfo
- struct unpack_nas_SLQSNasGetTxRxInfo_t
- struct nas_OperatorPLMNData
- struct nas_operatorPLMNList
- struct nas_serviceProviderName
- struct nas_PLMNNetworkNameData
- struct nas_PLMNNetworkName
- struct nas_operatorNameString
- struct unpack_nas_SLQSGetOperatorNameData_t
- struct pack_nas_SLQSNasGet3GPP2Subscription_t
- struct nas_namName
- struct nas_dirNum
- struct nas_sidNid
- struct nas_homeSIDNID
- struct nas_minBasedIMSI
- struct nas_trueIMSI
- struct nas_CDMAChannel
- struct nas_Mdn
- struct unpack_nas_SLQSNasGet3GPP2Subscription_t
- struct nas_protocolSubtypeElement
- struct unpack_nas_SLQSSwiGetHDRPersonality_t
- struct unpack_nas_SLQSSwiGetHDRProtSubtype_t
- struct pack_nas_SLQSSwiPSDetach_t
- struct unpack_nas_SLQSGetErrorRate_t
- struct nas_DRCParams
- struct nas_PilotSetParams
- struct nas_PilotSetData
- struct unpack_nas_SLQSSwiGetHRPDStats_t
- struct nas_ActPilotPNElement
- struct nas_NetworkStat1x
- struct nas_NetworkStatEVDO
- struct nas_DeviceConfigDetail
- struct nas_DataStatusDetail
- struct unpack_nas_SLQSSwiNetworkDebug_t
- struct nas_RSSIThresh
- struct nas_ECIOThresh
- struct nas_HDRSINRThresh
- struct nas_LTESNRThresh
- struct nas_IOThresh
- struct nas_RSRQThresh

- struct nas_RSRPThresh
- struct nas_LTESigRptCfg
- struct nas_TDSCDMASINRCONFThresh
- struct pack_nas_SLQSConfigSigInfo_t
- struct unpack_nas_GetHomeNetwork3GPP2_t
- struct nas_wcdmaUARFCN
- struct nas_lteEARFCN
- struct nas_ltePCI
- struct pack_nas_SLQSNASSwiSetChannelLock_t
- struct unpack_nas_SLQSNASSwiGetChannelLock_t
- struct unpack_nas_SLQSNASGeteDRXParams_t
- struct pack_nas_SLQSNASGeteDRXParamsExt_t
- struct unpack_nas_SLQSNASGeteDRXParamsExt_t
- struct pack_nas_SLQSNASSeteDRXParams_t
- struct nas_HDRPersonality_Ind_Data
- struct unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t
- struct nas_RankIndicatorTlv
- struct unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t
- struct nas_RFBandInfoElements
- struct nas_RfDedicatedBandInfoElements
- struct nas_RfBandInfoExtFormatElements
- struct nas_RfBandwidthInfoElements
- struct nas_RfBandInfoList
- struct nas_RfDedicatedBandInfo
- struct nas_RfBandInfoExtFormat
- struct nas_RfBandwidthInfo
- struct nas_LTEOperationMode
- struct unpack_nas_SLQSNasGetRFInfo_t
- struct nas_EmerModeTlv
- struct nas_ModePrefTlv
- struct nas_BandPrefTlv
- struct nas_PRLPrefTlv
- struct nas_RoamPrefTlv
- struct nas_LTEBandPrefTlv
- struct nas_NetSelPrefTlv
- struct nas_SrvDomainPrefTlv
- struct nas_GWAcqOrderPrefTlv
- struct nas_AcqOrderPrefTlv
- struct nas_RatDisabledMaskTlv
- struct nas_CiotLteOpModePrefTlv
- struct nas_LteM1BandPrefTlv
- struct nas_LteNb1BandPrefTlv
- struct nas_CiotAcqOrderPrefTlv
- struct nas_BandPrefInfoTlv
- struct unpack_nas_SLQSGetSysSelectionPrefExt_t
- struct nas_TDSCDMABandPrefTlv
- struct nas_SrvRegRestricTlv
- struct nas_UsageSettingTlv
- struct nas_VoiceDomainPrefTlv
- struct unpack_nas_SLQSGetSysSelectionPrefExtV2_t
- struct pack_nas_PerformNetworkScanPCI_t
- struct nas_EdrxEnableType
- struct nas_EdrxCycleLength
- struct nas_EdrxPagingTimeWindow
- struct nas_EdrxRatType

- struct nas_EdrxCiotLteMode
- struct unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t
- struct nas_CsgId
- struct nas_PlmnID
- struct nas_LteOpMode
- struct unpack_nas_SLQSNasNetworkRejectCallback_Ind_t
- struct nas_ForbiddenNetworks3GPP
- struct unpack_nas_SLQSNASGetForbiddenNetworks_t
- struct nas_RFBandInfoTlv
- struct nas_RFDedicatedBandInfoTlv
- struct nas_RFBandInfoExtTlv
- struct nas_RFBandwidthInfoTlv
- struct nas_LTEOperationalModeTlv
- struct unpack_nas_SLQSNasRFBandInfoCallback_Ind_t
- struct pack_nas_SLQSSetDDTMPref_t
- struct nas_DDTMSettings
- struct unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t
- struct pack_nas_SLQSSet3GPP2SubscriptionInfo_t
- struct nas_WCDMAARFCNTlv
- struct nas_LTEARFCNTlv
- struct nas_PrefPLMNListTlv
- struct pack_nas_SLQSPerformIncNetworkScanReq_t
- struct nas_NetworkScanStatusTlv
- struct nas_NetworkScan3GPPInfo
- struct nas_NetworkScan3GPPInfoTlv
- struct nas_CSGInfo
- struct nas_CSGInfoTlv
- struct nas_NetworkNameSrcTlv
- struct nas_SignalQualityTlv
- struct nas_SignalStrngthTlv
- struct unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t

Macros

- #define NAS_OTA_MESSAGE_MAX_BUF_SIZE 2048
- #define NAS_MAX_NUM_NETWORKS 30
- #define NAS_MAX_DESCRIPTION_LENGTH 255
- #define NAS_PLMN_LENGTH 3
- #define NAS_MAX_SCC_RX_INFO_INSTANCES 255
- #define NAS_NAM_NAME_LENGTH 12
- #define NAS_IMSI_M_S1_LENGTH 7
- #define NAS_IMSI_M_S2_LENGTH 3
- #define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST 255
- #define NAS_MAX_SLQS_NAS_PCI_INFO_PLMN_LENGTH 255
- #define NAS_MAX_SLQS_NAS_PCI_INFO_LENGTH 255
- #define NAS_MCC_MNC_INST_LENGTH 255
- #define LITE_NW_SCAN_LTE_OP_MODE_MAX_LENGTH 255
- #define NAS_MAX_PHY_CA_AGG_SCELL_ARRRY_SIZE 255
- #define LITE_MAX_NAS_3GPP2_MDN_LEN 15
- #define LITE_MAX_SLQS_NAS_NW_NAME_SRC_LENGTH 255
- #define LITE_MAX_ARFCN_NUM 255
- #define LITE_MAX_PLMNLIST_NUM 255
- #define LITE_MAX_NUM_OF_SET 255

Typedefs

- typedef unpack_result_t unpack_nas_SLQSSetSysSelectionPref_t
- typedef unpack_result_t unpack_nas_SLQSSetSysSelectionPrefExt_t
- typedef unpack_result_t unpack_nas_SLQSSetBandPreference_t
- typedef unpack_result_t unpack_nas_SLQSNasIndicationRegisterExt_t
- typedef unpack_result_t unpack_nas_SLQSNasIndicationRegisterV2_t
- typedef unpack_result_t unpack_nas_SLQSSetSignalStrengthsCallback_t
- typedef unpack_result_t unpack_nas_SetRFInfoCallback_t
- typedef unpack_result_t unpack_nas_SetLURejectCallback_t
- typedef unpack_result_t unpack_nas_SetACCOLC_t
- typedef unpack_result_t unpack_nas_SLQSNasConfigSigInfo2_t
- typedef unpack_result_t unpack_nas_SLQSInitiateNetworkRegistration_t
- typedef unpack_result_t unpack_nas_SLQSNasSwiIndicationRegister_t
- typedef unpack_result_t unpack_nas_InitiateDomainAttach_t
- typedef unpack_result_t unpack_nas_SetCDMANetworkParameters_t
- typedef unpack_result_t unpack_nas_SLQSSwiPSDetach_t
- typedef unpack_result_t unpack_nas_SLQSConfigSigInfo_t
- typedef unpack_result_t unpack_nas_SLQSNASSwiSetChannelLock_t
- typedef unpack_result_t unpack_nas_SLQSNASSeteDRXParams_t
- typedef nas_BandPrefInfoTlv nas_NR5gBandPrefTlv
- typedef nas_BandPrefInfoTlv nas_LTEBandPrefExtTlv
- typedef unpack_result_t unpack_nas_SLQSSetDDTMPref_t
- typedef unpack_result_t unpack_nas_SLQSSet3GPP2SubscriptionInfo_t
- typedef unpack_result_t unpack_nas_SLQSPerformIncNetworkScan_t

Enumerations

- enum LITEQMI_NAS_LTE_CPHY_SCELL_STATE {
  eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED =0x00,
  eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED =0x01,
  eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED =0x02 }
- enum LITEQMI_NAS_LTE_CPHY_CA_BW_NRB {
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6 =0x00,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15 =0x01,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25 =0x02,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50 =0x03,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75 =0x04,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100 =0x05 }
- enum LITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND {

eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_1 = 120,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_2 = 121,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_3 = 122,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_4 = 123,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_5 = 124,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_6 = 125,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_7 = 126,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_8 = 127,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_9 = 128,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_10 = 129,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_11 = 130,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_12 = 131,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_13 = 132,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_14 = 133,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_17 = 134,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_33 = 135,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_34 = 136,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_35 = 137,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_36 = 138,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_37 = 139,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_38 = 140,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_39 = 141,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_40 = 142,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_18 = 143,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_19 = 144,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_20 = 145,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_21 = 146,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_24 = 147,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_25 = 148,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_41 = 149,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_42 = 150,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_43 = 151,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_23 = 152,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_26 = 153,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_32 = 154,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_125 = 155,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_126 = 156,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_127 = 157,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_28 = 158,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_29 = 159,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_30 = 160,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_66 = 161,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_250 = 162,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_46 = 163,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_27 = 164,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_31 = 165,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_71 = 166,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_47 = 167,
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_48 = 168 }

- enum NAS_LTE_CPHY_CA_BW_NRB_LITE {
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_6 =0x00,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_15 =0x01,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_25 =0x02,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_50 =0x03,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_75 =0x04,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_100 =0x05 }

- enum NAS_LTE_CPHY_SCELL_STATE_LITE {

eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE =0x00,
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE =0x01,
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE =0x02 }

- enum LITE_TYPE_OF_SERVICE_DOMAIN {
LITE_SYS_SRV_DOMAIN_NO_SRV = 0x00,
LITE_SYS_SRV_DOMAIN_CS_ONLY = 0x01,
LITE_SYS_SRV_DOMAIN_PS_ONLY = 0x02,
LITE_SYS_SRV_DOMAIN_CS_PS = 0x03,
LITE_SYS_SRV_DOMAIN_CAMPED = 0x04 }

Functions

- int unpack_nas_GetSignalStrengths (uint8_t *pResp, uint16_t respLen, unpack_nas_GetSignalStrengths_t *pOutput)
- int pack_nas_GetSignalStrengths (pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen)
- int pack_nas_SLQSGetSysSelectionPref (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetSysSelectionPref (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysSelectionPref_t *pOutput)
- int pack_nas_SLQSSetSysSelectionPref (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSysSelectionPref_t *pReqParam)
- int unpack_nas_SLQSSetSysSelectionPref (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetSysSelectionPref_t *pOutput)
- int pack_nas_SLQSSetSysSelectionPrefExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSysSelectionPrefExt_t *pReqParam)
- int unpack_nas_SLQSSetSysSelectionPrefExt (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetSysSelectionPrefExt_t *pOutput)
- int pack_nas_SLQSSetBandPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetBandPreference_t *pReqParam)
- int unpack_nas_SLQSSetBandPreference (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetBandPreference_t *pOutput)
- int pack_nas_SLQSNasIndicationRegisterExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasIndicationRegisterExt_t *pReqParam)
- int unpack_nas_SLQSNasIndicationRegisterExt (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasIndicationRegisterExt_t *pOutput)
- int pack_nas_SLQSNasIndicationRegisterV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasIndicationRegisterV2_t *pReqParam)
- int unpack_nas_SLQSNasIndicationRegisterV2 (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasIndicationRegisterV2_t *pOutput)
- int pack_nas_GetRFInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetRFInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_GetRFInfo_t *pOutput)
- int pack_nas_SLQSNasGetSigInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetSigInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetSigInfo_t *pOutput)
- int unpack_valid_nas_SLQSNasGetSigInfo (uint8_t *pResp, uint8_t u8Info)
- int unpack_nas_SLQSNasSigInfoCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSigInfoCallback_ind_t *pOutput)
- int unpack_nas_GetHomeNetwork (uint8_t *pResp, uint16_t respLen, unpack_nas_GetHomeNetwork_t *pOutput)
- int pack_nas_GetHomeNetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetHomeNetwork (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetHomeNetwork_t *pOutput)
- int pack_nas_SLQSGetHomeNetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int pack_nas_SLQSGetSysInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetSysInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysInfo_t *pOutput)

- int unpack_nas_SLQSGetSysInfoV2 (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysInfoV2_t *pOutput)
- int pack_nas_SLQSGetSysInfoV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasSysInfoCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSysInfoCallback_ind_t *pOutput)
- int pack_nas_GetServingNetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetServingNetwork (uint8_t *pResp, uint16_t respLen, unpack_nas_GetServingNetwork_t *pOutput)
- int pack_nas_GetServingNetworkCapabilities (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetServingNetworkCapabilities (uint8_t *pResp, uint16_t respLen, unpack_nas_GetServingNetworkCapabilities_t *pOutput)
- int pack_nas_PerformNetworkScan (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_PerformNetworkScan (uint8_t *pResp, uint16_t respLen, unpack_nas_PerformNetworkScan_t *pOutput)
- int unpack_nas_SLQSPerformNetworkScanV2 (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSPerformNetworkScanV2_t *pOutput)
- int pack_nas_SLQSPerformNetworkScanV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int pack_nas_SLQSSwiGetLteCQI (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetLteCQI (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteCQI_t *pOutput)
- int pack_nas_SLQSNasSwiModemStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasSwiModemStatus (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSwiModemStatus_t *pOutput)
- int pack_nas_SLQSGetServingSystem (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetServingSystem (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetServingSystem_t *pOutput)
- int unpack_nas_SLQSGetServingSystemV2 (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetServingSystemV2_t *pOutput)
- int pack_nas_SLQSGetServingSystemV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_valid_nas_SLQSGetServingSystem (uint8_t *pResp, uint8_t u8Info)
- int pack_nas_SLQSGetSignalStrength (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint16_t reqMask)
- int unpack_nas_SLQSGetSignalStrength (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSignalStrength_t *pOutput)
- int unpack_valid_nas_SLQSGetSignalStrength (uint8_t *pResp, uint8_t u8Info)
- int pack_nas_SLQSSetSignalStrengthsCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSignalStrengthsCallback_t *pReqParam)
- int unpack_nas_SLQSSetSignalStrengthsCallback (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetSignalStrengthsCallback_t *pOutput)
- int pack_nas_SetRFInfoCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)
- int unpack_nas_SetRFInfoCallback (uint8_t *pResp, uint16_t respLen, unpack_nas_SetRFInfoCallback_t *pOutput)
- int pack_nas_SetLURejectCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)
- int unpack_nas_SetLURejectCallback (uint8_t *pResp, uint16_t respLen, unpack_nas_SetLURejectCallback_t *pOutput)
- int unpack_nas_SetEventReportInd (uint8_t *pResp, uint16_t respLen, unpack_nas_SetEventReportInd_t *pOutput)
- int pack_nas_GetCDMANetworkParameters (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetCDMANetworkParameters (uint8_t *pResp, uint16_t respLen, unpack_nas_GetCDMANetworkParameters_t *pOutput)
- int unpack_valid_nas_GetCDMANetworkParameters (uint8_t *pResp, uint8_t u8Info)
- int pack_nas_GetANAAAAuthenticationStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetANAAAAuthenticationStatus (uint8_t *pResp, uint16_t respLen, unpack_nas_GetANAAAAuthenticationStatus_t *pOutput)
- int pack_nas_GetACCOLC (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)

- int unpack_nas_GetACCOLC (uint8_t *pResp, uint16_t respLen, unpack_nas_GetACCOLC_t *pOutput)
- int pack_nas_SetACCOLC (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetACCOLC_t reqParam)
- int unpack_nas_SetACCOLC (uint8_t *pResp, uint16_t respLen, unpack_nas_SetACCOLC_t *pOutput)
- int pack_nas_SLQSNasConfigSigInfo2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasConfigSigInfo2_t *pReqParam)
- int unpack_nas_SLQSNasConfigSigInfo2 (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasConfigSigInfo2_t *pOutput)
- int unpack_nas_SetDataCapabilitiesCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetDataCapabilitiesCallback_ind_t *pOutput)
- int pack_nas_GetNetworkPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetNetworkPreference (uint8_t *pResp, uint16_t respLen, unpack_nas_GetNetworkPreference_t *pOutput)
- int pack_nas_SetNetworkPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetNetworkPreference_t *reqArg)
- int unpack_nas_SetNetworkPreference (uint8_t *pResp, uint16_t respLen, unpack_nas_SetNetworkPreference_t *pOutput)
- int unpack_nas_SetRoamingIndicatorCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetRoamingIndicatorCallback_ind_t *pOutput)
- int unpack_nas_SetServingSystemCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetServingSystemCallback_ind_t *pOutput)
- int pack_nas_SlqsGetLTECphyCAInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SlqsGetLTECphyCAInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SlqsGetLTECphyCAInfo_t *pOutput)
- int unpack_nas_SLQSSetSysSelectionPrefCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t *pOutput)
- int unpack_nas_SLQSNasSwiOTAMessageCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t *pOutput)
- int pack_nas_SLQSInitiateNetworkRegistration (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSInitiateNetworkRegistration_t *pReqParam)
- int unpack_nas_SLQSInitiateNetworkRegistration (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSInitiateNetworkRegistration_t *pOutput)
- int pack_nas_SLQSNasSwiIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasSwiIndicationRegister_t *pReqParam)
- int unpack_nas_SLQSNasSwiIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSwiIndicationRegister_t *pOutput)
- int pack_nas_SLQSGetPLMNName (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSGetPLMNName_t *reqArg)
- int unpack_nas_SLQSGetPLMNName (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetPLMNName_t *pOutput)
- int pack_nas_SLQSNasGetCellLocationInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetCellLocationInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetCellLocationInfo_t *pOutput)
- int unpack_nas_SLQSNasGetCellLocationInfoV2 (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetCellLocationInfoV2_t *pOutput)
- int pack_nas_SLQSNasGetCellLocationInfoV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int pack_nas_SLQSGetNetworkTime (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetNetworkTime (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetNetworkTime_t *pOutput)
- int unpack_nas_SLQSNasNetworkTimeCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasNetworkTimeCallBack_ind_t *pOutput)
- int unpack_nas_SetNasLTECphyCaIndCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetNasLTECphyCaIndCallback_ind_t *pOutput)
- int pack_nas_SLQSSwiGetLteSccRxInfo (pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetLteSccRxInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteSccRxInfo_t *pOutput)

- int unpack_nas_SLQSNasTimerCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasTimerCallback_ind_t *pOutput)
- int pack_nas_InitiateDomainAttach (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_InitiateDomainAttach_t *pReqParam)
- int unpack_nas_InitiateDomainAttach (uint8_t *pResp, uint16_t respLen, unpack_nas_InitiateDomainAttach_t *pOutput)
- int pack_nas_SetCDMANetworkParameters (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetCDMANetworkParameters_t *pReqParam)
- int unpack_nas_SetCDMANetworkParameters (uint8_t *pResp, uint16_t respLen, unpack_nas_SetCDMANetworkParameters_t *pOutput)
- int pack_nas_SLQSNasGetHDRColorCode (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetHDRColorCode (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetHDRColorCode_t *pOutput)
- int pack_nas_SLQSNasGetTxRxInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasGetTxRxInfo_t *reqArg)
- int unpack_nas_SLQSNasGetTxRxInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetTxRxInfo_t *pOutput)
- int pack_nas_SLQSGetOperatorNameData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetOperatorNameData (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetOperatorNameData_t *pOutput)
- int pack_nas_SLQSNasGet3GPP2Subscription (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasGet3GPP2Subscription_t *reqArg)
- int unpack_nas_SLQSNasGet3GPP2Subscription (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGet3GPP2Subscription_t *pOutput)
- int pack_nas_SLQSSwiGetHDRPersonality (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetHDRPersonality (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetHDRPersonality_t *pOutput)
- int pack_nas_SLQSSwiGetHDRProtSubtype (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetHDRProtSubtype (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetHDRProtSubtype_t *pOutput)
- int pack_nas_SLQSSwiPSDetach (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSwiPSDetach_t *pReqParam)
- int unpack_nas_SLQSSwiPSDetach (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiPSDetach_t *pOutput)
- int pack_nas_SLQSGetErrorRate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetErrorRate (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetErrorRate_t *pOutput)
- int pack_nas_SLQSSwiGetHRPDStats (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetHRPDStats (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetHRPDStats_t *pOutput)
- int pack_nas_SLQSSwiNetworkDebug (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiNetworkDebug (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiNetworkDebug_t *pOutput)
- int pack_nas_SLQSConfigSigInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSConfigSigInfo_t *pReqParam)
- int unpack_nas_SLQSConfigSigInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSConfigSigInfo_t *pOutput)
- int pack_nas_GetHomeNetwork3GPP2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetHomeNetwork3GPP2 (uint8_t *pResp, uint16_t respLen, unpack_nas_GetHomeNetwork3GPP2_t *pOutput)
- int pack_nas_SLQSNASSwiSetChannelLock (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNASSwiSetChannelLock_t *pReqParam)
- int unpack_nas_SLQSNASSwiSetChannelLock (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNASSwiSetChannelLock_t *pOutput)
- int pack_nas_SLQSNASSwiGetChannelLock (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)

- int unpack_nas_SLQSNASSwiGetChannelLock (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNASSwiGetChannelLock_t *pOutput)
- int pack_nas_SLQSNASGeteDRXParams (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNASGeteDRXParams (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNASGeteDRXParams_t *pOutput)
- int pack_nas_SLQSNASGeteDRXParamsExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNASGeteDRXParamsExt_t *pReqParam)
- int unpack_nas_SLQSNASGeteDRXParamsExt (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNASGeteDRXParamsExt_t *pOutput)
- int pack_nas_SLQSNASSeteDRXParams (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNASSeteDRXParams_t *pReqParam)
- int unpack_nas_SLQSNASSeteDRXParams (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNASSeteDRXParams_t *pOutput)
- int unpack_nas_SLQSSwiHDRPersonalityCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t *pOutput)
- int unpack_nas_SLQSSwiRandIndicatorCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t *pOutput)
- int pack_nas_SLQSNasGetRFInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetRFInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetRFInfo_t *pOutput)
- int pack_nas_SLQSGetSysSelectionPrefExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetSysSelectionPrefExt (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysSelectionPrefExt_t *pOutput)
- int unpack_nas_SLQSGetSysSelectionPrefExtV2 (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysSelectionPrefExtV2_t *pOutput)
- int pack_nas_SLQSGetSysSelectionPrefExtV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int pack_nas_PerformNetworkScanPCI (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_PerformNetworkScanPCI_t *pReqParam)
- int unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t *pOutput)
- int unpack_nas_SLQSNasNetworkRejectCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasNetworkRejectCallback_Ind_t *pOutput)
- int pack_nas_SLQSNASGetForbiddenNetworks (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNASGetForbiddenNetworks (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNASGetForbiddenNetworks_t *pOutput)
- int unpack_nas_SLQSNasRFBandInfoCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasRFBandInfoCallback_Ind_t *pOutput)
- int pack_nas_SLQSSetDDTMPref (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetDDTMPref_t *pReqParam)
- int unpack_nas_SLQSSetDDTMPref (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetDDTMPref_t *pOutput)
- int unpack_nas_SLQSDDTMPrefIndicationCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t *pOutput)
- int pack_nas_SLQSSet3GPP2SubscriptionInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSet3GPP2SubscriptionInfo_t *pReqParam)
- int unpack_nas_SLQSSet3GPP2SubscriptionInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSet3GPP2SubscriptionInfo_t *pOutput)
- int pack_nas_SLQSPerformIncNetworkScanReq (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSPerformIncNetworkScanReq_t *pReqParam)
- int unpack_nas_SLQSPerformIncNetworkScan (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSPerformIncNetworkScan_t *pOutput)
- int unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t *pOutput)

9.22.1 Macro Definition Documentation

9.22.1.1 #define LITE_MAX_ARFCN_NUM 255

9.22.1.2 #define LITE_MAX_NAS_3GPP2_MDN_LEN 15

9.22.1.3 #define LITE_MAX_NUM_OF_SET 255

9.22.1.4 #define LITE_MAX_PLMNLIST_NUM 255

9.22.1.5 #define LITE_MAX_SLQS_NAS_NW_NAME_SRC_LENGTH 255

9.22.1.6 #define LITE_NW_SCAN_LTE_OP_MODE_MAX_LENGTH 255

9.22.1.7 #define NAS_IMSI_M_S1_LENGTH 7

9.22.1.8 #define NAS_IMSI_M_S2_LENGTH 3

9.22.1.9 #define NAS_MAX_DESCRIPTION_LENGTH 255

9.22.1.10 #define NAS_MAX_NUM_NETWORKS 30

9.22.1.11 #define NAS_MAX_PHY_CA_AGG_SCELL_ARRRY_SIZE 255

9.22.1.12 #define NAS_MAX_SCC_RX_INFO_INSTANCES 255

9.22.1.13 #define NAS_MAX_SLQS_NAS_PCI_INFO_LENGTH 255

9.22.1.14 #define NAS_MAX_SLQS_NAS_PCI_INFO_PLMN_LENGTH 255

9.22.1.15 #define NAS_MCC_MNC_INST_LENGTH 255

9.22.1.16 #define NAS_NAM_NAME_LENGTH 12

9.22.1.17 #define NAS_OTA_MESSAGE_MAX_BUF_SIZE 2048

9.22.1.18 #define NAS_PLMN_LENGTH 3

9.22.1.19 #define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST 255

9.22.2 Typedef Documentation

9.22.2.1 typedef nas_BandPrefInfoTlv nas_LTEBandPrefExtTlv

Contain the LTE Band Preference Extended.

- See nas_BandPrefInfoTlv for more information.

9.22.2.2 typedef nas_BandPrefInfoTlv nas_NR5gBandPrefTlv

Contain the NR5G Band Preference.

- See nas_BandPrefInfoTlv for more information.

9.22.2.3 typedef unpack_result_t unpack_nas_InitiateDomainAttach_t

9.22.2.4 typedef unpack_result_t unpack_nas_SetACCOLC_t

9.22.2.5 typedef unpack_result_t unpack_nas_SetCDMANetworkParameters_t

9.22.2.6 typedef unpack_result_t unpack_nas_SetLURejectCallback_t

9.22.2.7 typedef unpack_result_t unpack_nas_SetRFInfoCallback_t

9.22.2.8 typedef unpack_result_t unpack_nas_SLQSConfigSigInfo_t

9.22.2.9 typedef unpack_result_t unpack_nas_SLQSInitiateNetworkRegistration_t

9.22.2.10 typedef unpack_result_t unpack_nas_SLQSNasConfigSigInfo2_t

9.22.2.11 typedef unpack_result_t unpack_nas_SLQSNasIndicationRegisterExt_t

9.22.2.12 typedef unpack_result_t unpack_nas_SLQSNasIndicationRegisterV2_t

9.22.2.13 typedef unpack_result_t unpack_nas_SLQSNASSeteDRXParams_t

9.22.2.14 typedef unpack_result_t unpack_nas_SLQSNasSwiIndicationRegister_t

9.22.2.15 typedef unpack_result_t unpack_nas_SLQSNASSwiSetChannelLock_t

9.22.2.16 typedef unpack_result_t unpack_nas_SLQSPerformIncNetworkScan_t

9.22.2.17 typedef unpack_result_t unpack_nas_SLQSSet3GPP2SubscriptionInfo_t

9.22.2.18 typedef unpack_result_t unpack_nas_SLQSSetBandPreference_t

9.22.2.19 typedef unpack_result_t unpack_nas_SLQSSetDDTMPref_t

9.22.2.20 typedef unpack_result_t unpack_nas_SLQSSetSignalStrengthsCallback_t

9.22.2.21 typedef unpack_result_t unpack_nas_SLQSSetSysSelectionPref_t

9.22.2.22 typedef unpack_result_t unpack_nas_SLQSSetSysSelectionPrefExt_t

9.22.2.23 typedef unpack_result_t unpack_nas_SLQSSwiPSDetach_t

9.22.3 Enumeration Type Documentation

9.22.3.1 enum LITE_TYPE_OF_SERVICE_DOMAIN

Enumerator

LITE_SYS_SRV_DOMAIN_NO_SRV

LITE_SYS_SRV_DOMAIN_CS_ONLY

LITE_SYS_SRV_DOMAIN_PS_ONLY

LITE_SYS_SRV_DOMAIN_CS_PS

LITE_SYS_SRV_DOMAIN_CAMPED

9.22.3.2 enum LITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND

NAS LTE BAND CLASS E-UTRA Operating BANDs

Enumerator

eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_1
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_2
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_3
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_4
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_5
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_6
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_7
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_8
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_9
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_10
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_11
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_12
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_13
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_14
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_17
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_33
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_34
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_35
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_36
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_37
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_38
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_39
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_40
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_18
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_19
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_20
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_21
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_24
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_25
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_41
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_42
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_43
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_23
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_26
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_32
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_125
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_126
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_127
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_28
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_29
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_30

eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_66
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_250
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_46
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_27
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_31
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_71
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_47
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_48

9.22.3.3 enum LITEQMI_NAS_LTE_CPHY_CA_BW_NRB

NAS LTE CPHY Downlink Values.

Enumerator

eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100

9.22.3.4 enum LITEQMI_NAS_LTE_CPHY_SCELL_STATE

NAS LTE CPHY Scell States.

Enumerator

eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED

9.22.3.5 enum NAS_LTE_CPHY_CA_BW_NRB_LITE

NAS LTE CPHY Downlink Values.

Enumerator

eNAS_LTE_CPHY_CA_BW_NRB_LITE_6
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100

9.22.3.6 enum NAS_LTE_CPHY_SCELL_STATE_LITE

NAS LTE CPHY Scell States.

Enumerator

eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE

9.22.4 Function Documentation

9.22.4.1 int pack_nas_GetACCOLC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Retrieves information about the access overload class (ACCOLC) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.2 int pack_nas_GetANAAAAuthenticationStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

AN-AAA authentication status of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.3 int pack_nas_GetCDMANetworkParameters (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get CDMA Network Parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.4 int pack_nas_GetHomeNetwork (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

get home network pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.5 int pack_nas_GetHomeNetwork3GPP2 (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Get Home Network 3GPP2 Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.6 int pack_nas_GetNetworkPreference (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Gets the network registration preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

9.22.4.7 int pack_nas_GetRFInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

get rf info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.8 int pack_nas_GetServingNetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack provides information about the system that provides service to the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.9 int pack_nas_GetServingNetworkCapabilities (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack returns information regarding the data capabilities of the system that currently provides service to the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.10 int pack_nas_GetSignalStrengths (pack_qmi_t * *pCtx*, uint8_t * *pReq*, uint16_t * *pLen*)

get signal strengths pack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15-C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use pack_nas_SLQSNasGetSigInfo() for new firmware versions and new modules

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.11 int pack_nas_InitiateDomainAttach (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_InitiateDomainAttach_t * *pReqParam*)

Initiates a domain attach/detach of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.12 int pack_nas_PerformNetworkScan (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack performs scan for available networks and scans for RAT info as well.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.13 int pack_nas_PerformNetworkScanPCI (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_PerformNetworkScanPCI_t * *pReqParam*)

Pack performs scan for available networks and scans for LTE RAT info (specific Network or Scan Type).

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Device Supported: MC75xx, EM75xx, EM74x1 and WP76xx.

9.22.4.14 int pack_nas_SetACCOLC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SetACCOLC_t *reqParam*)

Pack Sets the access overload class (ACCOLC)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.15 int pack_nas_SetCDMANetworkParameters (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SetCDMANetworkParameters_t * *pReqParam*)

CDMA Network Parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.16 int pack_nas_SetLURejectCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint8_t * *pBenable*)

Pack set registration reject Callback

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pBenable*	0/1 value to disable/enable indication respectively

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.17 int pack_nas_SetNetworkPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SetNetworkPreference_t * *reqArg*)

Sets the network registration preference. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use pack_nas_SLQSSetSys-SelectionPref() for new firmware versions and new modules

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.22.4.18 int pack_nas_SetRFInfoCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint8_t * *pBenable*)

Pack Set RF Band Information callback.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pBenable*	0/1 value to disable/enable indication respectively

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.19 int pack_nas_SLQSConfigSigInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSConfigSigInfo_t * *pReqParam*)

Config Sig Info Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.20 int pack_nas_SLQSGetErrorRate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Error Rate Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.21 int pack_nas_SLQSGetHomeNetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get home network pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.22 int pack_nas_SlqsGetLTECphyCAInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Pack get carrier aggregation event information.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

9.22.4.23 int pack_nas_SLQSGetNetworkTime (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Pack Get Network Time.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.24 int pack_nas_SLQSGetOperatorNameData (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Operator Name Data Parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.25 int pack_nas_SLQSGetPLMNName (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSGetPLMNName_t * reqArg)

Pack get operator name for specificed network.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.26 int pack_nas_SLQSGetServingSystem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack queries information regarding the system that currently provides service.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.27 int pack_nas_SLQSGetServingSystemV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack queries information regarding the system that currently provides service.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.28 int pack_nas_SLQSGetSignalStrength (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, uint16_t reqMask)

Queries the current signal strength as measured by the device. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use pack_nas_SLQSNasGetSigInfo() for new firmware versions and new modules

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqMask*	request mask for fetching extra signal info

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.29 int pack_nas_SLQSGetSysInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Pack get system information.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.30 int pack_nas_SLQSGetSysInfoV2 (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Pack get system information.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.31 int pack_nas_SLQSGetSysSelectionPref (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Queries the different system selection preferences of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.32 int pack_nas_SLQSGetSysSelectionPrefExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Queries the different system selection preferences of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.33 int pack_nas_SLQSGetSysSelectionPrefExtV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Queries the different system selection preferences of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.34 int pack_nas_SLQSInitiateNetworkRegistration (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSInitiateNetworkRegistration_t * *pReqParam*)

Pack initiates a network registration.This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use pack_nas_SLQSSetSysSelectionPrefExt() instead for new firmware versions and new modules.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.35 int pack_nas_SLQSNasConfigSigInfo2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasConfigSigInfo2_t * *pReqParam*)

Pack sets the signal strength reporting thresholds

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.36 int pack_nas_SLQSNasGet3GPP2Subscription (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasGet3GPP2Subscription_t * *reqArg*)

3GPP2 Subscription Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.37 int pack_nas_SLQSNasGetCellLocationInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack retrieves cell location-related information.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.38 int pack_nas_SLQSNasGetCellLocationInfoV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack retrieves cell location-related information.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.39 int pack_nas_SLQSNASGeteDRXParams (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get eDRX Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.40 int pack_nas_SLQSNASGeteDRXParamsExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNASGeteDRXParamsExt_t * *pReqParam*)

Get eDRX Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.41 int pack_nas_SLQSNASGetForbiddenNetworks (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get forbidden networks pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.42 int pack_nas_SLQSNasGetHDRColorCode (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HDR Color Code Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.43 int pack_nas_SLQSNasGetRFInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets radio band/channel information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.44 int pack_nas_SLQSNasGetSigInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get sig info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.45 int pack_nas_SLQSNasGetTxRxInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasGetTxRxInfo_t * *reqArg*)

Get Tx Rx Info Parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.46 int pack_nas_SLQSNasIndicationRegisterExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasIndicationRegisterExt_t * *pReqParam*)

Pack used to Registers/De-registers for different NAS (Network access service) indications.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This function is used by a control point to register/deregister for different QMI_NAS indications. The control point's registration state variables, controlling registration for indications, are modified to reflect the settings indicated in the parameters that are present in the request message. At least one optional parameter must be present in the request.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.47 int pack_nas_SLQSNasIndicationRegisterV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasIndicationRegisterV2_t * *pReqParam*)

Pack used to Registers/De-registers for different NAS (Network access service) indications.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This function is used by a control point to register/deregister for different QMI_NAS indications. The control point's registration state variables, controlling registration for indications, are modified to reflect the settings indicated in the parameters that are present in the request message. At least one optional parameter must be present in the request.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.48 int pack_nas_SLQSNASSeteDRXParams (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNASSeteDRXParams_t * *pReqParam*)

Set eDRX Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.49 int pack_nas_SLQSNASSwiGetChannelLock (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Channel Lock Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.50 int pack_nas_SLQSNasSwiIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasSwiIndicationRegister_t * *pReqParam*)

Pack sets the registration state for different QMI_NAS SWI indications.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.51 int pack_nas_SLQSNasSwiModemStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

This function requests the device to return the current status of modem.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.52 int pack_nas_SLQSNASSwiSetChannelLock (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNASSwiSetChannelLock_t * *pReqParam*)

Set Channel Lock Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.53 int pack_nas_SLQSPerformIncNetworkScanReq (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSPerformIncNetworkScanReq_t * pReqParam)

Pack sets the SLQSPerformIncNetworkScanReq

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.54 int pack_nas_SLQSPerformNetworkScanV2 (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Pack performs scan for available networks and scans for RAT info as well.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.55 int pack_nas_SLQSSet3GPP2SubscriptionInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSSet3GPP2SubscriptionInfo_t * pReqParam)

Pack sets the 3GPP2 subscription info

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.56 int pack_nas_SLQSSetBandPreference (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSSetBandPreference_t * pReqParam)

Pack sets the different system selection preferences of the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.57 int pack_nas_SLQSSetDDTMPref (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSSetDDTMPref_t * pReqParam)

Pack Sets the DDTM preference

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.58 int pack_nas_SLQSSetSignalStrengthsCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSSetSignalStrengthsCallback_t * pReqParam)

Pack set signal strength thresholds callback.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.59 int pack_nas_SLQSSetSysSelectionPref (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSetSysSelectionPref_t * *pReqParam*)

Sets the different system selection preferences of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.60 int pack_nas_SLQSSetSysSelectionPrefExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSetSysSelectionPrefExt_t * *pReqParam*)

Sets the different system selection preferences of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.61 int pack_nas_SLQSSwiGetHDRPersonality (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HDR Personality Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.62 int pack_nas_SLQSSwiGetHDRProtSubtype (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HDR Protocol Sub type Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.63 int pack_nas_SLQSSwiGetHRPDStats (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HRPD Stats Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.64 int pack_nas_SLQSSwiGetLteCQI (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack fetch CQI parameters for LTE data session.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.65 int pack_nas_SLQSSwiGetLteSccRxInfo (pack_qmi_t * *pCtx*, uint8_t * *pReq*, uint16_t * *pLen*)

get LTE Secondary carrier Rx signal level information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.66 int pack_nas_SLQSSwiNetworkDebug (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Network Debug Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.67 int pack_nas_SLQSSwiPSDetach (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSwiPSDetach_t * *pReqParam*)

PS Connection Detach Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.68 int unpack_nas_GetACCOLC (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetACCOLC_t * *pOutput*)

Retrieves information about the access overload class (ACCOLC) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.69 int unpack_nas_GetANAAAAuthenticationStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetANAAAAuthenticationStatus_t * *pOutput*)

AN-AAA authentication status of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.70 int unpack_nas_GetCDMANetworkParameters (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetCDMANetworkParameters_t * *pOutput*)

Get CDMA Network Parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	qmi output parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.71 int unpack_nas_GetHomeNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetHomeNetwork_t * *pOutput*)

Retrieves information about the home network of the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.72 int unpack_nas_GetHomeNetwork3GPP2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetHomeNetwork3GPP2_t * *pOutput*)

Get Home Network 3GPP2 Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.73 int unpack_nas_GetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetNetworkPreference_t * *pOutput*)

Gets the network registration preference unpack

Parameters

in	*pResp*	qmi resp from modem
in	*respLen*	qmi resp length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.74 int unpack_nas_GetRFInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetRFInfo_t * *pOutput*)

get rf info unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.75 int unpack_nas_GetServingNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetwork_t * *pOutput*)

Unpack provides information about the system that provides service to the device. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use API unpack_nas_SLQSGetSysInfo() for new firmware versions and new modules

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.76 int unpack_nas_GetServingNetworkCapabilities (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetworkCapabilities_t * *pOutput*)

Unpack returns information regarding the data capabilities of the system that currently provides service to the device.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.77 int unpack_nas_GetSignalStrengths (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetSignalStrengths_t * *pOutput*)

Returns the available signal strengths (in dBm) as measured by the device in an array. This will also provides the corresponding radio interfaces in an array. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use unpack_nas_SLQSNasGetSigInfo() for new firmware versions and new modules

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.78 int unpack_nas_InitiateDomainAttach (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_InitiateDomainAttach_t * *pOutput*)

Initiates a domain attach/detach of the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.79 int unpack_nas_PerformNetworkScan (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_PerformNetworkScan_t * *pOutput*)

Unpack performs scan for available networks and scans for RAT info as well.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.80 int unpack_nas_SetACCOLC (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetACCOLC_t * *pOutput*)

Unpack Sets the access overload class (ACCOLC)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.81 int unpack_nas_SetCDMANetworkParameters (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetCDMANetworkParameters_t * *pOutput*)

CDMA Network Parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.82 int unpack_nas_SetDataCapabilitiesCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetDataCapabilitiesCallback_ind_t * *pOutput*)

Data Capabilities indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.83 int unpack_nas_SetEventReportInd (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetEventReportInd_t * *pOutput*)

Unpack set event report indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.84 int unpack_nas_SetLURejectCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetLURejectCallback_t * *pOutput*)

Unpack set registration reject Callback

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.85 int unpack_nas_SetNasLTECphyCaIndCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetNasLTECphyCaIndCallback_ind_t * *pOutput*)

Unpack set LTE PHY CA indication callback.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.86 int unpack_nas_SetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetNetworkPreference_t * *pOutput*)

Sets the network registration preference unpack

Parameters

in	*pResp*	qmi resp from modem
in	*respLen*	qmi resp length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.87 int unpack_nas_SetRFInfoCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetRFInfoCallback_t * *pOutput*)

Unack Set RF Band Information callback.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.88 int unpack_nas_SetRoamingIndicatorCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetRoamingIndicatorCallback_ind_t * *pOutput*)

Roaming indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.89 int unpack_nas_SetServingSystemCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetServingSystemCallback_ind_t * *pOutput*)

Set Serving system indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.90 int unpack_nas_SLQSConfigSigInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSConfigSigInfo_t * *pOutput*)

Config Sig Info Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.91 int unpack_nas_SLQSDDTMPrefIndicationCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t * *pOutput*)

unpack nas RF Band Info Callback indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.92 int unpack_nas_SLQSGetErrorRate (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetErrorRate_t * *pOutput*)

Get Error Rate Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.93 int unpack_nas_SLQSGetHomeNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetHomeNetwork_t * *pOutput*)

Retrieves information about the home network of the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.94 int unpack_nas_SlqsGetLTECphyCAInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SlqsGetLTECphyCAInfo_t * *pOutput*)

Unack get carrier aggregation event information.

Parameters

in	*pResp*	qmi resp from modem
in	*respLen*	qmi resp length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.95 int unpack_nas_SLQSGetNetworkTime (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetNetworkTime_t * *pOutput*)

Unpack get network time.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.96 int unpack_nas_SLQSGetOperatorNameData (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetOperatorNameData_t * *pOutput*)

Operator Name Data Parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.97 int unpack_nas_SLQSGetPLMNName (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetPLMNName_t * *pOutput*)

unpack get operator name for specificed network.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.98 int unpack_nas_SLQSGetServingSystem (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetServingSystem_t * *pOutput*)

Unack queries information regarding the system that currently provides service.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.99 int unpack_nas_SLQSGetServingSystemV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetServingSystemV2_t * *pOutput*)

Unack queries information regarding the system that currently provides service.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.100 int unpack_nas_SLQSGetSignalStrength (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSignalStrength_t * *pOutput*)

Unpack get signal strength.This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15-C_05_xx_xx_xx and EM7430/EM7455 firmware versions. Please use unpack_nas_SLQSNasGetSigInfo() for new firmware versions and new modules

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.101 int unpack_nas_SLQSGetSysInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysInfo_t * *pOutput*)

Provides the system information. This API is preferred when trying to get the service status info and serving system info. The function unpack_nas_SLQSGetServingSystem_t() reports similar NAS information, but it is deprecated. Please refer to the header description of unpack_nas_SLQSGetServingSystem_t() for more information.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This API queries current serving system information, including registration information and system property. The registration information for all RATs specified in the mode capability setting are included regardless of registration status. The RAT-specific system property are included only for RATs that are specified in the mode capability setting and which are not in either No Service or Power Save modes.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.102 int unpack_nas_SLQSGetSysInfoV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysInfoV2_t * *pOutput*)

Provides the system information. This API is preferred when trying to get the service status info and serving system info. The function unpack_nas_SLQSGetServingSystem_t() reports similar NAS information, but it is deprecated. Please refer to the header description of unpack_nas_SLQSGetServingSystem_t() for more information.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This API queries current serving system information, including registration information and system property. The registration information for all RATs specified in the mode capability setting are included regardless of registration status. The RAT-specific system property are included only for RATs that are specified in the mode capability setting and which are not in either No Service or Power Save modes.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.103 int unpack_nas_SLQSGetSysSelectionPref (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysSelectionPref_t * *pOutput*)

Queries the different system selection preferences of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.104 int unpack_nas_SLQSGetSysSelectionPrefExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysSelectionPrefExt_t * *pOutput*)

Queries the different system selection preferences of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.105 int unpack_nas_SLQSGetSysSelectionPrefExtV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysSelectionPrefExtV2_t * *pOutput*)

Queries the different system selection preferences of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.106 int unpack_nas_SLQSInitiateNetworkRegistration (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSInitiateNetworkRegistration_t * *pOutput*)

Unpack initiates a network registration.This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM7430/EM7455 firmware versions. Please use unpack_nas_SLQSSetSysSelectionPrefExt() instead for new firmware versions and new modules.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.107 int unpack_nas_SLQSNasConfigSigInfo2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasConfigSigInfo2_t * *pOutput*)

Unpack sets the signal strength reporting thresholds

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.108 int unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t * *pOutput*)

unpack nas eDRX change Info indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.109 int unpack_nas_SLQSNasGet3GPP2Subscription (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGet3GPP2Subscription_t * *pOutput*)

3GPP2 Subscription Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.110 int unpack_nas_SLQSNasGetCellLocationInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetCellLocationInfo_t * *pOutput*)

Unpack retrieves cell location-related information.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.111 int unpack_nas_SLQSNasGetCellLocationInfoV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetCellLocationInfoV2_t * *pOutput*)

Unpack retrieves cell location-related information.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.112 int unpack_nas_SLQSNASGeteDRXParams (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNASGeteDRXParams_t * *pOutput*)

Get eDRX Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.113 int unpack_nas_SLQSNASGeteDRXParamsExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNASGeteDRXParamsExt_t * *pOutput*)

Get eDRX Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.114 int unpack_nas_SLQSNASGetForbiddenNetworks (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNASGetForbiddenNetworks_t * *pOutput*)

Get forbidden networks unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.115 int unpack_nas_SLQSNasGetHDRColorCode (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetHDRColorCode_t * *pOutput*)

HDR Color Code Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.116 int unpack_nas_SLQSNasGetRFInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetRFInfo_t * *pOutput*)

Gets radio band/channel information unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.117 int unpack_nas_SLQSNasGetSigInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetSigInfo_t * *pOutput*)

get sig info unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This command queries the signal strength information for currently active RATs. Information is reported only if the corresponding RATs have signal strength values to be reported. If no signal strength information is available for any RAT, the response message contains only the mandatory response message

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.118 int unpack_nas_SLQSNasGetTxRxInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetTxRxInfo_t * *pOutput*)

Get Tx Rx Info Parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.119 int unpack_nas_SLQSNasIndicationRegisterExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasIndicationRegisterExt_t * *pOutput*)

Unpack used to Registers/De-registers for different NAS (Network access service) indications.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.120 int unpack_nas_SLQSNasIndicationRegisterV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasIndicationRegisterV2_t * *pOutput*)

Unpack used to Registers/De-registers for different NAS (Network access service) indications.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.121 int unpack_nas_SLQSNasNetworkRejectCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasNetworkRejectCallback_Ind_t * *pOutput*)

unpack nas Network Reject Callback indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.122 int unpack_nas_SLQSNasNetworkTimeCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasNetworkTimeCallBack_ind_t * *pOutput*)

Unpack Network time callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.123 int unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t * *pOutput*)

unpack nas Incremental network scan Callback indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.124 int unpack_nas_SLQSNasRFBandInfoCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasRFBandInfoCallback_Ind_t * *pOutput*)

unpack nas RF Band Info Callback indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.125 int unpack_nas_SLQSNASSeteDRXParams (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNASSeteDRXParams_t * *pOutput*)

Set eDRX Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.126 int unpack_nas_SLQSNasSigInfoCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSigInfoCallback_ind_t * *pOutput*)

Unpack Signal Information callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.127 int unpack_nas_SLQSNASSwiGetChannelLock (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNASSwiGetChannelLock_t * *pOutput*)

Get Channel Lock Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.128 int unpack_nas_SLQSNasSwiIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSwiIndicationRegister_t * *pOutput*)

unpack sets the registration state for different QMI_NAS SWI indications.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.129 int unpack_nas_SLQSNasSwiModemStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSwiModemStatus_t * *pOutput*)

This function requests the device to return the current status of modem.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.130 int unpack_nas_SLQSNasSwiOTAMessageCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t * *pOutput*)

OTA message indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.131 int unpack_nas_SLQSNASSwiSetChannelLock (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNASSwiSetChannelLock_t * *pOutput*)

Set Channel Lock Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.132 int unpack_nas_SLQSNasSysInfoCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSysInfoCallback_ind_t * *pOutput*)

Unpack system information callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.133 int unpack_nas_SLQSNasTimerCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasTimerCallback_ind_t * *pOutput*)

Unpack timer callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	network timer indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.134 int unpack_nas_SLQSPerformIncNetworkScan (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSPerformIncNetworkScan_t * *pOutput*)

Unpack of SLQSPerformIncNetworkScan

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.135 int unpack_nas_SLQSPerformNetworkScanV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSPerformNetworkScanV2_t * *pOutput*)

Unpack performs scan for available networks and scans for RAT info as well.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.136 int unpack_nas_SLQSSet3GPP2SubscriptionInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSet3GPP2SubscriptionInfo_t * *pOutput*)

Unpack of Set 3GPP2 subscription info

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.137 int unpack_nas_SLQSSetBandPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSetBandPreference_t * *pOutput*)

Unpack sets the different system selection preferences of the device.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

9.22.4.138 int unpack_nas_SLQSSetDDTMPref (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSetDDTMPref_t * *pOutput*)

Unpack of Set DDTM pref

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.139 int unpack_nas_SLQSSetSignalStrengthsCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSetSignalStrengthsCallback_t * *pOutput*)

Unpack set signal strength thresholds callback.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.140 int unpack_nas_SLQSSetSysSelectionPref (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSetSysSelectionPref_t * *pOutput*)

Sets the different system selection preferences of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.141 int unpack_nas_SLQSSetSysSelectionPrefCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t * *pOutput*)

System Selection Preference indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.142 int unpack_nas_SLQSSetSysSelectionPrefExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSetSysSelectionPrefExt_t * *pOutput*)

Sets the different system selection preferences of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.143 int unpack_nas_SLQSSwiGetHDRPersonality (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetHDRPersonality_t * *pOutput*)

HDR Personality Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.144 int unpack_nas_SLQSSwiGetHDRProtSubtype (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetHDRProtSubtype_t * *pOutput*)

HDR Protocol Sub type Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.145 int unpack_nas_SLQSSwiGetHRPDStats (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetHRPDStats_t * *pOutput*)

HRPD Stats Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.146 int unpack_nas_SLQSSwiGetLteCQI (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetLteCQI_t * *pOutput*)

Unack fetch CQI parameters for LTE data session.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.147 int unpack_nas_SLQSSwiGetLteSccRxInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetLteSccRxInfo_t * *pOutput*)

get LTE Secondary carrier Rx signal level information unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.148 int unpack_nas_SLQSSwiHDRPersonalityCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t * *pOutput*)

unpack nas SWI Hdr Personality indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.149 int unpack_nas_SLQSSwiNetworkDebug (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiNetworkDebug_t * *pOutput*)

Network Debug Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.150 int unpack_nas_SLQSSwiPSDetach (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiPSDetach_t * *pOutput*)

PS Connection Detach Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.151 int unpack_nas_SLQSSwiRandIndicatorCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t * *pOutput*)

unpack nas SWI Rank indicator indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.152 int unpack_valid_nas_GetCDMANetworkParameters (uint8_t * *pResp*, uint8_t *u8Info*)

valid CDMA Network Parameters

Parameters

in	*pResp*	qmi response from modem
in	*u8Info*	value to check unpack_nas_GetCDMANetworkParameters_t param valid. • 0 - SCI • 1 - SCM • 2 - RegHomeSID, RegForeignSID and RegForeignNID • 3 - ForceRev0 • 4 - CustomSCP, Protocol ,Broadcast and Application • 5 - Roaming

Returns

eQCWWAN_ERR_NONE on signal info valid, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.153 int unpack_valid_nas_SLQSGetServingSystem (uint8_t * *pResp*, uint8_t *u8Info*)

valid queries information regarding the system that currently provides service

Parameters

in	*pResp*	qmi response from modem
in	*u8Info*	value to check unpack_nas_SLQSGetServingSystem_t param valid. • 0 - RoamIndicatorVal • 1 - DataSrvCapabilities • 2 - CurrentPLMN • 3 - SystemID and NetworkID • 4 - BaseStationID, BaseStationLatitude and BasestationLongitude • 5 - RoamingIndicatorList • 6 - DefaultRoamInd • 7 - Gpp2TimeZone • 8 - CDMA_P_Rev • 9 - GppTimeZone • 10 - GppNetworkDSTAdjustment • 11 - Lac • 12 - CellID • 13 - ConcSvcInfo • 14 - PRLInd • 15 - DTMInd • 16 - DetailedSvcInfo • 17 - CDMASystemInfoExt • 18 - HdrPersonality • 19 - TrackAreaCode • 20 - CallBarStatus

Returns

eQCWWAN_ERR_NONE on signal info valid, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.154 int unpack_valid_nas_SLQSGetSignalStrength (uint8_t * *pResp*, uint8_t *u8Info*)

valid queries information regarding the system that currently provides service

Parameters

in	*pResp*	qmi response from modem
in	*u8Info*	value to check unpack_nas_SLQSGetSignalStrength_t param valid. • 0 - rxSignalStrengthList and rxSignalStrengthListLen • 1 - ecioList and ecioListLen • 2 - Io • 3 - sinr • 4 - errorRateListLen, errorRateList • 5 - rsrqInfo • 6 - ltesnr • 7 - ltersrp

Returns

eQCWWAN_ERR_NONE on signal info valid, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.22.4.155 int unpack_valid_nas_SLQSNasGetSigInfo (uint8_t * *pResp*, uint8_t *u8Info*)

valid get sig info

Parameters

in	*pResp*	qmi response from modem
in	*u8Info*	value to check unpack_nas_SLQSNasGetSigInfo_t param valid. • 0 - CDMASSInfo • 1 - HDRSSInfo. • 2 - GSMSSInfo. • 3 - WCDMASSInfo. • 4 - LTESSInfo.

Returns

eQCWWAN_ERR_NONE on signal info valid, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23 pds.h File Reference

Data Structures

- struct unpack_pds_GetPDSState_t
- struct pack_pds_SetPDSState_t
- struct unpack_pds_GetPDSDefaults_t
- struct pack_pds_SetPDSDefaults_t
- struct unpack_pds_GetPortAutomaticTracking_t
- struct pack_pds_SetPortAutomaticTracking_t
- struct pack_pds_StartPDSTrackingSessionExt_t
- struct pack_pds_PDSInjectTimeReference_t
- struct unpack_pds_GetXTRAAutomaticDownload_t
- struct pack_pds_SetXTRAAutomaticDownload_t
- struct unpack_pds_GetXTRANetwork_t
- struct pack_pds_SetXTRANetwork_t
- struct unpack_pds_GetXTRAValidity_t
- struct unpack_pds_GetServiceAutomaticTracking_t
- struct pack_pds_SetServiceAutomaticTracking_t
- struct pack_pds_ResetPDSData_t
- struct pack_pds_SLQSSetAGPSConfig_t
- struct pack_pds_SLQSPDSInjectAbsoluteTimeReference_t
- struct pack_pds_SLQSGetAGPSConfig_t
- struct unpack_pds_SLQSGetAGPSConfig_t
- struct pack_pds_SLQSPDSInjectPositionData_t
- struct unpack_pds_SLQSGetGPSStateInfo_t
- struct pack_pds_SLQSSetPositionMethodState_t
- struct pack_pds_SetEventReportCallback_t
- struct unpack_pds_SetEventReport_Ind_t
- struct unpack_pds_SetPdsState_Ind_t

Typedefs

- typedef unpack_result_t unpack_pds_SetPDSState_t
- typedef unpack_result_t unpack_pds_SetPDSDefaults_t
- typedef unpack_result_t unpack_pds_SetPortAutomaticTracking_t
- typedef unpack_result_t unpack_pds_StartPDSTrackingSessionExt_t
- typedef unpack_result_t unpack_pds_StopPDSTrackingSession_t
- typedef unpack_result_t unpack_pds_PDSInjectTimeReference_t
- typedef unpack_result_t unpack_pds_SetXTRAAutomaticDownload_t
- typedef unpack_result_t unpack_pds_SetXTRANetwork_t
- typedef unpack_result_t unpack_pds_ForceXTRADownload_t
- typedef unpack_result_t unpack_pds_SetServiceAutomaticTracking_t
- typedef unpack_result_t unpack_pds_ResetPDSData_t
- typedef unpack_result_t unpack_pds_SLQSSetAGPSConfig_t
- typedef unpack_result_t unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t
- typedef unpack_result_t unpack_pds_SLQSPDSInjectPositionData_t
- typedef unpack_result_t unpack_pds_SLQSPDSDeterminePosition_t
- typedef unpack_result_t unpack_pds_SLQSSetPositionMethodState_t
- typedef unpack_result_t unpack_pds_SetEventReportCallback_t

Functions

- int pack_pds_GetPDSState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetPDSState (uint8_t *pResp, uint16_t respLen, unpack_pds_GetPDSState_t *pOutput)
- int pack_pds_SetPDSState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetPDSState_t *pReqParam)
- int unpack_pds_SetPDSState (uint8_t *pResp, uint16_t respLen, unpack_pds_SetPDSState_t *pOutput)
- int pack_pds_GetPDSDefaults (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetPDSDefaults (uint8_t *pResp, uint16_t respLen, unpack_pds_GetPDSDefaults_t *pOutput)
- int pack_pds_SetPDSDefaults (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetPDSDefaults_t *pReqParam)
- int unpack_pds_SetPDSDefaults (uint8_t *pResp, uint16_t respLen, unpack_pds_SetPDSDefaults_t *pOutput)
- int pack_pds_GetPortAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetPortAutomaticTracking (uint8_t *pResp, uint16_t respLen, unpack_pds_GetPortAutomaticTracking_t *pOutput)
- int pack_pds_SetPortAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetPortAutomaticTracking_t *pReqParam)
- int unpack_pds_SetPortAutomaticTracking (uint8_t *pResp, uint16_t respLen, unpack_pds_SetPortAutomaticTracking_t *pOutput)
- int pack_pds_StartPDSTrackingSessionExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_StartPDSTrackingSessionExt_t *pReqParam)
- int unpack_pds_StartPDSTrackingSessionExt (uint8_t *pResp, uint16_t respLen, unpack_pds_StartPDSTrackingSessionExt_t *pOutput)
- int pack_pds_StopPDSTrackingSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_StopPDSTrackingSession (uint8_t *pResp, uint16_t respLen, unpack_pds_StopPDSTrackingSession_t *pOutput)
- int pack_pds_PDSInjectTimeReference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_PDSInjectTimeReference_t *pReqParam)
- int unpack_pds_PDSInjectTimeReference (uint8_t *pResp, uint16_t respLen, unpack_pds_PDSInjectTimeReference_t *pOutput)
- int pack_pds_GetXTRAAutomaticDownload (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetXTRAAutomaticDownload (uint8_t *pResp, uint16_t respLen, unpack_pds_GetXTRAAutomaticDownload_t *pOutput)
- int pack_pds_SetXTRAAutomaticDownload (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetXTRAAutomaticDownload_t *pReqParam)
- int unpack_pds_SetXTRAAutomaticDownload (uint8_t *pResp, uint16_t respLen, unpack_pds_SetXTRAAutomaticDownload_t *pOutput)
- int pack_pds_GetXTRANetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetXTRANetwork (uint8_t *pResp, uint16_t respLen, unpack_pds_GetXTRANetwork_t *pOutput)
- int pack_pds_SetXTRANetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetXTRANetwork_t *pReqParam)
- int unpack_pds_SetXTRANetwork (uint8_t *pResp, uint16_t respLen, unpack_pds_SetXTRANetwork_t *pOutput)
- int pack_pds_GetXTRAValidity (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetXTRAValidity (uint8_t *pResp, uint16_t respLen, unpack_pds_GetXTRAValidity_t *pOutput)
- int pack_pds_ForceXTRADownload (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_ForceXTRADownload (uint8_t *pResp, uint16_t respLen, unpack_pds_ForceXTRADownload_t *pOutput)
- int pack_pds_GetServiceAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetServiceAutomaticTracking (uint8_t *pResp, uint16_t respLen, unpack_pds_GetServiceAutomaticTracking_t *pOutput)

- int pack_pds_SetServiceAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetServiceAutomaticTracking_t *pReqParam)
- int unpack_pds_SetServiceAutomaticTracking (uint8_t *pResp, uint16_t respLen, unpack_pds_SetServiceAutomaticTracking_t *pOutput)
- int pack_pds_ResetPDSData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_ResetPDSData_t *pReqParam)
- int unpack_pds_ResetPDSData (uint8_t *pResp, uint16_t respLen, unpack_pds_ResetPDSData_t *pOutput)
- int pack_pds_SLQSSetAGPSConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSSetAGPSConfig_t *pReqParam)
- int unpack_pds_SLQSSetAGPSConfig (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSSetAGPSConfig_t *pOutput)
- int pack_pds_SLQSPDSInjectAbsoluteTimeReference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSPDSInjectAbsoluteTimeReference_t *pReqParam)
- int unpack_pds_SLQSPDSInjectAbsoluteTimeReference (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t *pOutput)
- int pack_pds_SLQSGetAGPSConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSGetAGPSConfig_t *pReqParam)
- int unpack_pds_SLQSGetAGPSConfig (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSGetAGPSConfig_t *pOutput)
- int pack_pds_SLQSPDSInjectPositionData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSPDSInjectPositionData_t *pReqParam)
- int unpack_pds_SLQSPDSInjectPositionData (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSPDSInjectPositionData_t *pOutput)
- int pack_pds_SLQSPDSDeterminePosition (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_SLQSPDSDeterminePosition (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSPDSDeterminePosition_t *pOutput)
- int pack_pds_SLQSGetGPSStateInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_SLQSGetGPSStateInfo (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSGetGPSStateInfo_t *pOutput)
- int pack_pds_SLQSSetPositionMethodState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSSetPositionMethodState_t *pReqParam)
- int unpack_pds_SLQSSetPositionMethodState (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSSetPositionMethodState_t *pOutput)
- int pack_pds_SetEventReportCallback (pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen, pack_pds_SetEventReportCallback_t *pReqParam)
- int unpack_pds_SetEventReportCallback (uint8_t *pResp, uint16_t respLen, unpack_pds_SetEventReportCallback_t *pOutput)
- int unpack_pds_SetEventReport_Ind (uint8_t *pResp, uint16_t respLen, unpack_pds_SetEventReport_Ind_t *pOutput)
- int unpack_pds_SetPdsState_Ind (uint8_t *pResp, uint16_t respLen, unpack_pds_SetPdsState_Ind_t *pOutput)

9.23.1 Typedef Documentation

9.23.1.1 typedef unpack_result_t unpack_pds_ForceXTRADownload_t

9.23.1.2 typedef unpack_result_t unpack_pds_PDSInjectTimeReference_t

9.23.1.3 typedef unpack_result_t unpack_pds_ResetPDSData_t

9.23.1.4 typedef unpack_result_t unpack_pds_SetEventReportCallback_t

9.23.1.5 typedef unpack_result_t unpack_pds_SetPDSDefaults_t

9.23.1.6 typedef unpack_result_t unpack_pds_SetPDSState_t

9.23.1.7 typedef unpack_result_t unpack_pds_SetPortAutomaticTracking_t

9.23.1.8 typedef unpack_result_t unpack_pds_SetServiceAutomaticTracking_t

9.23.1.9 typedef unpack_result_t unpack_pds_SetXTRAAutomaticDownload_t

9.23.1.10 typedef unpack_result_t unpack_pds_SetXTRANetwork_t

9.23.1.11 typedef unpack_result_t unpack_pds_SLQSPDSDeterminePosition_t

9.23.1.12 typedef unpack_result_t unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t

9.23.1.13 typedef unpack_result_t unpack_pds_SLQSPDSInjectPositionData_t

9.23.1.14 typedef unpack_result_t unpack_pds_SLQSSetAGPSConfig_t

9.23.1.15 typedef unpack_result_t unpack_pds_SLQSSetPositionMethodState_t

9.23.1.16 typedef unpack_result_t unpack_pds_StartPDSTrackingSessionExt_t

9.23.1.17 typedef unpack_result_t unpack_pds_StopPDSTrackingSession_t

9.23.2 Function Documentation

9.23.2.1 int pack_pds_ForceXTRADownload (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Forces the XTRA database to be downloaded to the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.2 int pack_pds_GetPDSDefaults (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get the default tracking session configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.3 int pack_pds_GetPDSState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get current PDS state pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.4 int pack_pds_GetPortAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Returns the automatic tracking configuration for the NMEA COM port pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.5 int pack_pds_GetServiceAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

automatic tracking state for the service pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.6 int pack_pds_GetXTRAAutomaticDownload (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

XTRA automatic database download configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.7 int pack_pds_GetXTRANetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

XTRA WWAN network preference parameter pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.8 int pack_pds_GetXTRAValidity (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

XTRA database validity period parameter pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.9 int pack_pds_PDSInjectTimeReference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_PDSInjectTimeReference_t * *pReqParam*)

Inject system time parameters pack.

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.10 int pack_pds_ResetPDSData (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_ResetPDSData_t * *pReqParam*)

Resets the specified PDS data pack.

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.11 int pack_pds_SetEventReportCallback (pack_qmi_t * *pCtx*, uint8_t * *pReq*, uint16_t * *pLen*, pack_pds_SetEventReportCallback_t * *pReqParam*)

Enables/disables the PDS Event report callback function pack.

Parameters

in, out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.12 int pack_pds_SetPDSDefaults (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetPDSDefaults_t * *pReqParam*)

Sets the default tracking session configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.13 int pack_pds_SetPDSState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetPDSState_t * *pReqParam*)

Sets the PDS state pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.14 int pack_pds_SetPortAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetPortAutomaticTracking_t * *pReqParam*)

Sets the automatic tracking configuration for the NMEA COM port pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.15 int pack_pds_SetServiceAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetServiceAutomaticTracking_t * *pReqParam*)

Sets the automatic tracking state for the service pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.16 int pack_pds_SetXTRAAutomaticDownload (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetXTRAAutomaticDownload_t * *pReqParam*)

Sets the XTRA automatic database download configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.17 int pack_pds_SetXTRANetwork (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_pds_SetXTRANetwork_t * pReqParam)

Sets the XTRA WWAN network preference pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.18 int pack_pds_SLQSGetAGPSConfig (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_pds_SLQSGetAGPSConfig_t * pReqParam)

Gets the PDS AGPS (MS-based) configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.19 int pack_pds_SLQSGetGPSStateInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Queries the MSM GPS server for receiver state information pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.20 int pack_pds_SLQSPDSDeterminePosition (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Requests the MSM GPS service to obtain the current position for manually controlled tracking sessions pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.21 int pack_pds_SLQSPDSInjectAbsoluteTimeReference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSPDSInjectAbsoluteTimeReference_t * *pReqParam*)

Injects a absolute time reference into the PDS engine pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.22 int pack_pds_SLQSPDSInjectPositionData (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSPDSInjectPositionData_t * *pReqParam*)

Injects position data into the PDS engine pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.23 int pack_pds_SLQSSetAGPSConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSSetAGPSConfig_t * *pReqParam*)

Sets the PDS AGPS (MS-based) configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.24 int pack_pds_SLQSSetPositionMethodState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSSetPositionMethodState_t * *pReqParam*)

Sets the state of positioning methods for the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.25 int pack_pds_StartPDSTrackingSessionExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_StartPDSTrackingSessionExt_t * *pReqParam*)

Starts a PDS tracking session pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.26 int pack_pds_StopPDSTrackingSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Stops a PDS tracking session pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.27 int unpack_pds_ForceXTRADownload (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_ForceXTRADownload_t * *pOutput*)

Forces the XTRA database to be downloaded to the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.28 int unpack_pds_GetPDSDefaults (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetPDSDefaults_t * *pOutput*)

Get the default tracking session configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.29 int unpack_pds_GetPDSState (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetPDSState_t * *pOutput*)

Get current PDS state unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.30 int unpack_pds_GetPortAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetPortAutomaticTracking_t * *pOutput*)

Returns the automatic tracking configuration for the NMEA COM port unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.31 int unpack_pds_GetServiceAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetServiceAutomaticTracking_t * *pOutput*)

automatic tracking state for the service unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.32 int unpack_pds_GetXTRAAutomaticDownload (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetXTRAAutomaticDownload_t * *pOutput*)

XTRA automatic database download configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.33 int unpack_pds_GetXTRANetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetXTRANetwork_t * *pOutput*)

XTRA WWAN network preference parameter unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.34 int unpack_pds_GetXTRAValidity (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetXTRAValidity_t * *pOutput*)

XTRA database validity period parameter unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.35 int unpack_pds_PDSInjectTimeReference (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_PDSInjectTimeReference_t * *pOutput*)

Inject system time parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.36 int unpack_pds_ResetPDSData (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_ResetPDSData_t * *pOutput*)

Resets the specified PDS data unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.37 int unpack_pds_SetEventReport_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetEventReport_Ind_t * *pOutput*)

PDS Event Report Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.38 int unpack_pds_SetEventReportCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetEventReportCallback_t * *pOutput*)

Enables/disables the PDS Event report callback function unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.39 int unpack_pds_SetPDSDefaults (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetPDSDefaults_t * *pOutput*)

Sets the default tracking session configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.40 int unpack_pds_SetPDSState (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetPDSState_t * *pOutput*)

Sets the PDS state unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.41 int unpack_pds_SetPdsState_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetPdsState_Ind_t * *pOutput*)

PDS State Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.42 int unpack_pds_SetPortAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetPortAutomaticTracking_t * *pOutput*)

Sets the automatic tracking configuration for the NMEA COM port unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.43 int unpack_pds_SetServiceAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetServiceAutomaticTracking_t * *pOutput*)

Sets the automatic tracking state for the service unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.44 int unpack_pds_SetXTRAAutomaticDownload (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetXTRAAutomaticDownload_t * *pOutput*)

Sets the XTRA automatic database download configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.45 int unpack_pds_SetXTRANetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetXTRANetwork_t * *pOutput*)

Sets the XTRA WWAN network preference unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.46 int unpack_pds_SLQSGetAGPSConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSGetAGPSConfig_t * *pOutput*)

Gets the PDS AGPS (MS-based) configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.47 int unpack_pds_SLQSGetGPSStateInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSGetGPSStateInfo_t * *pOutput*)

Queries the MSM GPS server for receiver state information unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.48 int unpack_pds_SLQSPDSDeterminePosition (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSPDSDeterminePosition_t * *pOutput*)

Requests the MSM GPS service to obtain the current position for manually controlled tracking sessions unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.49 int unpack_pds_SLQSPDSInjectAbsoluteTimeReference (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t * *pOutput*)

Injects a absolute time reference into the PDS engine unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.50 int unpack_pds_SLQSPDSInjectPositionData (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSPDSInjectPositionData_t * *pOutput*)

Injects position data into the PDS engine unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.51 int unpack_pds_SLQSSetAGPSConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSSetAGPSConfig_t * *pOutput*)

Sets the PDS AGPS (MS-based) configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.52 int unpack_pds_SLQSSetPositionMethodState (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSSetPositionMethodState_t * *pOutput*)

Sets the state of positioning methods for the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.53 int unpack_pds_StartPDSTrackingSessionExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_StartPDSTrackingSessionExt_t * *pOutput*)

Starts a PDS tracking session unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.23.2.54 int unpack_pds_StopPDSTrackingSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_StopPDSTrackingSession_t * *pOutput*)

Stops a PDS tracking session unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.24 qaGobiApiTableBandClasses.h File Reference

Network Access Service API Band Classes table.

9.24.1 Detailed Description

Network Access Service API Band Classes table.

9.24.2 Band Classes (Value - Description)

- 0 - CDMA Band Class 0
- 1 - CDMA Band Class 1
- 3 - CDMA Band Class 3
- 4 - CDMA Band Class 4
- 5 - CDMA Band Class 5
- 6 - CDMA Band Class 6
- 7 - CDMA Band Class 7
- 8 - CDMA Band Class 8
- 9 - CDMA Band Class 9
- 10 - CDMA Band Class 10
- 11 - CDMA Band Class 11
- 12 - CDMA Band Class 12
- 13 - CDMA Band Class 13
- 14 - CDMA Band Class 14
- 15 - CDMA Band Class 15
- 16 - CDMA Band Class 16
- 17 - CDMA Band Class 17
- 18 - CDMA Band Class 18
- 19 - CDMA Band Class 19
- 40 - GSM 450
- 41 - GSM 480
- 42 - GSM 750
- 43 - GSM 850

- 44 - GSM 900 (Extended)
- 45 - GSM 900 (Primary)
- 46 - GSM 900 (Railways)
- 47 - GSM 1800
- 48 - GSM 1900
- 80 - WCDMA 2100
- 81 - WCDMA PCS 1900
- 82 - WCDMA DCS 1800
- 83 - WCDMA 1700 (US)
- 84 - WCDMA 850
- 85 - WCDMA 800
- 86 - WCDMA 2600
- 87 - WCDMA 900
- 88 - WCDMA 1700 (Japan)
- 90 - WCDMA 1500 band (Japan)
- 91 - WCDMA 850 band (Japan)
- < Reserved 89, 92-109 for WCDMA band classes>
- 110 - WLAN US 2400 MHz
- 111 - WLAN JAPAN 2400 MHz
- 112 - WLAN EUROPEAN 2400 MHz
- 113 - WLAN FRANCE 2400 MHz
- 114 - WLAN SPAIN 2400 MHz
- 115 - WLAN US 5000 MHz band
- 116 - WLAN JAPAN 5000 MHz
- 117 - WLAN EUROPEAN 5000 MHz
- 118 - WLAN FRANCE 5000 MHz
- 119 - WLAN SPAIN 5000 MHz

9.24.2.1 LTE Bands

- 28 - LTE Band Class 28
- 39 - LTE Band Class 39
- 40 - LTE Band Class 40
- 41 - LTE Band Class 41
- 120 - FDD UL:1920-1980; DL:2110-2170; E-UTRA Operating Band 1
- 121 - FDD UL:1850-1910; DL:1930-1990; E-UTRA Operating Band 2
- 122 - FDD UL:1710-1785; DL:1805-1880; E-UTRA Operating Band 3

- 123 - FDD UL:1710-1755; DL:2110-2155; E-UTRA Operating Band 4
- 124 - FDD UL: 824- 849; DL: 869- 894; E-UTRA Operating Band 5
- 125 - FDD UL: 830- 840; DL: 875- 885; E-UTRA Operating Band 6
- 126 - FDD UL:2500-2570; DL:2620-2690; E-UTRA Operating Band 7
- 127 - FDD UL: 880- 915; DL: 925- 960; E-UTRA Operating Band 8
- 128 - FDD UL:1749.9-1784.9; DL:1844.9-1879.9; E-UTRA Operating Band 9
- 129 - FDD UL:1710-1770; DL:2110-2170; E-UTRA Operating Band 10
- 130 - FDD UL:1427.9-1452.9; DL:1475.9-1500.9; E-UTRA Operating Band 11
- 131 - FDD UL:698-716; DL:728-746; E-UTRA Operating Band 12
- 132 - FDD UL: 777- 787; DL: 746-756; E-UTRA Operating Band 13
- 133 - FDD UL: 788- 798; DL: 758-768; E-UTRA Operating Band 14
- 134 - FDD UL: 704-716; DL: 734-746; E-UTRA Operating Band 17
- 135 - TDD LTE UL: 1900-1920; DL: 1900-1920; E-UTRA Operating Band 33
- 136 - TDD LTE UL: 2010-2025; DL: 2010-2025; E-UTRA Operating Band 34
- 137 - TDD LTE UL: 1850-1910; DL: 1850-1910; E-UTRA Operating Band 35
- 138 - TDD LTE UL: 1930-1990; DL: 1930-1990; E-UTRA Operating Band 36
- 139 - TDD LTE UL: 1910-1930; DL: 1910-1930; E-UTRA Operating Band 37
- 140 - TDD LTE UL: 2570-2620; DL: 2570-2620; E-UTRA Operating Band 38
- 141 - TDD LTE UL: 1880-1920; DL: 1880-1920; E-UTRA Operating Band 39
- 142 - TDD LTE UL: 2300-2400; DL: 2300-2400; E-UTRA Operating Band 40
- 143 - FDD LTE UL: 815-830; DL: 860-875; E-UTRA Operating Band 18
- 144 - FDD LTE UL: 830-845; DL: 875-890; E-UTRA Operating Band 19
- 145 - FDD LTE UL: 832-862; DL: 791-821; E-UTRA Operating Band 20
- 146 - FDD LTE UL: 1447.9-1462.9; DL: 1495.9-1510.9; E-UTRA Operating Band 21
- 147 - FDD LTE UL: 1626.5-1660.5; DL: 1525-1559; E-UTRA Operating Band 24
- 148 - FDD LTE UL: 1850-19195; DL: 1930-1995; E-UTRA Operating Band 25
- 149 - TDD LTE UL: 2496-2690; DL: 2496-2690; E-UTRA Operating Band 41
- 150 - TDD LTE UL: 3400-3600; DL: 3400-3600; E-UTRA Operating Band 42
- 151 - TDD LTE UL: 3600-3800; DL: 3600-3800; E-UTRA Operating Band 43
- 152 - E-UTRA Operating Band 23
- 153 - E-UTRA Operating Band 26
- 154 - E-UTRA Operating Band 32
- 155 - E-UTRA Operating Band 125
- 156 - E-UTRA Operating Band 126
- 157 - E-UTRA Operating Band 127
- 158 - E-UTRA Operating Band 28

- 159 - E-UTRA Operating Band 29
- 160 - E-UTRA Operating Band 30
- 161 - E-UTRA Operating Band 66
- 162 - E-UTRA Operating Band 250
- 163 - E-UTRA Operating Band 46
- 164 - E-UTRA Operating Band 27
- 165 - E-UTRA Operating Band 31
- 166 - E-UTRA Operating Band 71
- 167 - E-UTRA Operating Band 47
- 168 - E-UTRA Operating Band 48
- 200 - TD-SCDMA Band A
- 201 - TD-SCDMA Band B
- 202 - TD-SCDMA Band C
- 203 - TD-SCDMA Band D
- 204 - TD-SCDMA Band E
- 205 - TD-SCDMA Band F
- 250 - NR5G Band 1
- 251 - NR5G Band 2

Copyright: © 2018-2020 Sierra Wireless, Inc. all rights reserved

9.25 qaGobiApiTableCallControlReturnReasons.h File Reference

Call Control Return Reasons table.

9.25.1 Detailed Description

Call Control Return Reasons table.

9.25.2 S1

- 0x01 - QMI_VOICE_REASON_FWD_UNCONDITIONAL - Unconditional call forwarding
- 0x02 - QMI_VOICE_REASON_FWD_MOBILEBUSY - Forward when the mobile is busy
- 0x03 - QMI_VOICE_REASON_FWD_NOREPLY - Forward when there is no reply
- 0x04 - QMI_VOICE_REASON_FWD_UNREACHABLE - Forward when the call is unreachable
- 0x05 - QMI_VOICE_REASON_FWD_ALLFORWARDING - All forwarding
- 0x06 - QMI_VOICE_REASON_FWD_ALLCONDITIONAL - All conditional forwarding
- 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing
- 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal

- 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME - Outgoing external to home
- 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming
- 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming
- 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred
- 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred
- 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
- 0x0F - QMI_VOICE_REASON_CALLWAITING - Call waiting
- 0x10 - VOICE_CC_SUPS_RESULT_REASON_CLIP - CLIP
- 0x11 - VOICE_CC_SUPS_RESULT_REASON_CLIR - CLIR
- 0x12 - VOICE_CC_SUPS_RESULT_REASON_COLP - COLP
- 0x13 - VOICE_CC_SUPS_RESULT_REASON_COLR - COLR
- 0x14 - VOICE_CC_SUPS_RESULT_REASON_CNAP - CNAP
- 0xFF - Not Available

Copyright: © 2012-2020 Sierra Wireless, Inc. all rights reserved

9.26 qaGobiApiTableCallEndReasons.h File Reference

Wireless Data Service Call End Reasons.

9.26.1 Detailed Description

Wireless Data Service Call End Reasons.

9.26.2 Call end reason codes (Code - Reason)

9.26.2.1 Technology-agnostic call end reasons

- 1 - Reason unspecified, check the verbose call end reason
- 2 - Client ended the call
- 3 - Device has no service
- 4 - Call ended abnormally
- 5 - Received release from base station; no reason given
- 6 - Access attempt already in progress; SD2.0 only
- 7 - Access failure for reason other than the above
- 8 - Call rejected because of redirection or handoff
- 9 - Call failed because close is in progress
- 10 - Authentication failed, 3GPP equivalent ESM(EPS Session Management) cause code value 29, User authentication failed
- 11 - Call ended because of internal call end. This error code is returned when data call is brought down due to some unknown error, such as not specific to any RAT

- 12 - Call ended because of internal error. This error code is returned when data call is brought down due to some unspecified internal error, such as NULL pointer
- 13 - Internal unknown cause code

9.26.2.2 EVDO CDMA 1xEV-DO

- 500 - Device is CDMA-locked until power cycle
- 501 - Received intercept from base station; origination only
- 502 - Received reorder from base station; origination only
- 503 - Received release from base station; service option reject
- 504 - Received incoming call from base station
- 505 - Received alert stop from base station; incoming only
- 506 - Received end activation; OTASP call only
- 507 - Max access probes transmitted
- 508 - Concurrent service is not supported by base station
- 509 - No response received from base station
- 510 - Call rejected by the base station; CDMA only
- 511 - Concurrent services requested were not compatible; CDMA only
- 512 - Corresponds to CM CALL ORIG ERR ALREADY IN TC
- 513 - Used if Call manager subsystem is ending a GPS call in favor of a user call
- 514 - Used if Call manager subsystem is ending a SMS call in favor of a user call
- 515 - CDMA Only; Device has no service

9.26.2.3 WCDMA/GSM call end reasons

- 1000 - Call origination request failed; WCDMA/GSM Only
- 1001 - Client rejected the incoming call; WCDMA/GSM Only
- 1002 - Device has no UMTS service; WCDMA/GSM Only
- 1003 - Network ended the call, look in cc call; WCDMA/GSM Only
- 1004 - LLC(Logical Link Control) or SNDCP(Sub Network Dependent Convergence Protocol) failure
- 1005 - Insufficient resources, 3GPP equivalent ESM(EPS Session Management) cause code value 26, Insufficient resources
- 1006 - Service option temporarily out of order, 3GPP equivalent ESM(EPS Session Management) cause code value 34, Service option temporarily out of order
- 1007 - PTI already used, 3GPP equivalent ESM(EPS Session Management) cause code value 35, PTI(-Procedure Transaction Identity) already in use
- 1008 - Regular PDP context deactivation, 3GPP equivalent ESM(EPS Session Management) cause code value 36, Regular deactivation
- 1009 - Network failure, 3GPP equivalent ESM(EPS Session Management) cause code value 38, Network failure

- 1010 - Reactivation requested, 3GPP equivalent ESM(EPS Session Management) cause code value 39, Reactivation requested
- 1011 - Protocol error, unspecified, 3GPP equivalent ESM(EPS Session Management) cause code value 111, Protocol error, unspecified
- 1012 - Operator determined barring, 3GPP equivalent ESM(EPS Session Management) cause code value 8, Operator Determined Barring
- 1013 - Unknown or missing Access Point Name (APN), 3GPP equivalent ESM(EPS Session Management) cause code value 27, Missing or unknown APN
- 1014 - Unknown PDP address or PDP type, 3GPP equivalent ESM(EPS Session Management) cause code value 28, Unknown PDN type
- 1015 - Activation rejected by GGSN, 3GPP equivalent ESM(EPS Session Management) cause code value 30, Requested rejected by Serving GW or PDN GW
- 1016 - Activation rejected, unspecified, 3GPP equivalent ESM(EPS Session Management) cause code value 31, Request rejected, unspecified
- 1017 - Service option not supported, 3GPP equivalent ESM(EPS Session Management) cause code value 32, Service option not supported
- 1018 - Requested service option not subscribed, 3GPP equivalent ESM(EPS Session Management) cause code value 33, Requested service option not subscribed
- 1019 - EPS Quality of Service (QoS) not accepted, 3GPP equivalent ESM(EPS Session Management) cause code value 37, EPS QoS not accepted
- 1020 - Semantic error in the TFT operation, 3GPP equivalent ESM(EPS Session Management) cause code value 41, Semantic error in the TFT operation
- 1021 - Syntactical error in the TFT operation, 3GPP equivalent ESM(EPS Session Management) cause code value 42, Syntactical error in the TFT operation
- 1022 - Unknown PDP context, 3GPP equivalent ESM(EPS Session Management) cause code value 43, Invalid EPS bearer identity
- 1023 - Semantic errors in packet filter(s), 3GPP equivalent ESM(EPS Session Management) cause code value 44, Semantic errors in packet filter(s)
- 1024 - Syntactical error in packet filter(s), 3GPP equivalent ESM(EPS Session Management) cause code value 45, Syntactical errors in packet filter(s)
- 1025 - PDP context without TFT already activated, 3GPP equivalent ESM(EPS Session Management) cause code value 46, Unused
- 1026 - Invalid transaction identifier value, 3GPP equivalent ESM(EPS Session Management) cause code value 81, Invalid PTI value
- 1027 - Semantically incorrect message, 3GPP equivalent ESM(EPS Session Management) cause code value 95, Semantically incorrect message
- 1028 - Invalid mandatory information, 3GPP equivalent ESM(EPS Session Management) cause code value 96, Invalid mandatory information
- 1029 - Message type non-existent or not implemented, 3GPP equivalent ESM(EPS Session Management) cause code value 97, Message type non-existent or not implemented
- 1030 - Message not compatible with state, 3GPP equivalent ESM(EPS Session Management) cause code value 98, Message type not compatible with the protocol state
- 1031 - Information element nonexistent or not implemented, 3GPP equivalent ESM(EPS Session Management) cause code value 99, Information element non-existent or not implemented

- 1032 - Conditional information element error, 3GPP equivalent ESM(EPS Session Management) cause code value 100, Conditional IE error
- 1033 - Message not compatible with protocol state, 3GPP equivalent ESM(EPS Session Management) cause code value 101, Message not compatible with the protocol state
- 1034 - APN restriction value incompatible with active PDP context, 3GPP equivalent ESM(EPS Session Management) cause code value 112, APN restriction value incompatible with
  - active EPS bearer context
- 1035 - No GPRS context present
- 1036 - Requested feature not supported, 3GPP equivalent ESM(EPS Session Management) cause code value 40, Feature not supported
- 1037 - Illegal MS, 3GPP equivalent EMM(EPS Mobility Management) cause code value 3, Illegal UE (MS)
- 1038 - Illegal ME, 3GPP equivalent EMM(EPS Mobility Management) cause code value 6, Illegal ME. This error code is sent to the MS if the ME used is not acceptable
  - to the network, e.g. blacklisted
- 1039 - GPRS and non GPRS services not allowed
- 1040 - GPRS services not allowed
- 1041 - MS identity not derived by the network, 3GPP equivalent EMM(EPS Mobility Management) cause code value 9, UE (MS) Identify cannot be derived by the network
- 1042 - Implicitly detached, 3GPP equivalent EMM(EPS Mobility Management) cause code value 10, Implicitly Detached
- 1043 - PLMN not allowed, 3GPP equivalent EMM(EPS Mobility Management) cause code value 11, PLMN not allowed
- 1044 - LA not allowed, this cause is sent to the MS if it requests location updating in a location area where the HPLMN determines that the MS, by subscription, is not allowed to operate.
- 1045 - GPRS services not allowed in PLMN
- 1046 - PDP duplicate
- 1047 - UE radio access technology change
- 1048 - app preempted
- 1049 - Congestion, This cause is sent if the service request or LOCATION UPDATING REQUEST message cannot be actioned because of congestion (e.g. congestion of the MSC or SGSN or GGSN or PDN Gateway; no channel; facility busy/congested etc.).
- 1050 - No PDP context activated
- 1051 - Access class DSAC rejection

9.26.2.4 EVDO CDMA 1xEV-DO

- 1500 - Abort connection setup due to the reception of a Connection Deny message with deny code set to either general or network busy.
- 1501 - Abort connection setup due to the reception of a Connection Deny message with deny code set to either billing or authentication failure.
- 1502 - Change HDR system due to redirection or PRL not preferred
- 1503 - Exit HDR due to redirection or PRL not preferred

- 1504 - No HDR session
- 1505 - Used if Call manager is ending an HDR call origination in favor of a GPS fix
- 1506 - Connection setup timeout
- 1507 - Call manager released HDR call so 1x call can continue

9.26.2.5 call end reason type

- 1 - Mobile IP
- 2 - Internal
- 3 - Call Manager defined
- 6 - 3GPP specification defined
- 7 - PPP
- 8 - EHRPD
- 9 - IPv6

9.26.2.6 Mobile IP call end reasons (Type=1)

- 64 - MIP(Mobile IP) FA(Foreign Agent) ERR REASON UNSPECIFIED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration with unspecified reason
- 65 - MIP(Mobile IP) FA(Foreign Agent) ERR ADMINISTRATIVELY PROHIBITED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent administratively prohibited MIP registration
- 66 - MIP(Mobile IP) FA(Foreign Agent) ERR INSUFFICIENT RESOURCES, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to insufficient resources
- 67 - MIP(Mobile IP) FA(Foreign Agent) ERR MOBILE NODE AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because MN-AAA authenticator was wrong
- 68 - MIP(Mobile IP) FA(Foreign Agent) ERR HA AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because of home agent authentication failure
- 69 - MIP(Mobile IP) FA(Foreign Agent) ERR REQUESTED LIFETIME TOO LONG, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because requested lifetime is too long
- 70 - MIP(Mobile IP) FA(Foreign Agent) ERR MALFORMED REQUEST, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to malformed request
- 71 - MIP(Mobile IP) FA(Foreign Agent) ERR MALFORMED REPLY, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to malformed reply
- 72 - MIP(Mobile IP) FA(Foreign Agent) ERR ENCAPSULATION UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because requested encapsulation is unavailable
- 73 - MIP(Mobile IP) FA(Foreign Agent) ERR VJHC UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because VJ Header Compression is unavailable

- 74 - MIP(Mobile IP) FA(Foreign Agent) ERR REVERSE TUNNEL UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because reverse tunnel is unavailable
- 75 - MIP(Mobile IP) FA(Foreign Agent) ERR REVERSE TUNNEL IS MANDATORY AND T BIT NOT SET, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because reverse tunnel is mandatory but not requested by device
- 79 - MIP(Mobile IP) FA(Foreign Agent) ERR DELIVERY STYLE NOT SUPPORTED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because delivery style is not supported
- 97 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING NAI, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing NAI
- 98 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING HA, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing Home Agent
- 99 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING HOME ADDR, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing Home Address
- 104 - MIP(Mobile IP) FA(Foreign Agent) ERR UNKNOWN CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to unknown challenge
- 105 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing challenge
- 106 - MIP(Mobile IP) FA(Foreign Agent) ERR STALE CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to stale challenge
- 128 - MIP(Mobile IP) FA(Home Agent) ERR REASON UNSPECIFIED, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration with unspecified reason
- 129 - MIP(Mobile IP) FA(Home Agent) ERR ADMINISTRATIVELY PROHIBITED, this error code is returned when the data call bring up fails in MIP setup phase since home agent administratively prohibited MIP registration
- 130 - MIP(Mobile IP) FA(Home Agent) ERR INSUFFICIENT RESOURCES, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to insufficient resources
- 131 - MIP(Mobile IP) FA(Home Agent) ERR MOBILE NODE AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since home agent fails authentication because MN-HA authenticator was wrong
- 132 - MIP(Mobile IP) FA(Home Agent) ERR FA AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to FA authentication failure
- 133 - MIP(Mobile IP) FA(Home Agent) ERR REGISTRATION ID MISMATCH, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to registration id mismatch
- 134 - MIP(Mobile IP) FA(Home Agent) ERR MALFORMED REQUEST, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to malformed request
- 136 - MIP(Mobile IP) FA(Home Agent) ERR UNKNOWN HA ADDR, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to unknown Home Agent address. This code is returned by a home agent when the mobile node is performing dynamic home agent address resolution as described in RFC 3220 (IP Mobility Support for IPV4) Sections 3.6.1.1 and 3.6.1.2

- 137 - MIP(Mobile IP) FA(Home Agent) ERR REVERSE TUNNEL UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration because reverse tunnel is unavailable
- 138 - MIP(Mobile IP) FA(Home Agent) ERR REVERSE TUNNEL IS MANDATORY AND T BIT NOT SET, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration because reverse tunnel is mandatory but not requested by device
- 139 - MIP(Mobile IP) FA(Home Agent) ERR ENCAPSULATION UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to encapsulation unavailable
- 65536 - MIP ERR REASON UNKNOWN

9.26.2.7 Internal call end reasons (Type=2)

- 200 - INTERNAL MIN, internal error table offset value, no meaningful message to the error.
- 201 - INTERNAL ERROR, this error code is returned when data call is brought down due to some unspecified internal error
- 202 - CALL ENDED
- 203 - INTERNAL UNKNOWN CAUSE CODE, this error code is returned when data call is brought down due to some unknown error
- 204 - UNKNOWN CAUSE CODE, this error code is returned when data call is brought down due to some unknown error
- 205 - CLOSE IN PROGRESS
- 206 - NETWORK INITIATED TERMINATION
- 207 - APP PREEMPTED
- 208 - ERR PDN IPV4 CALL DISALLOWED, this error code is returned when V4 PDN is in throttled state due to network providing only V6 address during the previous VSNCP bring up (subs_limited_to_v6). The time for which the IPv4 PDN is throttled is determined by the IPv4 throttling timers maintained in the profile
- 209 - ERR PDN IPV4 CALL THROTTLED, this error code is returned when V4 PDN is in throttled state due to previous VSNCP bring up failure(s). The time for which the IPv4 PDN is throttled is determined by the IPv4 throttling timers maintained in the profile
- 210 - ERR PDN IPV6 CALL DISALLOWED, this error code is returned when V6 PDN is in throttled state due to network providing only V4 address during the previous VSNCP bring up (subs_limited_to_v4). The time for which the IPv6 PDN is throttled is determined by the IPv6 throttling timers maintained in the profile
- 211 - ERR PDN IPV6 CALL THROTTLED, this error code is returned when V6 PDN is in throttled state due to previous VSNCP bring up failure(s). The time for which the IPv6 PDN is throttled is determined by the IPv6 throttling timers maintained in the profile
- 212 - MODEM RESTART
- 213 - PDP PPP NOT SUPPORTED
- 214 - UNPREFERRED RAT, this error code is returned when data call is brought down since the RAT on which the data call is attempted/connected is no longer the preferred RAT
- 215 - PHYS LINK CLOSE IN PROGRESS, this error code is returned when data call bring up is rejected because physical link is in the process of cleanup
- 216 - APN PENDING HANDOVER, this error code is returned when interface bring up is attempted for an APN that is yet to be handed over to target RAT
- 217 - PROFILE BEARER INCOMPATIBLE

- 218 - MMGSDI CARD EVT, this error code is returned when data call is brought down because card got refreshed/removed
- 219 - LPM OR PWR DOWN, this error code is returned when data call is brought down because device is going into lower power mode or powering down
- 220 - APN DISABLED, this error code is returned when APN is disabled in card
- 221 - MPIT EXPIRED, this error code is returned when data call is brought down because maximum PPP inactivity timer expired
- 222 - IPV6 ADDR TRANSFER FAILED
- 223 - TRAT SWAP FAILED
- 224 - EHRPD TO HRPD FALLBACK, this error code is returned when data call is brought down because device falls back from eHRPD to HRPD (not because of OOS on eHRPD but due to operator/spec driven eHRPD to HRPD fallback requirements)
- 225 - MANDATORY APN DISABLED, this error code is returned when any mandatory APN is disabled, and MinApnList Disallow call config item is set to TRUE in device
- 226 - MIP CONFIG FAILURE, this error code is returned when UE is in MIP Only config (QCMIP=2) but MIP config fails on call bring up due to incorrect provisioning
- 227 - INTERNAL_PDN_INACTIVITY_TIMER_EXPIRED, this error code is returned when PDN inactivity timer expired due to no data transmission in a configurable duration of time
- 228 - MAX_V4_CONNECTIONS, this error code is returned when IPv4 data call bring up is rejected because the UE already maintains the allotted maximum number of IPv4 data connections
- 229 - MAX_V6_CONNECTIONS, this error code is returned when IPv6 data call bring up is rejected because the UE already maintains the allotted maximum number of IPv6 data connections
- 230 - APN_MISMATCH, this error code is returned when New PDN bring up is rejected during interface selection because the UE has already allotted the available interfaces for other PDNs
- 231 - IP_VERSION_MISMATCH, this error code is returned when New call bring up is rejected because the existing data call IP type does not match the requested IP type
- 232 - DUN_CALL_DISALLOWED, this error code is returned when DUN call bring up is rejected because the UE is in eHRPD RAT
- 233 - INVALID_PROFILE, this error code is returned when Call bring up was requested with an invalid profile
- 234 - INTERNAL_EPC_NONEPC_TRANSITION, this error code is returned when Data call is rejected or brought down because the UE is in transition between EPC and non-EPC RAT
- 235 - INVALID_PROFILE_ID, this error code is returned when Call bring up was requested with an invalid profile ID
- 236 - INTERNAL_CALL_ALREADY_PRESENT, this error code is returned when a data call with the same policy is already connected or in the process of bring up on another RmNet instance
- 237 - IFACE_IN_USE, this error code is returned when the current IFACE is in use
- 238 - IP_PDP_MISMATCH, this error code is returned when a PPP call is attempted on a PDP profile
- 239 - APN_DISALLOWED_ON_ROAMING, this error code is returned when PDN connection to the APN is disallowed on the roaming network. The DS generates DS_SYS_EVENT_3GPP_ROAMING_DISALLOWED_INFO to inform clients of the APNs that must deregister and disconnect (if the PDN is up) while the UE is on the roaming network. Clients must deregister or disconnect if the APN name corresponding to the client is listed in this system event. Clients can register again if the APN name corresponding to the client is not listed in this system event.

- 240 - APN_PARAM_CHANGE, this error code is returned when Failure to reestablish the PDN with the changed parameters; when APN-related parameters are changed, the PDN associated with the parameters must be brought down and reestablished with the changed parameters
- 241 - IFACE_IN_USE_CFG_MATCH, this error code is returned when the IFACE is already in use with a matching configuration
- 242 - NULL_APN_DISALLOWED, this error code is returned when a PDN is attempted to be brought up with a NULL APN and when a NULL APN is not supported
- 243 - THERMAL_MITIGATION, this error code is returned when the thermal level increases and causes calls to be torn down when normal mode of operation is not allowed
- 244 - SUBS_ID_MISMATCH, this error code is returned when a new call bring up is rejected due to a mismatch between the subs_id in the profile and the subs_id in the ACL policy information
- 245 - DATA_SETTINGS_DISABLED, this error code is returned when the PDN connection to a given APN is disallowed because data is disabled from the device user interface (UI) settings
- 246 - DATA_ROAMING_SETTINGS_DISABLED, this error code is returned when the PDN Connection to a given APN is disallowed because data roaming is disabled from the device UI settings and the UE is roaming
- 247 - APN_FORMAT_INVALID, this error code is returned when the APN specified in the policy does not follow the format specified in the 3GPP Specification
- 248 - DDS_CALL_ABORT, this error code is returned when a DDS switch occurs; the data call is brought down with the DDS abort reason
- 249 - VALIDATION_FAILURE, this error code is returned when a data call is brought down due to an internal validation failure

9.26.2.8 Call Manager defined call end reasons (Type=3)

- 500 - CDMA LOCK, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to device in CDMA locked state
- 501 - INTERCEPT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since it received an intercept order from the base station
- 502 - REORDER, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to receiving a reorder from base station
- 503 - REL SO REJ, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to receiving a release from base station with reason: SO Reject
- 504 - INCOM CALL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since it received an incoming call from base station
- 505 - ALERT STOP, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to RL/FL fade (or) receiving call release from base stations
- 506 - ACTIVATION, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to channel acquisition failures. This indicates that device has failed acquiring all the channels in the PRL
- 507 - MAX ACCESS PROBE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due maximum access probes transmitted
- 508 - CCS NOT SUPPORTED BY BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since concurrent service is not supported by base station
- 509 - NO RESPONSE FROM BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since there is no response received from base station

- 510 - REJECTED BY BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to base station rejecting the call
- 511 - INCOMPATIBLE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since concurrent services requested were not compatible
- 512 - ALREADY IN TC, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since traffic channel is already up for voice calls
- 513 - USER CAL ORIG DURING GPS
- 514 - USER CAL ORIG DURING SMS, this error code is returned when data call is brought down because traffic channel request got rejected since SMS is ongoing
- 515 - NO CDMA SRV, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device does not have CDMA service
- 516 - MC ABORT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since MC aborted the origination/conversation
- 517 - PSIST NG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to persistence test failure
- 518 - UIM NOT PRESENT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to RUIM not present
- 519 - RETRY ORDER, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to receiving a retry order from base station
- 520 - ACCESS BLOCK, this error code is returned when data call is brought down because traffic channel rejected/released due to Access blocked by base station
- 521 - ACCESS BLOCK ALL, this error code is returned when data call is brought down because traffic channel rejected due to Access blocked by the base station for all mobile devices
- 522 - IS707B MAX ACC, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due maximum access probes for IS-707B call
- 523 - THERMAL EMERGENCY, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) to put device in thermal emergency
- 524 - CALL ORIG THROTTLED, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since call origination is throttled by DCTM module
- 535 - USER CALL ORIGINATE DURING VOICE CALL, this error code is returned when data call is brought down because traffic channel got released by CM(Call Manager) in favor of voice call or SMS when concurrent voice and data are not supported
- 1000 - CONFERENCE FAILED
- 1001 - INCOMING REJECTED
- 1002 - NO GATEWAY SERVICE
- 1003 - NO GPRS CONTEXT
- 1004 - ILLEGAL MS, This cause is sent to the MS when the network refuses service to the MS either because an identity of the MS is not acceptable to the network or because the MS does not pass the authentication check, i.e. the SRES received from the MS is different from that generated by the network
- 1005 - ILLEGAL ME, This cause is sent to the MS if the ME used is not acceptable to the network, e.g. blacklisted
- 1006 - GPRS SERVICES AND NON GPRS SERVICES NOT ALLOWED
- 1007 - GPRS SERVICES NOT ALLOWED

- 1008 - MS IDENTITY CANNOT BE DERIVED BY THE NETWORK
- 1009 - IMPLICITLY DETACHED, this error code is sent to the MS either if the network has implicitly detached the MS, e.g. some while after the Mobile reachable timer has expired, or if the GMM context data related to the subscription does not exist in the SGSN e.g. because of a SGSN restart.
- 1010 - PLMN NOT ALLOWED, this error code is sent to the MS if it requests location updating in a PLMN where the MS, by subscription or due to operator determined barring is not allowed to operate
- 1011 - LOCAL AREA NOT ALLOWED
- 1012 - GPRS SERVICES NOT ALLOWED IN THIS PLMN
- 1013 - PDP DUPLICATE
- 1014 - USER EQUIPMENT RADIO ACCESS TECHNOLOGY CHANGE
- 1015 - CONGESTION
- 1016 - NO PDP CONEXT ACTIVATED
- 1017 - ACCESS CLASS DSAC REJECTION
- 1018 - PDP ACTIVATE MAX RETRY FAILED
- 1019 - RAB FAILURE
- 1020 - EPS SERVICE NOT ALLOWED
- 1021 - TRACKING AREA NOT ALLOWED
- 1022 - ROAMING NOT ALLOWED IN THIS TRACKING AREA
- 1023 - NO SUITABLE CELLS IN TRACKING AREA
- 1024 - NOT AUTHORIZED FOR THIS CLOSED SUBSCRIBER GROUP
- 1025 - ESM UNKNOWN EPS BEARER CONTEXT
- 1026 - DRB RELEASED AT RRC
- 1027 - NAS SIG CONN RELEASED
- 1028 - EPS MOBILITY MANAGEMENT DETACHED
- 1029 - EPS MOBILITY MANAGEMENT ATTACH FAILED
- 1030 - EPS MOBILITY MANAGEMENT ATTACH STARTED
- 1031 - LTE NAS SERVICE REQ FAILED
- 1032 - ESM(EPS Session Management) ACTIVE DEDICATED BEARER REACTIVATED BY NW
- 1033 - ESM(EPS Session Management) LOWER LAYER FAILURE
- 1034 - ESM(EPS Session Management) SYNC UP WITH NW
- 1035 - ESM(EPS Session Management) NW ACTIVATED DED BEARER WITH ID OF DEF BEARER
- 1036 - ESM(EPS Session Management) BAD OTA MESSAGE
- 1037 - ESM DS REJECTED THE CALL
- 1038 - ESM(EPS Session Management) CONTEXT TRANSFERRED DUE TO IRAT
- 1039 - DS EXPLICIT DEACT
- 1040 - ESM(EPS Session Management) LOCAL CAUSE NONE
- 1041 - LTE NAS SERVICE REQ FAILED NO THROTTLE

- 1042 - ACL FAILURE, This error code should rarely triggered and reported to the application
- 1043 - LTE NAS SERVICE REQ FAILED DS DISALLOW
- 1044 - EMM(EPS Mobility Management) T3417 EXPIRED
- 1045 - EMM(EPS Mobility Management) T3417 EXT EXPIRED
- 1046 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE TXN - Light Radio Resource Controller Uplink data confirmation failure
- 1047 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE HO
- 1048 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE CONN REL
- 1049 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE RLF
- 1050 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE CTRL NOT CONN
- 1051 - LRRC(LTE Radio Resource Control) CONN EST FAILURE
- 1052 - LRRC(LTE Radio Resource Control) CONN EST FAILURE ABORTED
- 1053 - LRRC(LTE Radio Resource Control) CONN EST FAILURE ACCESS BARRED
- 1054 - LRRC(LTE Radio Resource Control) CONN EST FAILURE CELL RESEL
- 1055 - LRRC(LTE Radio Resource Control) CONN EST FAILURE CONFIG FAILURE
- 1056 - LRRC(LTE Radio Resource Control) CONN EST FAILURE TIMER EXPIRED
- 1057 - LRRC(LTE Radio Resource Control) CONN EST FAILURE LINK FAILURE
- 1058 - LRRC(LTE Radio Resource Control) CONN EST FAILURE NOT CAMPED
- 1059 - LRRC(LTE Radio Resource Control) CONN EST FAILURE SI FAILURE
- 1060 - LRRC(LTE Radio Resource Control) CONN EST FAILURE CONN REJECT
- 1061 - LRRC(LTE Radio Resource Control) CONN REL NORMAL
- 1062 - LRRC(LTE Radio Resource Control) CONN REL RLF
- 1063 - LRRC(LTE Radio Resource Control) CONN REL CRE FAILURE
- 1064 - LRRC(LTE Radio Resource Control) CONN REL OOS DURING CRE
- 1065 - LRRC(LTE Radio Resource Control) CONN REL ABORTED
- 1066 - LRRC(LTE Radio Resource Control) CONN REL SIB READ ERROR
- 1067 - DETACH WITH REATTACH LTE NW DETACH
- 1068 - DETACH WITH OUT REATTACH LTE NW DETACH
- 1069 - ESM(EPS Session Management) PROC TIME OUT
- 1070 - MESSAGE EXCEED MAX L2 LIMIT
- 1500 - CD GEN OR BUSY, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to the reception of a Connection Deny message with a deny code of general or network busy
- 1501 - CD BILL OR AUTH, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to the reception of a Connection Deny message with a deny code of billing failure or authentication failure
- 1502 - CHG HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since there is a change to HDR system due to redirection or PRL not preferred

- 1503 - EXIT HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device exited HDR due to redirection or PRL not preferred
- 1504 - HDR NO SESSION, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device does not have a HDR session
- 1505 - HDR ORIG DURING GPS FIX, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since it is ending an HDR call origination in favor of a GPS fix
- 1506 - HDR CS TIMEOUT, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since connection setup on HDR system timed out
- 1507 - HDR RELEASED BY CM, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) when it wants to release a HDR call so a 1X call can continue
- 1508 - COLLOC ACQ FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) when device failed to acquire co-located HDR for origination
- 1509 - OTASP COMMIT IN PROG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since an OTASP commit is in progress
- 1510 - NO HYBR HDR SRV, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device has no Hybrid HDR service
- 1511 - HDR NO LOCK GRANTED, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since HDR module could not get the RF lock
- 1512 - HOLD OTHER IN PROG, this error code is returned when data call is brought down by CM(Call Manager) because DBM or SMS is in progress
- 1513 - HDR FADE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since HDR module released the call due to fade
- 1514 - HDR ACC FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to HDR system Access Failure
- 2000 - CLIENT END, this error code is returned when client ends the data call
- 2001 - NO SRV, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since device has no service
- 2002 - FADE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device lost the system due to fade
- 2003 - REL NORMAL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to receiving a release from base station with no reason
- 2004 - ACC IN PROG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to Access attempt already in progress
- 2005 - ACC FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to Access Failure
- 2006 - REDIR OR HANDOFF, this error code is returned when data call is brought down because device is in the process of redirecting/handing off to a different target system
- 2500 - OFFLINE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device went offline
- 2501 - EMERGENCY MODE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since device is operating in Emergency mode
- 2502 - PHONE IN USE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device is in use (e.g voice call)

- 2503 - INVALID MODE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since the device's operational mode is different from the mode requested in the traffic channel bring up
- 2504 - INVALID SIM STATE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since the SIM was marked by network as invalid for circuit and/or packet service domain
- 2505 - NO COLLOC HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since there is no collocated HDR
- 2506 - CALL CONTROL REJECTED, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since Call control module rejected the request

9.26.2.9 3GPP specification defined call end reasons (Type=6)

- 8 - OPERATOR DETERMINED BARRING, this reason code is posted by the MME(Mobility Management Entity) to indicate operator has barred the UE
- 25 - LLC SNDCP FAILURE, PDP context deactivation initiated by the MS or by the Network
- 26 - INSUFFICIENT RESOURCES, this reason is posted to indicate that the network cannot provide the requested service due to insufficient resources
- 27 - MISSING OR UNKNOWN APN, the APN was required and not specified or APN could not be resolved. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may endup throttling further requests to the same APN
- 28 - UNKNOWN PDN TYPE, the reason is posted by the network to indicate that the PDN type was not recognized
- 29 - AUTH FAILED, the reason is posted when authentication fails. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may end up throttling further requests to the same APN
- 30 - GGSN REJECT, the reason is posted when the request was rejected by Serving GW or PDN GW. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may end up throttling further requests to the same APN
- 31 - ACTIVATION REJECT, the reason is posted when the request is rejected by the network due to unspecified reasons
- 32 - OPTION NOT SUPPORTED, the reason is posted when UE requested a service not supported by the PLMN
- 33 - OPTION UNSUBSCRIBED, This cause is sent when the MS requests a service option for which it has no subscription
- 34 - OPTION TEMP OOO, service option temporarily out of order, this reason is posted when the network is temporarily out of resources to service the request
- 35 - PTI ALREADY USED, the reason is posted to indicate that PTI (Procedure Transaction Identifier) used in the request is already active via another UE requested procedure
- 36 - REGULAR DEACTIVATION, this reason is posted by the network to initiate a regular release of bearer resources
- 37 - EPS QOS NOT ACCEPTED, this reason is posted by the network to indicate that the QoS requested by the UE could not be accepted
- 38 - NETWORK FAILURE, this reason is posted when an error occurs in the network
- 39 - UMTS REACTIVATION REQ, this reason is posted by the network to request for bearer reactivation. This code may be posted during network congestion

- 40 - FEATURE NOT SUPPORTED, Unsuccessful MBMS context activation requested by the network
- 41 - TFT SEMANTIC ERROR, the reason is posted by the network to indicate semantic error(s) in specifying TFT operation included in the request
- 42 - TFT SYNTAX ERROR, the reason is posted by the network to indicate syntactic error(s) in specifying TFT operation included in the request
- 43 - UNKNOWN PDP CONTEXT, the reason is posted when the bearer identity (or linked bearer identity) in the request is invalid (or inactive)
- 44 - FILTER SEMANTIC ERROR, the reason is posted by the network to indicate semantic error(s) in specifying packet filter(s) associated with a TFT
- 45 - FILTER SYNTAX ERROR, the reason is posted by the network to indicate syntactic error(s) in specifying packet filter(s) associated with a TFT
- 46 - PDP WITHOUT ACTIVE TFT, the reason is posted by the network when UW requested more than one PDP connection without TFT
- 50 - IPV4 ONLY ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 50, PDN type IPv4 only allowed.
- 51 - IPV6 ONLY ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 51, PDN type IPv6 only allowed
- 52 - SINGLE ADDRESS BEARER ONLY, 3GPP equivalent ESM(EPS Session Management) cause code value 52, Single address bearers only allowed. The reason is posted when the network supports single address bearers only, meaning dual IP bearers are not supported
- 53 - ESM INFORMATION NOT RECEIVED, 3GPP equivalent ESM(EPS Session Management) cause code value 53, ESM information not received. The reason is posted by the network to indicate that the PDN connection request was rejected because ESM information was not received
- 54 - PND CONNECTION DOES NOT EXIST, 3GPP equivalent ESM(EPS Session Management) cause code value 54, PDN connection does not exist The reason is posted by the network during handover from a non-3GPP network to indicate that the MME does not have any information regarding the requested PDN connection
- 55 - MULTIPLE CONNECTION TO SAME PDN NOT ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 55, Multiple PDN connections for a given APN not allowed. The reason is posted by the network to indicate that the UE is already connected to the requested APN via another PDN/PDN connection
- 81 - INVALID TRANSACTION ID, the reason is posted by the network to indicate that the PTI used in the request is unassigned or reserved
- 95 - MESSAGE INCORRECT SEMANTIC, the reason is posted by the network to indicate receipt of an invalid message
- 96 - INVALID MANDATORY INFO, the reason is posted by the network to indicate receipt of a message with semantic error in a mandatory information element
- 97 - MESSAGE TYPE UNSUPPORTED, the reason is posted by the network to indicate receipt of a message that is either undefined or defined but not implemented by the equipment sending this ESM cause
- 98 - MSG TYPE NONCOMPATIBLE STATE, the reason is posted by the network to indicate receipt of a message type that cannot be handled in the current network protocol state
- 99 - UNKNOWN INFO ELEMENT, the reason is posted by the network to indicate receipt of a message that includes and information element that is either not defined or defined but not implemented by the equipment sending the ESM cause
- 100 - CONDITIONAL IE ERROR, the reason is posted by the network to indicate receipt of a message that includes a syntactically incorrect information element. This message is ignored by the network.

- 101 - MSG AND PROTOCOL STATE UNCOMPATIBLE, the reason is posted by the network to indicate receipt of a message that cannot be handled in the current network protocol state
- 111 - PROTOCOL ERROR, the reason is posted by the network to indicate a protocol error when no other error applies
- 112 - APN TYPE CONFLICT
- 113 - INVALID PROXY-CALL SESSION CONTROL FUNCTION ADDRESS

9.26.2.10 PPP call end reasons (Type=7)

- 1 - TIMEOUT, this error code is returned when the data call bring up fails in PPP setup due to timeout (For e.g: LCP Conf Ack not received from network)
- 2 - AUTH FAILURE, this error code is returned when the data call bring up fails in PPP setup due to authentication failure
- 3 - OPTION MISMATCH, this error code is returned when the data call bring up fails in PPP setup due option mismatch (e.g: Authentication is required, but not negotiated with network during LCP phase)
- 31 - PAP FAILURE, this error code is returned when the data call bring up fails in PPP setup due to PAP failure
- 32 - CHAP FAILURE, this error code is returned when the data call bring up fails in PPP setup due to CHAP failure
- 33 - CLOSE IN PROGRESS, this error code is returned when the data call bring up fails in PPP setup since PPP is in the process of cleaning the previous PPP session
- -1 - UNKNOWN, this error code is unused

9.26.2.11 EHRPD call end reasons (Type=8)

- 1 - SUBS LIMITED TO V4, this error code is returned when the V6 interface bring up fails because network provided only V4 address for the upcoming PDN
- 2 - SUBS LIMITED TO V6, this error code is returned when the V4 interface bring up fails because network provided only V6 address for the upcoming PDN
- 4 - VSNCP(Vendor Specific Network Control Protocol) TIMEOUT, this error code is returned when the data call bring up fails in VSNCP phase due to VSNCP timeout error
- 5 - VSNCP(Vendor Specific Network Control Protocol) FAILURE, this error code is returned when VSNCP configuration failed during call bring up
- 6 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I GEN ERROR, this error code is returned when the data call bring up fails in VSNCP phase due to general error
- 7 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I UNAUTH APN, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason requested APN is unauthorized
- 8 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN LIMIT EXCEED, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN limit exceeded
- 9 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I NO PDN GW, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason no PDN gateway

- 10 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN GW UNREACH, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN gateway unreachable

- 11 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN GW REJ, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN gateway reject

- 12 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I INSUFF PARAM, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason insufficient parameter

- 13 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I RESOURCE UNAVAIL, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason resource unavailable

- 14 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I ADMIN PROHIBIT, this error code is returned when the data call bring up fails in SNCP phase since network rejected VSNCP config request with reason admin prohibited

- 15 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN ID IN USE, this error code is returned when the data call bring up fails in VSNCP phase because network rejected with reason PDN ID IN USE (or) All existing PDNs are brought down with this end reason because one of the PDN bring up got rejected by network with reason PDN ID IN USE

- 16 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I SUBSCR LIMITATION, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason subscriber limitation

- 17 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN EXISTS FOR THIS APN, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN exists for this APN

9.26.2.12 IPV6 call end reasons (Type=9)

- 1 - PREFIX UNAVAILABLE, this error code is returned when V6 data call is brought down because device failed to get the prefix from network

- 2 - IPV6 ERR HRPD IPV6 DISABLED, this error code is returned when V6 data call bring up is rejected because IPV6 is disabled in 1X/HRPD mode

- 3 - IPV6 DISABLED, this error code is returned when IPv6 data call bring up is rejected because NV1896 (IPV6 enable) is disabled

Copyright: © 2011-2020 Sierra Wireless, Inc. all rights reserved

9.27 qaGobiApiTableCarrierCodes.h File Reference

Carrier Codes table.

9.27.1 Detailed Description

Carrier Codes table.

9.27.2 Carrier Codes (Number - Carrier)

- 0 - no carrier specified
- 1 - Generic
- 2 - Telstra
- 4 - AT&T
- 5 - Verizon
- 11 - Sprint
- 12 - Telefonica
- 101 - Verizon
- 102 - Sprint
- 103 - Alltel
- 104 - Bell Mobility
- 105 - Telus
- 106 - U.S. Cellular
- 107 - Telstra
- 108 - China Unicom
- 109 - Telecom New Zealand
- 110 - SK Telecom
- 111 - Reliance Communications
- 112 - Tata Communications
- 113 - MetroPCS Communications
- 114 - Leap Wireless
- 115 - KDDI
- 116 - Grupo Iusacell
- 117 - China Telecom
- 118 - Open Mobile Handset
- 176 - Rogers
- 177 - NetIndex
- 178 - DNA
- 179 - Big Pond
- 201 - AT&T
- 202 - Vodafone
- 203 - T-Mobile
- 204 - Orange
- 205 - Telefonica

- 206 - Telecom Italia
- 207 - 3
- 208 - O2
- 209 - SFR
- 210 - Swisscom AG
- 211 - China Mobile
- 212 - Telstra
- 213 - Singapore Telecommunications
- 214 - Reliance Telecommunications
- 215 - Bharti Airtel
- 216 - NTT docomo
- 217 - E Mobile
- 218 - Softbank
- 219 - Korea Telecom Freetel
- 220 - SK Telecom
- 221 - Telenor
- 222 - NetCom Norway
- 223 - TeliaSonera
- 224 - América Móvil
- 225 - Brasil Vivo
- 0xFFFFFFFF - Unknown

Copyright: © 2011-2020 Sierra Wireless, Inc. all rights reserved

9.28 qaGobiApiTableCodingScheme.h File Reference

Data Coding Scheme.

9.28.1 Detailed Description

Data Coding Scheme.

9.28.2 S1

9.28.2.1 Use of bits 3..0\n\n

- Language using the GSM 7 bit default alphabet Bits 3..0
  indicate the language:
  0000 German
  0001 English

0010 Italian

0011 French

0100 Spanish

0101 Dutch

0110 Swedish

0111 Danish

1000 Portuguese

1001 Finnish

1010 Norwegian

1011 Greek

1100 Turkish

1101 Hungarian

1110 Polish

1111 Language unspecified

9.28.3 Coding Group Bits 7..4(0001)

9.28.3.1 use of bits 3..0

- 0000 GSM 7 bit default alphabet; message preceded by language indication.
  The first 3 characters of the message are a two-character representation of the language encoded according to ISO 639 [12], followed by a CR character. The CR character is then followed by 90 characters of text.
- 0001 UCS2; message preceded by language indication
  The message starts with a two GSM 7-bit default alphabet character representation of the language encoded according to ISO 639. This is padded to the octet boundary with two bits set to 0 and then followed by 40 characters of UCS2-encoded message.
  An MS not supporting UCS2 coding will present the two character language identifier followed by improperly interpreted user data.

9.28.4 Coding Group Bits 7..4(0010)

9.28.4.1 use of bits 3..0

- 0000 Czech
  0001 Hebrew
  0010 Arabic
  0011 Russian
  0100 Icelandic
  0101..1111 Reserved for other languages using the GSM 7 bit default alphabet, with unspecified handling at the MS

9.28.5 Coding Group Bits 7..4(0011)

9.28.5.1 use of bits 3..0

- 0000..1111 Reserved for other languages using the GSM 7 bit
  default alphabet, with unspecified handling at the MS

9.28.6 Coding Group Bits 7..4(01xx)

9.28.6.1 use of bits 3..0

- General Data Coding indication
  - Bits 5..0 indicate the following:
  - Bit 5, if set to 0, indicates the text is uncompressed
  - Bit 5, if set to 1, indicates the text is compressed using the compression algorithm defined in 3GPP TS 23.042
  - Bit 4, if set to 0, indicates that bits 1 to 0 are reserved and have no message class meaning
  - Bit 4, if set to 1, indicates that bits 1 to 0 have a message class meaning: Bit 1 Bit 0 Message Class:
    0 0 Class 0
    0 1 Class 1 Default meaning: ME-specific.
    1 0 Class 2 (U)SIM specific message.
    1 1 Class 3 Default meaning: TE-specific (see 3GPP TS 27.005)
    Bits 3 and 2 indicate the character set being used, as follows:
    Bit 3 Bit 2 Character set:
    0 0 GSM 7 bit default alphabet
    0 1 8 bit data
    1 0 UCS2 (16 bit) [10]
    1 1 Reserved

9.28.7 Coding Group Bits 7..4(1001)

9.28.7.1 Reserved coding groups

- Message with User Data Header (UDH) structure:
  Bit 1 Bit 0 Message Class:
  0 0 Class 0
  0 1 Class 1 Default meaning: ME-specific.
  1 0 Class 2 (U)SIM specific message.
  1 1 Class 3 Default meaning: TE-specific (see 3GPP TS 27.005 [8])
  Bits 3 and 2 indicate the alphabet being used, as follows:
  Bit 3 Bit 2 Alphabet:
  0 0 GSM 7 bit default alphabet
  0 1 8 bit data
  1 0 USC2 (16 bit) [10]
  1 1 Reserved

9.28.8 Coding Group Bits 7..4(1010..1101)

9.28.8.1 Reserved coding groups

9.28.9 Coding Group Bits 7..4(1110)

9.28.9.1 Defined by the WAP Forum

9.28.10 Coding Group Bits 7..4 (1111)

9.28.10.1 Data coding / message handling

- Bit 3 is reserved, set to 0.
  Bit 2 Message coding:
  0 GSM 7 bit default alphabet
  1 8 bit data
  Bit 1 Bit 0 Message Class:
  0 0 No message class.
  0 1 Class 1 user defined.
  1 0 Class 2 user defined.
  1 1 Class 3
  default meaning: TE specific(3GPP TS 27.005)
  Copyright: © 2012-2020 Sierra Wireless, Inc. all rights reserved

9.29 qaGobiApiTableFwDldErrorCodes.h File Reference

FOTA update agent errors and FDT/SSDP reported errors.

9.29.1 Detailed Description

FOTA update agent errors and FDT/SSDP reported errors.

9.29.2 FOTA update agent errors

9.29.2.1 defined error code (in hexadecimal)

- 01 - Flash driver init failed (invisible)
- 02 - could not open/read workspace
- 03 - could not open FOTO
- 04 - could not open FOTO header
- 05 - bad FOTO magic number
- 06 - bad FOTO format
- 08 - could not allocate memory for bitmap
- 09 - could not read restart bitmap from FOTO
- 0a - could not read FOTO digest

- 0b - FOTO checksum error
- 0d - could not allocate memory for FOTO DGD
- 0f - flash driver SSP_FD_query for size failed
- 10 - flash driver SSP_FD_query for DGD failed
- 11 - could not allocate memory for driver DGD
- 12 - FOTO DGD did not match driver DGD
- 13 - DGD_init() failed
- 14 - reference image checksum error
- 15 - could not initialize decompressor
- 16 - flash read failed during verification
- 18 - assertion failed (debug builds only)
- 19 - invalid FOTO extension header
- 1a - could not read xheader data from FOTO
- 1b - could not allocate memory for xheader
- 1c - could not get state from workspace
- 1d - FOTO image base block ID out of range
- 1e - FOTO read failed during verification
- 1f - could not open backup block/file
- 20 - unsupported FOTO header revision
- 21 - FOTO size invalid: too small
- 22 - unsupported FOTO_flags value
- 23 - illegal FOTO_DIGEST_OFFSET value
- 24 - illegal FOTO_PARTITION value
- 25 - FOTO_BODY too small for expected data
- 26 - could not allocate digest storage
- 27 - could not allocate image regions data
- 29 - badly formed restart bitmap in FOTO
- 2a - illegal/unsupported restart map type
- 2b - invalid I/RDATA section offsets in FOTO
- 2c - new restart map not clean
- 2d - FOTO requires feature unsupported by UA
- 2e - could not allocate memory for RAM block
- 2f - could not read package header
- 30 - bad package format
- 31 - bad component package magic number
- 32 - file has unknown/unsupported status

- 33 - source or target file missing
- 34 - target parent directory missing
- 35 - target file exists
- 36 - target directory exists
- 37 - source file is a directory
- 38 - extant file digest mismatch
- 39 - buffer overflow
- 3a - digest mismatch (internal only)
- 3b - internal pseudo-error => skip to next pkg
- 3c - internal pseudo-error => op is complete
- 3d - internal pseudo-error => nothing to do
- 3e - could not reset decompressor
- 3f - reference or version image too large
- 40 - FOTO ref. img size did not match img store
- 41 - insufficient free space in image store
- 42 - could not allocate memory for image cache
- 43 - ZBM blockmap inconsistency
- 44 - ZBM could not find vBlock in image store
- 80 - version image digest/checksum error
- 81 - bad FCP command
- 82 - premature end of FCP stream
- 83 - error reading fFCP stream from FOTO
- 84 - decompression error
- 85 - error reading flash memory
- 86 - error writing flash memory
- 87 - error erasing flash memory
- 88 - error writing to restart bitmap in FOTO
- 89 - flash block number out of range
- 8a - flash byte offset out of range
- 8b - error unlocking flash memory
- 8c - error reading backup block
- 8d - error writing backup block
- 8e - error erasing backup block
- 8f - error writing to FOTO
- 90 - error writing to workspace
- 91 - could not set workspace state (internal)

- 92 - UA re-enetered after fatal error (internal)
- 93 - could not initialize compressor
- 94 - could not reset compressor
- 95 - compression error
- 96 - ZBM ran out of free blocks in image store
- c0 - management client error
- c1 - DL auth failure
- c2 - DL download timeout
- c3 - Not implemented operation
- c4 - Malformed or bad DL URL
- c5 - DL server unavailable
- c6 - DL server error
- c7 - DL fails due to out of memory

9.29.2.2 defined error code (in hexadecimal)

- 100 - CWE hdr allocation error
- 101 - CWE hdr read error
- 102 - CWE hdr check error
- 103 - package descriptor error
- 104 - CWE full image prog error
- 105 - no reverse package found
- 106 - reverse package applied already
- 107 - package preprocessing error
- 108 - package postprocessing error

9.29.3 FDT/SSDP reported errors

9.29.3.1 Code(in hex) - Description - Reference BUT error Message

- 80 - Invalid image type(in CWE header) - Download Failed(Invalid Image Type)
- 81 - Invalid product type(in CWE header) - Download Failed(Wrong Image for this product)
- 82 - Invalid size mismatch(with size specified in CWE header) - Download Failed(File size inconsistent with file header)
- 83 - Image size out of range(could not fit into flash partition) - Download Failed (Firmware image is out of range)
- 84 - Application image not compatible with boot loader or target - Download Failed (App not compatible with bootloader or target)
- 85 - Boot loader not compatible with target - Download Failed(Bootloader not compatible with target)
- 86 - Package not compatible with target - Download Failed(Package not compatible with target)

- 87 - Wrong Signature in application image - Download Failed(Wrong signature for this application)
- 88 - Flash read error - Download Failed(Read flash error)
- 89 - CRC32 check failed - Download Failed(CRC32 check failed)
- 8A - CRC16 check failed - Download Failed(Failed file checksum)
- 8B - Invalid CWE header(too short or other check failed) - Download Failed(Start download packet too short)
- 8C - Decompression error - Could not decompress the zipped image
- 8F - Unspecified CWE processing error - Download Failed(Unknown error)

Copyright: © 2014-2020 Sierra Wireless, Inc. all rights reserved

9.30 qaGobiApiTableGpsCapabilityCodes.h File Reference

Position Determination Service API GPS Capability Codes.

9.30.1 Detailed Description

Position Determination Service API GPS Capability Codes.

9.30.2 GPS capability (Value - Capability)

- 0 - None
- 1 - Standalone
- 2 - Assisted (including XTRA and implying standalone is also supported)
- 3 - Assisted (without XTRA and implying standalone is also supported)
- 0xFFFFFFFF - Unknown

Copyright: © 2011-2020 Sierra Wireless, Inc. all rights reserved

9.31 qaGobiApiTablePowerModes.h File Reference

Device Management Service API Power Modes table.

9.31.1 Detailed Description

Device Management Service API Power Modes table.

9.31.2 Power Modes (Value - Description)

- 0 - Online (default)
- 1 - Low power (airplane) mode
- 2 - Factory test mode
- 3 - Offline

- 4 - Resetting
- 5 - Power off
- 6 - Persistent low power (airplane) mode
- 7 - Mode - only low power

Valid transitions for Power Modes

- Online to Low Power, Persistent low power, Factory test, Offline or Shut Down
- Low power to online, Persistent low power, Offline, or Shut Down
- Persistent low power to Online, Low power, Offline or Shut down
- Factory test to online
- Offline to Reset

Copyright: © 2011-2020 Sierra Wireless, Inc. all rights reserved

9.32 qaGobiApiTableRadioInterfaces.h File Reference

Network Access Service API Radio Interfaces table.

9.32.1 Detailed Description

Network Access Service API Radio Interfaces table.

9.32.2 Radio interface

9.32.2.1 Technology (Value - Radio Interface Technology)

- 0 - No service
- 1 - CDMA 1xRTT
- 2 - CDMA 1xEV-DO
- 3 - AMPS (Unsupported)
- 4 - GSM
- 5 - UMTS
- 6 - WLAN
- 7 - GPS
- 8 - LTE
- 9 - TD-SCDMA
- 10 - LTE-M1
- 11 - LTE-NB1
- 12 - NR5G

Copyright: © 2011-2020 Sierra Wireless, Inc. all rights reserved

9.33 qaGobiApiTableRegionCodes.h File Reference

Region Codes table.

9.33.1 Detailed Description

Region Codes table.

9.33.2 Region Codes (Code - Region)

- 0 - North America
- 1 - Latin America
- 2 - Europe
- 3 - Asia
- 4 - Australia
- 5 - Global
- 0xFFFFFFFF - Unknown

Copyright: © 2011-2020 Sierra Wireless, Inc. all rights reserved

9.34 qaGobiApiTableServiceOptions.h File Reference

Voice Service Options.

9.34.1 Detailed Description

Voice Service Options.

9.34.2 Service Option codes (Code - Reason)

9.34.2.1 Description

- 0x0001 - Basic variable rate voice service (8 kbps)
- 0x0002 - Mobile station loopback (8 kbps)
- 0x0003 - Enhanced variable rate voice service (8 kbps)
- 0x0004 - Asynchronous data service (9.6 kbps)
- 0x0005 - Group 3 facsimile (9.6 kbps)
- 0x0006 - Short message service (rate set 1)
- 0x0007 - Packet data service: Internet or ISO Protocol stack (9.6 kbps)
- 0x0008 - Packet data service: CDPD Protocol stack (9.6 kbps)
- 0x0009 - Mobile station loopback (13 kbps)
- 0x000A - transparent service

- 0x000B - III nontransparent service
- 0x000C - Asynchronous data service (14.4 or 9.6 kbps)
- 0x000D - Group 3 facsimile (14.4 or 9.6 kbps)
- 0x000E - Short message service (rate set 2)
- 0x000F - Packet data service: Internet or ISO Protocol stack (14.4 kbps)
- 0x0010 - Packet data service: CDPD Protocol stack (14.4 kbps)
- 0x0011 - High-rate voice service (13 kbps)
- 0x0012 - Over-the-air parameter administration (rate set 1)
- 0x0013 - Over-the-air parameter administration (rate set 2)
- 0x0014 - Group 3 analog facsimile (rate set 1)
- 0x0015 - Group 3 analog facsimile (rate set 2)
- 0x0016 - High-speed packet data service: Internet or ISO Protocol stack (RS1 forward, RS1 reverse)
- 0x0017 - High-speed packet data service: Internet or ISO Protocol stack (RS1 forward, RS2 reverse)
- 0x0018 - High-speed packet data service: Internet or ISO Protocol stack (RS2 forward, RS1 reverse)
- 0x0019 - High-speed packet data service: Internet or ISO Protocol stack (RS2 forward, RS2 reverse)
- 0x001A - High-speed packet data service: CDPD Protocol stack (RS1 forward, RS1 reverse)
- 0x001B - High-speed packet data service: CDPD Protocol stack (RS1 forward, RS2 reverse)
- 0x001C - High-speed packet data service: CDPD Protocol stack (RS2 forward, RS1 reverse)
- 0x001D - High-speed packet data service: CDPD Protocol stack (RS2 forward, RS2 reverse)
- 0x001E - RATE_SET_1 Supplemental channel loopback test for rate set 1
- 0x001F - RATE_SET_2 Supplemental channel loopback test for rate set 2
- 0x0020 - Test Data Service Option (TDSO)
- 0x0021 - cdma2000 high-speed packet data service, Internet or ISO Protocol stack
- 0x0022 - cdma2000 high-speed packet data service, CDPD Protocol
- 0x0023 - Location services, rate set 1 (9.6 kbps)
- 0x0024 - Location services, rate set 2 (14.4 kbps)
- 0x0025 - ISDN interworking service (64 kbps)
- 0x0026 - GSM voice
- 0x0027 - GSM circuit data
- 0x0028 - GSM packet data
- 0x0029 - GSM short message service
- 0x0036 - Markov Service Option (MSO)
- 0x0037 - Loopback Service Option (LSO)
- 0x0038 - Selectable mode vocoder
- 0x0039 - 32 kbps circuit video conferencing
- 0x003A - CONFERENCING 64 kbps circuit video conferencing

- 0x003B - HRPD packet data service, which when used in paging over the 1X air interface, a page response is not required
- 0x003C - Link Layer Assisted Robust Header Compression (LLA ROHC) - header removal
- 0x003D - LLA ROHC - Header Compression
- 0x003E - Source-controlled Variable-Rate Multimode Wideband (VMR-WB) speech codec rate set 2
- 0x003F - Source-controlled VMR-WB speech codec rate set 1
- 0x0040 - HRPD auxiliary packet data service instance
- 0x0041 - cdma2000/GPRS interworking
- 0x0042 - ISO_PROTOCOL_SO_66 cdma2000 high-speed packet data service, Internet or ISO Protocol stack
- 0x0043 - HRPD packet data IP service where higher layer protocol is IP or ROHC
- 0x0044 - Enhanced variable rate voice service (EVRC-B)
- 0x0045 - HRPD packet data service, which when used in paging over the 1X air interface, a page response is required
- 0x0046 - Enhanced variable rate voice service (EVRC-WB)
- 0x1004 - Asynchronous data service, Revision 1 (9.6 or 14.4 kbps)
- 0x1005 - Group 3 facsimile, Revision 1 (9.6 or 14.4 kbps)
- 0x1007 - Packet data service: Internet or ISO Protocol stack, Revision 1 (9.6 or 14.4 kbps)
- 0x1008 - Packet data service: CDPD Protocol stack, Revision 1 (9.6 or 14.4 kbps)
- 0x7FF8 - Identifies service reference identifier 0
- 0x7FF9 - Identifies service reference identifier 1
- 0x7FFA - Identifies service reference identifier 2
- 0x7FFB - Identifies service reference identifier 3
- 0x7FFC - Identifies service reference identifier 4
- 0x7FFD - Identifies service reference identifier 5
- 0x7FFE - Identifies service reference identifier 6
- 0x7FFF - Identifies service reference identifier 7

Copyright: © 2012-2020 Sierra Wireless, Inc. all rights reserved

9.35 qaGobiApiTableSupServiceInfoClasses.h File Reference

Voice Supplementary Service Information Classes.

9.35.1 Detailed Description

Voice Supplementary Service Information Classes.

9.35.2 Supplementary Service Information Classes (Value - Service Class)

- 0X00 - CLASS_NONE
- 0X01 - CLASS_VOICE
- 0X02 - CLASS_DATA
- 0X04 - CLASS_FAX
- 0X08 - CLASS_SMS
- 0X10 - CLASS_DATACIRCUITSYNC
- 0X20 - CLASS_DATACIRCUITASYNC
- 0X40 - CLASS_PACKETACCESS
- 0X80 - CLASS_PADACCESS

Copyright: © 2012-2020 Sierra Wireless, Inc. all rights reserved

9.36 qaGobiApiTableSwiAudio.h File Reference

Swi Audio related tables.

9.36.1 Detailed Description

Swi Audio related tables.

9.36.2 ACDB Device (Device ID - description)

- 0 - Vehicle HF
- 1 - Handset
- 2 - TTY
- 3 - USB
- 4 - NA

9.36.3 FDT/SSDP reported errors

- 0 - PCM - Mode: 0-slave, 1-master, 2-Auxiliary PCM; Rate: 0-8k, 1-16k; Format: 0-linear, 1-u-law, 2-A-law; Padding: 0-disable, 1-enable; Bits-frame: 0-8BPF, 1-16BPF, 2-32BPF, 3-64BPF, 4-128BPF, 5-256BPF;
- 1 - I2S - None
- 2 - Analog(with internal codec) - None
- 3 - USB - None

Copyright: © 2013-2020 Sierra Wireless, Inc. all rights reserved

9.37 qaGobiApiTableSwiOMADMSessionStatus.h File Reference

Session Status table.

9.37.1 Detailed Description

Session Status table.

9.37.2 DSDSOMask SO Mask

- 0x0000 - Successful - Session succeeded
- 0x0001 - Break - Session succeeded
- 0x0002 - Large Object Handled - Session succeeded
- 0x0003 - No status - Session succeeded
- 0x0004 - No more commands - Session succeeded
- 0x0005 - User cancel - Session cancelled
- 0x0100-0x01FF - General errors
- 0x0200-0x02FF - Syncml errors
- 0x0300-0x03FF - Authentication errors
- 0x0400-0x04FF - Protocol errors
- 0x0500-0x05FF - Tree errors (DM Only)
- 0x0600-0x06FF - Not applicable
- 0x0700-0x07FF - Trigger errors
- 0x0800-0x08FF - FUMO errors
- 0x0900-0x09FF - Communication errors
- 0x0A00-0x0AFF - Parsing errors
- 0x0B00-0x0CFF - Not applicable
- 0x7F00-0x7F12 - Insignia errors
- 0x7F13 - Illegal text - Text received contains illegal characters
- 0x7F14 - Download failure - Failed to download FOTA image
- 0x7F15 - Empty session - Session ran successfully, but no information was updated
- 0x7F16 - Factory reset successful - Factory reset succeeded
- 0x7F17 - Factory reset fail - Factory reset failed

Copyright: © 2017-2020 Sierra Wireless, Inc. all rights reserved

9.38 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference

Update Complete Status table.

9.38.1 Detailed Description

Update Complete Status table.

9.38.2 OMA DM Update Complete Status (Update Complete Status - Meaning - Usage)

- 200 - Successful - The request has succeeded
- 250-299 - Successful(vendor specified) - successful operation with vendor specified ResultCode
- 400 - Management Client Error - Management Client error - based on User or Device behavior
- 401 - User Cancelled - User chose not to accept the operation when prompted
- 402 - Corrupted Firmware Update Package - Corrupted firmware update package did not store correctly. Detected for example, by mismatch CRCs between actual and expected
- 403 - Firmware UpdatePackage(Device Mismatch) - Wrong firmware update package delivered to device based on current device characteristics
- 404 - Failed Firmware Update Package Validation - Failure to positively validate digital signature of firmware update package
- 405 - Firmware Update Package Not acceptable - firmware update package is not acceptable
- 406 - Alternate Download Authentication Failure - authentication required but authentication failure was encountered when downloading firmware update package
- 407 - Alternate Download Request Timeout - client has encountered a timeout when downloading firmware update package
- 408 - Not Implemented - the device does not support the requested operation
- 409 - Undefined Error - indicates failure not defined by any other error code
- 410 - Firmware Update Failed - firmware update operation failed in device
- 411 - Malformed or Bad URL - the URL provided for alternate download is bad
- 412 - Alternate Download Server Unavailable - the alternate download server is unavailable or does not respond
- 450 - Client Error (OMADM General) - Vendor defined client error
- 451 - Client Error (OMADM SyncML) - Vendor defined client error
- 452 - Client Error (OMADM Auth) - Vendor defined client error
- 453 - Client Error (OMADM Protocol)- Vendor defined client error
- 454 - Client Error (OMADM Tree) - Vendor defined client error
- 455 - Client Error (OMADM DStore) - Vendor defined client error
- 456 - Client Error (OMADM Trigger) - Vendor defined client error
- 457 - Client Error (OMADM Fumo) - Vendor defined client error
- 458 - Client Error (OMADM Comms) - Vendor defined client error
- 459 - Client Error (OMADM Parse) - Vendor defined client error
- 460 - Client Error (OMADM TNDS) - Vendor defined client error
- 461 - Client Error (OMADM SCM) - Vendor defined client error
- 462 - Client Error (OMADM Impl) - Vendor defined client error
- 463-499 - Client Error (Vendor Specified) - client error encountered for operation with vendor specified result code
- 500 - Alternate Download Server Error - Alternate download server error encountered

- 501 - Download fails due to device out of memory - The download fails due to insufficient memory in the device to save the firmware update package
- 502 - Firmware update fails due to device out of memory - The update fails because there isn't sufficient memory to update the device
- 503 - Download fails due to network issues - The download fails due to network/transport level errors
- 550-599 - Alternate Download Server Error (vendor specified)- Alternate download server error encountered for operation with vendor specified result code

Copyright: © 2013-2020 Sierra Wireless, Inc. all rights reserved

9.39 qaGobiApiTableVoiceCallEndReasons.h File Reference

Voice Service Call and supplementary services end reasons.

9.39.1 Detailed Description

Voice Service Call and supplementary services end reasons.

9.39.2 Voice Call and supplementary services end reason codes (Code - Reason)

9.39.2.1 General

- 0 - Phone is offline
- 20 - Phone is CDMA locked until a power cycle; CDMA only
- 21 - Phone has no service, this is for backward compatibility
- 22 - Call has ended abnormally; CDMA only
- 23 - Received intercept from the base station; originating only; CDMA only
- 24 - Received reorder from the base station; originating only; CDMA only
- 25 - Received release from the base station; no reason was given
- 26 - Received release from the base station; SO reject; CDMA only
- 27 - Received incoming call from the base station
- 28 - Received alert stop from the base station; incoming only; CDMA only
- 29 - Client ended the call
- 30 - Received end activation; OTASP call only; CDMA only
- 31 - MC aborted the origination/conversation; CDMA only
- 32 - Maximum access probes were transmitted; CDMA only
- 33 - Persistence test failure; FEATURE_JCDMA only; CDMA only
- 34 - R-UIM is not present
- 35 - Access attempt is already in progress
- 36 - Access failure for a reason other than the above
- 37 - Received retry order; originating only; IS 2000; CDMA only

- 38 - BYBS Concurrent service is not supported by the base station
- 39 - No response was received from the base station
- 40 - Call was rejected by the base station; CDMA only
- 41 - Concurrent services requested were not compatible; CDMA only
- 42 - Access is blocked by the base station; CDMA only
- 43 - Corresponds to CM_CALL_ORIG_ ERR_ALREADY_IN_TC
- 44 - Call is ended because an emergency call is flashed over this call; CDMA only
- 45 - Used if CM is ending a GPS call in preference of a user call
- 46 - Used if CM is ending an SMS call in preference of a user call
- 47 - Used if CM is ending a data call in preference of an emergency call
- 48 - Call was rejected because of a redirection or handoff
- 49 - Access is blocked by the base station for all mobiles; KDDI-specific; CDMA only
- 50 - To support OTASP SPC Error indication
- 51 - Maximum access probes for an IS-707B call; CDMA only
- 52 - Base station reject order
- 53 - Base station retry order
- 54 - Timer T42 is expired
- 55 - Timer T40 is expired
- 56 - Service initialization failure - Traffic Channel Initialization
- 57 - Timer T50m is expired - Traffic Channel Initialization
- 58 - Timer T51m is expired - Traffic Channel Initialization
- 59 - Acknowledgement timeout due to 12 retransmissions
- 60 - Bad forward link or timer T5M is expired
- 61 - Transceiver Resource Manager request failed
- 62 - Timer T41 is expired
- 100 - WCDMA/GSM/TDS only; call end LL cause, Received a reason for ending the call from the lower layer
- 101 - WCDMA/GSM only; Call origination request failed
- 102 - WCDMA/GSM only; client rejected an incoming call
- 103 - WCDMA/GSM only; client rejected a setup indication
- 104 - WCDMA/GSM only; network ended the call
- 105 - WCDMA/GSM only
- 106 - GWM/WCDMA only; phone has no service
- 107 - 1X only; phone has no service
- 108 - Full service is unavailable
- 109 - Indicates resources are not available to handle a new MO/MT PS call

9.39.2.2 service Errors

- 110 - Unknown subscriber
- 111 - Illegal subscriber
- 112 - Bearer service not provisioned
- 113 - Tele service not provisioned
- 114 - Illegal equipment
- 115 - Call barred
- 116 - Illegal ss operation
- 117 - Ss error status
- 118 - Ss not available
- 119 - Ss subscription violation
- 120 - Ss incompatibility
- 121 - Facility not supported
- 122 - Absent subscriber
- 123 - Short term denial
- 124 - Long term denial
- 125 - System failure
- 126 - Data missing
- 127 - Unexpected data value
- 128 - Pwd registration failure
- 129 - Negative pwd check
- 130 - Num of pwd attempts violation
- 131 - Position method failure
- 132 - Unknown alphabet
- 133 - Ussd busy
- 134 - Rejected by user
- 135 - Rejected by network
- 136 - Deflection to served subscriber
- 137 - Special service code
- 138 - Invalid deflected to number
- 139 - Mpty participants exceeded
- 140 - Resources not available

9.39.2.3 control cause values

- 141 - Unassigned number
- 142 - No route to destination
- 143 - Channel unacceptable
- 144 - Operator determined barring
- 145 - Normal call clearing
- 146 - User busy sEE [s3, aNNEX h]
- 147 - No user responding sEE [s3, aNNEX h]
- 148 - User alerting no answer
- 149 - Call rejected sEE [s3, aNNEX h]
- 150 - Number changed sEE [s3, aNNEX h]
- 151 - Preemption sEE [s3, aNNEX h]
- 152 - Destination out of order
- 153 - Invalid number format
- 154 - Facility rejected
- 155 - Resp to status enquiry
- 156 - Normal unspecified
- 157 - No circuit or channel available
- 158 - Network out of order
- 159 - Temporary failure
- 160 - Switching equipment congestion
- 161 - Access information discarded
- 162 - Requested circuit or channel not available
- 163 - Resources unavailable or unspecified
- 164 - Qos unavailable
- 165 - Requested facility not subscribed
- 166 - Incoming calls barred within cug
- 167 - Bearer capability not auth
- 168 - Bearer capability unavailable
- 169 - Service option not available
- 170 - Acm limit exceeded
- 171 - Bearer service not implemented
- 172 - Requested facility not implemented
- 173 - Only digital information bearer available
- 174 - Service or option not implemented

- 175 - Invalid transaction identifier
- 176 - USER NOT MEMBER OF CUG
- 177 - Incompatible destination
- 178 - Invalid transit nw selection
- 179 - Semantically incorrect message
- 180 - Invalid mandatory information
- 181 - Message type non implemented
- 182 - Message type not compatible with protocol statE
- 183 - Information element non existent
- 184 - Conditonal ie error
- 185 - Message not compatible with protocol state
- 186 - Recovery on timer expired
- 187 - Protocol error unspecified
- 188 - Interworking unspecified
- 189 - Outgoing calls barred within cug
- 190 - No cug selection
- 191 - Unknown cug index
- 192 - Cug index incompatible
- 193 - Cug call failure unspecified
- 194 - Clir not subscribed
- 195 - Ccbs possible sEE
- 196 - Ccbs not possible

9.39.2.4 reject causes

- 197 - Imsi unknown in hlr
- 198 - Illegal ms
- 199 - Imsi unknown in vlr
- 200 - Imei not accepted
- 201 - Illegal me sEE
- 202 - Plmn not allowed
- 203 - Location area not allowed
- 204 - Roaming not allowed in this location area
- 205 - No suitable cells in location area
- 206 - Network failure sEE
- 207 - mac failure sEE
- 208 - Synch failure

- 209 - Network congestion
- 210 - GSM authentication unacceptable
- 211 - Service not subscribed
- 212 - Service temporarily out of order
- 213 - Call cannot be identified
- 214 - Incorrect semantics in message
- 215 - Mandatory information invalid
- 216 - Call failed due to other access stratum failures
- 217 - SIM is invalid
- 218 - Invalid call state
- 219 - Access class is blocked
- 220 - No resources are in the protocol stack to allow the call
- 221 - Invalid user data was received

9.39.2.5 reject causes

- 222 - Timer T3230 is expired
- 223 - No cell is available
- 224 - Abort message was received
- 225 - Radio link was lost due to other lower layer causes

9.39.2.6 reject causes

- 226 - Timer T303 is expired
- 227 - CNM MM release is pending

9.39.2.7 stratum reject causes

- 228 - Access stratum RR release indication
- 229 - Access stratum random access failure
- 230 - RRC_REL_IND Access stratum RRC release indication
- 231 - Access stratum close session indication
- 232 - Access stratum open session failure
- 233 - Access stratum low level failure
- 234 - Access stratum low level failure redial is not allowed
- 235 - Access stratum low level immediate retry
- 236 - Access stratum abort radio is unavailable

9.39.2.8 reject causes

- 237 - Service option is not supported

9.39.2.9 IP end reasons

- 300 - Received SIP 400 bad request;waiting for INVITE response
- 301 - Received SIP 400 bad request;waiting for INVITE response
- 302 - Received SIP 404 not found; call failed; called party does not exist
- 303 - Received SIP 415 unsupported media type; call failed; called party does not support media
- 304 - Received SIP 480 temporarily unavailable; call failed; called party is not in the LTE area
- 305 - No network response; call failed
- 306 - No network response; unable to put call on hold
- 307 - Moved to eHRPD; call failed or dropped; not in the LTE area
- 308 - Upgrade/downgrade rejected (200 OK with the current call SDP)
- 309 - Received 403 call forbidden; waiting for INVITE response
- 310 - Generic timeout; did not receive a response from the server or other end
- 311 - Reported on the MO side for generic internal software errors; user can try again if the call still exists
- 312 - Reported on the MT side if the upgrade timer has been cancelled or cannot complete the request for some reason after notifying the user of a re-invite request
- 313 - Call origination is rejected due to a Service-Specific Access Control (SSAC) barring
- 314 - Phone was put in thermal emergency
- 315 - 1XCSFB call ended because of a soft failure
- 316 - 1XCSFB call ended because of a hard failure

Copyright: © 2012-2020 Sierra Wireless, Inc. all rights reserved

9.40 qmerrno.h File Reference

Enumerations

- enum eQCWWANError {
  eQCWWAN_ERR_ENUM_BEGIN = -1,
  eQCWWAN_ERR_NONE,
  eQCWWAN_ERR_GENERAL,
  eQCWWAN_ERR_INTERNAL,
  eQCWWAN_ERR_MEMORY,
  eQCWWAN_ERR_INVALID_ARG,
  eQCWWAN_ERR_BUFFER_SZ,
  eQCWWAN_ERR_NO_DEVICE,
  eQCWWAN_ERR_INVALID_DEVID,
  eQCWWAN_ERR_NO_CONNECTION,
  eQCWWAN_ERR_QMI_IFACE,
  eQCWWAN_ERR_QMI_CONNECT,
  eQCWWAN_ERR_QMI_REQ_SCH,
  eQCWWAN_ERR_QMI_REQ,
  eQCWWAN_ERR_QMI_RSP,
  eQCWWAN_ERR_QMI_REQ_TO,
  eQCWWAN_ERR_QMI_RSP_TO,
  eQCWWAN_ERR_MALFORMED_QMI_RSP,
  eQCWWAN_ERR_INVALID_QMI_RSP,
  eQCWWAN_ERR_INVALID_FILE,
  eQCWWAN_ERR_FILE_OPEN,
  eQCWWAN_ERR_FILE_COPY,
  eQCWWAN_ERR_OFFLINE = 27,
  eQCWWAN_ERR_RESET,
  eQCWWAN_ERR_NO_SIGNAL,
  eQCWWAN_ERR_MULTIPLE_DEVICES,
  eQCWWAN_ERR_DRIVER,
  eQCWWAN_ERR_NO_CANCELABLE_OP,
  eQCWWAN_ERR_CANCEL_OP,
  eQCWWAN_ERR_API_MUTEX_TIMEOUT,
  eQCWWAN_ERR_PDU_GENERATION,
  eQCWWAN_ERR_INVALID_XID,
  eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED,
  eQCWWAN_ERR_ENUM_END,
  eQCWWAN_ERR_QMI_OFFSET = 1000,
  eQCWWAN_ERR_QMI_MALFORMED_MSG = 1001,
  eQCWWAN_ERR_QMI_NO_MEMORY,
  eQCWWAN_ERR_QMI_INTERNAL,
  eQCWWAN_ERR_QMI_ABORTED,
  eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED,
  eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION,
  eQCWWAN_ERR_QMI_INVALID_CLIENT_ID,
  eQCWWAN_ERR_QMI_NO_THRESHOLDS,
  eQCWWAN_ERR_QMI_INVALID_HANDLE,
  eQCWWAN_ERR_QMI_INVALID_PROFILE,
  eQCWWAN_ERR_QMI_INVALID_PINID,
  eQCWWAN_ERR_QMI_INCORRECT_PIN,
  eQCWWAN_ERR_QMI_NO_NETWORK_FOUND,
  eQCWWAN_ERR_QMI_CALL_FAILED,
  eQCWWAN_ERR_QMI_OUT_OF_CALL,
  eQCWWAN_ERR_QMI_NOT_PROVISIONED,
  eQCWWAN_ERR_QMI_MISSING_ARG,
  eQCWWAN_ERR_QMI_ARG_TOO_LONG = 1019,
  eQCWWAN_ERR_QMI_INVALID_TX_ID = 1022,
  eQCWWAN_ERR_QMI_DEVICE_IN_USE,
  eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED,
  eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED,
  eQCWWAN_ERR_QMI_NO_EFFECT,
  eQCWWAN_ERR_QMI_NO_FREE_PROFILE,
  eQCWWAN_ERR_QMI_INVALID_PDP_TYPE,
  eQCWWAN_ERR_QMI_INVALID_TECH_PREF,
  eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE,

eQCWWAN_ERR_QMI_WIDTH = 0xFFFF }
- enum qm_wds_ds_profile_extended_err_codes {
  eWDS_ERR_PROFILE_REG_RESULT_FAIL = 1,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID,
  eWDS_ERR_PROFILE_REG_RESULT_LIST_END,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID,
  eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY,
  eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY = 1001,
  eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR,
  eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED,
  eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET,
  eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET,
  eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES,
  eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE = 1101,
  eWDS_ERR_PROFILE_REG_END }

9.40.1 Enumeration Type Documentation

9.40.1.1 enum eQCWWANError

QMI Error Code Enumeration

Enumerator

eQCWWAN_ERR_ENUM_BEGIN

eQCWWAN_ERR_NONE 00 - Success

eQCWWAN_ERR_GENERAL 01 - General error

eQCWWAN_ERR_INTERNAL 02 - Internal error

eQCWWAN_ERR_MEMORY 03 - Memory error

eQCWWAN_ERR_INVALID_ARG 04 - Invalid argument

eQCWWAN_ERR_BUFFER_SZ 05 - Buffer too small

eQCWWAN_ERR_NO_DEVICE 06 - Unable to detect WWAN device

eQCWWAN_ERR_INVALID_DEVID 07 - Invalid WWAN device ID

eQCWWAN_ERR_NO_CONNECTION 08 - No connection to WWAN device

eQCWWAN_ERR_QMI_IFACE 09 - Unable to obtain QMI interace

eQCWWAN_ERR_QMI_CONNECT 10 - Unable to connect to QMI interface

eQCWWAN_ERR_QMI_REQ_SCH 11 - Unable to schedule QMI request

eQCWWAN_ERR_QMI_REQ 12 - Error sending QMI request

eQCWWAN_ERR_QMI_RSP 13 - Error receiving QMI response

eQCWWAN_ERR_QMI_REQ_TO 14 - Timeout while sending QMI request

eQCWWAN_ERR_QMI_RSP_TO 15 - Timeout while receiving QMI response

eQCWWAN_ERR_MALFORMED_QMI_RSP 16 - Malformed QMI response received

eQCWWAN_ERR_INVALID_QMI_RSP 17 - Invalid QMI response received

eQCWWAN_ERR_INVALID_FILE 18 - Invalid file path

eQCWWAN_ERR_FILE_OPEN 19 - Unable to open file

eQCWWAN_ERR_FILE_COPY 20 - Unable to copy file

eQCWWAN_ERR_OFFLINE 27 - Unable to set WWAN device offline

eQCWWAN_ERR_RESET 28 - Unable to reset WWAN device

eQCWWAN_ERR_NO_SIGNAL 29 - No available signal

eQCWWAN_ERR_MULTIPLE_DEVICES 30 - Multiple WWAN devices detected

eQCWWAN_ERR_DRIVER 31 - Error interfacing to driver

eQCWWAN_ERR_NO_CANCELABLE_OP 32 - No cancelable operation is pending

eQCWWAN_ERR_CANCEL_OP 33- Error canceling outstanding operation

eQCWWAN_ERR_API_MUTEX_TIMEOUT 34- api mutex lock timeout

eQCWWAN_ERR_PDU_GENERATION 35- PDU generation error

eQCWWAN_ERR_INVALID_XID 36- Invalid transaction id

eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED 37- Unsupported multiple SMS

eQCWWAN_ERR_ENUM_END End of SLQS SDK specific error codes

eQCWWAN_ERR_QMI_OFFSET 1000 - This is not an error code but the offset from which mapped QMI error codes start from

eQCWWAN_ERR_QMI_MALFORMED_MSG 1001 - Malformed or Corrupted QMI msg

eQCWWAN_ERR_QMI_NO_MEMORY 1002 - Device could not allocate memory for QMI Resp

eQCWWAN_ERR_QMI_INTERNAL 1003 - Unexpected error occurred during processing

eQCWWAN_ERR_QMI_ABORTED 1004 - Processing aborted

eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED 1005 - QMI client IDs have been exhausted

eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION 1006 - Unable to abort QMI transaction

eQCWWAN_ERR_QMI_INVALID_CLIENT_ID 1007 - Invalid QMI client ID

eQCWWAN_ERR_QMI_NO_THRESHOLDS 1008 - No thresholds were provided

eQCWWAN_ERR_QMI_INVALID_HANDLE 1009 - Invalid Handle provided in the QMI request

eQCWWAN_ERR_QMI_INVALID_PROFILE 1010 - Profile specified is invalid

eQCWWAN_ERR_QMI_INVALID_PINID 1011 - Invalid PIN ID specified

eQCWWAN_ERR_QMI_INCORRECT_PIN 1012 - Incorrect PIN ID specified

eQCWWAN_ERR_QMI_NO_NETWORK_FOUND 1013 - No network found

eQCWWAN_ERR_QMI_CALL_FAILED 1014 - Call failed

eQCWWAN_ERR_QMI_OUT_OF_CALL 1015 - Device is not in a call

eQCWWAN_ERR_QMI_NOT_PROVISIONED 1016 - Requested information element not provisioned on device

eQCWWAN_ERR_QMI_MISSING_ARG 1017 - Mandatory QMI TLV not provided

eQCWWAN_ERR_QMI_ARG_TOO_LONG 1019 - Arg passed in QMI TLV larger than available storage in device

eQCWWAN_ERR_QMI_INVALID_TX_ID 1022 - Invalid TX ID specified

eQCWWAN_ERR_QMI_DEVICE_IN_USE 1023 - Device currently in a call

eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED 1024 - The selected operation is not supported by the network

eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED 1025 - The selected operation is not supported by the device

eQCWWAN_ERR_QMI_NO_EFFECT 1026 - Requested operation would have no effect

eQCWWAN_ERR_QMI_NO_FREE_PROFILE 1027 - No space for a profile is available

eQCWWAN_ERR_QMI_INVALID_PDP_TYPE 1028 - Invalid PDP type specified

eQCWWAN_ERR_QMI_INVALID_TECH_PREF 1029 - Invalid technology preference specified

eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE 1030 - Invalid profile type specified

eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE 1031 - Invalid service type specified

eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION 1032 - Invalid register action specified

eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION 1033 - Invalid PS attach/detach action specified

eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED 1034 - Authentication of supplied information element failed

eQCWWAN_ERR_QMI_PIN_BLOCKED 1035 - PIN is blocked; an unblock operation needs to be issued

eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED 1036 - PIN is permanently blocked; the UIM is unusable

eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED 1037 - UIM initialization has not completed

eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE 1038 - Max QOS requests are used

eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER 1039 - The Flow filter is incorrect

eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE 1040 - Network unaware of the QOS requested

eQCWWAN_ERR_QMI_INVALID_ID 1041 - Invalid QOS ID

eQCWWAN_ERR_QMI_INVALID_QOS_ID 1041 - Invalid QOS ID

eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED 1042 - The request number is not supported

eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND 1043 - Unable to find the interface

eQCWWAN_ERR_QMI_FLOW_SUSPENDED 1044- Flow suspended

eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT 1045 - Data format is invalid

eQCWWAN_ERR_QMI_GENERAL 1046 - General error

eQCWWAN_ERR_QMI_UNKNOWN 1047 - Unknown error

eQCWWAN_ERR_QMI_INVALID_ARG 1048 - A specified argument is invalid

eQCWWAN_ERR_QMI_INVALID_INDEX 1049 - A specified index is invalid

eQCWWAN_ERR_QMI_NO_ENTRY 1050 - No information element exists at specified memory designation

eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL 1051 - The memory storage specified in the request is full

eQCWWAN_ERR_QMI_DEVICE_NOT_READY 1052 - Device not in a ready state

eQCWWAN_ERR_QMI_NETWORK_NOT_READY 1053 - Network not in a ready state

eQCWWAN_ERR_QMI_CAUSE_CODE 1054 - Error provided in SMS cause code

eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT 1055 - The message could not be sent

eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE 1056 - The message could not be delivered

eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID 1057 - The message ID specified for the message is invalid

eQCWWAN_ERR_QMI_ENCODING 1058 - The message is not encoded properly

eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK 1059 - Maximum number of authentication failures has been reached

eQCWWAN_ERR_QMI_INVALID_TRANSITION 1060 - Operating mode transition from the current mode is invalid

eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE 1061 - The intercace is not muticast

eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE 1062 - Maximum requests in use

eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE 1063 - Invalid muticast handle

eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF 1064 - Invalid IP family preference

eQCWWAN_ERR_QMI_SESSION_INACTIVE 1065 - No tracking session has been started

eQCWWAN_ERR_QMI_SESSION_INVALID 1066 - Current session does not allow this operation

eQCWWAN_ERR_QMI_SESSION_OWNERSHIP 1067 - Current tracking session not started by this QMI control point

eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES 1068 - Device GPS service resources insufficient for request

eQCWWAN_ERR_QMI_DISABLED 1069 - Support for QMI message is disabled

eQCWWAN_ERR_QMI_INVALID_OPERATION 1070 - Invalid operation specified

eQCWWAN_ERR_QMI_INVALID_QMI_CMD 1071 - Invalid/unknown QMI command specified

eQCWWAN_ERR_QMI_TPDU_TYPE 1072 - Message contains TPDU type that cannot be read as raw message

eQCWWAN_ERR_QMI_SMSC_ADDR 1073 - The SMSC address specified is invalid

eQCWWAN_ERR_QMI_INFO_UNAVAILABLE 1074 - Information element is unavailable at this point

eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG 1075 - Segment size too large

eQCWWAN_ERR_QMI_SEGMENT_ORDER 1076 - Segment order is incorrect

eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED 1077 - Bundling not supported

eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE 1078 - The operation failed partially

eQCWWAN_ERR_QMI_POLICY_MISMATCH 1079 - Policy mismatch

eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND 1080 - SIM file not found

eQCWWAN_ERR_QMI_EXTENDED_INTERNAL 1081 - Extended internal error

eQCWWAN_ERR_QMI_ACCESS_DENIED 1082 - Access to a required entity is not available

eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED 1083 - Selected operating mode is invalid with current hardware setting

eQCWWAN_ERR_QMI_ACK_NOT_SENT 1084 - ACK not sent

eQCWWAN_ERR_QMI_INJECT_TIMEOUT 1084 - Inject a timeout for the request

eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE 1090 - Incompatible state

eQCWWAN_ERR_QMI_FDN_RESTRICT 1091 - FDN Restrict

eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE 1092 - SUPS failure cause

eQCWWAN_ERR_QMI_NO_RADIO 1093 - No Radio

eQCWWAN_ERR_QMI_NOT_SUPPORTED 1094 - Not Supported

eQCWWAN_ERR_QMI_NO_SUBSCRIPTION 1095 - No Subscription

eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED 1096 - Card call control failed

eQCWWAN_ERR_QMI_NETWORK_ABORTED 1097 - Network Aborted

eQCWWAN_ERR_QMI_MSG_BLOCKED 1098 - Open Error

eQCWWAN_ERR_QMI_MAX Error - End of QMI specific defines

eQCWWAN_ERR_SWICM_START **Vendor defines - Connection Manager error codes**

eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED 0xE001 - The API is yet to be implemented

eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED 0xE002 - The service is not supported

eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED 0xE003 - The client is not supported

eQCWWAN_ERR_SWICM_TIMEOUT 0xE004 - API Timeout

eQCWWAN_ERR_SWICM_SOCKET_IN_USE 0xE005 - The communication socket is in use

eQCWWAN_ERR_SWICM_AM_VERS_ERROR 0xE006 - SLQS API and SDK version mismatch

eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS 0xE007 - Failed to kill SDK process

eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS 0xE008 - Call in progress

eQCWWAN_ERR_SWICM_V4DWN_V6DWN 0xE009 - IPV4 and IPV6 is down

eQCWWAN_ERR_SWICM_V4DWN_V6UP 0xE00A - IPV4 is down and IPV6 is up

eQCWWAN_ERR_SWICM_V4UP_V6DWN 0xE00B - IPV4 is up and IPV6 is down

eQCWWAN_ERR_SWICM_V4UP_V6UP 0xE00C - IPV4 and IPV6 is up

eQCWWAN_ERR_SWICM_INVALID_SESSION_ID 0xE00D - Invalid V4 Session ID

eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID 0xE00E - Invalid V4 Session ID

eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID 0xE00F - Invalid V6 Session ID

eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS 0xE010 - No available Session Manager slots for additional data sessions

eQCWWAN_ERR_SWICM_END 0xE011 - End of connection manager specific codes

eQCWWAN_ERR_SWISMS_START **Vendor defines - SMS Error codes**

eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG 0xE101 - SMS message length is long

eQCWWAN_ERR_SWISMS_MSG_CORRUPTED 0xE102 - The SMS message is corrupted (encoding wrong)

eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED 0xE103 - The SMS number is corrupted (incorrect number)

eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND 0xE104 - The SMS bearer data is not available

eQCWWAN_ERR_SWISM_END

eQCWWAN_ERR_SWIIM_START **Vendor defines - Image Management error codes**

eQCWWAN_ERR_SWIIM_INVALID_PATH 0xE801 - Invalid directory path

eQCWWAN_ERR_SWIIM_OPENING_DIR 0xE802 - Unable to open the directory

eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND 0xE803 - No Firmware image present in the path

eQCWWAN_ERR_SWIIM_OPENING_FILE 0xE804 - Unable to open the file

eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE 0xE805 - Firmware image is corrupted

eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED 0xE806 - No Firmware image download needed

eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL 0xE807 - Firmware update failed

eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH 0xE808 - Update success but pri/fw preference mismatch

eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS 0xE809 - Update successful

eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE 0xE80A - Enter Download Mode

eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE 0xE80B - File transfer to modem complete

eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT 0xE80C - Wait for modem to reboot

eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE 0xE80D - Invalid Crash State for Firmware Download

eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE 0xE80E - Same as current active image

eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX 0xE80F - invalid slot index

eQCWWAN_ERR_SWIIM_FW_SECBOOT_IMAGE_NOT_SIGNED 0xE810 - image not signed

eQCWWAN_ERR_SWIIM_FW_SECBOOT_INVALID_CERT_CHAIN 0xE811 - invalid certificate chain

eQCWWAN_ERR_SWIIM_FW_TOO_MANY_FILES 0xE812 - too many files

eQCWWAN_ERR_SWIIM_END

eQCWWAN_ERR_SWIDCS_START **Vendor defines - Device Connectivity error codes**

eQCWWAN_ERR_SWIDCS_IOCTL_ERR 0xE901 - IO Control error

eQCWWAN_ERR_SWIDCS_FILEIO_ERR 0xE902 - file open/read/write error

eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND 0xE903 - The device is not found

eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED 0xE904 - Application is disconnected from SDK

eQCWWAN_ERR_SWIDCS_END

eQCWWAN_ERR_QMI_CAT_START **QMI errors related to CAT**

eQCWWAN_ERR_QMI_EVENT_REG_FAILED 62441 - CAT event registration failed

eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP 62442 - Invalid terminal response

eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD 62443 - Invalid envelope command

eQCWWAN_ERR_QMI_CARD_BUSY_RSP 62444 - Card busy response for envelope command

eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE 62445 - Envelope command failure

eQCWWAN_ERR_QMI_CAT_END

eQCWWAN_ERR_NULL_TLV

eQCWWAN_ERR_QMI_WIDTH 0xFFFF - Not an error, represent the end of QMI errors

9.40.1.2 enum qm_wds_ds_profile_extended_err_codes

WDS DS profile extented error codes

Enumerator

eWDS_ERR_PROFILE_REG_RESULT_FAIL 1 - General Failure

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL 2 - The request contains an invalid profile handle

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP 3 - An invalid operation was requested.

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE 4 - The request contains an invalid technology type

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM 5 - The request contains an invalid profile number

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT 6 - The request contains an invalid profile identifier

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL 7 - The request contains an invalid argument other than profile number and profile identifier received.

eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED 8 - Profile registry has not been initialized yet

eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID 9 - The request contains a parameter with invalid length.

eWDS_ERR_PROFILE_REG_RESULT_LIST_END 10 - End of the profile list was reached while searching for the requested profile.

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID 11 - The request contains an invalid subscrition identifier.

eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY 12 - The request contains an invalid profile family.

eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY 1001 - The request contains an invalid 3GPP profile family.

eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR 1002 - An error was encountered while accessing the 3GPP profiles.

eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED 1003 - The given 3GPP profile doesn't have a valid context.

eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET 1004 - The given 3GPP profile is marked invalid.

eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET 1005 - The given 3GPP profile is marked read-only.

eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES 1006 - Creation of a new 3GPP profile failed because the limit of 16 profiles has already been reached.

eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE 1101 - An invalid profile identifier was received as part of the 3GPP2 profile modification request.

eWDS_ERR_PROFILE_REG_END

9.41 QmiService.h File Reference

Data Structures

- struct QmiService

Macros

- #define QMI_TRANSACTION_POOL_SIZE 10

Typedefs

- typedef void(∗ QMI_INDICATION_CALLBACK)(uint8_t ∗qmiPacket, uint16_t qmiPacketSize, void ∗pIndicationCallbackContext)
- typedef struct QmiService QmiService

Functions

- void QmiService_Initialize (QmiService ∗pThis, uint8_t serviceType, uint8_t clientId, struct QmuxTransport ∗pTransport, QMI_INDICATION_CALLBACK pIndicationCallback, void ∗pIndicationCallbackContext)
- void QmiService_ShutDown (QmiService ∗pThis)
- uint16_t QmiService_GetNextTransactionId (QmiService ∗pThis)
- int QmiService_SendRequest (QmiService ∗pThis, uint16_t transactionId, uint8_t ∗qmiRequest, uint16_t qmiRequestSize, QMI_RESPONSE_CALLBACK pResponseCallback, void ∗pResponseCallbackContext)
- void QmiService_CancelTransaction (QmiService ∗pThis, uint16_t transactionId)
- bool QmiService_HandleQmiPacket (QmiService ∗pThis, uint8_t serviceType, uint8_t clientId, uint8_t ∗qmiPacket, uint16_t qmiPacketSize)

9.41.1 Macro Definition Documentation

9.41.1.1 #define QMI_TRANSACTION_POOL_SIZE 10

Number of concurrent QMI transactions that may be outstanding per QMI service.

9.41.2 Typedef Documentation

9.41.2.1 typedef void(∗ QMI_INDICATION_CALLBACK)(uint8_t ∗qmiPacket, uint16_t qmiPacketSize, void ∗pIndicationCallbackContext)

This funcion prototype represents the callback which is called by the QMI service when a QMI indication is received.

Parameters

in	*qmiPacket*	indication's QMI packet (note: this does not include QMUX header).
in	*qmiPacketSize*	Number of bytes in indication's QMI packet.
in	*pIndication-CallbackContext*	User context supplied when QMI service was initialized.

9.41.2.2 typedef struct QmiService QmiService

This structure abstracts a QMI service. It is responsible issuing transaction IDs. It tracks QMI requests and responses using transaction objects from a pool. It invokes a user callback when QMI indications which match its client ID (or a broadcast ID) are received from the transport layer.

Parameters

pNext	• A link so we can store this object in a linked list by the transport layer.

serviceType	• QMI service type, CTL, NAS, etc.
clientId	• QMI client ID
pTransport	• Pointer to underlying QMUX transport object.
transactionId	• QMI transaction ID. Upper limit 0xffff for regular QMI services, 0xff for CTL service.
transactionId-Lock	• Mutex to make transactionId access thread-safe.
transactionPool	• Pool of transaction objects which may be assigned to QMI requests made by clients of this service.
transactionPool-Lock	• Mutex to make transactionPool access thread-safe
pIndication-Callback	• User callback for QMI indications of this service.
pIndication-CallbackContext	• User context for user callback.
deviceFd	• Store service fd.
pQcqmitransport	• Pointer to QMI service transport object.

9.41.3 Function Documentation

9.41.3.1 void QmiService_CancelTransaction (QmiService * *pThis,* uint16_t *transactionId*)

Cancel an outstanding QMI request.

Parameters

in	*pThis*	The primary object of this call.
in	*transactionId*	Transaction ID of outstanding request.

9.41.3.2 uint16_t QmiService_GetNextTransactionId (QmiService * *pThis*)

Returns next transaction ID for a QMI service.

Parameters

in	*pThis*	The primary object of this call.

Returns

transaction ID

9.41.3.3 bool QmiService_HandleQmiPacket (QmiService * *pThis*, uint8_t *serviceType*, uint8_t *clientId*, uint8_t * *qmiPacket*, uint16_t *qmiPacketSize*)

Process a received QMI packet. This function is called by the QMUX transport layer.

Parameters

in	*pThis*	The primary object of this call.
in	*serviceType*	QMI service type of received packet.
in	*clientId*	QMI Client ID of received packet.
in	*qmiPacket*	Pointer to QMI packet.
in	*qmiPacketSize*	Pointer to size in bytes of QMI packet.

Returns

true if packet is consumed by function, false if it should be offered to another QMI service

9.41.3.4 void QmiService_Initialize (QmiService * *pThis*, uint8_t *serviceType*, uint8_t *clientId*, struct QmuxTransport * *pTransport*, QMI_INDICATION_CALLBACK *pIndicationCallback*, void * *pIndicationCallbackContext*)

Initialize QMI service.

Parameters

in	*pThis*	The primary object of this call.
in	*serviceType*	QMI service e.g. NAS, etc.
in	*clientId*	Value previously obtained by invoking CtlService_GetClientId
in	*pTransport*	Pointer to underlying transport layer which is responsible for sending and receiving QMUX packets
in	*pIndication-Callback*	User function to be be called when QMI indications are received.
in	*pIndication-CallbackContext*	A user context for callback.

9.41.3.5 int QmiService_SendRequest (QmiService * *pThis*, uint16_t *transactionId*, uint8_t * *qmiRequest*, uint16_t *qmiRequestSize*, QMI_RESPONSE_CALLBACK *pResponseCallback*, void * *pResponseCallbackContext*)

Send a QMI request to the device and return immediately.

Parameters

in	*pThis*	The primary object of this call.
in	*transactionId*	transaction ID obtained by previous call to QmiService_GetNextTransactionId.
in	*qmiRequest*	QMI request (this becomes the QMUX SDU).
in	*qmiRequestSize*	Size in bytes of QMI request.
in	*pResponse-Callback*	Callback called when QMI request completes.
in	*pResponse-CallbackContext*	Context for callback.

Returns

0 on success, < 0 on failure.

9.41.3.6 void QmiService_ShutDown (QmiService * *pThis*)

Shut down QMI service. Brings service to an uninitiaized state.

Parameters

in	*pThis*	The primary object of this call.

9.42 QmiSyncObject.h File Reference

Data Structures

- struct QmiSyncObject

Macros

- #define MAX_QMI_PACKET_SIZE (2048 - 6)

Typedefs

- typedef struct QmiSyncObject QmiSyncObject

Functions

- int QmiSyncObject_Initialize (QmiSyncObject *pThis)
- int QmiSyncObject_Destroy (QmiSyncObject *pThis)
- int QmiSyncObject_Lock (QmiSyncObject *pThis)
- int QmiSyncObject_Unlock (QmiSyncObject *pThis)
- int QmiSyncObject_Wait (QmiSyncObject *pThis)
- int QmiSyncObject_TimedWait (QmiSyncObject *pThis, time_t timeoutSeconds)
- int QmiSyncObject_Signal (QmiSyncObject *pThis)
- void QmiSyncObject_ResponseCallback (uint8_t *qmiPacket, uint16_t qmiPacketSize, void *pResponseCallbackContext)

9.42.1 Macro Definition Documentation

9.42.1.1 #define MAX_QMI_PACKET_SIZE (2048 - 6)

Maximum size in bytes of a QMI packet. This does not include QMUX header. It becomes the maximum SDU of a QMUX packet, MAX_QMUX_PACKET_SIZE - sizeof(QMUX_HDR)

9.42.2 Typedef Documentation

9.42.2.1 typedef struct QmiSyncObject QmiSyncObject

This structure is a wrapper around a mutex and condition object and a buffer. It is a convenience object which may be used to create synchronous QMI APIs.

Parameters

mutex	• For thread safety.
condition	• Triggered on completion.
buffer	• QMI request packet is written to this buffer. On request completion, contains the QMI response packet.
bufferSize	• Set to size in bytes of QMI request packet. On request completion, set to size of QMI response packet.

Note

The buffer is intended for QMI packet only i.e. QMI header and QMI payload. This buffer becomes the SDU of a QMUX packet.

9.42.3 Function Documentation

9.42.3.1 int QmiSyncObject_Destroy (QmiSyncObject * *pThis*)

Destroy sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, $<$ 0 on failure.

9.42.3.2 int QmiSyncObject_Initialize (QmiSyncObject * *pThis*)

Initialize sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, $<$ 0 on failure.

9.42.3.3 int QmiSyncObject_Lock (QmiSyncObject * *pThis*)

Lock sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, $<$ 0 on failure.

9.42.3.4 void QmiSyncObject_ResponseCallback (uint8_t * *qmiPacket,* uint16_t *qmiPacketSize,* void * *pResponseCallbackContext*)

This funcion copies QMI response packet to buffer and signals the waiting thread.

Parameters

in	*qmiPacket*	Received QMI packet.
in	*qmiPacketSize*	Received QMI packet size in bytes.
in	*pResponse-CallbackContext*	Pointer to this sync object.

9.42.3.5 int QmiSyncObject_Signal (QmiSyncObject * *pThis*)

Signal sync object. Waiting thread should wake up.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, ETIMEDOUT on timeout, or other negative on failure.

9.42.3.6 int QmiSyncObject_TimedWait (QmiSyncObject * *pThis,* time_t *timeoutSeconds*)

Block, waiting on sync object for a number of seconds.

Parameters

in	*pThis*	The primary object of this call.
in	*timeoutSeconds*	Timeout in seconds before abondoning wait.

Returns

0 on success, $<$ 0 on failure.

9.42.3.7 int QmiSyncObject_Unlock (QmiSyncObject * *pThis*)

Unlock sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, $<$ 0 on failure.

9.42.3.8 int QmiSyncObject_Wait (QmiSyncObject * *pThis*)

Block, waiting on sync object.

Parameters

in	*pThis*	The primary object of this call.

Returns

0 on success, < 0 on failure.

9.43 QmiTransaction.h File Reference

Data Structures

- struct QmiTransaction

Typedefs

- typedef void(* QMI_RESPONSE_CALLBACK)(uint8_t *qmiPacket, uint16_t qmiPacketSize, void *pResponseCallbackContext)
- typedef struct QmiTransaction QmiTransaction

Functions

- void QmiTransaction_Initialize (QmiTransaction *pThis, uint16_t transactionId, uint16_t messageId, QMI_RESPONSE_CALLBACK pResponseCallback, void *pResponseCallbackContext)

9.43.1 Typedef Documentation

9.43.1.1 typedef void(* QMI_RESPONSE_CALLBACK)(uint8_t *qmiPacket, uint16_t qmiPacketSize, void *pResponseCallbackContext)

This funcion prototype represents the callback which is called by the QMI service when a QMI response is received.

Parameters

in	*qmiPacket*	Response's QMI packet (note: this does not include QMUX header).
in	*qmiPacketSize*	Number of bytes in response's QMI packet.
in	*pResponseCallbackContext*	User context supplied when QMI transaction was initialized.

9.43.1.2 typedef struct QmiTransaction QmiTransaction

This structure abstracts a QMI transaction. It is responsible for tracking a QMI request to its response.

Parameters

inUse	• true indicates that this object has been allocated by the QMI service and is un use.

transactionId	• QMI transaction ID. Upper limit 0xffff for regular QMI services, 0xff for CTL service.
messageId	• QMI message ID.
pResponse-Callback	• User callback for QMI response.
pResponse-CallbackContext	• User context for user callback.

9.43.2 Function Documentation

9.43.2.1 void QmiTransaction_Initialize (QmiTransaction * *pThis*, uint16_t *transactionId*, uint16_t *messageId*, QMI_RESPONSE_CALLBACK *pResponseCallback*, void * *pResponseCallbackContext*)

Initialize QMI transaction.

Parameters

in	*pThis*	The primary object of this call.
in	*transactionId*	Transaction ID obtained from a call to QmiService_GetNextTransactionId.
in	*messageId*	QMI message ID.
in	*pResponse-Callback*	User function to be be called when QMI response is received.
in	*pResponse-CallbackContext*	A user context for callback.

9.44 QmuxInternal.h File Reference

Data Structures

- struct QMUX_HDR
- struct QmiTransactionHdr
- struct QmiHdr
- struct ResultTlv
- struct CtlTransactionHdr
- struct GetClientIdReqMsg
- struct GetClientIdRspMsg
- struct ReleaseClientIdReqMsg
- struct ReleaseClientIdRspMsg

Macros

- #define MAX(x, y) (((x) > (y)) ? (x) : (y))
- #define SUCCESS 0
- #define FAILURE (-1)
- #define MAX_QMUX_PACKET_SIZE 2048
- #define QMUX_MSG_FLAG 0x01
- #define IF_TYPE_SIZE 1

- #define QMUX_CTL_FLAG_CTL_POINT 0x00
- #define QMUX_CTL_FLAG_SERVICE 0x80
- #define eInvalidService 255
- #define eCTL_REQ 0
- #define eCTL_RSP 1
- #define eCTL_IND 2
- #define BROADCAST_CLIENT_ID 255
- #define GET_CLIENT_ID 0x0022
- #define RELEASE_CLIENT_ID 0x0023
- #define TLV_TYPE_REQUIRED_PARAMETER 0x01
- #define TLV_TYPE_RESULT_CODE 0x02
- #define QMI_RESULT_SUCCESS 0
- #define QMI_RESULT_FAILURE 1
- #define QMI_ERR_GENERAL 46

Typedefs

- typedef struct QmiTransactionHdr QmiTransactionHdr
- typedef struct QmiHdr QmiHdr
- typedef struct ResultTlv ResultTlv
- typedef struct CtlTransactionHdr CtlTransactionHdr
- typedef struct GetClientIdReqMsg GetClientIdReqMsg
- typedef struct GetClientIdRspMsg GetClientIdRspMsg
- typedef struct ReleaseClientIdReqMsg ReleaseClientIdReqMsg
- typedef struct ReleaseClientIdRspMsg ReleaseClientIdRspMsg

9.44.1 Macro Definition Documentation

9.44.1.1 #define BROADCAST_CLIENT_ID 255

9.44.1.2 #define eCTL_IND 2

9.44.1.3 #define eCTL_REQ 0

9.44.1.4 #define eCTL_RSP 1

9.44.1.5 #define eInvalidService 255

9.44.1.6 #define FAILURE (-1)

9.44.1.7 #define GET_CLIENT_ID 0x0022

9.44.1.8 #define IF_TYPE_SIZE 1

9.44.1.9 #define MAX(*x*, *y*) (((x) > (y)) ? (x) : (y))

9.44.1.10 #define MAX_QMUX_PACKET_SIZE 2048

9.44.1.11 #define QMI_ERR_GENERAL 46

9.44.1.12 #define QMI_RESULT_FAILURE 1

9.44.1.13 #define QMI_RESULT_SUCCESS 0

9.44.1.14 #define QMUX_CTL_FLAG_CTL_POINT 0x00

9.44.1.15 #define QMUX_CTL_FLAG_SERVICE 0x80

9.44.1.16 #define QMUX_MSG_FLAG 0x01

9.44.1.17 #define RELEASE_CLIENT_ID 0x0023

9.44.1.18 #define SUCCESS 0

9.44.1.19 #define TLV_TYPE_REQUIRED_PARAMETER 0x01

9.44.1.20 #define TLV_TYPE_RESULT_CODE 0x02

9.44.2 Typedef Documentation

9.44.2.1 typedef struct CtlTransactionHdr CtlTransactionHdr

9.44.2.2 typedef struct GetClientIdReqMsg GetClientIdReqMsg

9.44.2.3 typedef struct GetClientIdRspMsg GetClientIdRspMsg

9.44.2.4 typedef struct QmiHdr QmiHdr

9.44.2.5 typedef struct QmiTransactionHdr QmiTransactionHdr

9.44.2.6 typedef struct ReleaseClientIdReqMsg ReleaseClientIdReqMsg

9.44.2.7 typedef struct ReleaseClientIdRspMsg ReleaseClientIdRspMsg

9.44.2.8 typedef struct ResultTlv ResultTlv

9.45 QmuxTransport.h File Reference

Data Structures

- struct QcqmiService
- struct QmuxTransport

Macros

- #define QCQMI_NODE_NAME_PREFIX "qcqmi"
- #define MAX_QMI_CLIENT 255
- #define MAX_QMI_CLIENT_ID 255
- #define INVALID_CLIENT_ID 255
- #define INVALID_TRANSACTION_ID 0

Typedefs

- typedef struct QmuxTransport QmuxTransport

Functions

- int QmuxTransport_Initialize (QmuxTransport *pThis, char *devicePath, bool useMbim, void *pInterfaceSpecific)
- void QmuxTransport_ShutDown (QmuxTransport *pThis)
- void QmuxTransport_AttachQmiService (QmuxTransport *pThis, struct QmiService *pQmiService)
- void QmuxTransport_DetachQmiService (QmuxTransport *pThis, struct QmiService *pQmiService)
- int QmuxTransport_SendQmiPacket (QmuxTransport *pThis, uint8_t qmiServiceType, uint8_t qmiClientId, uint16_t qmiTransactionId, uint8_t *qmiPacket, uint16_t qmiPacketSize)

9.45.1 Macro Definition Documentation

9.45.1.1 #define INVALID_CLIENT_ID 255

9.45.1.2 #define INVALID_TRANSACTION_ID 0

9.45.1.3 #define MAX_QMI_CLIENT 255

9.45.1.4 #define MAX_QMI_CLIENT_ID 255

9.45.1.5 #define QCQMI_NODE_NAME_PREFIX "qcqmi"

9.45.2 Typedef Documentation

9.45.2.1 typedef struct QmuxTransport QmuxTransport

This structure abstracts a QMUX transport layer. It is responsible for applying QMUX headers to QMI packets from QMI services and forwarding them to the device. It is responsible for receiving QMUX packets from the device. On the basis of QMUX headers, it distributes the SDUs (QMI packets) to the appropriate QMI service. It maintains a list of connected QMI service objects.

Parameters

useMbim	• Using MBIM interface as opposed to direct QMUX interface for data layer.
pDataLayer	• Pointer to instantiated data layer.
deviceFd	• File descriptor of device.
shutdownFd	• Event descriptor. Used to signal read thread to terminate.
qmiServiceList	• List of connected QMI service objects.
writeLock	• Provides mutual exclusion for write operations.
qmiServiceListLock	• Provides thread safety for the service list.

readThread	• Thread responsible for reading from device.
qcqmi	• QCQMI data.

9.45.3 Function Documentation

9.45.3.1 void QmuxTransport_AttachQmiService (QmuxTransport * *pThis*, struct QmiService * *pQmiService*)

Attaches a QMI service to the transport layer.

Parameters

in	*pThis*	The primary object of this call.
in	*pQmiService*	The QMI service to be attached.

9.45.3.2 void QmuxTransport_DetachQmiService (QmuxTransport * *pThis*, struct QmiService * *pQmiService*)

Detaches a QMI service from the transport layer.

Parameters

in	*pThis*	The primary object of this call.
in	*pQmiService*	The QMI service to be detached.

9.45.3.3 int QmuxTransport_Initialize (QmuxTransport * *pThis*, char * *devicePath*, bool *useMbim*, void * *pInterfaceSpecific*)

Initialize the transport layer.

Parameters

in	*pThis*	The primary object of this call.
in	*devicePath*	Absolute path to device.
in	*useMbim*	Use MBIM interface as opposed to direct interface for data layer.
in	*pInterface-Specific*	A pointer to an object (may be NULL) related to the specific interface used. Currently ignored for useMbim = false. If useMbim = true, it may point to an already initialized MbimTransport object, in which case, this object will use that object rather than try to create its own MbimTransport object.

Returns

0 on success, < 0 on failure.

9.45.3.4 int QmuxTransport_SendQmiPacket (QmuxTransport * *pThis*, uint8_t *qmiServiceType*, uint8_t *qmiClientId*, uint16_t *qmiTransactionId*, uint8_t * *qmiPacket*, uint16_t *qmiPacketSize*)

Send a QMI request to the device and return immediately.

Parameters

in	*pThis*	The primary object of this call.
in	*qmiServiceType*	QMI service type.
in	*qmiClientId*	QMI client ID.
in	*qmiTransactionId*	QMI transaction ID. Can be used for tracking purposes.
in	*qmiPacket*	Pointer to QMI request.
in	*qmiPacketSize*	Size in bytes of QMI packet.

Returns

0 on success, $<$ 0 on failure.

9.45.3.5 void QmuxTransport_ShutDown (QmuxTransport * *pThis*)

Shut down the transport layer.

Parameters

in	*pThis*	The primary object of this call.

9.46 qos.h File Reference

Data Structures

- struct unpack_qos_SLQSQosGetNetworkStatus_t
- struct pack_qos_SLQSQosSwiReadApnExtraParams_t
- struct unpack_qos_SLQSQosSwiReadApnExtraParams_t
- struct pack_qos_SLQSQosSwiReadDataStats_t
- struct unpack_QosFlowStat_t
- struct unpack_qos_SLQSQosSwiReadDataStats_t
- struct unpack_qos_SLQSSetQosNWStatusCallback_ind_t
- struct unpack_qos_SLQSSetQosStatusCallback_ind_t
- struct unpack_qos_SLQSSetQosPriEventCallback_ind_t
- struct pack_qos_SLQSSetQosEventCallback_t
- struct unpack_qos_SLQSSetQosEventCallback_t
- struct unpack_qos_QosFlowInfoState_t
- struct unpack_qos_dataRate_t
- struct unpack_qos_tokenBucket_t
- struct unpack_qos_pktErrRate_t
- struct unpack_qos_swiQosFlow_t
- struct unpack_qos_IPv4Addr_t
- struct unpack_qos_Tos_t
- struct unpack_qos_IPv6Addr_t
- struct unpack_qos_IPv6TrafCls_t
- struct unpack_qos_Port_t
- struct unpack_qos_swiQosFilter_t
- struct unpack_qos_QosFlowInfo_t
- struct unpack_qos_SLQSSetQosEventCallback_ind_t
- struct qos_BindDataPortPeripheralEndPointID_t
- struct qos_BindDataPortMuxID_t
- struct qos_BindDataPortSIODataPort_t
- struct pack_qos_BindDataPort_t
- struct unpack_qos_BindDataPort_t

Macros

- #define LITEQMI_MAX_QOS_FLOW_PER_APN_STATS 10
- #define LITEQMI_MAX_QOS_FILTERS 25
- #define LITEQMI_MAX_QOS_FLOWS 8

Functions

- int pack_qos_SLQSQosGetNetworkStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_qos_SLQSQosGetNetworkStatus (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosGetNetworkStatus_t *pOutput)
- int pack_qos_SLQSQosSwiReadApnExtraParams (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSQosSwiReadApnExtraParams_t reqParam)
- int unpack_qos_SLQSQosSwiReadApnExtraParams (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosSwiReadApnExtraParams_t *pOutput)
- int pack_qos_SLQSQosSwiReadDataStats (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSQosSwiReadDataStats_t reqParam)
- int unpack_qos_SLQSQosSwiReadDataStats (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosSwiReadDataStats_t *pOutput)
- int unpack_qos_SLQSSetQosNWStatusCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosNWStatusCallback_ind_t *pOutput)
- int unpack_qos_SLQSSetQosStatusCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosStatusCallback_ind_t *pOutput)
- int unpack_qos_SLQSSetQosPriEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosPriEventCallback_ind_t *pOutput)
- int pack_qos_SLQSSetQosEventCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSSetQosEventCallback_t reqParam)
- int unpack_qos_SLQSSetQosEventCallback (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosEventCallback_t *pOutput)
- int unpack_qos_SLQSSetQosEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosEventCallback_ind_t *pOutput)
- int pack_qos_BindDataPort (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_BindDataPort_t req)
- int unpack_qos_BindDataPort (uint8_t *pResp, uint16_t respLen, unpack_qos_BindDataPort_t *pOutput)

9.46.1 Macro Definition Documentation

9.46.1.1 #define LITEQMI_MAX_QOS_FILTERS 25

9.46.1.2 #define LITEQMI_MAX_QOS_FLOW_PER_APN_STATS 10

9.46.1.3 #define LITEQMI_MAX_QOS_FLOWS 8

9.46.2 Function Documentation

9.46.2.1 int pack_qos_BindDataPort (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_qos_BindDataPort_t *req*)

Function to pack command to Binds a control point to a data port. This maps to PkQmiQosBindDataPort

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*req*	• See pack_qos_BindDataPort_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds

9.46.2.2 int pack_qos_SLQSQosGetNetworkStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack command to retrieve QoS status of the network. This maps to SLQSQosGetNetworkStatus

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
- Technology Supported: CDMA
- PDN Specific: No

9.46.2.3 int pack_qos_SLQSQosSwiReadApnExtraParams (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_qos_SLQSQosSwiReadApnExtraParams_t *reqParam*)

Function to pack QMI command to query extra APN parameters This maps to SLQSQosSwiReadApnExtraParams

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_qos_SLQSQosSwiReadApnExtraParams_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
- PDN Specific: Yes

9.46.2.4 int pack_qos_SLQSQosSwiReadDataStats (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_qos_SLQSQosSwiReadDataStats_t *reqParam*)

Function to pack QMI command to query APN data statistics This maps to SLQSQosSwiReadDataStats

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes

in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_qos_SLQSQosSwiReadDataStats_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
  - PDN Specific: Yes

9.46.2.5 int pack_qos_SLQSSetQosEventCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_qos_SLQSSetQosEventCallback_t *reqParam*)

Function to pack QMI command to enable QoS event indications This maps to SLQSSetQosEventCallback

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_qos_SLQSSetQosEventCallback_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
  - PDN Specific: Yes

9.46.2.6 int unpack_qos_BindDataPort (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_BindDataPort_t * *pOutput*)

Function to unpack the response to Binds a control point to a data port. This maps to UpkQmiQosBindDataPort

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_BindDataPort_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.46.2.7 int unpack_qos_SLQSQosGetNetworkStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSQosGetNetworkStatus_t * *pOutput*)

Function to unpack the response to get NW QoS status command This maps to SLQSQosGetNetworkStatus

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSQosGetNetworkStatus_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.46.2.8 int unpack_qos_SLQSQosSwiReadApnExtraParams (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSQosSwiReadApnExtraParams_t * *pOutput*)

Function to unpack the response to get NW QoS status command This maps to SLQSQosSwiReadApnExtraParams

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSQosSwiReadApnExtraParams_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.46.2.9 int unpack_qos_SLQSQosSwiReadDataStats (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSQosSwiReadDataStats_t * *pOutput*)

Function to unpack APN data statistics response This maps to SLQSQosSwiReadDataStats

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSQosSwiReadDataStats_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.46.2.10 int unpack_qos_SLQSSetQosEventCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosEventCallback_t * *pOutput*)

Function to unpack enable QoS event indications command's response This maps to SLQSSetQosEventCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosEventCallback_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.46.2.11 int unpack_qos_SLQSSetQosEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosEventCallback_ind_t * *pOutput*)

Function to unpack QoS event indications This maps to SLQSSetQosEventCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosEventCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_NETWORK_STATUS_IND indication to identify this event from QOS service read function

9.46.2.12 int unpack_qos_SLQSSetQosNWStatusCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosNWStatusCallback_ind_t * *pOutput*)

Function to unpack QoS NW status indication. This maps to SLQSSetQosNWStatusCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosNWStatusCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Technology Supported: CDMA
- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_NETWORK_STATUS_IND indication to identify this event from QOS service read function

9.46.2.13 int unpack_qos_SLQSSetQosPriEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosPriEventCallback_ind_t * *pOutput*)

Function to unpack QoS primary flow events. This maps to SLQSSetQosPriEventCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosPriEventCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_PRIMARY_QOS_EVENT_IND indication to identify this event from QOS service read function
- This is only generated when the primary flow is modified by the host

9.46.2.14 int unpack_qos_SLQSSetQosStatusCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosStatusCallback_ind_t * *pOutput*)

Function to unpack QoS status indications. This maps to SLQSSetQosStatusCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosStatusCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_FLOW_STATUS_IND indication to identify this event from QOS service read function

9.47 rms.h File Reference

Data Structures

- struct unpack_rms_GetSMSWake_t
- struct pack_rms_SetSMSWake_t
- struct unpack_rms_SetSMSWake_t

Macros

- #define __LITEQMI_RMS_H__

Functions

- int pack_rms_GetSMSWake (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_rms_GetSMSWake (uint8_t *pResp, uint16_t respLen, unpack_rms_GetSMSWake_t *pOutput)
- int pack_rms_SetSMSWake (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_rms_SetSMSWake_t *reqArg)
- int unpack_rms_SetSMSWake (uint8_t *pResp, uint16_t respLen, unpack_rms_SetSMSWake_t *pOutput)

9.47.1 Macro Definition Documentation

9.47.1.1 #define __LITEQMI_RMS_H__

9.47.2 Function Documentation

9.47.2.1 int pack_rms_GetSMSWake (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get SMS Wake pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.47.2.2 int pack_rms_SetSMSWake (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_rms_SetSMSWake_t * *reqArg*)

Set SMS Wake pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.47.2.3 int unpack_rms_GetSMSWake (uint8_t * *pResp*, uint16_t *respLen*, unpack_rms_GetSMSWake_t * *pOutput*)

Get SMS Wake unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.47.2.4 int unpack_rms_SetSMSWake (uint8_t * *pResp*, uint16_t *respLen*, **unpack_rms_SetSMSWake_t** * *pOutput*)

Set SMS Wake unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.48 sar.h File Reference

Data Structures

- struct unpack_sar_SLQSGetRfSarState_t
- struct pack_sar_SLQSSetRfSarState_t

Typedefs

- typedef unpack_result_t unpack_sar_SLQSSetRfSarState_t

Functions

- int pack_sar_SLQSGetRfSarState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sar_SLQSGetRfSarState (uint8_t *pResp, uint16_t respLen, unpack_sar_SLQSGetRfSarState_t *pOutput)
- int pack_sar_SLQSSetRfSarState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sar_SLQSSetRfSarState_t *reqArg)
- int unpack_sar_SLQSSetRfSarState (uint8_t *pResp, uint16_t respLen, unpack_sar_SLQSSetRfSarState_t *pOutput)

9.48.1 Typedef Documentation

9.48.1.1 typedef unpack_result_t unpack_sar_SLQSSetRfSarState_t

9.48.2 Function Documentation

9.48.2.1 int pack_sar_SLQSGetRfSarState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the specified RF SAR state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.48.2.2 int pack_sar_SLQSSetRfSarState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sar_SLQSSetRfSarState_t * *reqArg*)

Sets the specified RF SAR state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.48.2.3 int unpack_sar_SLQSGetRfSarState (uint8_t * *pResp*, uint16_t *respLen*, unpack_sar_SLQSGetRfSarState_t * *pOutput*)

Gets the specified RF SAR state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.48.2.4 int unpack_sar_SLQSSetRfSarState (uint8_t * *pResp*, uint16_t *respLen*, unpack_sar_SLQSSetRfSarState_t * *pOutput*)

Sets the specified RF SAR state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49 sms.h File Reference

Data Structures

- struct pack_sms_SLQSGetSMS_t
- struct unpack_sms_SLQSGetSMS_t
- struct pack_sms_SLQSGetSMSList_t
- struct qmiSmsMessageList
- struct unpack_sms_SLQSGetSMSList_t
- struct pack_sms_SLQSModifySMSStatus_t
- struct unpack_sms_SLQSModifySMSStatus_t
- struct pack_sms_SLQSDeleteSMS_t
- struct unpack_sms_SLQSDeleteSMS_t
- struct pack_sms_SendSMS_t
- struct unpack_sms_SendSMS_t
- struct pack_sms_SetNewSMSCallback_t
- struct unpack_sms_SetNewSMSCallback_t
- struct sMSMTMessageInfo
- struct newMTMessageTlv
- struct sMSTransferRouteMTMessageInfo
- struct transferRouteMessageTlv
- struct sMSMessageModeInfo
- struct messageModeTlv
- struct sMSEtwsMessageInfo
- struct sMSEtwsMessageTlv
- struct sMSEtwsPlmnInfo
- struct eTWSPLMNInfoTlv

- struct sMSCAddressInfo
- struct sMSCAddressTlv
- struct sMSOnIMSInfo
- struct sMSOnIMSTlv
- struct unpack_sms_SetNewSMSCallback_ind_t
- struct unpack_sms_SLQSWmsMemoryFullCallBack_ind_t
- struct unpack_sms_GetSMSCAddress_t
- struct pack_sms_SetSMSCAddress_t
- struct unpack_sms_SetSMSCAddress_t
- struct pack_sms_SaveSMS_t
- struct unpack_sms_SaveSMS_t
- struct sms_BroadcastConfig
- struct sms_qaQmi3GPPBroadcastCfgInfo
- struct sms_CDMABroadcastConfig
- struct sms_qaQmi3GPP2BroadcastCfgInfo
- struct unpack_sms_SLQSGetSmsBroadcastConfig_t
- struct pack_sms_SLQSGetSmsBroadcastConfig_t
- struct pack_sms_SLQSSetSmsBroadcastConfig_t
- struct unpack_sms_SLQSSetSmsBroadcastConfig_t
- struct pack_sms_SLQSSetSmsBroadcastActivation_t
- struct unpack_sms_SLQSSetSmsBroadcastActivation_t
- struct sms_transLayerInfo
- struct sms_getTransLayerInfo
- struct unpack_sms_SLQSGetTransLayerInfo_t
- struct sms_getTransNWRegInfo
- struct unpack_sms_SLQSGetTransNWRegInfo_t
- struct sms_getIndicationReg
- struct unpack_sms_SLQSGetIndicationRegister_t
- struct sms_setIndicationReg
- struct pack_sms_SLQSSetIndicationRegister_t
- struct unpack_sms_SLQSSetIndicationRegister_t
- struct sms_routeEntry
- struct sms_setRoutesReq
- struct pack_sms_SLQSSmsSetRoutes_t
- struct unpack_sms_SLQSSmsSetRoutes_t
- struct sms_msgProtocolResp
- struct unpack_sms_SLQSSmsGetMessageProtocol_t
- struct sms_maxStorageSizeReq
- struct sms_maxStorageSizeResp
- struct pack_sms_SLQSSmsGetMaxStorageSize_t
- struct unpack_sms_SLQSSmsGetMaxStorageSize_t
- struct sms_messageWaitingInfoContent
- struct sms_getMsgWaitingInfo
- struct unpack_sms_SLQSGetMessageWaiting_t
- struct sms_sendAsyncsmsParams
- struct pack_sms_SLQSSendAsyncSMS_t
- struct unpack_sms_SLQSSendAsyncSMS_t
- struct pack_sms_SLQSSetSmsStorage_t
- struct unpack_sms_SLQSSetSmsStorage_t
- struct unpack_sms_SLQSSwiGetSMSStorage_t
- struct unpack_sms_SLQSTransLayerInfoCallback_ind_t
- struct unpack_sms_SLQSNWRegInfoCallback_ind_t
- struct unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t
- struct unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t

Macros

- #define MAX_SMS_MESSAGE_SIZE 2048
- #define MAX_SMS_LIST_SIZE 255
- #define MAX_MS_TRANSFER_ROUTE_MSG 256
- #define MAX_MSE_TWS_MSG 1254
- #define MAX_MSC_ADDRESS_SIZE 256
- #define MAX_CDMA_ENC_MO_TXT_MSG_SIZE 255
- #define SMSC_TYPE_LEN 0x03
- #define SMS_CONFIG_LEN 0x05
- #define SMS_MAX_SMS_ROUTES 0x0A
- #define SMS_NUM_OF_SET 0xFF

Enumerations

- enum eqmiCbkSetStatus {
  LITEQMI_QMI_CBK_PARAM_RESET = 0,
  LITEQMI_QMI_CBK_PARAM_SET = 1,
  LITEQMI_QMI_CBK_PARAM_NOCHANGE }

Functions

- int pack_sms_SLQSGetSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSMS_t *reqParam)
- int unpack_sms_SLQSGetSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSMS_t *pOutput)
- int pack_sms_SLQSGetSMSList (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSMSList_t *reqParam)
- int unpack_sms_SLQSGetSMSList (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSMSList_t *pOutput)
- int pack_sms_SLQSModifySMSStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSModifySMSStatus_t *reqParam)
- int unpack_sms_SLQSModifySMSStatus (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSModifySMSStatus_t *pOutput)
- int pack_sms_SLQSDeleteSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSDeleteSMS_t *reqParam)
- int unpack_sms_SLQSDeleteSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSDeleteSMS_t *pOutput)
- int pack_sms_SendSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SendSMS_t *reqParam)
- int unpack_sms_SendSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SendSMS_t *pOutput)
- int pack_sms_SetNewSMSCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SetNewSMSCallback_t reqParam)
- int unpack_sms_SetNewSMSCallback (uint8_t *pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_t *Output)
- int unpack_sms_SetNewSMSCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_ind_t *pOutput)
- int unpack_sms_SLQSWmsMemoryFullCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSWmsMemoryFullCallBack_ind_t *pOutput)
- int pack_sms_GetSMSCAddress (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_GetSMSCAddress (uint8_t *pResp, uint16_t respLen, unpack_sms_GetSMSCAddress_t *pOutput)
- int pack_sms_SetSMSCAddress (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SetSMSCAddress_t *reqParam)
- int unpack_sms_SetSMSCAddress (uint8_t *pResp, uint16_t respLen, unpack_sms_SetSMSCAddress_t *pOutput)

- int pack_sms_SaveSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SaveSMS_t *reqParam)
- int unpack_sms_SaveSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SaveSMS_t *pOutput)
- int pack_sms_SLQSGetSmsBroadcastConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSmsBroadcastConfig_t *reqParam)
- int unpack_sms_SLQSGetSmsBroadcastConfig (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSmsBroadcastConfig_t *pOutput)
- int pack_sms_SLQSSetSmsBroadcastConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetSmsBroadcastConfig_t *reqParam)
- int unpack_sms_SLQSSetSmsBroadcastConfig (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSetSmsBroadcastConfig_t *pOutput)
- int pack_sms_SLQSSetSmsBroadcastActivation (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetSmsBroadcastActivation_t *reqParam)
- int unpack_sms_SLQSSetSmsBroadcastActivation (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSetSmsBroadcastActivation_t *pOutput)
- int pack_sms_SLQSGetTransLayerInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSGetTransLayerInfo (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetTransLayerInfo_t *pOutput)
- int pack_sms_SLQSGetTransNWRegInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSGetTransNWRegInfo (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetTransNWRegInfo_t *pOutput)
- int pack_sms_SLQSGetIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSGetIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetIndicationRegister_t *pOutput)
- int pack_sms_SLQSSetIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetIndicationRegister_t *reqParam)
- int unpack_sms_SLQSSetIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSetIndicationRegister_t *pOutput)
- int pack_sms_SLQSSmsSetRoutes (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSmsSetRoutes_t *reqParam)
- int unpack_sms_SLQSSmsSetRoutes (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSmsSetRoutes_t *pOutput)
- int pack_sms_SLQSSmsGetMessageProtocol (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSSmsGetMessageProtocol (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSmsGetMessageProtocol_t *pOutput)
- int pack_sms_SLQSSmsGetMaxStorageSize (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSmsGetMaxStorageSize_t *reqParam)
- int unpack_sms_SLQSSmsGetMaxStorageSize (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSmsGetMaxStorageSize_t *pOutput)
- int pack_sms_SLQSGetMessageWaiting (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSGetMessageWaiting (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetMessageWaiting_t *pOutput)
- int pack_sms_SLQSSendAsyncSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSendAsyncSMS_t *reqParam)
- int unpack_sms_SLQSSendAsyncSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSendAsyncSMS_t *pOutput)
- int pack_sms_SLQSSetSmsStorage (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetSmsStorage_t *reqParam)
- int unpack_sms_SLQSSetSmsStorage (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSetSmsStorage_t *pOutput)
- int pack_sms_SLQSSwiGetSMSStorage (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSSwiGetSMSStorage (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSwiGetSMSStorage_t *pOutput)
- int unpack_sms_SLQSTransLayerInfoCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSTransLayerInfoCallback_ind_t *pOutput)

- int unpack_sms_SLQSNWRegInfoCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSNWRegInfoCallback_ind_t *pOutput)
- int unpack_sms_SLQSWmsMessageWaitingCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t *pOutput)
- int unpack_sms_SLQSWmsAsyncRawSendCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t *pOutput)

9.49.1 Macro Definition Documentation

9.49.1.1 #define MAX_CDMA_ENC_MO_TXT_MSG_SIZE 255

9.49.1.2 #define MAX_MS_TRANSFER_ROUTE_MSG 256

9.49.1.3 #define MAX_MSC_ADDRESS_SIZE 256

9.49.1.4 #define MAX_MSE_TWS_MSG 1254

9.49.1.5 #define MAX_SMS_LIST_SIZE 255

9.49.1.6 #define MAX_SMS_MESSAGE_SIZE 2048

9.49.1.7 #define SMS_CONFIG_LEN 0x05

9.49.1.8 #define SMS_MAX_SMS_ROUTES 0x0A

9.49.1.9 #define SMS_NUM_OF_SET 0xFF

9.49.1.10 #define SMSC_TYPE_LEN 0x03

9.49.2 Enumeration Type Documentation

9.49.2.1 enum eqmiCbkSetStatus

Enumerator

LITEQMI_QMI_CBK_PARAM_RESET

LITEQMI_QMI_CBK_PARAM_SET

LITEQMI_QMI_CBK_PARAM_NOCHANGE

9.49.3 Function Documentation

9.49.3.1 int pack_sms_GetSMSCAddress (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the SMS center address pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.2 int pack_sms_SaveSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SaveSMS_t * *reqParam*)

Saves an SMS message to device memory pack.

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.3 int pack_sms_SendSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SendSMS_t * *reqParam*)

send sms list pack

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.4 int pack_sms_SetNewSMSCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SetNewSMSCallback_t *reqParam*)

set new sms callback pack

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.5 int pack_sms_SetSMSCAddress (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SetSMSCAddress_t * *reqParam*)

Set the SMS center address pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.6 int pack_sms_SLQSDeleteSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSDeleteSMS_t * *reqParam*)

delete sms pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.7 int pack_sms_SLQSGetIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets registration state of different WMS indications pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.8 int pack_sms_SLQSGetMessageWaiting (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the message waiting information pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.9 int pack_sms_SLQSGetSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSGetSMS_t * *reqParam*)

get sms pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.10 int pack_sms_SLQSGetSmsBroadcastConfig (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_sms_SLQSGetSmsBroadcastConfig_t * reqParam)

Provides Information about the SMS BroadcastConfiguration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.11 int pack_sms_SLQSGetSMSList (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_sms_SLQSGetSMSList_t * reqParam)

get sms list pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.12 int pack_sms_SLQSGetTransLayerInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Gets information about the transport layer pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.13 int pack_sms_SLQSGetTransNWRegInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets transport layer network registration info pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.14 int pack_sms_SLQSModifySMSStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSModifySMSStatus_t * *reqParam*)

modify sms status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.15 int pack_sms_SLQSSendAsyncSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSendAsyncSMS_t * *reqParam*)

Sends an SMS message for immediate over-the-air transmission pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.16 int pack_sms_SLQSSetIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetIndicationRegister_t * *reqParam*)

Sets the registration state of different WMS indications pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.17 int pack_sms_SLQSSetSmsBroadcastActivation (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetSmsBroadcastActivation_t * *reqParam*)

Enables or disables the reception of broadcast SMS messages pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.18 int pack_sms_SLQSSetSmsBroadcastConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetSmsBroadcastConfig_t * *reqParam*)

Sets the information about the SMS BroadcastConfiguration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.19 int pack_sms_SLQSSetSmsStorage (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetSmsStorage_t * *reqParam*)

Sets the SMS Storage on the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.20 int pack_sms_SLQSSmsGetMaxStorageSize (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSmsGetMaxStorageSize_t * *reqParam*)

Get the maximum number of messages that can be stored in the specified memory storage. Also it provides the number of slots currently available pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.21 int pack_sms_SLQSSmsGetMessageProtocol (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Gets the message protocol currently in use for the WMS client pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.22 int pack_sms_SLQSSmsSetRoutes (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_sms_SLQSSmsSetRoutes_t * reqParam)

Sets the action performed on SMS message receipt for specified message routes. It also specifies the action performed on SMS receipt of status reports pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.23 int pack_sms_SLQSSwiGetSMSStorage (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Gets the current SMS configuration that is applied to all incoming and outgoing messages pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.24 int unpack_sms_GetSMSCAddress (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_GetSMSCAddress_t * *pOutput*)

Gets the SMS center address unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.25 int unpack_sms_SaveSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SaveSMS_t * *pOutput*)

Saves an SMS message to device memory unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.26 int unpack_sms_SendSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SendSMS_t * *pOutput*)

send sms unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.27 int unpack_sms_SetNewSMSCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SetNewSMSCallback_t * *Output*)

set new sms callback unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*Output*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.28 int unpack_sms_SetNewSMSCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SetNewSMSCallback_ind_t * *pOutput*)

set new sms callback indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.29 int unpack_sms_SetSMSCAddress (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SetSMSCAddress_t * *pOutput*)

Set the SMS center address unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.30 int unpack_sms_SLQSDeleteSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSDeleteSMS_t * *pOutput*)

delete sms unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.31 int unpack_sms_SLQSGetIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetIndicationRegister_t * *pOutput*)

Gets registration state of different WMS indications unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.32 int unpack_sms_SLQSGetMessageWaiting (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetMessageWaiting_t * *pOutput*)

Gets the message waiting information unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.33 int unpack_sms_SLQSGetSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetSMS_t * *pOutput*)

get sms unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.34 int unpack_sms_SLQSGetSmsBroadcastConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetSmsBroadcastConfig_t * *pOutput*)

Provides Information about the SMS BroadcastConfiguration unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.35 int unpack_sms_SLQSGetSMSList (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetSMSList_t * *pOutput*)

get sms list unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.36 int unpack_sms_SLQSGetTransLayerInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetTransLayerInfo_t * *pOutput*)

Gets information about the transport layer unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.37 int unpack_sms_SLQSGetTransNWRegInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetTransNWRegInfo_t * *pOutput*)

Gets transport layer network registration info unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.38 int unpack_sms_SLQSModifySMSStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSModifySMSStatus_t * *pOutput*)

modify sms status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.39 int unpack_sms_SLQSNWRegInfoCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSNWRegInfoCallback_ind_t * *pOutput*)

Unpack indication about change in transport layer info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.40 int unpack_sms_SLQSSendAsyncSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSendAsyncSMS_t * *pOutput*)

Sends an SMS message for immediate over-the-air transmission unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.41 int unpack_sms_SLQSSetIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSetIndicationRegister_t * *pOutput*)

Sets the registration state of different WMS indications unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.42 int unpack_sms_SLQSSetSmsBroadcastActivation (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSetSmsBroadcastActivation_t * *pOutput*)

Enables or disables the reception of broadcast SMS messages unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.43 int unpack_sms_SLQSSetSmsBroadcastConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSetSmsBroadcastConfig_t * *pOutput*)

Sets the information about the SMS BroadcastConfiguration unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.44 int unpack_sms_SLQSSetSmsStorage (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSetSmsStorage_t * *pOutput*)

Sets the SMS Storage on the device unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.45 int unpack_sms_SLQSSmsGetMaxStorageSize (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSmsGetMaxStorageSize_t * *pOutput*)

Get the maximum number of messages that can be stored in the specified memory storage. Also it provides the number of slots currently available unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.46 int unpack_sms_SLQSSmsGetMessageProtocol (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSmsGetMessageProtocol_t * *pOutput*)

Gets the message protocol currently in use for the WMS client unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.47 int unpack_sms_SLQSSmsSetRoutes (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSmsSetRoutes_t * *pOutput*)

Sets the action performed on SMS message receipt for specified message routes. It also specifies the action performed on SMS receipt of status reports unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.48 int unpack_sms_SLQSSwiGetSMSStorage (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSwiGetSMSStorage_t * *pOutput*)

Gets the current SMS configuration that is applied to all incoming and outgoing messages unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.49 int unpack_sms_SLQSTransLayerInfoCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSTransLayerInfoCallback_ind_t * *pOutput*)

Unpack indication about change in transport layer info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.50 int unpack_sms_SLQSWmsAsyncRawSendCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t * *pOutput*)

Unpack indication for sms async raw send

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.51 int unpack_sms_SLQSWmsMemoryFullCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSWmsMemoryFullCallBack_ind_t * *pOutput*)

sms full callback indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.49.3.52 int unpack_sms_SLQSWmsMessageWaitingCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t * *pOutput*)

Unpack indication for message waiting information

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50 swiaudio.h File Reference

Data Structures

- struct pack_swiaudio_SLQSGetM2MAudioProfile_t
- struct unpack_swiaudio_SLQSGetM2MAudioProfile_t
- struct pack_swiaudio_SLQSSetM2MAudioProfile_t
- struct pack_swiaudio_SLQSGetM2MAudioVolume_t
- struct unpack_swiaudio_SLQSGetM2MAudioVolume_t
- struct pack_swiaudio_SLQSSetM2MAudioVolume_t
- struct swiaudio_PCMparams
- struct pack_swiaudio_SLQSSetM2MAudioAVCFG_t
- struct pack_swiaudio_SLQSSetM2MAudioLPBK_t
- struct pack_swiaudio_SLQSGetM2MSpkrGain_t
- struct unpack_swiaudio_SLQSGetM2MSpkrGain_t
- struct pack_swiaudio_SLQSSetM2MSpkrGain_t
- struct pack_swiaudio_SLQSGetM2MAVMute_t
- struct unpack_swiaudio_SLQSGetM2MAVMute_t
- struct pack_swiaudio_SLQSSetM2MAVMute_t

Macros

- #define SWIAUDIO_MAX_LEN_IFACE_TABLE 255

Typedefs

- typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioProfile_t
- typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioVolume_t
- typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioAVCFG_t
- typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioLPBK_t
- typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioNVDef_t
- typedef unpack_result_t unpack_swiaudio_SLQSSetM2MSpkrGain_t
- typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAVMute_t

Functions

- int pack_swiaudio_SLQSGetM2MAudioProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSGetM2MAudioProfile_t *pReqParam)
- int unpack_swiaudio_SLQSGetM2MAudioProfile (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSGetM2MAudioProfile_t *pOutput)
- int pack_swiaudio_SLQSSetM2MAudioProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSSetM2MAudioProfile_t *pReqParam)
- int unpack_swiaudio_SLQSSetM2MAudioProfile (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSSetM2MAudioProfile_t *pOutput)
- int pack_swiaudio_SLQSGetM2MAudioVolume (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSGetM2MAudioVolume_t *pReqParam)
- int unpack_swiaudio_SLQSGetM2MAudioVolume (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSGetM2MAudioVolume_t *pOutput)

- int pack_swiaudio_SLQSSetM2MAudioVolume (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSSetM2MAudioVolume_t *pReqParam)
- int unpack_swiaudio_SLQSSetM2MAudioVolume (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSSetM2MAudioVolume_t *pOutput)
- int pack_swiaudio_SLQSSetM2MAudioAVCFG (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSSetM2MAudioAVCFG_t *pReqParam)
- int unpack_swiaudio_SLQSSetM2MAudioAVCFG (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSSetM2MAudioAVCFG_t *pOutput)
- int pack_swiaudio_SLQSSetM2MAudioLPBK (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSSetM2MAudioLPBK_t *pReqParam)
- int unpack_swiaudio_SLQSSetM2MAudioLPBK (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSSetM2MAudioLPBK_t *pOutput)
- int pack_swiaudio_SLQSSetM2MAudioNVDef (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiaudio_SLQSSetM2MAudioNVDef (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSSetM2MAudioNVDef_t *pOutput)
- int pack_swiaudio_SLQSGetM2MSpkrGain (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSGetM2MSpkrGain_t *pReqParam)
- int unpack_swiaudio_SLQSGetM2MSpkrGain (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSGetM2MSpkrGain_t *pOutput)
- int pack_swiaudio_SLQSSetM2MSpkrGain (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSSetM2MSpkrGain_t *pReqParam)
- int unpack_swiaudio_SLQSSetM2MSpkrGain (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSSetM2MSpkrGain_t *pOutput)
- int pack_swiaudio_SLQSGetM2MAVMute (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSGetM2MAVMute_t *pReqParam)
- int unpack_swiaudio_SLQSGetM2MAVMute (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSGetM2MAVMute_t *pOutput)
- int pack_swiaudio_SLQSSetM2MAVMute (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiaudio_SLQSSetM2MAVMute_t *pReqParam)
- int unpack_swiaudio_SLQSSetM2MAVMute (uint8_t *pResp, uint16_t respLen, unpack_swiaudio_SLQSSetM2MAVMute_t *pOutput)

9.50.1 Macro Definition Documentation

9.50.1.1 #define SWIAUDIO_MAX_LEN_IFACE_TABLE 255

9.50.2 Typedef Documentation

9.50.2.1 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioAVCFG_t

9.50.2.2 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioLPBK_t

9.50.2.3 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioNVDef_t

9.50.2.4 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioProfile_t

9.50.2.5 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioVolume_t

9.50.2.6 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAVMute_t

9.50.2.7 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MSpkrGain_t

9.50.3 Function Documentation

9.50.3.1 int pack_swiaudio_SLQSGetM2MAudioProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSGetM2MAudioProfile_t * *pReqParam*)

Gets the profile content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.2 int pack_swiaudio_SLQSGetM2MAudioVolume (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSGetM2MAudioVolume_t * *pReqParam*)

Gets the Volume content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.3 int pack_swiaudio_SLQSGetM2MAVMute (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSGetM2MAVMute_t * *pReqParam*)

Gets the AV Mute content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.4 int pack_swiaudio_SLQSGetM2MSpkrGain (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSGetM2MSpkrGain_t * *pReqParam*)

Gets the SPKRGAIN content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.5 int pack_swiaudio_SLQSSetM2MAudioAVCFG (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSSetM2MAudioAVCFG_t * *pReqParam*)

Sets the AVCFG content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.6 int pack_swiaudio_SLQSSetM2MAudioLPBK (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSSetM2MAudioLPBK_t * *pReqParam*)

Sets the LPBK content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.7 int pack_swiaudio_SLQSSetM2MAudioNVDef (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Sets the NVDef content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.8 int pack_swiaudio_SLQSSetM2MAudioProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSSetM2MAudioProfile_t * *pReqParam*)

sets an audio profile content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.9 int pack_swiaudio_SLQSSetM2MAudioVolume (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSSetM2MAudioVolume_t * *pReqParam*)

Sets the Volume content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.10 int pack_swiaudio_SLQSSetM2MAVMute (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSSetM2MAVMute_t * *pReqParam*)

Sets the AV Mute content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.11 int pack_swiaudio_SLQSSetM2MSpkrGain (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiaudio_SLQSSetM2MSpkrGain_t * *pReqParam*)

Sets the SPKRGAIN content pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.12 int unpack_swiaudio_SLQSGetM2MAudioProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSGetM2MAudioProfile_t * *pOutput*)

Gets the profile content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.13 int unpack_swiaudio_SLQSGetM2MAudioVolume (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSGetM2MAudioVolume_t * *pOutput*)

Gets the Volume content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.14 int unpack_swiaudio_SLQSGetM2MAVMute (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSGetM2MAVMute_t * *pOutput*)

Gets the AV Mute content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.15 int unpack_swiaudio_SLQSGetM2MSpkrGain (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSGetM2MSpkrGain_t * *pOutput*)

Gets the SPKRGAIN content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.16 int unpack_swiaudio_SLQSSetM2MAudioAVCFG (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSSetM2MAudioAVCFG_t * *pOutput*)

Sets the AVCFG content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.17 int unpack_swiaudio_SLQSSetM2MAudioLPBK (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSSetM2MAudioLPBK_t * *pOutput*)

Sets the LPBK content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.18 int unpack_swiaudio_SLQSSetM2MAudioNVDef (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSSetM2MAudioNVDef_t * *pOutput*)

Sets the NVDef content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.19 int unpack_swiaudio_SLQSSetM2MAudioProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSSetM2MAudioProfile_t * *pOutput*)

sets an audio profile content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.20 int unpack_swiaudio_SLQSSetM2MAudioVolume (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSSetM2MAudioVolume_t * *pOutput*)

Sets the Volume content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.21 int unpack_swiaudio_SLQSSetM2MAVMute (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSSetM2MAVMute_t * *pOutput*)

Sets the AV Mute content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.50.3.22 int unpack_swiaudio_SLQSSetM2MSpkrGain (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiaudio_SLQSSetM2MSpkrGain_t * *pOutput*)

Sets the SPKRGAIN content unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51 swiavms.h File Reference

Data Structures

- struct pack_swiavms_SLQSAVMSStartSession_t
- struct unpack_swiavms_SLQSAVMSStartSession_t
- struct pack_swiavms_SLQSAVMSStopSession_t

- struct unpack_swiavms_SLQSAVMSStopSession_t
- struct PackSwiAvmsSetSettingsPeriodInfo
- struct PackSwiAvmsSetSettingsAPNInfo
- struct PackSwiAvmsSetSettingsConnectionRetryTimers
- struct pack_swiavms_SLQSAVMSSetSettings_t
- struct unpack_swiavms_SLQSAVMSSetSettings_t
- struct pack_swiavms_SLQSAVMSSetSettings_v2_t
- struct unpack_swiavms_SLQSAVMSSetSettings_v2_t
- struct PackSwiAVMSSettingsAPNInfo
- struct PackSwiAVMSSettingsConnectionRetryTimers
- struct PackSwiAVMSSettingsPeriodsInfo
- struct unpack_swiavms_SLQSAVMSGetSettings_t
- struct unpack_swiavms_SLQSAVMSGetSettings_v2_t
- struct pack_swiavms_SLQSAVMSSendSelection_t
- struct unpack_swiavms_SLQSAVMSSendSelection_t
- struct unpack_swiavms_SLQSAvmsSetEventReport_t
- struct UnpackSwiAvmsEventReportBinaryUpdateSessionInfo
- struct UnpackSwiAvmsEventReportConfig
- struct UnpackSwiAvmsEventReportNotification
- struct UnpackSwiAvmsEventReportConnectionRequest
- struct UnpackSwiAvmsEventReportWAMSParamChange
- struct UnpackSwiAvmsEventReportPackageID
- struct UnpackSwiAvmsEventReportRegStatus
- struct UnpackSwiAvmsEventReportDataSessionStatus
- struct UnpackSwiAvmsEventReportSessionType
- struct UnpackSwiAvmsEventReportHTTPStatus
- struct unpack_swiavms_SLQSAVMSSessionGetInfo_t
- struct unpack_swiavms_SLQSAVMSEventReportInd_t

Macros

- #define LITEQMI_MAX_SWIOMA_STR_LEN 255
- #define LITEQMI_MAX_GET_SETTINGS_AVMS_APN_STRING_LENGTH 49
- #define LITEQMI_MAX_GET_SETTINGS_AVMS_UNAME_STRING_LENGTH 29
- #define LITEQMI_MAX_GET_SETTINGS_AVMS_PWD_STRING_LENGTH 29
- #define MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH 49
- #define MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 29
- #define MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 29
- #define MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS 8
- #define MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER 8
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH 128
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH 128
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH 1024
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH 200

Typedefs

- typedef unpack_swiavms_SLQSAVMSStopSession_t unpack_swiavms_SLQSAVMSStopSession_avc2_t

Functions

- int pack_swiavms_SLQSAVMSStartSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSStartSession_t reqParam)
- int unpack_swiavms_SLQSAVMSStartSession (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSStartSession_t *pResponse)
- int pack_swiavms_SLQSAVMSStopSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSStopSession_t reqParam)
- int unpack_swiavms_SLQSAVMSStopSession (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSStopSession_t *pResponse)
- int unpack_swiavms_SLQSAVMSStopSession_avc2 (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSStopSession_t *pResponse)
- int pack_swiavms_SLQSAVMSSetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSSetSettings_t reqParam)
- int unpack_swiavms_SLQSAVMSSetSettings (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSSetSettings_t *pResponse)
- int pack_swiavms_SLQSAVMSSetSettings_v2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSSetSettings_v2_t reqParam)
- int unpack_swiavms_SLQSAVMSSetSettings_v2 (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSSetSettings_v2_t *pResponse)
- int pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSSetSettings_t reqParam)
- int pack_swiavms_SLQSAVMSGetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiavms_SLQSAVMSGetSettings (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSGetSettings_t *pResponse)
- int pack_swiavms_SLQSAVMSGetSettings_v2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiavms_SLQSAVMSGetSettings_v2 (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSGetSettings_v2_t *pResponse)
- int pack_swiavms_SLQSAVMSSendSelection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSSendSelection_t reqParam)
- int unpack_swiavms_SLQSAVMSSendSelection (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSSendSelection_t *pResponse)
- int pack_swiavms_SLQSAvmsSetEventReport (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiavms_SLQSAvmsSetEventReport (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAvmsSetEventReport_t *pResponse)
- int pack_swiavms_SLQSAVMSSessionGetInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiavms_SLQSAVMSSessionGetInfo (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSSessionGetInfo_t *pResponse)
- int unpack_swiavms_SLQSAVMSEventReportInd (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSEventReportInd_t *pResponse)

9.51.1 Macro Definition Documentation

9.51.1.1 #define LITEQMI_MAX_GET_SETTINGS_AVMS_APN_STRING_LENGTH 49

9.51.1.2 #define LITEQMI_MAX_GET_SETTINGS_AVMS_PWD_STRING_LENGTH 29

9.51.1.3 #define LITEQMI_MAX_GET_SETTINGS_AVMS_UNAME_STRING_LENGTH 29

9.51.1.4 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.51.1.5 #define MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER 8

9.51.1.6 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH 1024

9.51.1.7 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH 128

9.51.1.8 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH 128

9.51.1.9 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH 200

9.51.1.10 #define MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH 49

9.51.1.11 #define MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS 8

9.51.1.12 #define MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 29

9.51.1.13 #define MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 29

9.51.2 Typedef Documentation

9.51.2.1 typedef unpack_swiavms_SLQSAVMSStopSession_t unpack_swiavms_SLQSAVMSStopSession_avc2_t

9.51.3 Function Documentation

9.51.3.1 int pack_swiavms_SLQSAVMSGetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack AVMS Get settings command This maps to SLQSAVMSSetSettings

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.2 int pack_swiavms_SLQSAVMSGetSettings_v2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack AVMS Get settings command This maps to SLQSAVMSSetSettings_v2 (For AVC2 service)

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.3 int pack_swiavms_SLQSAVMSSendSelection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSSendSelection_t *reqParam*)

Function to pack AVMS send selection command This maps to SLQSAVMSSendSelection

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSSendSelection_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.4 int pack_swiavms_SLQSAVMSSessionGetInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack AVMS get session info command This maps to SLQSAVMSSessionGetInfo

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.5 int pack_swiavms_SLQSAvmsSetEventReport (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack AVMS set event report command This maps to SLQSAVMSSetEventReport

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.6 int pack_swiavms_SLQSAVMSSetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSSetSettings_t *reqParam*)

Function to pack AVMS Set settings command This maps to SLQSAVMSSetSettings

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.7 int pack_swiavms_SLQSAVMSSetSettings_v2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSSetSettings_v2_t *reqParam*)

Function to pack AVMS Set settings command This maps to SLQSAVMSSetSettings_v2 (For AVC2 service)

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSSetSettings_v2_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.8 int pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSSetSettings_t *reqParam*)

Function to pack AVMS Set settings without Auto Reobot field command This maps to SLQSAVMSSetSettings-IgnoreAutoReboot

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.9 int pack_swiavms_SLQSAVMSStartSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSStartSession_t *reqParam*)

Function to pack Start AVMS session command This maps to SLQSAVMSStartSession

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.10 int pack_swiavms_SLQSAVMSStopSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSStopSession_t *reqParam*)

Function to pack cancel AVMS session command This maps to SLQSAVMSStopSession

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSStopSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.51.3.11 int unpack_swiavms_SLQSAVMSEventReportInd (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSEventReportInd_t * *pResponse*)

Function to unpack AVMS event report Indication command This maps to SLQSAVMSSetEventReport

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSEventReportInd_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.12 int unpack_swiavms_SLQSAVMSGetSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSGetSettings_t * *pResponse*)

Function to unpack AVMS set settings command This maps to SLQSAVMSGetSettings

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.13 int unpack_swiavms_SLQSAVMSGetSettings_v2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSGetSettings_v2_t * *pResponse*)

Function to unpack AVMS set settings command This maps to SLQSAVMSGetSettings_v2 (For AVC2 service)

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.14 int unpack_swiavms_SLQSAVMSSendSelection (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSSendSelection_t * *pResponse*)

Function to unpack AVMS send selection command This maps to SLQSAVMSSendSelection

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSendSelection_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.15 int unpack_swiavms_SLQSAVMSSessionGetInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSSessionGetInfo_t * *pResponse*)

Function to unpack AVMS event report Indication command This maps to SLQSAVMSSetEventReport

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSessionGetInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.16 int unpack_swiavms_SLQSAvmsSetEventReport (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAvmsSetEventReport_t * *pResponse*)

Function to unpack AVMS set event report command This maps to SLQSAVMSSetEventReport

Parameters

in	*pResp*	• Response from modem

in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAvmsSetEventReport_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.17 int unpack_swiavms_SLQSAVMSSetSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSSetSettings_t * *pResponse*)

Function to unpack AVMS Set settings command This maps to SLQSAVMSSetSettings

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.18 int unpack_swiavms_SLQSAVMSSetSettings_v2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSSetSettings_v2_t * *pResponse*)

Function to unpack AVMS Set settings command This maps to SLQSAVMSSetSettings_v2 (For AVC2 service)

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

out	*pResponse*	• See unpack_swiavms_SLQSAVMSSetSettings_v2_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.19 int unpack_swiavms_SLQSAVMSStartSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSStartSession_t * *pResponse*)

Function to unpack Start AVMS session response from modem This maps to SLQSAVMSStartSession

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.20 int unpack_swiavms_SLQSAVMSStopSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSStopSession_t * *pResponse*)

Function to pack stop AVMS session command This maps to SLQSAVMSStopSession

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSStopSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.51.3.21 int unpack_swiavms_SLQSAVMSStopSession_avc2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSStopSession_t * *pResponse*)

Function to pack stop AVMS session command This maps to SLQSAVMSStopSession Note: this targets WP760x as the stop response doesn't contains mandatory session_type(0x01) TLV

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSStopSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.52 SwiDataTypes.h File Reference

SWI data types.

Data Structures

- struct qmTlvResult

Macros

- #define SWI_API
- #define QMI_NO_LTE_FW_SUPPORT 0
- #define QMI_TLV_PLACEHOLDER 0x8F
- #define UNUSEDPARAM(x) (void)x
- #define MEMSET_STATIC_OUTPUT_STRUCT(OUTPUT_PARA, STRUCT_OF_OUTPUT_PARA) memset(O-UTPUT_PARA,0,sizeof(STRUCT_OF_OUTPUT_PARA));\

Typedefs

- typedef unsigned int ULONG
- typedef unsigned long long ULONGLONG
- typedef signed char INT8
- typedef unsigned char BYTE
- typedef char CHAR
- typedef unsigned short WORD
- typedef unsigned short USHORT
- typedef const char * LPCSTR
- typedef int BOOL
- typedef signed short SHORT
- typedef signed int INT32
- typedef float FLOAT
- typedef unsigned short qmuint16
- typedef unsigned long qmulong

9.52.1 Detailed Description

SWI data types.

9.52.2 Macro Definition Documentation

9.52.2.1 #define MEMSET_STATIC_OUTPUT_STRUCT(*OUTPUT_PARA, STRUCT_OF_OUTPUT_PARA*) memset(OUTPUT_PARA,0,sizeof(STRUCT_OF_OUTPUT_PARA));\

Macro used to memset unpack output paramter. To prevent output parameter is not inititalized.

9.52.2.2 #define QMI_NO_LTE_FW_SUPPORT 0

9.52.2.3 #define QMI_TLV_PLACEHOLDER 0x8F

9.52.2.4 #define SWI_API

9.52.2.5 #define UNUSEDPARAM(*x*) (void)x

Macro used to avoid “unused variable” compiler warnings generated due to the inclusion of the "-Wextra" flag in our make files.

9.52.3 Typedef Documentation

9.52.3.1 typedef int BOOL

9.52.3.2 typedef unsigned char BYTE

9.52.3.3 typedef char CHAR

9.52.3.4 typedef float FLOAT

9.52.3.5 typedef signed int INT32

9.52.3.6 typedef signed char INT8

9.52.3.7 typedef const char* LPCSTR

9.52.3.8 typedef unsigned short qmuint16

9.52.3.9 typedef unsigned long qmulong

9.52.3.10 typedef signed short SHORT

9.52.3.11 typedef unsigned int ULONG

9.52.3.12 typedef unsigned long long ULONGLONG

9.52.3.13 typedef unsigned short USHORT

9.52.3.14 typedef unsigned short WORD

9.53 swidms.h File Reference

Data Structures

- struct unpack_swidms_SLQSSwiDmsGetUsbNetNum_t
- struct pack_swidms_SLQSSwiDmsSetUsbNetNum_t
- struct pack_swidms_SLQSSwiDmsSetMTU_t
- struct unpack_swidms_SLQSSwiDmsSetMTU_t
- struct swidms_mtuSize3gppTlv
- struct swidms_hrpdMTUSizeTlv
- struct swidms_ehrpdMTUSizeTlv
- struct swidms_usbMTUSizeTlv
- struct unpack_swidms_SLQSSwiDmsGetMTU_t
- struct swidms_intfaceCfgTlv
- struct swidms_supportedIntBitmaskTlv
- struct unpack_swidms_SLQSSwiDmsGetUsbComp_t
- struct pack_swidms_SLQSSwiDmsSetUsbComp_t
- struct unpack_swidms_SLQSSwiDmsSetUsbComp_t
- struct swidms_SwiDmsGetHWWatchdog
- struct unpack_swidms_SLQSSwiDmsGetHWWatchdog_t
- struct pack_swidms_SLQSSwiDmsSetHWWatchdog_t
- struct unpack_swidms_SLQSSwiDmsSetHWWatchdog_t
- struct unpack_swidms_SLQSSwiDmsGetSecureInfo_t
- struct swidms_UimSlotSelectedTlv
- struct swidms_MultiplexedSimModeTlv
- struct swidms_AutoUimSwitchModeTlv
- struct unpack_swidms_SLQSSwiDmsGetUimSlot_t
- struct swidms_SetMultiplexedModeTlv
- struct pack_swidms_SLQSSwiDmsSetUimSlot_t
- struct unpack_swidms_SLQSSwiDmsSetUimSlot_t

Typedefs

- typedef unpack_result_t unpack_swidms_SLQSSwiDmsSetUsbNetNum_t

Functions

- int pack_swidms_SLQSSwiDmsGetUsbNetNum (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swidms_SLQSSwiDmsGetUsbNetNum (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsGetUsbNetNum_t *pOutput)
- int pack_swidms_SLQSSwiDmsSetUsbNetNum (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swidms_SLQSSwiDmsSetUsbNetNum_t *pReqParam)
- int unpack_swidms_SLQSSwiDmsSetUsbNetNum (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsSetUsbNetNum_t *pOutput)
- int pack_swidms_SLQSSwiDmsSetMTU (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swidms_SLQSSwiDmsSetMTU_t *reqArg)
- int unpack_swidms_SLQSSwiDmsSetMTU (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsSetMTU_t *pOutput)
- int pack_swidms_SLQSSwiDmsGetMTU (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swidms_SLQSSwiDmsGetMTU (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsGetMTU_t *pOutput)
- int pack_swidms_SLQSSwiDmsGetUsbComp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swidms_SLQSSwiDmsGetUsbComp (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsGetUsbComp_t *pOutput)
- int pack_swidms_SLQSSwiDmsSetUsbComp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swidms_SLQSSwiDmsSetUsbComp_t *reqArg)
- int unpack_swidms_SLQSSwiDmsSetUsbComp (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsSetUsbComp_t *pOutput)
- int pack_swidms_SLQSSwiDmsGetHWWatchdog (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swidms_SLQSSwiDmsGetHWWatchdog (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsGetHWWatchdog_t *pOutput)
- int pack_swidms_SLQSSwiDmsSetHWWatchdog (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swidms_SLQSSwiDmsSetHWWatchdog_t *reqArg)
- int unpack_swidms_SLQSSwiDmsSetHWWatchdog (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsSetHWWatchdog_t *pOutput)
- int pack_swidms_SLQSSwiDmsGetSecureInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swidms_SLQSSwiDmsGetSecureInfo (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsGetSecureInfo_t *pOutput)
- int pack_swidms_SLQSSwiDmsGetUimSlot (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swidms_SLQSSwiDmsGetUimSlot (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsGetUimSlot_t *pOutput)
- int pack_swidms_SLQSSwiDmsSetUimSlot (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swidms_SLQSSwiDmsSetUimSlot_t *reqArg)
- int unpack_swidms_SLQSSwiDmsSetUimSlot (uint8_t *pResp, uint16_t respLen, unpack_swidms_SLQSSwiDmsSetUimSlot_t *pOutput)

9.53.1 Typedef Documentation

9.53.1.1 typedef unpack_result_t unpack_swidms_SLQSSwiDmsSetUsbNetNum_t

9.53.2 Function Documentation

9.53.2.1 int pack_swidms_SLQSSwiDmsGetHWWatchdog (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the hardware watchdog settings pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.2 int pack_swidms_SLQSSwiDmsGetMTU (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get the preferred MTU Size of the 3GPP, HRPD, EHRPD and USB descriptor Interfaces pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.3 int pack_swidms_SLQSSwiDmsGetSecureInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

pack function to get secure boot config and other capabilities

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.4 int pack_swidms_SLQSSwiDmsGetUimSlot (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

pack function to get uim slot info

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.5 int pack_swidms_SLQSSwiDmsGetUsbComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the usb interface composition pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.6 int pack_swidms_SLQSSwiDmsGetUsbNetNum (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

pack function to get usb net numbers for QMAP configuration

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.7 int pack_swidms_SLQSSwiDmsSetHWWatchdog (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swidms_SLQSSwiDmsSetHWWatchdog_t * *reqArg*)

Sets the usb interface config value pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values.

9.53.2.8 int pack_swidms_SLQSSwiDmsSetMTU (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swidms_SLQSSwiDmsSetMTU_t * *reqArg*)

Sets the preferred MTU size for 3GPP, HRPD, EHRPD Interfaces pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.9 int pack_swidms_SLQSSwiDmsSetUimSlot (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swidms_SLQSSwiDmsSetUimSlot_t * *reqArg*)

Sets the uim slot value pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values.

9.53.2.10 int pack_swidms_SLQSSwiDmsSetUsbComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swidms_SLQSSwiDmsSetUsbComp_t * *reqArg*)

Sets the usb interface config value pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values. This setting will be effective after modem reboot

9.53.2.11 int pack_swidms_SLQSSwiDmsSetUsbNetNum (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swidms_SLQSSwiDmsSetUsbNetNum_t * *pReqParam*)

pack function to set usb net numbers for QMAP configuration

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request params

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.12 int unpack_swidms_SLQSSwiDmsGetHWWatchdog (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsGetHWWatchdog_t * *pOutput*)

Gets the hardware watchdog settings unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.13 int unpack_swidms_SLQSSwiDmsGetMTU (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsGetMTU_t * *pOutput*)

Get the MTU Size of the 3GPP, HRPD, EHRPD and USB descriptor Interfaces unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.14 int unpack_swidms_SLQSSwiDmsGetSecureInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsGetSecureInfo_t * *pOutput*)

unpack function to get secure boot config

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.15 int unpack_swidms_SLQSSwiDmsGetUimSlot (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsGetUimSlot_t * *pOutput*)

unpack function to get UIM Slot info

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.16 int unpack_swidms_SLQSSwiDmsGetUsbComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsGetUsbComp_t * *pOutput*)

Gets the usb interface composition unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.17 int unpack_swidms_SLQSSwiDmsGetUsbNetNum (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsGetUsbNetNum_t * *pOutput*)

unpack function to get usb net numbers for QMAP configuration

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.18 int unpack_swidms_SLQSSwiDmsSetHWWatchdog (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsSetHWWatchdog_t * *pOutput*)

Sets the hardware watchdog settings value unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values. This setting will be effective after modem reboot

9.53.2.19 int unpack_swidms_SLQSSwiDmsSetMTU (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsSetMTU_t * *pOutput*)

Sets preferred MTU size for 3GPP, HRPD, EHRPD Interfaces unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.53.2.20 int unpack_swidms_SLQSSwiDmsSetUimSlot (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsSetUimSlot_t * *pOutput*)

Sets the set UIM slot info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values. This setting will be effective after modem reboot

9.53.2.21 int unpack_swidms_SLQSSwiDmsSetUsbComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsSetUsbComp_t * *pOutput*)

Sets the usb interface config value unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values. This setting will be effective after modem reboot

9.53.2.22 int unpack_swidms_SLQSSwiDmsSetUsbNetNum (uint8_t * *pResp*, uint16_t *respLen*, unpack_swidms_SLQSSwiDmsSetUsbNetNum_t * *pOutput*)

unpack function to set usb net numbers for QMAP configuration

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.54 swiloc.h File Reference

Data Structures

- struct unpack_swiloc_SwiLocGetAutoStart_t
- struct pack_swiloc_SwiLocSetAutoStart_t

Typedefs

- typedef unpack_result_t unpack_swiloc_SwiLocSetAutoStart_t

Functions

- int pack_swiloc_SwiLocGetAutoStart (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiloc_SwiLocGetAutoStart (uint8_t *pResp, uint16_t respLen, unpack_swiloc_SwiLocGetAutoStart_t *pOutput)
- int pack_swiloc_SwiLocSetAutoStart (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiloc_SwiLocSetAutoStart_t *reqArg)
- int unpack_swiloc_SwiLocSetAutoStart (uint8_t *pResp, uint16_t respLen, unpack_swiloc_SwiLocSetAutoStart_t *pOutput)

9.54.1 Typedef Documentation

9.54.1.1 typedef unpack_result_t unpack_swiloc_SwiLocSetAutoStart_t

9.54.2 Function Documentation

9.54.2.1 int pack_swiloc_SwiLocGetAutoStart (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Auto Start pack

Parameters

in	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.54.2.2 int pack_swiloc_SwiLocSetAutoStart (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiloc_SwiLocSetAutoStart_t * *reqArg*)

Set Auto Start pack

Parameters

in	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.54.2.3 int unpack_swiloc_SwiLocGetAutoStart (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiloc_SwiLocGetAutoStart_t * *pOutput*)

Get Auto Start unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.54.2.4 int unpack_swiloc_SwiLocSetAutoStart (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiloc_SwiLocSetAutoStart_t * *pOutput*)

Set Auto Start unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.55 swioma.h File Reference

Data Structures

- struct pack_swioma_SLQSOMADMStartSession_t
- struct unpack_swioma_SLQSOMADMStartSession_t
- struct pack_swioma_SLQSOMADMCancelSession_t
- struct unpack_swioma_SLQSOMADMGetSettings_t
- struct pack_swioma_SLQSOMADMSetSettings_t
- struct pack_swioma_SLQSOMADMSendSelection_t
- struct pack_swioma_SLQSOMADMGetSessionInfo_t
- struct unpack_swioma_SLQSOMADMGetSessionInfo_t
- struct unpack_omaDmFotaTlv_t
- struct unpack_omaDmConfigTlv_t
- struct unpack_omaDmNotificationsTlv_t
- struct unpack_swioma_SLQSOMADMAlertCallback_ind_t

Macros

- #define LITEQMI_MAX_SWIOMA_STR_LEN 255

Typedefs

- typedef unpack_result_t unpack_swioma_SLQSOMADMCancelSession_t
- typedef unpack_result_t unpack_swioma_SLQSOMADMSetSettings_t
- typedef unpack_result_t unpack_swioma_SLQSOMADMSendSelection_t
- typedef unpack_result_t unpack_swioma_SLQSOMADMAlertCallback_t

Functions

- int pack_swioma_SLQSOMADMStartSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMStartSession_t reqParam)
- int unpack_swioma_SLQSOMADMStartSession (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMStartSession_t *pOutput)
- int pack_swioma_SLQSOMADMCancelSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMCancelSession_t reqParam)
- int unpack_swioma_SLQSOMADMCancelSession (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMCancelSession_t *pOutput)
- int pack_swioma_SLQSOMADMGetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swioma_SLQSOMADMGetSettings (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMGetSettings_t *pOutput)
- int pack_swioma_SLQSOMADMSetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSetSettings_t reqParam)
- int unpack_swioma_SLQSOMADMSetSettings (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMSetSettings_t *pOutput)
- int pack_swioma_SLQSOMADMSendSelection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSendSelection_t reqParam)
- int unpack_swioma_SLQSOMADMSendSelection (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMSendSelection_t *pOutput)
- int pack_swioma_SLQSOMADMGetSessionInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMGetSessionInfo_t reqParam)
- int unpack_swioma_SLQSOMADMGetSessionInfo (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMGetSessionInfo_t *pOutput)
- int pack_swioma_SLQSOMADMAlertCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swioma_SLQSOMADMAlertCallback (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMAlertCallback_t *pOutput)
- int unpack_swioma_SLQSOMADMAlertCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMAlertCallback_ind_t *pOutput)

9.55.1 Macro Definition Documentation

9.55.1.1 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.55.2 Typedef Documentation

9.55.2.1 typedef unpack_result_t unpack_swioma_SLQSOMADMAlertCallback_t

9.55.2.2 typedef unpack_result_t unpack_swioma_SLQSOMADMCancelSession_t

9.55.2.3 typedef unpack_result_t unpack_swioma_SLQSOMADMSendSelection_t

9.55.2.4 typedef unpack_result_t unpack_swioma_SLQSOMADMSetSettings_t

9.55.3 Function Documentation

9.55.3.1 int pack_swioma_SLQSOMADMAlertCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack QMI command to enable the SWIOMADM network-initiated alert callback function. This maps to SetSLQSOMADMAlertCallback

Parameters

pCtx	[out] • See pack_qmi_t for more information
pReqBuf	[in,out] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[in,out] • On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.55.3.2 int pack_swioma_SLQSOMADMCancelSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMCancelSession_t *reqParam*)

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSession

Parameters

pCtx	[out] • See pack_qmi_t for more information
pReqBuf	[in,out] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[in,out] • On input, size of pReqBuf • On output, number of bytes actually packed
reqParam	[in] • See pack_swioma_SLQSOMADMCancelSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.55.3.3 int pack_swioma_SLQSOMADMGetSessionInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMGetSessionInfo_t *reqParam*)

Function to pack QMI command to return information related to the current (or previous if no session is active) OMA-DM session. This maps to SLQSOMADMGetSessionInfo

Parameters

pCtx	[out] • See pack_qmi_t for more information
pReqBuf	[in,out] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[in,out] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[in] • See pack_swioma_SLQSOMADMGetSessionInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.55.3.4 int pack_swioma_SLQSOMADMGetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack command to retrieve the OMA-DM settings from the device. This maps to SLQSOMADMGetSettings2

Parameters

pCtx	[out] • See pack_qmi_t for more information

pReqBuf	[in,out] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[in,out] • On input, size of pReqBuf • On output, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.55.3.5 int pack_swioma_SLQSOMADMSendSelection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMSendSelection_t *reqParam*)

Function to pack OMA-DM send selection command This maps to SLQSOMADMSendSelection2

Parameters

pCtx	[out] • See pack_qmi_t for more information
pReqBuf	[in,out] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[in,out] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[in] • See pack_swioma_SLQSOMADMSendSelection_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.55.3.6 int pack_swioma_SLQSOMADMSetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMSetSettings_t *reqParam*)

Function to pack OMA-DM set settings command This maps to SLQSOMADMSetSettings3

Parameters

pCtx	[out] • See pack_qmi_t for more information
pReqBuf	[in,out] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[in,out] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[in] • See pack_swioma_SLQSOMADMSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.55.3.7 int pack_swioma_SLQSOMADMStartSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMStartSession_t *reqParam*)

Function to pack Start OMA-DM session command This maps to SLQSOMADMStartSession2

Parameters

pCtx	[out] • See pack_qmi_t for more information
pReqBuf	[in,out] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[in,out] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[in] • See pack_swioma_SLQSOMADMStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.55.3.8 int unpack_swioma_SLQSOMADMAlertCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMAlertCallback_t * *pOutput*)

Function to unpack response of QMI command to enable the SWIOMADM network-initiated alert callback function. This maps to SetSLQSOMADMAlertCallback

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Please use eQMI_SWIOMA_EVENT_IND indication to identify this event from SWIOMA service read function

9.55.3.9 int unpack_swioma_SLQSOMADMAlertCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMAlertCallback_ind_t * *pOutput*)

Function to unpack SWIOMADM alert indications This maps to SetSLQSOMADMAlertCallback

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem

pOutput	[out] • See unpack_swioma_SLQSOMADMAlertCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.55.3.10 int unpack_swioma_SLQSOMADMCancelSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMCancelSession_t * *pOutput*)

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSession

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.55.3.11 int unpack_swioma_SLQSOMADMGetSessionInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMGetSessionInfo_t * *pOutput*)

Function to unpack information related to the current (or previous if no session is active) OMA-DM session. This maps to SLQSOMADMGetSessionInfo

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • See unpack_swioma_SLQSOMADMGetSessionInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.55.3.12 int unpack_swioma_SLQSOMADMGetSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMGetSettings_t * *pOutput*)

Function to unpack OMA-DM get settings response from modem This maps to SLQSOMADMGetSettings2

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • See unpack_swioma_SLQSOMADMGetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.55.3.13 int unpack_swioma_SLQSOMADMSendSelection (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMSendSelection_t * *pOutput*)

Function to unpack OMA-DM send selection command This maps to SLQSOMADMSendSelection2

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.55.3.14 int unpack_swioma_SLQSOMADMSetSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMSetSettings_t * *pOutput*)

Function to unpack OMA-DM set settings command This maps to SLQSOMADMSetSettings3

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.55.3.15 int unpack_swioma_SLQSOMADMStartSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMStartSession_t * *pOutput*)

Function to unpack Start OMA-DM session response from modem This maps to SLQSOMADMStartSession2

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • See unpack_swioma_SLQSOMADMStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.56 swiomaext.h File Reference

Data Structures

- struct pack_swioma_SLQSOMADMStartSessionExt_t
- struct pack_swioma_SLQSOMADMCancelSessionExt_t
- struct pack_swioma_SLQSOMADMSetSettingsExt_t
- struct pack_swioma_SLQSOMADMSendSelectionExt_t
- struct unpack_swioma_SLQSOMADMGetSessionInfoExt_t

Macros

- #define LITEQMI_MAX_SWIOMA_STR_LEN 255
- #define LITE_SWIOMAEXT_MAX_UCS2_DATA_LEN 512

Typedefs

- typedef unpack_result_t unpack_swioma_SLQSOMADMStartSessionExt_t
- typedef unpack_result_t unpack_swioma_SLQSOMADMCancelSessionExt_t
- typedef unpack_result_t unpack_swioma_SLQSOMADMSetSettingsExt_t
- typedef unpack_result_t unpack_swioma_SLQSOMADMSendSelectionExt_t

Functions

- int pack_swioma_SLQSOMADMStartSessionExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMStartSessionExt_t reqParam)
- int unpack_swioma_SLQSOMADMStartSessionExt (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMStartSessionExt_t *pOutput)
- int pack_swioma_SLQSOMADMCancelSessionExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMCancelSessionExt_t reqParam)
- int unpack_swioma_SLQSOMADMCancelSessionExt (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMCancelSessionExt_t *pOutput)
- int pack_swioma_SLQSOMADMSetSettingsExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSetSettingsExt_t reqParam)
- int unpack_swioma_SLQSOMADMSetSettingsExt (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMSetSettingsExt_t *pOutput)
- int pack_swioma_SLQSOMADMSendSelectionExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSendSelectionExt_t reqParam)
- int unpack_swioma_SLQSOMADMSendSelectionExt (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMSendSelectionExt_t *pOutput)
- int pack_swioma_SLQSOMADMGetSessionInfoExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swioma_SLQSOMADMGetSessionInfoExt (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMGetSessionInfoExt_t *pOutput)

9.56.1 Macro Definition Documentation

9.56.1.1 #define LITE_SWIOMAEXT_MAX_UCS2_DATA_LEN 512

9.56.1.2 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.56.2 Typedef Documentation

9.56.2.1 typedef unpack_result_t unpack_swioma_SLQSOMADMCancelSessionExt_t

9.56.2.2 typedef unpack_result_t unpack_swioma_SLQSOMADMSendSelectionExt_t

9.56.2.3 typedef unpack_result_t unpack_swioma_SLQSOMADMSetSettingsExt_t

9.56.2.4 typedef unpack_result_t unpack_swioma_SLQSOMADMStartSessionExt_t

9.56.3 Function Documentation

9.56.3.1 int pack_swioma_SLQSOMADMCancelSessionExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMCancelSessionExt_t *reqParam*)

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSessionExt

Parameters

pCtx	[OUT] • See pack_qmi_t for more information
pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[IN] • See pack_swioma_SLQSOMADMCancelSessionExt_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.56.3.2 int pack_swioma_SLQSOMADMGetSessionInfoExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack OMA-DM session info command This maps to SLQSOMADMGetSessionInfoExt

Parameters

pCtx	[OUT] • See pack_qmi_t for more information
pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes

pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.56.3.3 int pack_swioma_SLQSOMADMSendSelectionExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMSendSelectionExt_t *reqParam*)

Function to pack OMA-DM send selection command This maps to SLQSOMADMSendSelectionExt

Parameters

pCtx	[OUT] • See pack_qmi_t for more information
pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[IN] • See pack_swioma_SLQSOMADMSendSelectionExt_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.56.3.4 int pack_swioma_SLQSOMADMSetSettingsExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMSetSettingsExt_t *reqParam*)

Function to pack OMA-DM set settings command This maps to SLQSOMADMSetSettingsExt

Parameters

pCtx	[OUT] • See pack_qmi_t for more information
pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[IN] • See pack_swioma_SLQSOMADMSetSettingsExt_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.56.3.5 int pack_swioma_SLQSOMADMStartSessionExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMStartSessionExt_t *reqParam*)

Function to pack Start OMA-DM session command This maps to SLQSOMADMStartSessionExt

Parameters

pCtx	[OUT] • See pack_qmi_t for more information
pReqBuf	[IN/OUT] • Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen	[IN/OUT] • On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam	[IN] • See pack_swioma_SLQSOMADMStartSessionExt_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.56.3.6 int unpack_swioma_SLQSOMADMCancelSessionExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMCancelSessionExt_t * *pOutput*)

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSessionExt

Parameters

pResp	[IN] • Response from modem
respLen	[IN] • Length of pResp from modem
pOutput	[OUT] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.56.3.7 int unpack_swioma_SLQSOMADMGetSessionInfoExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMGetSessionInfoExt_t * *pOutput*)

Function to unpack information related to the current (or previous if no session is active) OMA-DM session. This maps to SLQSOMADMGetSessionInfoExt

Parameters

pResp	[in] • Response from modem
respLen	[in] • Length of pResp from modem
pOutput	[out] • See unpack_swioma_SLQSOMADMGetSessionInfoExt_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.56.3.8 int unpack_swioma_SLQSOMADMSendSelectionExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMSendSelectionExt_t * *pOutput*)

Function to unpack OMA-DM send selection command This maps to SLQSOMADMSendSelectionExt

Parameters

pResp	[IN] • Response from modem
respLen	[IN] • Length of pResp from modem
pOutput	[OUT] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.56.3.9 int unpack_swioma_SLQSOMADMSetSettingsExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMSetSettingsExt_t * *pOutput*)

Function to unpack OMA-DM set settings command This maps to SLQSOMADMSetSettingsExt

Parameters

pResp	[IN] • Response from modem
respLen	[IN] • Length of pResp from modem
pOutput	[OUT] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.56.3.10 int unpack_swioma_SLQSOMADMStartSessionExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMStartSessionExt_t * *pOutput*)

Function to unpack Start OMA-DM session response from modem This maps to SLQSOMADMStartSessionExt

Parameters

pResp	[IN] • Response from modem
respLen	[IN] • Length of pResp from modem
pOutput	[OUT] • response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.57 switype_256bit.h File Reference

Data Structures

- struct swi_uint256_t

Macros

- #define SWI_UINT256_WORD_COUNT 16
- #define SWI_UINT256_BITS_PER_WORD 16
- #define SWI_UINT256_WORD_OFFSET(val) ((val) / SWI_UINT256_BITS_PER_WORD)
- #define SWI_UINT256_BIT_OFFSET(val) ((val) % SWI_UINT256_BITS_PER_WORD)
- #define SWI_UINT256_INT_VALUE {{0}}

Typedefs

- typedef void(* logger)(uint8_t lvl, const char *buff)

Functions

- void swi_uint256_set_bit (swi_uint256_t *pMask, uint8_t bit)
- void swi_uint256_clear_bit (swi_uint256_t *pMask, uint8_t bit)
- uint8_t swi_uint256_get_bit (swi_uint256_t Mask, uint8_t bit)
- void swi_uint256_print_mask (swi_uint256_t mask)

9.57.1 Macro Definition Documentation

9.57.1.1 #define SWI_UINT256_BIT_OFFSET(*val*) ((val) % SWI_UINT256_BITS_PER_WORD)

9.57.1.2 #define SWI_UINT256_BITS_PER_WORD 16

9.57.1.3 #define SWI_UINT256_INT_VALUE {{0}}

9.57.1.4 #define SWI_UINT256_WORD_COUNT 16

9.57.1.5 #define SWI_UINT256_WORD_OFFSET(*val*) ((val) / SWI_UINT256_BITS_PER_WORD)

9.57.2 Typedef Documentation

9.57.2.1 typedef void(∗ logger)(uint8_t lvl, const char ∗buff)

9.57.3 Function Documentation

9.57.3.1 void swi_uint256_clear_bit (swi_uint256_t ∗ *pMask,* uint8_t *bit*)

Clear relevant bit in provided mask

Parameters

in	*pMask*	Pointer to bitmask
in	*bit*	Bit to clear

9.57.3.2 uint8_t swi_uint256_get_bit (swi_uint256_t *Mask,* uint8_t *bit*)

Get relevant bit from provided mask

Parameters

in	*Mask*	Bit mask
in	*bit*	Bit to extract

Returns

Bit value

- 0 - bit is not set
- 1 - bit is set

9.57.3.3 void swi_uint256_print_mask (swi_uint256_t *mask*)

Print the log mask to syslog. Use only for debugging

Parameters

in	*mask*	Bit mask

9.57.3.4 void swi_uint256_set_bit (swi_uint256_t ∗ *pMask,* uint8_t *bit*)

Set relevant bit in provided mask

Parameters

in	*pMask*	Pointer to bit mask
in	*bit*	Bit to set

9.58 SWIWWANCMAPI.h File Reference

9.59 tmd.h File Reference

Data Structures

- struct tmd_mitigationDevList
- struct unpack_tmd_SLQSTmdGetMitigationDevList_t
- struct pack_tmd_SLQSTmdGetMitigationLvl_t
- struct unpack_tmd_SLQSTmdGetMitigationLvl_t
- struct pack_tmd_SLQSTmdRegNotMitigationLvl_t
- struct unpack_tmd_SLQSTmdRegNotMitigationLvl_t
- struct pack_tmd_SLQSTmdDeRegNotMitigationLvl_t
- struct unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t
- struct unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t

Macros

- #define TMD_MAX_DEV_LIST 255
- #define MAX_MITIGATION_DEV_LIST_LEN 255
- #define MAX_MITIGATION_DEV_ID_LEN 255

Functions

- int pack_tmd_SLQSTmdGetMitigationDevList (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_tmd_SLQSTmdGetMitigationDevList (uint8_t *pResp, uint16_t respLen, unpack_tmd_SLQSTmdGetMitigationDevList_t *pOutput)
- int pack_tmd_SLQSTmdGetMitigationLvl (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_tmd_SLQSTmdGetMitigationLvl_t *reqArg)
- int unpack_tmd_SLQSTmdGetMitigationLvl (uint8_t *pResp, uint16_t respLen, unpack_tmd_SLQSTmdGetMitigationLvl_t *pOutput)
- int pack_tmd_SLQSTmdRegNotMitigationLvl (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_tmd_SLQSTmdRegNotMitigationLvl_t *reqArg)
- int unpack_tmd_SLQSTmdRegNotMitigationLvl (uint8_t *pResp, uint16_t respLen, unpack_tmd_SLQSTmdRegNotMitigationLvl_t *pOutput)
- int pack_tmd_SLQSTmdDeRegNotMitigationLvl (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_tmd_SLQSTmdDeRegNotMitigationLvl_t *reqArg)
- int unpack_tmd_SLQSTmdDeRegNotMitigationLvl (uint8_t *pResp, uint16_t respLen, unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t *pOutput)
- int unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t *pOutput)

9.59.1 Macro Definition Documentation

9.59.1.1 #define MAX_MITIGATION_DEV_ID_LEN 255

9.59.1.2 #define MAX_MITIGATION_DEV_LIST_LEN 255

9.59.1.3 #define TMD_MAX_DEV_LIST 255

9.59.2 Function Documentation

9.59.2.1 int pack_tmd_SLQSTmdDeRegNotMitigationLvl (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_tmd_SLQSTmdDeRegNotMitigationLvl_t * reqArg)

To DeRegister notification mitigation level pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*reqArg*	request argument.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.2 int pack_tmd_SLQSTmdGetMitigationDevList (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

To get mitigation device list pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.3 int pack_tmd_SLQSTmdGetMitigationLvl (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_tmd_SLQSTmdGetMitigationLvl_t * reqArg)

To get mitigation level pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*reqArg*	request argument.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.4 int pack_tmd_SLQSTmdRegNotMitigationLvl (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_tmd_SLQSTmdRegNotMitigationLvl_t * *reqArg*)

To Register notification mitigation level pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*reqArg*	request argument.

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.5 int unpack_tmd_SLQSTmdDeRegNotMitigationLvl (uint8_t * *pResp*, uint16_t *respLen*, unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t * *pOutput*)

To De-Register notification mitigation level unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.6 int unpack_tmd_SLQSTmdGetMitigationDevList (uint8_t * *pResp*, uint16_t *respLen*, unpack_tmd_SLQSTmdGetMitigationDevList_t * *pOutput*)

To mitigation device list unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.7 int unpack_tmd_SLQSTmdGetMitigationLvl (uint8_t * *pResp*, uint16_t *respLen*, unpack_tmd_SLQSTmdGetMitigationLvl_t * *pOutput*)

To get mitigation level unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.8 int unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t * *pOutput*)

Mitigiation Level Report Indication unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.59.2.9 int unpack_tmd_SLQSTmdRegNotMitigationLvl (uint8_t * *pResp*, uint16_t *respLen*, unpack_tmd_SLQSTmdRegNotMitigationLvl_t * *pOutput*)

To Register notification mitigation level unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60 uim.h File Reference

Data Structures

- struct uim_appStatus
- struct uim_slotInfo
- struct uim_cardStatus
- struct uim_hotSwapStatus
- struct uim_validCardStatus
- struct uim_simBusyStatus
- struct unpack_uim_GetCardStatus_t
- struct unpack_uim_GetCardStatusV2_t
- struct uim_encryptedPIN1
- struct uim_remainingRetries
- struct uim_sessionInformation
- struct uim_verifyUIMPIN
- struct uim_unblockUIMPIN
- struct uim_cardResult
- struct uim_setPINProtection
- struct uim_changeUIMPIN
- struct uim_fileInfo
- struct uim_UIMSessionInformation
- struct uim_readTransparentInfo
- struct uim_readResult
- struct pack_uim_VerifyPin_t
- struct unpack_uim_VerifyPin_t
- struct pack_uim_UnblockPin_t
- struct unpack_uim_UnblockPin_t
- struct unpack_uim_UnblockPinV2_t
- struct pack_uim_SetPinProtection_t
- struct unpack_uim_SetPinProtection_t
- struct pack_uim_ChangePin_t
- struct unpack_uim_ChangePin_t
- struct pack_uim_ReadTransparent_t
- struct unpack_uim_ReadTransparent_t

- struct pack_uim_SLQSUIMEventRegister_t
- struct unpack_uim_SLQSUIMEventRegister_t
- struct appStats
- struct slotInf
- struct unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t
- struct slot_t
- struct slots_t
- struct uim_physlotInfo
- struct uim_physlotsInfo
- struct unpack_uim_SLQSUIMGetSlotsStatus_t
- struct uim_GetSlotsStatusTlv
- struct uim_GetSlotsInfoTlv
- struct unpack_uim_SLQSUIMGetSlotsStatusV2_t
- struct pack_uim_SLQSUIMSwitchSlot_t
- struct unpack_uim_SetUimSlotStatusChangeCallback_ind_t
- struct pack_uim_SLQSUIMPowerUp_t
- struct pack_uim_SLQSUIMPowerDown_t
- struct pack_uim_SLQSUIMRefreshOK_t
- struct uim_registerRefresh
- struct pack_uim_SLQSUIMRefreshRegister_t
- struct pack_uim_SLQSUIMRefreshComplete_t
- struct uim_refreshevent
- struct pack_uim_SLQSUIMRefreshGetLastEvent_t
- struct unpack_uim_SLQSUIMRefreshGetLastEvent_t
- struct pack_uim_SLQSUIMGetFileAttributes_t
- struct uim_fileAttributes
- struct unpack_uim_SLQSUIMGetFileAttributes_t
- struct uim_depersonalizationInformation
- struct pack_uim_SLQSUIMDepersonalization_t
- struct unpack_uim_SLQSUIMDepersonalization_t
- struct uim_authenticationData
- struct pack_uim_SLQSUIMAuthenticate_t
- struct uim_authenticateResult
- struct unpack_uim_SLQSUIMAuthenticate_t
- struct uim_personalizationStatus
- struct pack_uim_SLQSUIMGetConfiguration_t
- struct unpack_uim_SLQSUIMGetConfiguration_t
- struct unpack_uim_SLQSUIMRefreshCallback_Ind_t
- struct pack_uim_SLQSUIMSetServiceStatus_t
- struct pack_uim_SLQSUIMGetServiceStatus_t
- struct uim_UIMGetFDNStatus
- struct uim_UIMGetHiddenKeyStatus
- struct uim_UIMGetIndex
- struct unpack_uim_SLQSUIMGetServiceStatus_t
- struct uim_readRecordInfo
- struct pack_uim_SLQSUIMReadRecord_t
- struct uim_cardResultInfo
- struct uim_readResultInfo
- struct uim_additionalReadResult
- struct uim_indToken
- struct unpack_uim_SLQSUIMReadRecord_t
- struct uim_writeRecordInfo
- struct pack_uim_SLQSUIMWriteRecord_t
- struct unpack_uim_SLQSUIMWriteRecord_t
- struct uim_writeTransparentInfo

- struct pack_uim_SLQSUIMWriteTransparent_t
- struct unpack_uim_SLQSUIMWriteTransparent_t
- struct pack_uim_SLQSUIMGetEID_t
- struct uim_eidInfo
- struct unpack_uim_SLQSUIMGetEID_t
- struct uim_UIMProfileCode
- struct pack_uim_SLQSUIMAddProfile_t
- struct pack_uim_SLQSUIMDeleteProfile_t
- struct uim_UIMNickName
- struct pack_uim_SLQSUIMUpdateProfile_t
- struct pack_uim_SLQSUIMGetSIMProfile_t
- struct uim_getSimProfileInfo
- struct uim_activeProfileInfo
- struct unpack_uim_SLQSUIMGetSIMProfile_t
- struct pack_uim_SLQSUIMSetSIMProfile_t
- struct uim_profileICCIDTlv
- struct uim_profileStateTlv
- struct uim_profileNicknameTlv
- struct uim_profileSPNTlv
- struct uim_profileNameTlv
- struct uim_profileIconTypeTlv
- struct uim_profileIconTlv
- struct uim_profileClassTlv
- struct uim_profilePolicyRulesTlv
- struct pack_uim_SLQSUIMGetProfileInfo_t
- struct unpack_uim_SLQSUIMGetProfileInfo_t
- struct pack_uim_SLQSUIMeUICCMemReset_t

Macros

- #define UIM_UINT8_MAX_STRING_SZ 255
- #define UIM_MAX_DESCRIPTION_LENGTH 255
- #define UIM_MAX_NO_OF_SLOTS 5
- #define UIM_MAX_NO_OF_APPLICATIONS 10
- #define MAX_NO_OF_SLOTS 5
- #define MAX_NO_OF_APPLICATIONS 10
- #define MAX_DESCRIPTION_LENGTH 255
- #define MAX_SLOTS_STATUS 255
- #define MAX_ICCID_LENGTH 255
- #define UIM_MAX_CONTENT_LENGTH 1024
- #define UIM_MAX_ACTIVE_PERS_FEATURES 12
- #define MAX_ATR_LENGTH 255
- #define MAX_PHY_SLOTS_INFO 255

Typedefs

- typedef unpack_result_t unpack_uim_SLQSUIMSwitchSlot_t
- typedef unpack_result_t unpack_uim_SLQSUIMPowerUp_t
- typedef unpack_result_t unpack_uim_SLQSUIMPowerDown_t
- typedef unpack_result_t unpack_uim_SLQSUIMReset_t
- typedef unpack_result_t unpack_uim_SLQSUIMRefreshOK_t
- typedef unpack_result_t unpack_uim_SLQSUIMRefreshRegister_t
- typedef unpack_result_t unpack_uim_SLQSUIMRefreshComplete_t
- typedef unpack_result_t unpack_uim_SLQSUIMSetServiceStatus_t

- typedef unpack_result_t unpack_uim_SLQSUIMAddProfile_t
- typedef unpack_result_t unpack_uim_SLQSUIMDeleteProfile_t
- typedef unpack_result_t unpack_uim_SLQSUIMUpdateProfile_t
- typedef unpack_result_t unpack_uim_SLQSUIMSetSIMProfile_t
- typedef unpack_result_t unpack_uim_SLQSUIMeUICCMemReset_t

Functions

- int pack_uim_GetCardStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_uim_GetCardStatus (uint8_t *pResp, uint16_t respLen, unpack_uim_GetCardStatus_t *pOutput)
- int unpack_uim_GetCardStatusV2 (uint8_t *pResp, uint16_t respLen, unpack_uim_GetCardStatusV2_t *pOutput)
- int pack_uim_VerifyPin (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_VerifyPin_t *reqArg)
- int unpack_uim_VerifyPin (uint8_t *pResp, uint16_t respLen, unpack_uim_VerifyPin_t *pOutput)
- int pack_uim_UnblockPin (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_UnblockPin_t *reqArg)
- int unpack_uim_UnblockPin (uint8_t *pResp, uint16_t respLen, unpack_uim_UnblockPin_t *pOutput)
- int unpack_uim_UnblockPinV2 (uint8_t *pResp, uint16_t respLen, unpack_uim_UnblockPinV2_t *pOutput)
- int pack_uim_SetPinProtection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SetPinProtection_t *reqArg)
- int unpack_uim_SetPinProtection (uint8_t *pResp, uint16_t respLen, unpack_uim_SetPinProtection_t *pOutput)
- int pack_uim_ChangePin (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_ChangePin_t *reqArg)
- int unpack_uim_ChangePin (uint8_t *pResp, uint16_t respLen, unpack_uim_ChangePin_t *pOutput)
- int pack_uim_ReadTransparent (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_ReadTransparent_t *reqArg)
- int unpack_uim_ReadTransparent (uint8_t *pResp, uint16_t respLen, unpack_uim_ReadTransparent_t *pOutput)
- int pack_uim_SLQSUIMEventRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMEventRegister_t *reqArg)
- int unpack_uim_SLQSUIMEventRegister (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMEventRegister_t *pOutput)
- int unpack_uim_SLQSUIMSetStatusChangeCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t *pOutput)
- int pack_uim_SLQSUIMGetSlotsStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_uim_SLQSUIMGetSlotsStatus (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetSlotsStatus_t *pOutput)
- int unpack_uim_SLQSUIMGetSlotsStatusV2 (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetSlotsStatusV2_t *pOutput)
- int pack_uim_SLQSUIMSwitchSlot (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMSwitchSlot_t *reqArg)
- int unpack_uim_SLQSUIMSwitchSlot (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMSwitchSlot_t *pOutput)
- int unpack_uim_SetUimSlotStatusChangeCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_uim_SetUimSlotStatusChangeCallback_ind_t *pOutput)
- int pack_uim_SLQSUIMPowerUp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMPowerUp_t *reqArg)
- int unpack_uim_SLQSUIMPowerUp (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMPowerUp_t *pOutput)
- int pack_uim_SLQSUIMPowerDown (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMPowerDown_t *reqArg)
- int unpack_uim_SLQSUIMPowerDown (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMPowerDown_t *pOutput)

- int pack_uim_SLQSUIMReset (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_uim_SLQSUIMReset (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMReset_t *pOutput)
- int pack_uim_SLQSUIMRefreshOK (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshOK_t *reqArg)
- int unpack_uim_SLQSUIMRefreshOK (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMRefreshOK_t *pOutput)
- int pack_uim_SLQSUIMRefreshRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshRegister_t *reqArg)
- int unpack_uim_SLQSUIMRefreshRegister (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMRefreshRegister_t *pOutput)
- int pack_uim_SLQSUIMRefreshComplete (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshComplete_t *reqArg)
- int unpack_uim_SLQSUIMRefreshComplete (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMRefreshComplete_t *pOutput)
- int pack_uim_SLQSUIMRefreshGetLastEvent (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshGetLastEvent_t *reqArg)
- int unpack_uim_SLQSUIMRefreshGetLastEvent (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMRefreshGetLastEvent_t *pOutput)
- int pack_uim_SLQSUIMGetFileAttributes (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetFileAttributes_t *reqArg)
- int unpack_uim_SLQSUIMGetFileAttributes (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetFileAttributes_t *pOutput)
- int pack_uim_SLQSUIMDepersonalization (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMDepersonalization_t *reqArg)
- int unpack_uim_SLQSUIMDepersonalization (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMDepersonalization_t *pOutput)
- int pack_uim_SLQSUIMAuthenticate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMAuthenticate_t *reqArg)
- int unpack_uim_SLQSUIMAuthenticate (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMAuthenticate_t *pOutput)
- int pack_uim_SLQSUIMGetConfiguration (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetConfiguration_t *reqArg)
- int unpack_uim_SLQSUIMGetConfiguration (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetConfiguration_t *pOutput)
- int unpack_uim_SLQSUIMRefreshCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMRefreshCallback_Ind_t *pOutput)
- int pack_uim_SLQSUIMSetServiceStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMSetServiceStatus_t *reqArg)
- int unpack_uim_SLQSUIMSetServiceStatus (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMSetServiceStatus_t *pOutput)
- int pack_uim_SLQSUIMGetServiceStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetServiceStatus_t *reqArg)
- int unpack_uim_SLQSUIMGetServiceStatus (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetServiceStatus_t *pOutput)
- int pack_uim_SLQSUIMReadRecord (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMReadRecord_t *reqArg)
- int unpack_uim_SLQSUIMReadRecord (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMReadRecord_t *pOutput)
- int pack_uim_SLQSUIMWriteRecord (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMWriteRecord_t *reqArg)
- int unpack_uim_SLQSUIMWriteRecord (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMWriteRecord_t *pOutput)
- int pack_uim_SLQSUIMWriteTransparent (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMWriteTransparent_t *reqArg)
- int unpack_uim_SLQSUIMWriteTransparent (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMWriteTransparent_t *pOutput)

- int pack_uim_SLQSUIMGetEID (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetEID_t *reqArg)
- int unpack_uim_SLQSUIMGetEID (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetEID_t *pOutput)
- int pack_uim_SLQSUIMAddProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMAddProfile_t *reqArg)
- int unpack_uim_SLQSUIMAddProfile (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMAddProfile_t *pOutput)
- int pack_uim_SLQSUIMDeleteProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMDeleteProfile_t *reqArg)
- int unpack_uim_SLQSUIMDeleteProfile (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMDeleteProfile_t *pOutput)
- int pack_uim_SLQSUIMUpdateProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMUpdateProfile_t *reqArg)
- int unpack_uim_SLQSUIMUpdateProfile (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMUpdateProfile_t *pOutput)
- int pack_uim_SLQSUIMGetSIMProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetSIMProfile_t *reqArg)
- int unpack_uim_SLQSUIMGetSIMProfile (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetSIMProfile_t *pOutput)
- int pack_uim_SLQSUIMSetSIMProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMSetSIMProfile_t *reqArg)
- int unpack_uim_SLQSUIMSetSIMProfile (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMSetSIMProfile_t *pOutput)
- int pack_uim_SLQSUIMGetProfileInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetProfileInfo_t *reqArg)
- int unpack_uim_SLQSUIMGetProfileInfo (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetProfileInfo_t *pOutput)
- int pack_uim_SLQSUIMeUICCMemReset (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMeUICCMemReset_t *reqArg)
- int unpack_uim_SLQSUIMeUICCMemReset (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMeUICCMemReset_t *pOutput)

9.60.1 Macro Definition Documentation

9.60.1.1 #define MAX_ATR_LENGTH 255

9.60.1.2 #define MAX_DESCRIPTION_LENGTH 255

9.60.1.3 #define MAX_ICCID_LENGTH 255

9.60.1.4 #define MAX_NO_OF_APPLICATIONS 10

9.60.1.5 #define MAX_NO_OF_SLOTS 5

9.60.1.6 #define MAX_PHY_SLOTS_INFO 255

9.60.1.7 #define MAX_SLOTS_STATUS 255

9.60.1.8 #define UIM_MAX_ACTIVE_PERS_FEATURES 12

9.60.1.9 #define UIM_MAX_CONTENT_LENGTH 1024

9.60.1.10 #define UIM_MAX_DESCRIPTION_LENGTH 255

9.60.1.11 #define UIM_MAX_NO_OF_APPLICATIONS 10

9.60.1.12 #define UIM_MAX_NO_OF_SLOTS 5

9.60.1.13 #define UIM_UINT8_MAX_STRING_SZ 255

9.60.2 Typedef Documentation

9.60.2.1 typedef unpack_result_t unpack_uim_SLQSUIMAddProfile_t

9.60.2.2 typedef unpack_result_t unpack_uim_SLQSUIMDeleteProfile_t

9.60.2.3 typedef unpack_result_t unpack_uim_SLQSUIMeUICCMemReset_t

9.60.2.4 typedef unpack_result_t unpack_uim_SLQSUIMPowerDown_t

9.60.2.5 typedef unpack_result_t unpack_uim_SLQSUIMPowerUp_t

9.60.2.6 typedef unpack_result_t unpack_uim_SLQSUIMRefreshComplete_t

9.60.2.7 typedef unpack_result_t unpack_uim_SLQSUIMRefreshOK_t

9.60.2.8 typedef unpack_result_t unpack_uim_SLQSUIMRefreshRegister_t

9.60.2.9 typedef unpack_result_t unpack_uim_SLQSUIMReset_t

9.60.2.10 typedef unpack_result_t unpack_uim_SLQSUIMSetServiceStatus_t

9.60.2.11 typedef unpack_result_t unpack_uim_SLQSUIMSetSIMProfile_t

9.60.2.12 typedef unpack_result_t unpack_uim_SLQSUIMSwitchSlot_t

9.60.2.13 typedef unpack_result_t unpack_uim_SLQSUIMUpdateProfile_t

9.60.3 Function Documentation

9.60.3.1 int pack_uim_ChangePin (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_ChangePin_t * *reqArg*)

Change Pin pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.2 int pack_uim_GetCardStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Card Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.3 int pack_uim_ReadTransparent (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_ReadTransparent_t * *reqArg*)

SLQS ReadTransparent pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.4 int pack_uim_SetPinProtection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SetPinProtection_t * *reqArg*)

Set Pin Protection pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.5 int pack_uim_SLQSUIMAddProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMAddProfile_t * *reqArg*)

Pack add profile for eSIM

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.6 int pack_uim_SLQSUIMAuthenticate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMAuthenticate_t * *reqArg*)

Send a security command to the card pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.7 int pack_uim_SLQSUIMDeleteProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMDeleteProfile_t * *reqArg*)

Pack delete profile for eSIM

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.8 int pack_uim_SLQSUIMDepersonalization (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMDepersonalization_t * *reqArg*)

De-activates or unblocks the personalization on the phone pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.9 int pack_uim_SLQSUIMeUICCMemReset (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMeUICCMemReset_t * *reqArg*)

Pack UIM eUICC memory reset

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.10 int pack_uim_SLQSUIMEventRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMEventRegister_t * *reqArg*)

UIM Status Change callback enable pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.11 int pack_uim_SLQSUIMGetConfiguration (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMGetConfiguration_t * *reqArg*)

Gets the modem configuration for the UIM module pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.12 int pack_uim_SLQSUIMGetEID (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMGetEID_t * *reqArg*)

Pack UIM Get EID

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.13 int pack_uim_SLQSUIMGetFileAttributes (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMGetFileAttributes_t * reqArg)

Retrieves the file attributes for any EF(Elementary File) or DF(Dedicated File) in the card and provides access by the path pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.14 int pack_uim_SLQSUIMGetProfileInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMGetProfileInfo_t * reqArg)

Pack Get profile Info

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.15 int pack_uim_SLQSUIMGetServiceStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMGetServiceStatus_t * reqArg)

Pack Get Service Status.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.16 int pack_uim_SLQSUIMGetSIMProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMGetSIMProfile_t * *reqArg*)

Pack Get SIM profile

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.17 int pack_uim_SLQSUIMGetSlotsStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Pack get slots status.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.18 int pack_uim_SLQSUIMPowerDown (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMPowerDown_t * *reqArg*)

Powers down the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.19 int pack_uim_SLQSUIMPowerUp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMPowerUp_t * *reqArg*)

Powers up the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.20 int pack_uim_SLQSUIMReadRecord (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMReadRecord_t * *reqArg*)

Pack Read Record Status

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.21 int pack_uim_SLQSUIMRefreshComplete (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshComplete_t * *reqArg*)

Enables the terminal response to be sent to the card When the client finished Refresh procedure pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.22 int pack_uim_SLQSUIMRefreshGetLastEvent (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshGetLastEvent_t * *reqArg*)

Retrieve the last refresh event pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.23 int pack_uim_SLQSUIMRefreshOK (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshOK_t * *reqArg*)

Enables the client to indicate whether it is OK to start the Refresh procedure pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.24 int pack_uim_SLQSUIMRefreshRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshRegister_t * *reqArg*)

Registers for file change notifications triggered by the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.25 int pack_uim_SLQSUIMReset (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Resets the issuing control points state kept by the service pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.26 int pack_uim_SLQSUIMSetServiceStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMSetServiceStatus_t * *reqArg*)

Pack Set Service Status.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.27 int pack_uim_SLQSUIMSetSIMProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMSetSIMProfile_t * *reqArg*)

Pack UIM Set SIM Profile

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.28 int pack_uim_SLQSUIMSwitchSlot (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMSwitchSlot_t * *reqArg*)

switch slot pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.29 int pack_uim_SLQSUIMUpdateProfile (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMUpdateProfile_t * reqArg)

Pack update profile for eSIM

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.30 int pack_uim_SLQSUIMWriteRecord (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMWriteRecord_t * reqArg)

Pack Write Record data

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.31 int pack_uim_SLQSUIMWriteTransparent (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_uim_SLQSUIMWriteTransparent_t * reqArg)

Pack Write Transparent

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.32 int pack_uim_UnblockPin (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_UnblockPin_t * *reqArg*)

Unblock Pin pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.33 int pack_uim_VerifyPin (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_VerifyPin_t * *reqArg*)

Verify Pin Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.34 int unpack_uim_ChangePin (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_ChangePin_t * *pOutput*)

Change Pin unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.35 int unpack_uim_GetCardStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_GetCardStatus_t * *pOutput*)

Get Card Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.36 int unpack_uim_GetCardStatusV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_GetCardStatusV2_t * *pOutput*)

Get Card Status unpack V2

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.37 int unpack_uim_ReadTransparent (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_ReadTransparent_t * *pOutput*)

SLQS ReadTransparent unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.38 int unpack_uim_SetPinProtection (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SetPinProtection_t * *pOutput*)

Set Pin Protection unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.39 int unpack_uim_SetUimSlotStatusChangeCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SetUimSlotStatusChangeCallback_ind_t * *pOutput*)

UIM Slot Status Change indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

use pack_uim_SLQSUIMEventRegister to subscribe

9.60.3.40 int unpack_uim_SLQSUIMAddProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMAddProfile_t * *pOutput*)

Unpack Add pofile for eSIM

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.41 int unpack_uim_SLQSUIMAuthenticate (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIM-Authenticate_t * *pOutput*)

Unpack the status code received from the card when card responded to the read request

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.42 int unpack_uim_SLQSUIMDeleteProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMDelete-Profile_t * *pOutput*)

Unpack delete pofile for eSIM

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.43 int unpack_uim_SLQSUIMDepersonalization (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMDepersonalization_t * *pOutput*)

De-activates or unblocks the personalization on the phone unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.44 int unpack_uim_SLQSUIMeUICCMemReset (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMeUICCMemReset_t * *pOutput*)

Unpack UIM eUICC memory reset

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.45 int unpack_uim_SLQSUIMEventRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMEvent-Register_t * *pOutput*)

UIM Status Change callback enable unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.46 int unpack_uim_SLQSUIMGetConfiguration (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetConfiguration_t * *pOutput*)

Gets the modem configuration for the UIM module unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.47 int unpack_uim_SLQSUIMGetEID (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetEID_t * *pOutput*)

UIM Get EID unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.48 int unpack_uim_SLQSUIMGetFileAttributes (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetFileAttributes_t * *pOutput*)

Retrieves the file attributes for any EF(Elementary File) or DF(Dedicated File) in the card and provides access by the path unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.49 int unpack_uim_SLQSUIMGetProfileInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetProfileInfo_t * *pOutput*)

UIM Get Profile Info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.50 int unpack_uim_SLQSUIMGetServiceStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetServiceStatus_t * *pOutput*)

Get Service Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.51 int unpack_uim_SLQSUIMGetSIMProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetSIMProfile_t * *pOutput*)

UIM Get SIM Profile unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.52 int unpack_uim_SLQSUIMGetSlotsStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetSlotsStatus_t * *pOutput*)

get slot status unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.53 int unpack_uim_SLQSUIMGetSlotsStatusV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetSlotsStatusV2_t * *pOutput*)

get slot status unpack V2

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.54 int unpack_uim_SLQSUIMPowerDown (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMPowerDown_t * *pOutput*)

Powers down the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.55 int unpack_uim_SLQSUIMPowerUp (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMPowerUp_t * *pOutput*)

Powers up the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.56 int unpack_uim_SLQSUIMReadRecord (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMReadRecord_t * *pOutput*)

Read Record Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.57 int unpack_uim_SLQSUIMRefreshCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMRefreshCallback_Ind_t * *pOutput*)

Unpack UIM refresh event indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.58 int unpack_uim_SLQSUIMRefreshComplete (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMRefreshComplete_t * *pOutput*)

Enables the terminal response to be sent to the card When the client finished Refresh procedure unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.59 int unpack_uim_SLQSUIMRefreshGetLastEvent (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMRefreshGetLastEvent_t * *pOutput*)

Retrieve the last refresh event unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.60 int unpack_uim_SLQSUIMRefreshOK (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMRefreshOK_t * *pOutput*)

Enables the client to indicate whether it is OK to start the Refresh procedure unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.61 int unpack_uim_SLQSUIMRefreshRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMRefreshRegister_t * *pOutput*)

Registers for file change notifications triggered by the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.62 int unpack_uim_SLQSUIMReset (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMReset_t * *pOutput*)

Resets the issuing control points state kept by the service unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.63 int unpack_uim_SLQSUIMSetServiceStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMSetServiceStatus_t * *pOutput*)

Set Service Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.64 int unpack_uim_SLQSUIMSetSIMProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMSetSIMProfile_t * *pOutput*)

Unpack UIM Set SIM profile

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.65 int unpack_uim_SLQSUIMSetStatusChangeCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t * *pOutput*)

UIM Status Change indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

use pack_uim_SLQSUIMEventRegister to subscribe

9.60.3.66 int unpack_uim_SLQSUIMSwitchSlot (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMSwitchSlot_t * *pOutput*)

switch slot unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.67 int unpack_uim_SLQSUIMUpdateProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMUpdate-Profile_t * *pOutput*)

Unpack update pofile for eSIM

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.68 int unpack_uim_SLQSUIMWriteRecord (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMWrite-Record_t * *pOutput*)

Write Record Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.69 int unpack_uim_SLQSUIMWriteTransparent (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMWriteTransparent_t * *pOutput*)

Write Transparent unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.70 int unpack_uim_UnblockPin (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_UnblockPin_t * *pOutput*)

Unblock Pin unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.71 int unpack_uim_UnblockPinV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_UnblockPinV2_t * *pOutput*)

Unblock Pin unpack V2

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.60.3.72 int unpack_uim_VerifyPin (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_VerifyPin_t * *pOutput*)

Verify Pin unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61 voice.h File Reference

Data Structures

- struct voice_USSInfo
- struct pack_voice_OriginateUSSD_t
- struct pack_voice_AnswerUSSD_t
- struct voice_UUSInfo
- struct voice_CUGInfo
- struct voice_calledPartySubAdd
- struct pack_voice_SLQSVoiceDialCall_t
- struct voice_alphaIDInfo
- struct voice_ccSUPSType
- struct unpack_voice_SLQSVoiceDialCall_t
- struct pack_voice_SLQSVoiceEndCall_t
- struct unpack_voice_SLQSVoiceEndCall_t
- struct voice_callFwdTypeAndPlan
- struct pack_voice_SLQSVoiceSetSUPSService_t
- struct unpack_voice_SLQSVoiceSetSUPSService_t
- struct voice_airTimer
- struct voice_roamTimer
- struct voice_prefVoiceSO

- struct pack_voice_SLQSVoiceSetConfig_t
- struct unpack_voice_SLQSVoiceSetConfig_t
- struct pack_voice_SLQSVoiceAnswerCall_t
- struct unpack_voice_SLQSVoiceAnswerCall_t
- struct voice_CLIRResp
- struct unpack_voice_SLQSVoiceGetCLIR_t
- struct voice_CLIPResp
- struct unpack_voice_SLQSVoiceGetCLIP_t
- struct pack_voice_SLQSVoiceGetCallWaiting_t
- struct unpack_voice_SLQSVoiceGetCallWaiting_t
- struct pack_voice_SLQSVoiceGetCallBarring_t
- struct unpack_voice_SLQSVoiceGetCallBarring_t
- struct pack_voice_SLQSVoiceGetCallForwardingStatus_t
- struct voice_callFWInfo
- struct voice_callFWExtInfo
- struct voice_getCallFWInfo
- struct voice_getCallFWExtInfo
- struct unpack_voice_SLQSVoiceGetCallForwardingStatus_t
- struct pack_voice_SLQSVoiceSetCallBarringPassword_t
- struct unpack_voice_SLQSVoiceSetCallBarringPassword_t
- struct pack_voice_SLQSVoiceGetCallInfo_t
- struct voice_callInfo
- struct voice_remotePartyNum
- struct voice_remotePartyName
- struct voice_connectNumInfo
- struct voice_diagInfo
- struct unpack_voice_SLQSVoiceGetCallInfo_t
- struct voice_getAllCallInformation
- struct voice_getAllCallRmtPtyNum
- struct voice_getAllCallRmtPtyName
- struct voice_allCallsUUSInfo
- struct voice_allCallsAlphaIDInfo
- struct voice_allCallsDiagInfo
- struct voice_peerNumberInfo
- struct voice_arrCallInfo
- struct voice_arrRemotePartyNum
- struct voice_arrRemotePartyName
- struct voice_arrAlertingType
- struct voice_arrUUSInfo
- struct voice_arrSvcOption
- struct voice_arrCallEndReason
- struct voice_arrAlphaID
- struct voice_arrConnectPartyNum
- struct voice_arrDiagInfo
- struct voice_arrCalledPartyNum
- struct voice_arrRedirPartyNum
- struct voice_arrAlertingPattern
- struct unpack_voice_SLQSVoiceGetAllCallInfo_t
- struct pack_voice_SLQSVoiceManageCalls_t
- struct unpack_voice_SLQSVoiceManageCalls_t
- struct voice_burstDTMFInfo
- struct voice_DTMFLengths
- struct pack_voice_SLQSVoiceBurstDTMF_t
- struct unpack_voice_SLQSVoiceBurstDTMF_t
- struct pack_voice_SLQSVoiceStartContDTMF_t

- struct unpack_voice_SLQSVoiceStartContDTMF_t
- struct pack_voice_SLQSVoiceStopContDTMF_t
- struct unpack_voice_SLQSVoiceStopContDTMF_t
- struct pack_voice_SLQSVoiceSendFlash_t
- struct unpack_voice_SLQSVoiceSendFlash_t
- struct pack_voice_SLQSVoiceSetPreferredPrivacy_t
- struct pack_voice_SLQSVoiceIndicationRegister_t
- struct pack_voice_SLQSVoiceGetConfig_t
- struct voice_curAMRConfig
- struct unpack_voice_SLQSVoiceGetConfig_t
- struct pack_voice_SLQSVoiceOrigUSSDNoWait_t
- struct pack_voice_SLQSVoiceBindSubscription_t
- struct pack_voice_SLQSVoiceALSSetLineSwitching_t
- struct pack_voice_SLQSVoiceALSSelectLine_t
- struct voice_COLPResp
- struct unpack_voice_SLQSVoiceGetCOLP_t
- struct voice_COLRResp
- struct unpack_voice_SLQSVoiceGetCOLR_t
- struct voice_CNAPResp
- struct unpack_voice_SLQSVoiceGetCNAP_t
- struct pack_voice_SLQSOriginateUSSD_t
- struct unpack_voice_SLQSOriginateUSSD_t
- struct voice_USSDNotificationNetworkInfo
- struct unpack_voice_USSDNotificationCallback_ind_t
- struct voice_SUPSInfo
- struct voice_newPwdData
- struct unpack_voice_SLQSVoiceSUPSCallback_ind_t
- struct unpack_voice_allCallStatusCallback_ind_t
- struct unpack_voice_voicePrivacyChangeCallback_ind_t
- struct voice_DTMFInfo
- struct unpack_voice_DTMFEventCallback_ind_t
- struct voice_ECTNum
- struct unpack_voice_SUPSNotificationCallback_ind_t
- struct unpack_voice_OTASPStatusCallback_ind_t
- struct voice_signalInfo
- struct voice_callerIDInfo
- struct voice_callingPartyInfo
- struct voice_calledPartyInfo
- struct voice_redirNumInfo
- struct voice_NSSAudioCtrl
- struct voice_lineCtrlInfo
- struct voice_extDispRecInfo
- struct unpack_voice_VoiceInfoRecCallback_ind_t
- struct voice_VoiceUssdNoWaitErrorTlv
- struct voice_VoiceUssdNoWaitFailCauseTlv
- struct voice_VoiceUssdNoWaitUSSINFOTlv
- struct voice_VoiceUssdNoWaitAlphaIDTlv
- struct unpack_voice_USSDNoWaitIndicationCallback_ind_t

Macros

- #define MAXVOICEUSSDLENGTH 182
- #define MAX_VOICE_CALL_NO_LEN 81
- #define MAX_VOICE_DESCRIPTION_LENGTH 255
- #define VOICE_MAX_NO_OF_CALLS 20
- #define BARRING_PASSWORD_LENGTH 4

Typedefs

- typedef unpack_result_t unpack_voice_OriginateUSSD_t
- typedef unpack_result_t unpack_voice_AnswerUSSD_t
- typedef unpack_result_t unpack_voice_CancelUSSD_t
- typedef unpack_result_t unpack_voice_SLQSVoiceSetPreferredPrivacy_t
- typedef unpack_result_t unpack_voice_SLQSVoiceIndicationRegister_t
- typedef unpack_result_t unpack_voice_SLQSVoiceOrigUSSDNoWait_t
- typedef unpack_result_t unpack_voice_SLQSVoiceBindSubscription_t
- typedef unpack_result_t unpack_voice_SLQSVoiceALSSetLineSwitching_t
- typedef unpack_result_t unpack_voice_SLQSVoiceALSSelectLine_t

Enumerations

- enum liteServiceClassInformation {
  LITE_VOICE_SUPS_SRV_CLASS_NONE = 0x00,
  LITE_VOICE_SUPS_SRV_CLASS_VOICE = 0x01,
  LITE_VOICE_SUPS_SRV_CLASS_DATA = 0x02,
  LITE_VOICE_SUPS_SRV_CLASS_FAX = 0x04,
  LITE_VOICE_SUPS_SRV_CLASS_SMS = 0x08,
  LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC = 0x10,
  LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC = 0x20,
  LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS = 0x40,
  LITE_VOICE_SUPS_SRV_CLASS_PADACCESS = 0x80 }

Functions

- int pack_voice_OriginateUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_OriginateUSSD_t *reqArg)
- int unpack_voice_OriginateUSSD (uint8_t *pResp, uint16_t respLen, unpack_voice_OriginateUSSD_t *pOutput)
- int pack_voice_AnswerUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_AnswerUSSD_t *reqArg)
- int unpack_voice_AnswerUSSD (uint8_t *pResp, uint16_t respLen, unpack_voice_AnswerUSSD_t *pOutput)
- int pack_voice_CancelUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_CancelUSSD (uint8_t *pResp, uint16_t respLen, unpack_voice_CancelUSSD_t *pOutput)
- int pack_voice_SLQSVoiceDialCall (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceDialCall_t *pReq)
- int unpack_voice_SLQSVoiceDialCall (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceDialCall_t *pOutput)
- int pack_voice_SLQSVoiceEndCall (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceEndCall_t *pReq)
- int unpack_voice_SLQSVoiceEndCall (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceEndCall_t *pOutput)
- int pack_voice_SLQSVoiceSetSUPSService (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetSUPSService_t *pReq)
- int unpack_voice_SLQSVoiceSetSUPSService (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSetSUPSService_t *pOutput)
- int pack_voice_SLQSVoiceSetConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetConfig_t *pReq)
- int unpack_voice_SLQSVoiceSetConfig (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSetConfig_t *pOutput)
- int pack_voice_SLQSVoiceAnswerCall (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceAnswerCall_t *pReq)

- int unpack_voice_SLQSVoiceAnswerCall (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceAnswerCall_t *pOutput)
- int pack_voice_SLQSVoiceGetCLIR (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCLIR (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCLIR_t *pOutput)
- int pack_voice_SLQSVoiceGetCLIP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCLIP (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCLIP_t *pOutput)
- int pack_voice_SLQSVoiceGetCallWaiting (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallWaiting_t *pReq)
- int unpack_voice_SLQSVoiceGetCallWaiting (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallWaiting_t *pOutput)
- int pack_voice_SLQSVoiceGetCallBarring (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallBarring_t *pReq)
- int unpack_voice_SLQSVoiceGetCallBarring (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallBarring_t *pOutput)
- int pack_voice_SLQSVoiceGetCallForwardingStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallForwardingStatus_t *pReq)
- int unpack_voice_SLQSVoiceGetCallForwardingStatus (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallForwardingStatus_t *pOutput)
- int pack_voice_SLQSVoiceSetCallBarringPassword (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetCallBarringPassword_t *pReq)
- int unpack_voice_SLQSVoiceSetCallBarringPassword (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSetCallBarringPassword_t *pOutput)
- int pack_voice_SLQSVoiceGetCallInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallInfo_t *pReq)
- int unpack_voice_SLQSVoiceGetCallInfo (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallInfo_t *pOutput)
- int pack_voice_SLQSVoiceGetAllCallInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetAllCallInfo (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetAllCallInfo_t *pOutput)
- int pack_voice_SLQSVoiceManageCalls (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceManageCalls_t *pReq)
- int unpack_voice_SLQSVoiceManageCalls (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceManageCalls_t *pOutput)
- int pack_voice_SLQSVoiceBurstDTMF (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceBurstDTMF_t *pReq)
- int unpack_voice_SLQSVoiceBurstDTMF (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceBurstDTMF_t *pOutput)
- int pack_voice_SLQSVoiceStartContDTMF (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceStartContDTMF_t *pReq)
- int unpack_voice_SLQSVoiceStartContDTMF (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceStartContDTMF_t *pOutput)
- int pack_voice_SLQSVoiceStopContDTMF (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceStopContDTMF_t *pReq)
- int unpack_voice_SLQSVoiceStopContDTMF (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceStopContDTMF_t *pOutput)
- int pack_voice_SLQSVoiceSendFlash (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSendFlash_t *pReq)
- int unpack_voice_SLQSVoiceSendFlash (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSendFlash_t *pOutput)
- int pack_voice_SLQSVoiceSetPreferredPrivacy (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetPreferredPrivacy_t *reqArg)
- int unpack_voice_SLQSVoiceSetPreferredPrivacy (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSetPreferredPrivacy_t *pOutput)
- int pack_voice_SLQSVoiceIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceIndicationRegister_t *reqArg)

- int unpack_voice_SLQSVoiceIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceIndicationRegister_t *pOutput)
- int pack_voice_SLQSVoiceGetConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetConfig_t *reqArg)
- int unpack_voice_SLQSVoiceGetConfig (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetConfig_t *pOutput)
- int pack_voice_SLQSVoiceOrigUSSDNoWait (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceOrigUSSDNoWait_t *reqArg)
- int unpack_voice_SLQSVoiceOrigUSSDNoWait (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceOrigUSSDNoWait_t *pOutput)
- int pack_voice_SLQSVoiceBindSubscription (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceBindSubscription_t *reqArg)
- int unpack_voice_SLQSVoiceBindSubscription (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceBindSubscription_t *pOutput)
- int pack_voice_SLQSVoiceALSSetLineSwitching (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceALSSetLineSwitching_t *reqArg)
- int unpack_voice_SLQSVoiceALSSetLineSwitching (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceALSSetLineSwitching_t *pOutput)
- int pack_voice_SLQSVoiceALSSelectLine (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceALSSelectLine_t *reqArg)
- int unpack_voice_SLQSVoiceALSSelectLine (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceALSSelectLine_t *pOutput)
- int pack_voice_SLQSVoiceGetCOLP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCOLP (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCOLP_t *pOutput)
- int pack_voice_SLQSVoiceGetCOLR (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCOLR (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCOLR_t *pOutput)
- int pack_voice_SLQSVoiceGetCNAP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCNAP (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCNAP_t *pOutput)
- int pack_voice_SLQSOriginateUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSOriginateUSSD_t *reqArg)
- int unpack_voice_SLQSOriginateUSSD (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSOriginateUSSD_t *pOutput)
- int unpack_voice_USSDNotificationCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_USSDNotificationCallback_ind_t *pOutput)
- int unpack_voice_SLQSVoiceSUPSCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSUPSCallback_ind_t *pOutput)
- int unpack_voice_allCallStatusCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_allCallStatusCallback_ind_t *pOutput)
- int unpack_voice_voicePrivacyChangeCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_voicePrivacyChangeCallback_ind_t *pOutput)
- int unpack_voice_DTMFEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_DTMFEventCallback_ind_t *pOutput)
- int unpack_voice_SUPSNotificationCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_SUPSNotificationCallback_ind_t *pOutput)
- int unpack_voice_OTASPStatusCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_OTASPStatusCallback_ind_t *pOutput)
- int unpack_voice_VoiceInfoRecCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_VoiceInfoRecCallback_ind_t *pOutput)
- int unpack_voice_USSDNoWaitIndicationCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_voice_USSDNoWaitIndicationCallback_ind_t *pOutput)

9.61.1 Detailed Description

9.61.2 S1

- 3GPP Mobile Radio Interface Layer 3 Specification : Core Network Protocols; Stage 3 (Release 7)
- 3GPP TS 24.008 V7.0.0 (2005-06)

9.61.3 Macro Definition Documentation

9.61.3.1 #define BARRING_PASSWORD_LENGTH 4

9.61.3.2 #define MAX_VOICE_CALL_NO_LEN 81

9.61.3.3 #define MAX_VOICE_DESCRIPTION_LENGTH 255

9.61.3.4 #define MAXVOICEUSSDLENGTH 182

9.61.3.5 #define VOICE_MAX_NO_OF_CALLS 20

9.61.4 Typedef Documentation

9.61.4.1 typedef unpack_result_t unpack_voice_AnswerUSSD_t

9.61.4.2 typedef unpack_result_t unpack_voice_CancelUSSD_t

9.61.4.3 typedef unpack_result_t unpack_voice_OriginateUSSD_t

9.61.4.4 typedef unpack_result_t unpack_voice_SLQSVoiceALSSelectLine_t

9.61.4.5 typedef unpack_result_t unpack_voice_SLQSVoiceALSSetLineSwitching_t

9.61.4.6 typedef unpack_result_t unpack_voice_SLQSVoiceBindSubscription_t

9.61.4.7 typedef unpack_result_t unpack_voice_SLQSVoiceIndicationRegister_t

9.61.4.8 typedef unpack_result_t unpack_voice_SLQSVoiceOrigUSSDNoWait_t

9.61.4.9 typedef unpack_result_t unpack_voice_SLQSVoiceSetPreferredPrivacy_t

9.61.5 Enumeration Type Documentation

9.61.5.1 enum liteServiceClassInformation

Service Class information

Enumerator

LITE_VOICE_SUPS_SRV_CLASS_NONE

LITE_VOICE_SUPS_SRV_CLASS_VOICE

LITE_VOICE_SUPS_SRV_CLASS_DATA

LITE_VOICE_SUPS_SRV_CLASS_FAX

LITE_VOICE_SUPS_SRV_CLASS_SMS

LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC

LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC

LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS

LITE_VOICE_SUPS_SRV_CLASS_PADACCESS

9.61.6 Function Documentation

9.61.6.1 int pack_voice_AnswerUSSD (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_AnswerUSSD_t * *reqArg*)

Responds to a USSD request from the network pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.2 int pack_voice_CancelUSSD (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Cancels an in-progress USSD operation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.3 int pack_voice_OriginateUSSD (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_OriginateUSSD_t * *reqArg*)

Initiates a USSD operation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.4 int pack_voice_SLQSOriginateUSSD (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSOriginateUSSD_t * *reqArg*)

Initiates a USSD session pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.5 int pack_voice_SLQSVoiceALSSelectLine (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceALSSelectLine_t * *reqArg*)

allows the user to select the preferred line pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.6 int pack_voice_SLQSVoiceALSSetLineSwitching (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceALSSetLineSwitching_t * *reqArg*)

sets the line switch setting on the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.7 int pack_voice_SLQSVoiceAnswerCall (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceAnswerCall_t * *pReq*)

Answers an incoming voice call pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.8 int pack_voice_SLQSVoiceBindSubscription (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceBindSubscription_t * *reqArg*)

binds a subscription type to a specific voice client ID pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.9 int pack_voice_SLQSVoiceBurstDTMF (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceBurstDTMF_t * *pReq*)

Sends a burst Dual-Tone Multi frequency (DTMF) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.10 int pack_voice_SLQSVoiceDialCall (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceDialCall_t * *pReq*)

Originates a voice call (MO call) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.11 int pack_voice_SLQSVoiceEndCall (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceEndCall_t * *pReq*)

Ends a voice call pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.12 int pack_voice_SLQSVoiceGetAllCallInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

information associated with all the calls originating or terminating pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.13 int pack_voice_SLQSVoiceGetCallBarring (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceGetCallBarring_t * *pReq*)

Status of Call Barring Supplementary Service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.14 int pack_voice_SLQSVoiceGetCallForwardingStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceGetCallForwardingStatus_t * *pReq*)

Status of Call Forwarding Supplementary Service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.15 int pack_voice_SLQSVoiceGetCallInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceGetCallInfo_t * *pReq*)

information associated with a call pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.16 int pack_voice_SLQSVoiceGetCallWaiting (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceGetCallWaiting_t * *pReq*)

Status of Call Waiting Supplementary Service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.17 int pack_voice_SLQSVoiceGetCLIP (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

status of the Calling Line Identification Presentation (CLIP) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.18 int pack_voice_SLQSVoiceGetCLIR (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

status of the Calling Line Identification Restriction (CLIR) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.19 int pack_voice_SLQSVoiceGetCNAP (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

status of the Calling Name Presentation(CNAP) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.20 int pack_voice_SLQSVoiceGetCOLP (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

status of the Connected Line Identification Presentation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.21 int pack_voice_SLQSVoiceGetCOLR (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

status of the Connected Line Identification Restriction pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.22 int pack_voice_SLQSVoiceGetConfig (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_voice_SLQSVoiceGetConfig_t * reqArg)

retrieves various configuration parameters that control the modem behavior pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.23 int pack_voice_SLQSVoiceIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceIndicationRegister_t * *reqArg*)

Sets the registration state for different QMI_VOICE indications pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.24 int pack_voice_SLQSVoiceManageCalls (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceManageCalls_t * *pReq*)

Manages the calls by using the supplementary service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.25 int pack_voice_SLQSVoiceOrigUSSDNoWait (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceOrigUSSDNoWait_t * *reqArg*)

initiates a USSD operation No Wait pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.26 int pack_voice_SLQSVoiceSendFlash (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSendFlash_t * *pReq*)

sends a simple flash message pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.27 int pack_voice_SLQSVoiceSetCallBarringPassword (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetCallBarringPassword_t * *pReq*)

Sets a Call Barring Password pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.28 int pack_voice_SLQSVoiceSetConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetConfig_t * *pReq*)

configuration parameters that control the modem behavior pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.29 int pack_voice_SLQSVoiceSetPreferredPrivacy (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetPreferredPrivacy_t * *reqArg*)

sets the voice privacy preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.30 int pack_voice_SLQSVoiceSetSUPSService (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetSUPSService_t * *pReq*)

call-independent supplementary services pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.31 int pack_voice_SLQSVoiceStartContDTMF (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceStartContDTMF_t * *pReq*)

Starts a continuous DTMF pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.32 int pack_voice_SLQSVoiceStopContDTMF (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceStopContDTMF_t * *pReq*)

Stops a continuous DTMF pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.33 int unpack_voice_allCallStatusCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_allCallStatusCallback_ind_t * *pOutput*)

Unpack all call status callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	all call status callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.34 int unpack_voice_AnswerUSSD (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_AnswerUSSD_t * *pOutput*)

Responds to a USSD request from the network unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.35 int unpack_voice_CancelUSSD (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_CancelUSSD_t * *pOutput*)

Cancels an in-progress USSD operation unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.36 int unpack_voice_DTMFEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_DTMFEventCallback_ind_t * *pOutput*)

Unpack DTMF event callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	DTMF event callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.37 int unpack_voice_OriginateUSSD (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_OriginateUSSD_t * *pOutput*)

Initiates a USSD operation unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.38 int unpack_voice_OTASPStatusCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_OTASPStatusCallback_ind_t * *pOutput*)

Unpack voice OTASP status callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	voice OTASP status callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.39 int unpack_voice_SLQSOriginateUSSD (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSOriginateUSSD_t * *pOutput*)

Initiates a USSD session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.40 int unpack_voice_SLQSVoiceALSSelectLine (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceALSSelectLine_t * *pOutput*)

allows the user to select the preferred line unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.41 int unpack_voice_SLQSVoiceALSSetLineSwitching (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceALSSetLineSwitching_t * *pOutput*)

sets the line switch setting on the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.42 int unpack_voice_SLQSVoiceAnswerCall (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceAnswerCall_t * *pOutput*)

Answers an incoming voice call unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.43 int unpack_voice_SLQSVoiceBindSubscription (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceBindSubscription_t * *pOutput*)

binds a subscription type to a specific voice client ID unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.44 int unpack_voice_SLQSVoiceBurstDTMF (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceBurstDTMF_t * *pOutput*)

Sends a burst Dual-Tone Multi frequency (DTMF) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.45 int unpack_voice_SLQSVoiceDialCall (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceDialCall_t * *pOutput*)

Originates a voice call (MO call) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.46 int unpack_voice_SLQSVoiceEndCall (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceEndCall_t * *pOutput*)

Ends a voice call unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.47 int unpack_voice_SLQSVoiceGetAllCallInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetAllCallInfo_t * *pOutput*)

information associated with all the calls originating or terminating unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.48 int unpack_voice_SLQSVoiceGetCallBarring (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallBarring_t * *pOutput*)

Status of Call Barring Supplementary Service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.49 int unpack_voice_SLQSVoiceGetCallForwardingStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallForwardingStatus_t * *pOutput*)

Status of Call Forwarding Supplementary Service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.50 int unpack_voice_SLQSVoiceGetCallInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallInfo_t * *pOutput*)

information associated with a call unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.51 int unpack_voice_SLQSVoiceGetCallWaiting (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallWaiting_t * *pOutput*)

Status of Call Waiting Supplementary Service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.52 int unpack_voice_SLQSVoiceGetCLIP (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCLIP_t * *pOutput*)

status of the Calling Line Identification Presentation (CLIP) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.53 int unpack_voice_SLQSVoiceGetCLIR (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCLIR_t * *pOutput*)

status of the Calling Line Identification Restriction (CLIR) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.54 int unpack_voice_SLQSVoiceGetCNAP (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCNAP_t * *pOutput*)

status of the Calling Name Presentation(CNAP) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.55 int unpack_voice_SLQSVoiceGetCOLP (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCOLP_t * *pOutput*)

status of the Connected Line Identification Presentation unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.56 int unpack_voice_SLQSVoiceGetCOLR (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCOLR_t * *pOutput*)

status of the Connected Line Identification Restriction unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.57 int unpack_voice_SLQSVoiceGetConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetConfig_t * *pOutput*)

retrieves various configuration parameters that control the modem behavior unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.58 int unpack_voice_SLQSVoiceIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceIndicationRegister_t * *pOutput*)

Sets the registration state for different QMI_VOICE indications unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.59 int unpack_voice_SLQSVoiceManageCalls (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceManageCalls_t * *pOutput*)

Manages the calls by using the supplementary service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.60 int unpack_voice_SLQSVoiceOrigUSSDNoWait (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceOrigUSSDNoWait_t * *pOutput*)

initiates a USSD operation No Wait unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.61 int unpack_voice_SLQSVoiceSendFlash (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSendFlash_t * *pOutput*)

sends a simple flash message unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.62 int unpack_voice_SLQSVoiceSetCallBarringPassword (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSetCallBarringPassword_t * *pOutput*)

Sets a Call Barring Password unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.63 int unpack_voice_SLQSVoiceSetConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSetConfig_t * *pOutput*)

configuration parameters that control the modem behavior unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.64 int unpack_voice_SLQSVoiceSetPreferredPrivacy (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSetPreferredPrivacy_t * *pOutput*)

sets the voice privacy preference unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.65 int unpack_voice_SLQSVoiceSetSUPSService (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSetSUPSService_t * *pOutput*)

call-independent supplementary services unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.66 int unpack_voice_SLQSVoiceStartContDTMF (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceStartContDTMF_t * *pOutput*)

Starts a continuous DTMF unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.67 int unpack_voice_SLQSVoiceStopContDTMF (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceStopContDTMF_t * *pOutput*)

Stops a continuous DTMF unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

return eQCWWAN_ERR_NULL_TLV then callID value is invalid.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.68 int unpack_voice_SLQSVoiceSUPSCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSUPSCallback_ind_t * *pOutput*)

Unpack voice sups callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	voice susp callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.69 int unpack_voice_SUPSNotificationCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SUPSNotificationCallback_ind_t * *pOutput*)

Unpack SUPS notification callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	SUPS notification callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.70 int unpack_voice_USSDNotificationCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_USSDNotificationCallback_ind_t * *pOutput*)

Unpack USSD notification callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	USSD notification unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.71 int unpack_voice_USSDNoWaitIndicationCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_USSDNoWaitIndicationCallback_ind_t * *pOutput*)

Unpack USSD No Wait callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	SUPS notification callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.72 int unpack_voice_VoiceInfoRecCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_VoiceInfoRecCallback_ind_t * *pOutput*)

Unpack SUPS notification callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	SUPS notification callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.61.6.73 int unpack_voice_voicePrivacyChangeCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_voicePrivacyChangeCallback_ind_t * *pOutput*)

Unpack voice privacy change callback indication.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	voice privacy change callback unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62 wda.h File Reference

Data Structures

- struct pack_wda_SetDataFormat_t
- struct unpack_wda_SetDataFormat_t
- struct pack_wda_GetDataFormat_t
- struct unpack_wda_GetDataFormat_t
- struct pack_wda_SetQmapSettings_t
- struct unpack_wda_SetQmapSettings_t
- struct pack_wda_GetQmapSettings_t
- struct unpack_wda_GetQmapSettings_t

Functions

- void helper_get_wda_resp_ctx (uint8_t *rsp, uint32_t len, unpack_qmi_t *pCtx)
- int pack_wda_SetDataFormat (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wda_SetDataFormat_t *reqArg)
- int unpack_wda_SetDataFormat (uint8_t *pResp, uint16_t respLen, unpack_wda_SetDataFormat_t *pOutput)
- int pack_wda_GetDataFormat (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wda_GetDataFormat_t *reqArg)
- int unpack_wda_GetDataFormat (uint8_t *pResp, uint16_t respLen, unpack_wda_GetDataFormat_t *pOutput)
- int pack_wda_SetQmapSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wda_SetQmapSettings_t *reqArg)
- int unpack_wda_SetQmapSettings (uint8_t *pResp, uint16_t respLen, unpack_wda_SetQmapSettings_t *pOutput)
- int pack_wda_GetQmapSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wda_GetQmapSettings_t *reqArg)
- int unpack_wda_GetQmapSettings (uint8_t *pResp, uint16_t respLen, unpack_wda_GetQmapSettings_t *pOutput)

9.62.1 Function Documentation

9.62.1.1 void helper_get_wda_resp_ctx (uint8_t * *rsp*, uint32_t *len*, **unpack_qmi_t** * *pCtx*)

9.62.1.2 int pack_wda_GetDataFormat (**pack_qmi_t** * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, **pack_wda_GetDataFormat_t** * *reqArg*)

Get data format pack Read data format settings for use with QMAP

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62.1.3 int pack_wda_GetQmapSettings (**pack_qmi_t** * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, **pack_wda_GetQmapSettings_t** * *reqArg*)

Get qmap settings pack Read qmap settings for use with QMAP

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62.1.4 int pack_wda_SetDataFormat (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wda_SetDataFormat_t * *reqArg*)

Set data format pack Configure data format to use with QMAP

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62.1.5 int pack_wda_SetQmapSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wda_SetQmapSettings_t * *reqArg*)

Set qmap settings pack Configure qmap settings to use with QMAP

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62.1.6 int unpack_wda_GetDataFormat (uint8_t * *pResp*, uint16_t *respLen*, unpack_wda_GetDataFormat_t * *pOutput*)

Get data format unpack Read configured data format to use with QMAP

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62.1.7 int unpack_wda_GetQmapSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wda_GetQmapSettings_t * *pOutput*)

Get qmap settings unpack Read configured qmap settings to use with QMAP

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62.1.8 int unpack_wda_SetDataFormat (uint8_t * *pResp*, uint16_t *respLen*, unpack_wda_SetDataFormat_t * *pOutput*)

Set data format unpack Configure data format to use with QMAP

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.62.1.9 int unpack_wda_SetQmapSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wda_SetQmapSettings_t * *pOutput*)

Set qmap settings unpack Configure qmap settings to use with QMAP

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63 wds.h File Reference

Data Structures

- struct LibPackQosClassID
- struct LibPackTFTIDParams
- struct LibPackGPRSRequestedQoS
- struct LibPackUMTSQoS
- struct LibPackUMTSReqQoSSigInd
- struct LibPackProfileMnc
- struct LibPackPDNThrottleTimer
- struct LibPackPCOIDList
- struct pack_wds_SLQSStartDataSession_t
- struct unpack_wds_SLQSStartDataSession_t
- struct unpack_wds_SLQSSetPacketSrvStatusCallback_t
- struct pack_wds_SLQSStopDataSession_t
- struct wds_ProfileIdentifier
- struct wds_GPRSQoS
- struct wds_PCSCFIPv4ServerAddressList
- struct wds_PCSCFFQDNAddress
- struct wds_PCSCFFQDNAddressList
- struct wds_Domain
- struct wds_DomainNameList
- struct wds_IPV6AddressInfo
- struct wds_IPV6GWAddressInfo
- struct unpack_wds_SLQSGetRuntimeSettings_t
- struct wds_currNetworkInfo
- struct unpack_wds_SLQSSetWdsEventCallback_ind_t
- struct pack_wds_SLQSSetWdsEventCallback_t
- struct pack_wds_SLQSGetRuntimeSettings_t
- struct wds_UMTSMinQoS
- struct LibPackprofile_3GPP
- struct LibPackprofile_3GPP2
- union wds_profileInfo
- struct pack_wds_SLQSCreateProfile_t

- struct PackCreateProfileOut
- struct unpack_wds_SLQSCreateProfile_t
- struct pack_wds_SLQSModifyProfile_t
- struct unpack_wds_SLQSModifyProfile_t
- struct pack_wds_SLQSGetProfileSettings_t
- struct LibpackProfile3GPP
- struct LibpackProfile3GPP2
- union unpackWdsProfileParam
- struct UnPackGetProfileSettingOut
- struct unpack_wds_SLQSGetProfileSettings_t
- struct LibpackProfile3GPPV2
- union unpackWdsProfileParamV2
- struct UnPackGetProfileSettingOutV2
- struct unpack_wds_SLQSGetProfileSettingsV2_t
- struct unpack_wds_GetSessionState_t
- struct pack_wds_GetDefaultProfile_t
- struct unpack_wds_GetDefaultProfile_t
- struct pack_wds_GetDefaultProfileV2_t
- struct unpack_wds_GetDefaultProfileV2_t
- struct unpack_wds_GetConnectionRate_t
- struct pack_wds_GetPacketStatus_t
- struct unpack_wds_GetPacketStatus_t
- struct unpack_wds_GetSessionDuration_t
- struct pack_wds_GetSessionDuration_t
- struct unpack_wds_GetSessionDurationV2_t
- struct unpack_wds_GetDormancyState_t
- struct pack_wds_GetDormancyState_t
- struct pack_wds_SLQSDeleteProfile_t
- struct unpack_wds_SLQSDeleteProfile_t
- struct pack_wds_SetDefaultProfile_t
- struct unpack_wds_SLQSGet3GPPConfigItem_t
- struct pack_wds_SLQSSet3GPPConfigItem_t
- struct unpack_wds_GetMobileIP_t
- struct pack_wds_GetMobileIP_t
- struct pack_wds_GetMobileIPProfile_t
- struct unpack_wds_GetMobileIPProfile_t
- struct currNetworkInfo
- struct unpack_wds_SLQSGetCurrDataSystemStat_t
- struct pack_wds_SLQSGetCurrDataSystemStat_t
- struct unpack_wds_GetLastMobileIPError_t
- struct pack_wds_GetLastMobileIPError_t
- struct rmTrasnferStaticsReq
- struct pack_wds_RMSetTransferStatistics_t
- struct unpack_wds_RMSetTransferStatistics_t
- struct pack_wds_SetMobileIPProfile_t
- struct unpack_wds_SetMobileIPProfile_t
- struct pack_wds_SLQSWdsSwiPDPRuntimeSettings_t
- struct ipv6AddressInfo
- struct unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t
- struct wds_transferStatInd
- struct pack_wds_SLQSGetDUNCallInfo_t
- struct connectionStatus
- struct dunchannelRate
- struct unpack_wds_SLQSGetDUNCallInfo_t
- struct qmiWDSDataBearerTechnology

- struct unpack_wds_SLQSGetDataBearerTechnology_t
- struct pack_wds_SLQSGetDataBearerTechnology_t
- struct pack_wds_SLQSSetIPFamilyPreference_t
- struct unpack_wds_SLQSSetIPFamilyPreference_t
- struct pack_wds_SetDefaultProfileNum_t
- struct pack_wds_GetDefaultProfileNum_t
- struct unpack_wds_GetDefaultProfileNum_t
- struct wdsDhcpv4ProfileId
- struct wdsDhcpv4HwConfig
- struct wdsDhcpv4Option
- struct wdsDhcpv4OptionList
- struct pack_wds_SLQSSGetDHCPv4ClientConfig_t
- struct unpack_wds_SLQSSGetDHCPv4ClientConfig_t
- struct pack_wds_GetPacketStatistics_t
- struct unpack_wds_GetPacketStatistics_t
- struct unpack_wds_GetByteTotals_t
- struct unpack_wds_SLQSGetCurrentChannelRate_t
- struct unpack_wds_SLQSSGetLoopback_t
- struct pack_wds_SLQSSSetLoopback_t
- struct wds_DataULongTlv
- struct wds_DataULongLongTlv
- struct unpack_RMTransferStatistics_ind_t
- struct pack_wds_DHCPv4ClientLeaseChange_t
- struct wds_DHCPProfileIdTlv
- struct wds_DHCPLeaseStateTlv
- struct wds_IPv4AdTlv
- struct wds_DHCPOpt
- struct wds_DHCPLeaseOptTlv
- struct unpack_wds_DHCPv4ClientLease_ind_t
- struct pack_wds_SetMobileIP_t
- struct pack_wds_SetMobileIPParameters_t
- struct pack_wds_SetAutoconnect_t
- struct unpack_wds_GetAutoconnect_t
- struct wds_TrStatInd
- struct pack_wds_SLQSWdsSetEventReport_t
- struct wds_DHCPv4ProfileId
- struct wds_DHCPv4HWConfig
- struct wds_DHCPv4Option
- struct wds_DHCPv4OptionList
- struct pack_wds_SLQSSSetDHCPv4ClientConfig_t
- struct unpack_wds_GetDataBearerTechnology_t
- struct wds_ConnStatusTlv
- struct wds_LastMdmCallEndRsnTlv
- struct wds_TXBytesOKTlv
- struct wds_RXBytesOKTlv
- struct wds_DormStatTlv
- struct wds_DataBearTechTlv
- struct wds_channelRate
- struct wds_ChannelRateTlv
- struct unpack_wds_SLQSDUNCallInfoCallBack_ind_t
- struct pack_wds_SLQSSwiProfileChangeCallback_t
- struct wds_profileChange
- struct wds_sourceOfChange
- struct unpack_wds_SLQSSwiProfileChangeCallback_Ind_t

Macros

- #define IPV6_ADDRESS_ARRAY_SIZE 8
- #define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24
- #define PACK_WDS_IPV4 4
- #define PACK_WDS_IPV6 6
- #define BYT_STAT_STAT_MASK 0X000000C0
- #define WDS_DHCP_MAX_NUM_OPTIONS 30
- #define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /∗ current max size of raw message in SDK process is 2048 ∗/
- #define WDS_TFTID_SOURCE_IP_SIZE 8
- #define WDS_PROFILE_3GPP 0
- #define WDS_PROFILE_3GPP2 1
- #define LITE_MAX_PDN_THROTTLE_TIMER 10
- #define LITE_MAX_PCOID_LIST 10
- #define MAX_WDS_NAME_ARRAY_SIZE 255

Typedefs

- typedef unpack_result_t unpack_wds_SLQSStopDataSession_t
- typedef unpack_result_t unpack_wds_SLQSSetWdsEventCallback_t
- typedef union unpackWdsProfileParam UnpackQmiProfileInfo
- typedef union unpackWdsProfileParamV2 UnpackQmiProfileInfoV2
- typedef unpack_result_t unpack_wds_SetDefaultProfile_t
- typedef unpack_result_t unpack_wds_SLQSSet3GPPConfigItem_t
- typedef unpack_result_t unpack_wds_SetDefaultProfileNum_t
- typedef unpack_result_t unpack_wds_SLQSSSetLoopback_t
- typedef struct unpack_RMTransferStatistics_ind_t unpack_wds_RMTransferStatistics_ind_t
- typedef unpack_result_t unpack_wds_DHCPv4ClientLeaseChange_t
- typedef unpack_result_t unpack_wds_SetMobileIP_t
- typedef unpack_result_t unpack_wds_SetMobileIPParameters_t
- typedef unpack_result_t unpack_wds_SetAutoconnect_t
- typedef unpack_result_t unpack_wds_SLQSWdsSetEventReport_t
- typedef unpack_result_t unpack_wds_SLQSWdsGoDormant_t
- typedef unpack_result_t unpack_wds_SLQSWdsGoActive_t
- typedef unpack_result_t unpack_wds_SLQSResetPacketStatics_t
- typedef unpack_result_t unpack_wds_SLQSSSetDHCPv4ClientConfig_t
- typedef unpack_result_t unpack_wds_SetMuxID_t
- typedef unpack_result_t unpack_wds_SLQSSwiProfileChangeCallback_t

Enumerations

- enum liteQmiDataBearerMasks { QMI_LITE_WDS_CURRENT_CALL_DB_MASK = 0x01, QMI_LITE_WDS_LAST_CALL_DB_MASK = 0x02 }

Functions

- int pack_wds_SLQSStartDataSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSStartDataSession_t *reqArg)
- int unpack_wds_SLQSStartDataSession (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSStartDataSession_t *pOutput)
- int unpack_wds_SLQSSetPacketSrvStatusCallback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetPacketSrvStatusCallback_t *pOutput)
- int pack_wds_SLQSStopDataSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSStopDataSession_t *reqArg)
- int unpack_wds_SLQSStopDataSession (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSStopDataSession_t *pOutput)
- int unpack_wds_SLQSGetRuntimeSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetRuntimeSettings_t *pOutput)
- int unpack_wds_SLQSSetWdsEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetWdsEventCallback_ind_t *pOutput)
- int unpack_wds_SLQSSetWdsEventCallback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetWdsEventCallback_t *pOutput)
- int pack_wds_SLQSSetWdsEventCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetWdsEventCallback_t *reqArg)
- int pack_wds_SLQSGetRuntimeSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetRuntimeSettings_t *reqArg)
- int pack_wds_SLQSCreateProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSCreateProfile_t *reqArg)
- int unpack_wds_SLQSCreateProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSCreateProfile_t *pOutput)
- int pack_wds_SLQSModifyProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSModifyProfile_t *reqArg)
- int unpack_wds_SLQSModifyProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSModifyProfile_t *pOutput)
- int pack_wds_SLQSGetProfileSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetProfileSettings_t *reqArg)
- int unpack_wds_SLQSGetProfileSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetProfileSettings_t *pOutput)
- int pack_wds_SLQSGetProfileSettingsV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetProfileSettings_t *reqArg)
- int unpack_wds_SLQSGetProfileSettingsV2 (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetProfileSettingsV2_t *pOutput)
- int pack_wds_GetSessionState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetSessionState (uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionState_t *pOutput)
- int pack_wds_GetDefaultProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfile_t *reqParam)
- int unpack_wds_GetDefaultProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfile_t *pOutput)
- int pack_wds_GetDefaultProfileV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfileV2_t *reqParam)
- int unpack_wds_GetDefaultProfileV2 (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfileV2_t *pOutput)
- int pack_wds_GetConnectionRate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetConnectionRate (uint8_t *pResp, uint16_t respLen, unpack_wds_GetConnectionRate_t *pOutput)
- int pack_wds_GetPacketStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetPacketStatus_t *reqParam)
- int unpack_wds_GetPacketStatus (uint8_t *pResp, uint16_t respLen, unpack_wds_GetPacketStatus_t *pOutput)

- int pack_wds_GetSessionDuration (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetSessionDuration_t *reqParam)
- int unpack_wds_GetSessionDuration (uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionDuration_t *pOutput)
- int pack_wds_GetSessionDurationV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetSessionDuration_t *reqParam)
- int unpack_wds_GetSessionDurationV2 (uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionDurationV2_t *pOutput)
- int pack_wds_GetDormancyState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDormancyState_t *reqParam)
- int unpack_wds_GetDormancyState (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDormancyState_t *pOutput)
- int pack_wds_SLQSDeleteProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSDeleteProfile_t *reqParam)
- int unpack_wds_SLQSDeleteProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSDeleteProfile_t *pOutput)
- int pack_wds_SetDefaultProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetDefaultProfile_t *reqParam)
- int unpack_wds_SetDefaultProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SetDefaultProfile_t *pOutput)
- int pack_wds_SLQSGet3GPPConfigItem (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSGet3GPPConfigItem (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGet3GPPConfigItem_t *pOutput)
- int pack_wds_SLQSSet3GPPConfigItem (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSet3GPPConfigItem_t *reqParam)
- int unpack_wds_SLQSSet3GPPConfigItem (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSet3GPPConfigItem_t *pOutput)
- int pack_wds_GetMobileIP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetMobileIP_t *pReqParam)
- int unpack_wds_GetMobileIP (uint8_t *pResp, uint16_t respLen, unpack_wds_GetMobileIP_t *pOutput)
- int pack_wds_GetMobileIPProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetMobileIPProfile_t *reqParam)
- int unpack_wds_GetMobileIPProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_GetMobileIPProfile_t *pOutput)
- int pack_wds_SLQSGetCurrDataSystemStat (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetCurrDataSystemStat_t *pReqParam)
- int unpack_wds_SLQSGetCurrDataSystemStat (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetCurrDataSystemStat_t *pOutput)
- int pack_wds_GetLastMobileIPError (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetLastMobileIPError_t *pReqParam)
- int unpack_wds_GetLastMobileIPError (uint8_t *pResp, uint16_t respLen, unpack_wds_GetLastMobileIPError_t *pOutput)
- int pack_wds_RMSetTransferStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_RMSetTransferStatistics_t *reqParam)
- int unpack_wds_RMSetTransferStatistics (uint8_t *pResp, uint16_t respLen, unpack_wds_RMSetTransferStatistics_t *pOutput)
- int pack_wds_SetMobileIPProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIPProfile_t *reqParam)
- int unpack_wds_SetMobileIPProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SetMobileIPProfile_t *pOutput)
- int pack_wds_SLQSWdsSwiPDPRuntimeSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSWdsSwiPDPRuntimeSettings_t *reqParam)
- int unpack_wds_SLQSWdsSwiPDPRuntimeSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t *pOutput)
- int pack_wds_SLQSGetDUNCallInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetDUNCallInfo_t *reqParam)

- int unpack_wds_SLQSGetDUNCallInfo (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetDUNCallInfo_t *pOutput)
- int pack_wds_SLQSGetDataBearerTechnology (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetDataBearerTechnology_t *pReqParam)
- int unpack_wds_SLQSGetDataBearerTechnology (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetDataBearerTechnology_t *pOutput)
- int pack_wds_SLQSSetIPFamilyPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetIPFamilyPreference_t *pReqParam)
- int unpack_wds_SLQSSetIPFamilyPreference (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetIPFamilyPreference_t *pOutput)
- int pack_wds_SetDefaultProfileNum (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetDefaultProfileNum_t *pReqParam)
- int unpack_wds_SetDefaultProfileNum (uint8_t *pResp, uint16_t respLen, unpack_wds_SetDefaultProfileNum_t *pOutput)
- int pack_wds_GetDefaultProfileNum (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfileNum_t *pReqParam)
- int unpack_wds_GetDefaultProfileNum (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfileNum_t *pOutput)
- int pack_wds_SLQSSGetDHCPv4ClientConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSGetDHCPv4ClientConfig_t *pReq)
- int unpack_wds_SLQSSGetDHCPv4ClientConfig (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSGetDHCPv4ClientConfig_t *pOutput)
- int pack_wds_GetPacketStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetPacketStatistics_t *pReq)
- int unpack_wds_GetPacketStatistics (uint8_t *pResp, uint16_t respLen, unpack_wds_GetPacketStatistics_t *pOutput)
- int pack_wds_GetByteTotals (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetByteTotals (uint8_t *pResp, uint16_t respLen, unpack_wds_GetByteTotals_t *pOutput)
- int pack_wds_SLQSGetCurrentChannelRate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSGetCurrentChannelRate (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetCurrentChannelRate_t *pOutput)
- int pack_wds_SLQSSGetLoopback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSSGetLoopback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSGetLoopback_t *pOutput)
- int pack_wds_SLQSSSetLoopback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSSetLoopback_t *reqArg)
- int unpack_wds_SLQSSSetLoopback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSSetLoopback_t *pOutput)
- int unpack_wds_RMTransferStatistics_ind (uint8_t *pResp, uint16_t respLen, unpack_RMTransferStatistics_ind_t *pOutput)
- int pack_wds_DHCPv4ClientLeaseChange (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_DHCPv4ClientLeaseChange_t *reqArg)
- int unpack_wds_DHCPv4ClientLeaseChange (uint8_t *pResp, uint16_t respLen, unpack_wds_DHCPv4ClientLeaseChange_t *pOutput)
- int unpack_wds_DHCPv4ClientLease_ind (uint8_t *pResp, uint16_t respLen, unpack_wds_DHCPv4ClientLease_ind_t *pOutput)
- int pack_wds_SetMobileIP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIP_t *reqArg)
- int unpack_wds_SetMobileIP (uint8_t *pResp, uint16_t respLen, unpack_wds_SetMobileIP_t *pOutput)
- int pack_wds_SetMobileIPParameters (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIPParameters_t *reqArg)
- int unpack_wds_SetMobileIPParameters (uint8_t *pResp, uint16_t respLen, unpack_wds_SetMobileIPParameters_t *pOutput)
- int pack_wds_SetAutoconnect (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetAutoconnect_t *reqArg)

- int unpack_wds_SetAutoconnect (uint8_t *pResp, uint16_t respLen, unpack_wds_SetAutoconnect_t *pOutput)
- int pack_wds_GetAutoconnect (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetAutoconnect (uint8_t *pResp, uint16_t respLen, unpack_wds_GetAutoconnect_t *pOutput)
- int pack_wds_SLQSWdsSetEventReport (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSWdsSetEventReport_t *reqArg)
- int unpack_wds_SLQSWdsSetEventReport (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSWdsSetEventReport_t *pOutput)
- int pack_wds_SLQSWdsGoDormant (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSWdsGoDormant (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSWdsGoDormant_t *pOutput)
- int pack_wds_SLQSWdsGoActive (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSWdsGoActive (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSWdsGoActive_t *pOutput)
- int pack_wds_SLQSResetPacketStatics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSResetPacketStatics (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSResetPacketStatics_t *pOutput)
- int pack_wds_SLQSSSetDHCPv4ClientConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSSetDHCPv4ClientConfig_t *reqArg)
- int unpack_wds_SLQSSSetDHCPv4ClientConfig (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSSetDHCPv4ClientConfig_t *pOutput)
- int pack_wds_GetDataBearerTechnology (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetDataBearerTechnology (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDataBearerTechnology_t *pOutput)
- int pack_wds_SetMuxID (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pMuxID)
- int unpack_wds_SetMuxID (uint8_t *pResp, uint16_t respLen, unpack_wds_SetMuxID_t *pOutput)
- int unpack_wds_SLQSDUNCallInfoCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSDUNCallInfoCallBack_ind_t *pOutput)
- int pack_wds_SLQSSwiProfileChangeCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSwiProfileChangeCallback_t *reqArg)
- int unpack_wds_SLQSSwiProfileChangeCallback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSwiProfileChangeCallback_t *pOutput)
- int unpack_wds_SLQSSwiProfileChangeCallback_Ind (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSwiProfileChangeCallback_Ind_t *pOutput)

9.63.1 Detailed Description

9.63.2 SO Mask

- SO mask to indicate the service option or type of application.
  - An SO mask value of zero indicates that this field is ignored.
  - Values:
    - 0x00 - DONT_CARE
  - CDMA 1X SO mask:
    - 0x01 - CDMA_1X_IS95
    - 0x02 - CDMA_1X_IS2000
    - 0x04 - CDMA_1X_IS2000_REL_A
  - CDMA EV-DO Rev 0 SO mask:
    - 0x01 - DPA
  - CDMA EV-DO Rev A SO mask:
    - 0x01 - DPA

    - 0x02 - MFPA
    - 0x04 - EMPA
    - 0x08 - EMPA_EHRPD
  - CDMA EV-DO Rev B SO mask:
    - 0x01 - DPA
    - 0x02 - MFPA
    - 0x04 - EMPA
    - 0x08 - EMPA_EHRPD
    - 0x10 - MMPA
    - 0x20 - MMPA_EHRPD

9.63.3 RAT Mask

- RAT mask to indicate the type of technology. A RAT mask value of zero indicates that this field is ignored. Values:
  - 0x00 - DONT_CARE
  - 0x8000 - NULL_BEARER
- CDMA RAT mask:
  - 0x01 - CDMA_1X
  - 0x02 - EVDO_REV0
  - 0x04 - EVDO_REVA
  - 0x08 - EVDO_REVB
  - 0x10 - EHRPD
  - 0x20 - FMC
- UMTS RAT mask:
  - 0x01 - WCDMA
  - 0x02 - GPRS
  - 0x04 - HSDPA
  - 0x08 - HSUPA
  - 0x10 - EDGE
  - 0x20 - LTE
  - 0x40 - HSDPA+
  - 0x80 - DC_HSDPA+
  - 0x100 - 64_QAM
  - 0x200 - TD-SCDMA

9.63.4 Macro Definition Documentation

9.63.4.1 #define BYT_STAT_STAT_MASK 0X000000C0

9.63.4.2 #define IPV6_ADDRESS_ARRAY_SIZE 8

9.63.4.3 #define LITE_MAX_PCOID_LIST 10

9.63.4.4 #define LITE_MAX_PDN_THROTTLE_TIMER 10

9.63.4.5 #define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24

9.63.4.6 #define MAX_WDS_NAME_ARRAY_SIZE 255

9.63.4.7 #define PACK_WDS_IPV4 4

9.63.4.8 #define PACK_WDS_IPV6 6

9.63.4.9 #define WDS_DHCP_MAX_NUM_OPTIONS 30

9.63.4.10 #define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /* current max size of raw message in SDK process is 2048 */

9.63.4.11 #define WDS_PROFILE_3GPP 0

9.63.4.12 #define WDS_PROFILE_3GPP2 1

9.63.4.13 #define WDS_TFTID_SOURCE_IP_SIZE 8

9.63.5 Typedef Documentation

9.63.5.1 typedef unpack_result_t unpack_wds_DHCPv4ClientLeaseChange_t

9.63.5.2 typedef struct unpack_RMTransferStatistics_ind_t unpack_wds_RMTransferStatistics_ind_t

9.63.5.3 typedef unpack_result_t unpack_wds_SetAutoconnect_t

9.63.5.4 typedef unpack_result_t unpack_wds_SetDefaultProfile_t

9.63.5.5 typedef unpack_result_t unpack_wds_SetDefaultProfileNum_t

9.63.5.6 typedef unpack_result_t unpack_wds_SetMobileIP_t

9.63.5.7 typedef unpack_result_t unpack_wds_SetMobileIPParameters_t

9.63.5.8 typedef unpack_result_t unpack_wds_SetMuxID_t

9.63.5.9 typedef unpack_result_t unpack_wds_SLQSResetPacketStatics_t

9.63.5.10 typedef unpack_result_t unpack_wds_SLQSSet3GPPConfigItem_t

9.63.5.11 typedef unpack_result_t unpack_wds_SLQSSetWdsEventCallback_t

9.63.5.12 typedef unpack_result_t unpack_wds_SLQSSSetDHCPv4ClientConfig_t

9.63.5.13 typedef unpack_result_t unpack_wds_SLQSSSetLoopback_t

9.63.5.14 typedef unpack_result_t unpack_wds_SLQSStopDataSession_t

9.63.5.15 typedef unpack_result_t unpack_wds_SLQSSwiProfileChangeCallback_t

9.63.5.16 typedef unpack_result_t unpack_wds_SLQSWdsGoActive_t

9.63.5.17 typedef unpack_result_t unpack_wds_SLQSWdsGoDormant_t

9.63.5.18 typedef unpack_result_t unpack_wds_SLQSWdsSetEventReport_t

9.63.5.19 typedef union unpackWdsProfileParam UnpackQmiProfileInfo

This union WdsProfileParam consists of Profile3GPP and Profile3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile • See LibpackProfile3GPP
SlqsProfile3GP-P2	3GPP2 profile • See LibpackProfile3GPP2

9.63.5.20 typedef union unpackWdsProfileParamV2 UnpackQmiProfileInfoV2

This union WdsProfileParam consists of Profile3GPP and Profile3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile • See LibpackProfile3GPP
SlqsProfile3GP-P2	3GPP2 profile • See LibpackProfile3GPP2

9.63.6 Enumeration Type Documentation

9.63.6.1 enum liteQmiDataBearerMasks

Bit mask values to indicate the presence of data bearer information for the current and last data calls

Enumerator

QMI_LITE_WDS_CURRENT_CALL_DB_MASK

QMI_LITE_WDS_LAST_CALL_DB_MASK

9.63.7 Function Documentation

9.63.7.1 int pack_wds_DHCPv4ClientLeaseChange (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_DHCPv4ClientLeaseChange_t * *reqArg*)

DHCPv4 lease state changes pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.2 int pack_wds_GetAutoconnect (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets auto connect data session setting pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.3 int pack_wds_GetByteTotals (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get Rx/Tx byte counts since the start of the last packet data session pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.4 int pack_wds_GetConnectionRate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get connection rate pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.63.7.5 int pack_wds_GetDataBearerTechnology (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get current data bearer technology pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.6 int pack_wds_GetDefaultProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetDefaultProfile_t * *reqParam*)

get default profile pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.7 int pack_wds_GetDefaultProfileNum (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetDefaultProfileNum_t * *pReqParam*)

get default profile number pack

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.8 int pack_wds_GetDefaultProfileV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetDefaultProfileV2_t * *reqParam*)

get default profile pack V2.

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.9 int pack_wds_GetDormancyState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetDormancyState_t * *reqParam*)

get dormancy state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.63.7.10 int pack_wds_GetLastMobileIPError (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetLastMobileIPError_t * *pReqParam*)

get current data system pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.63.7.11 int pack_wds_GetMobileIP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetMobileIP_t * *pReqParam*)

get mobile ip mode pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.63.7.12 int pack_wds_GetMobileIPProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetMobileIPProfile_t * *reqParam*)

get mobile ip profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.13 int pack_wds_GetPacketStatistics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetPacketStatistics_t * *pReq*)

gets current packet transfer counter values pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.14 int pack_wds_GetPacketStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetPacketStatus_t * *reqParam*)

get packet status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.15 int pack_wds_GetSessionDuration (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetSessionDuration_t * *reqParam*)

get session duration pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.63.7.16 int pack_wds_GetSessionDurationV2 (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_GetSessionDuration_t * reqParam)

get session duration pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.63.7.17 int pack_wds_GetSessionState (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

get session state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.18 int pack_wds_RMSetTransferStatistics (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_RMSetTransferStatistics_t * reqParam)

RM set transfer statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Note

PDN Specific: No

9.63.7.19 int pack_wds_SetAutoconnect (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetAutoconnect_t * *reqArg*)

Auto connect data session parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.20 int pack_wds_SetDefaultProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetDefaultProfile_t * *reqParam*)

set default profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.21 int pack_wds_SetDefaultProfileNum (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetDefaultProfileNum_t * *pReqParam*)

set default profile number pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.22 int pack_wds_SetMobileIP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetMobileIP_t * *reqArg*)

Sets the current mobile IP setting pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.23 int pack_wds_SetMobileIPParameters (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetMobileIPParameters_t * *reqArg*)

Sets the specified mobile IP parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.24 int pack_wds_SetMobileIPProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetMobileIPProfile_t * *reqParam*)

set mobile ip profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.25 int pack_wds_SetMuxID (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint8_t * *pMuxID*)

Set MUX ID pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pMuxID*	MUX ID. • 0x80 to 0x88

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.26 int pack_wds_SLQSCreateProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSCreateProfile_t * *reqArg*)

Create Profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.27 int pack_wds_SLQSDeleteProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSDeleteProfile_t * *reqParam*)

delete stored profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.28 int pack_wds_SLQSGet3GPPConfigItem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get 3Gpp config items pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.29 int pack_wds_SLQSGetCurrDataSystemStat (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetCurrDataSystemStat_t * *pReqParam*)

get current data system pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.30 int pack_wds_SLQSGetCurrentChannelRate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get current Tx/Rx channel bitrate of the current packet data pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.31 int pack_wds_SLQSGetDataBearerTechnology (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetDataBearerTechnology_t * *pReqParam*)

get data bearer technology pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.32 int pack_wds_SLQSGetDUNCallInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetDUNCallInfo_t * *reqParam*)

get dun call info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.33 int pack_wds_SLQSGetProfileSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetProfileSettings_t * *reqArg*)

Get Profile Settings pack

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.34 int pack_wds_SLQSGetProfileSettingsV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetProfileSettings_t * *reqArg*)

Get Profile Settings pack V2

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.35 int pack_wds_SLQSGetRuntimeSettings (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSGetRuntimeSettings_t * reqArg)

get runtime settings pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.36 int pack_wds_SLQSModifyProfile (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSModifyProfile_t * reqArg)

Modify Profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.37 int pack_wds_SLQSResetPacketStatics (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Reset packet data transfer statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.38 int pack_wds_SLQSSet3GPPConfigItem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSet3GPPConfigItem_t * *reqParam*)

set 3Gpp config items pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.39 int pack_wds_SLQSSetIPFamilyPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSetIPFamilyPreference_t * *pReqParam*)

Set IP Family Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.40 int pack_wds_SLQSSetWdsEventCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSSetWdsEventCallback_t * reqArg)

set event callback pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.63.7.41 int pack_wds_SLQSSGetDHCPv4ClientConfig (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSSGetDHCPv4ClientConfig_t * pReq)

get DHCPv4 Client Config pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.42 int pack_wds_SLQSSGetLoopback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

get the value of loopback mode and multiplier pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.43 int pack_wds_SLQSSSetDHCPv4ClientConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSSetDHCPv4ClientConfig_t * *reqArg*)

Gets the DHCP Client V4 Configuration pack

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

see qmerrno.h for eQCWWAN_xxx error values

9.63.7.44 int pack_wds_SLQSSSetLoopback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSSetLoopback_t * *reqArg*)

Enable/disable Data Loopback Mode and set the value of loopback multiplier pack

Parameters

in, out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.45 int pack_wds_SLQSStartDataSession (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSStartDataSession_t * reqArg)

Start data session

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.46 int pack_wds_SLQSStopDataSession (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSStopDataSession_t * reqArg)

stop data session pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.47 int pack_wds_SLQSSwiProfileChangeCallback (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSSwiProfileChangeCallback_t * reqArg)

Gets profile change information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.48 int pack_wds_SLQSWdsGoActive (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the device into Active state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.49 int pack_wds_SLQSWdsGoDormant (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the device into dormant state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.50 int pack_wds_SLQSWdsSetEventReport (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSWdsSetEventReport_t * *reqArg*)

Sets the event report parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.51 int pack_wds_SLQSWdsSwiPDPRuntimeSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSWdsSwiPDPRuntimeSettings_t * *reqParam*)

swi pdp runtime settings pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.63.7.52 int unpack_wds_DHCPv4ClientLease_ind (uint8_t * *pResp*, uint16_t *respLen*, **unpack_wds_DHCPv4ClientLease_ind_t** * *pOutput*)

DHCP lease state has changed indication unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.53 int unpack_wds_DHCPv4ClientLeaseChange (uint8_t * *pResp*, uint16_t *respLen*, **unpack_wds_DHCPv4ClientLeaseChange_t** * *pOutput*)

DHCPv4 lease state changes unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.54 int unpack_wds_GetAutoconnect (uint8_t * *pResp*, uint16_t *respLen*, **unpack_wds_GetAutoconnect_t** * *pOutput*)

Gets auto connect data session setting unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.55 int unpack_wds_GetByteTotals (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetByteTotals_t * *pOutput*)

get Rx/Tx byte counts since the start of the last packet data session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.56 int unpack_wds_GetConnectionRate (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetConnectionRate_t * *pOutput*)

get connection rate unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.57 int unpack_wds_GetDataBearerTechnology (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDataBearerTechnology_t * *pOutput*)

get current data bearer technology unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.58 int unpack_wds_GetDefaultProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDefaultProfile_t * *pOutput*)

get default profile unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.59 int unpack_wds_GetDefaultProfileNum (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDefaultProfile-Num_t * *pOutput*)

get default profile number unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.60 int unpack_wds_GetDefaultProfileV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDefaultProfileV2_t * *pOutput*)

get default profile unpack V2.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.61 int unpack_wds_GetDormancyState (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDormancyState_t * *pOutput*)

get dormancy state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.62 int unpack_wds_GetLastMobileIPError (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetLastMobileIP-Error_t * *pOutput*)

get current data system unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.63 int unpack_wds_GetMobileIP (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetMobileIP_t * *pOutput*)

get mobile ip mode unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.64 int unpack_wds_GetMobileIPProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetMobileIPProfile_t * *pOutput*)

get mobile ip profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.65 int unpack_wds_GetPacketStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetPacketStatistics_t * *pOutput*)

gets current packet transfer counter values unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.66 int unpack_wds_GetPacketStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetPacketStatus_t * *pOutput*)

get packet status unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.67 int unpack_wds_GetSessionDuration (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetSessionDuration_t * *pOutput*)

get session duration unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.68 int unpack_wds_GetSessionDurationV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetSessionDurationV2_t * *pOutput*)

get session duration unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.69 int unpack_wds_GetSessionState (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetSessionState_t * *pOutput*)

get session state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.70 int unpack_wds_RMSetTransferStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_RMSetTransferStatistics_t * *pOutput*)

RM set transfer statistics unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.71 int unpack_wds_RMTransferStatistics_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_RMTransferStatistics_ind_t * *pOutput*)

RM transfer statistics indication unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.72 int unpack_wds_SetAutoconnect (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetAutoconnect_t * *pOutput*)

Auto connect data session parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.73 int unpack_wds_SetDefaultProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetDefaultProfile_t * *pOutput*)

set default profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.74 int unpack_wds_SetDefaultProfileNum (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetDefaultProfile-Num_t * *pOutput*)

set default profile number unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.75 int unpack_wds_SetMobileIP (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetMobileIP_t * *pOutput*)

Sets the current mobile IP setting unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.76 int unpack_wds_SetMobileIPParameters (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetMobileIP-Parameters_t * *pOutput*)

Sets the specified mobile IP parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.77 int unpack_wds_SetMobileIPProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetMobileIPProfile_t * *pOutput*)

set mobile ip profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.78 int unpack_wds_SetMuxID (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetMuxID_t * *pOutput*)

Set MUX ID unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.79 int unpack_wds_SLQSCreateProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSCreateProfile_t * *pOutput*)

Create Profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.80 int unpack_wds_SLQSDeleteProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSDeleteProfile_t * *pOutput*)

delete stored profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.81 int unpack_wds_SLQSDUNCallInfoCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSDUNCallInfoCallBack_ind_t * *pOutput*)

unpack DUN call info indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.82 int unpack_wds_SLQSGet3GPPConfigItem (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGet3GPPConfigItem_t * *pOutput*)

get 3GPP config items unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.83 int unpack_wds_SLQSGetCurrDataSystemStat (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetCurrDataSystemStat_t * *pOutput*)

get current data system unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.84 int unpack_wds_SLQSGetCurrentChannelRate (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetCurrentChannelRate_t * *pOutput*)

get current Tx/Rx channel bitrate of the current packet data unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.85 int unpack_wds_SLQSGetDataBearerTechnology (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetDataBearerTechnology_t * *pOutput*)

get data bearer technology unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.86 int unpack_wds_SLQSGetDUNCallInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetDUNCallInfo_t * *pOutput*)

get dun call info unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.87 int unpack_wds_SLQSGetProfileSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetProfileSettings_t * *pOutput*)

get session state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.88 int unpack_wds_SLQSGetProfileSettingsV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetProfileSettingsV2_t * *pOutput*)

get profile settings unpack V2

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.89 int unpack_wds_SLQSGetRuntimeSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetRuntimeSettings_t * *pOutput*)

get runtime settings unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.90 int unpack_wds_SLQSModifyProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSModifyProfile_t * *pOutput*)

Modify Profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.91 int unpack_wds_SLQSResetPacketStatics (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSResetPacketStatics_t * *pOutput*)

Reset packet data transfer statistics unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.92 int unpack_wds_SLQSSet3GPPConfigItem (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSet3GPPConfigItem_t * *pOutput*)

set 3GPP config items unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

9.63.7.93 int unpack_wds_SLQSSetIPFamilyPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSetIPFamilyPreference_t * *pOutput*)

Set IP Family Preference unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.94 int unpack_wds_SLQSSetPacketSrvStatusCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSetPacketSrvStatusCallback_t * *pOutput*)

set packet srv status callback unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.95 int unpack_wds_SLQSSetWdsEventCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSetWdsEventCallback_t * *pOutput*)

set event callback unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.96 int unpack_wds_SLQSSetWdsEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSetWdsEventCallback_ind_t * *pOutput*)

set event callback unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.97 int unpack_wds_SLQSSGetDHCPv4ClientConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSGetDHCPv4ClientConfig_t * *pOutput*)

get DHCPv4 Client Config unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.98 int unpack_wds_SLQSSGetLoopback (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSGetLoopback_t * *pOutput*)

get the value of loopback mode and multiplier unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.99 int unpack_wds_SLQSSSetDHCPv4ClientConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSSetDHCPv4ClientConfig_t * *pOutput*)

Gets the DHCP Client V4 Configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.100 int unpack_wds_SLQSSSetLoopback (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSSetLoopback_t * *pOutput*)

Enable/disable Data Loopback Mode and set the value of loopback multiplier unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.101 int unpack_wds_SLQSStartDataSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSStartDataSession_t * *pOutput*)

start data session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.102 int unpack_wds_SLQSStopDataSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSStopDataSession_t * *pOutput*)

stop data session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.103 int unpack_wds_SLQSSwiProfileChangeCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSwiProfileChangeCallback_t * *pOutput*)

unpack profile change info

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.104 int unpack_wds_SLQSSwiProfileChangeCallback_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSwiProfileChangeCallback_Ind_t * *pOutput*)

unpack profile change indication

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.105 int unpack_wds_SLQSWdsGoActive (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSWdsGoActive_t * *pOutput*)

Gets the device into Active state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.106 int unpack_wds_SLQSWdsGoDormant (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSWdsGoDormant_t * *pOutput*)

Gets the device into dormant state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.107 int unpack_wds_SLQSWdsSetEventReport (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSWdsSetEventReport_t * *pOutput*)

Sets the event report parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.63.7.108 int unpack_wds_SLQSWdsSwiPDPRuntimeSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t * *pOutput*)

get current data system unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Index

_3gppRelease
    unpack_wds_SLQSGet3GPPConfigItem_t, 1166
_litefw_FirmwareFileInfo, 39
    carrierStr, 40
    fullPath, 40
    headerType, 40
    imageMask, 40
    imageType, 40
    modelIdStr, 40
    packageIdStr, 40
    partNoStr, 40
    priVersionStr, 40
    releaseDate, 40
    skuStr, 40
    versionStr, 40
_litefw_FirmwareInfo_, 40
    szCarrier_str, 41
    szCarrierPriversion_str, 41
    szFwversion_str, 41
    szModelid_str, 41
    szPackageid_str, 41
    szSku_str, 41
_litefw_FirmwarePartNo_, 41
    szPartno_str, 42

AAASPI
    unpack_wds_GetMobileIPProfile_t, 1155
AAAState
    unpack_wds_GetMobileIPProfile_t, 1155
ACT_CODE_MAX_SIZE
    dms.h, 1306
AIDTlv
    unpack_voice_USSDNoWaitIndicationCallback_-
        ind_t, 1130
ALS
    voice_getAllCallInformation, 1235
AMSSString
    unpack_dms_GetFirmwareRevision_t, 754
    unpack_dms_GetFirmwareRevisions_t, 756
AMTlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
AOATlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
AOPTlv
    NASQmiCbkNasSystemSelPrefInd, 402
APNName
    unpack_wds_SLQSGetRuntimeSettings_t, 1176
AWMTlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
AWOATlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
accelTemp
    pack_loc_SLQSLOCInjectSensorData_t, 456
acceleroData
    pack_loc_SLQSLOCInjectSensorData_t, 456
acceleroTimeSrc
    pack_loc_SLQSLOCInjectSensorData_t, 456
AccessMac
    nas_protocolSubtypeElement, 322
accolc
    pack_nas_SetACCOLC_t, 465
accuracy
    loc_HorAccuracyLvlTlv, 172
    pack_pds_SetPDSDefaults_t, 519
ackIndicator
    sMSTransferRouteMTMessageInfo, 676
acqOrdeLen
    nas_acqOrderPref, 193
    nas_AcqOrderPrefTlv, 194
AcqOrderLen
    NASAcqOrderPrefTlv, 386
AcqOrderPref
    NASAcqOrderPrefTlv, 386
acroamsetting
    pack_wds_SetAutoconnect_t, 616
acsetting
    pack_wds_SetAutoconnect_t, 616
actCode
    pack_dms_ActivateAutomatic_t, 416
ActSetCnt
    nas_NetworkStat1x, 292
ActSetPilotPN
    nas_ActPilotPNElement, 195
ActSetPilotPNStrength
    nas_ActPilotPNElement, 195
action
    pack_nas_InitiateDomainAttach_t, 461
activated_ind
    sms_qaQmi3GPP2BroadcastCfgInfo, 665
    sms_qaQmi3GPPBroadcastCfgInfo, 665
activationState
    pack_dms_SetActivationStatusCallback_t, 417
activationStatus
    dms_ActivationStatusTlv, 61
ActivationStatusTlv
    unpack_dms_SetEventReport_ind_t, 772
activeBand
    nas_RfBandInfoExtFormatElements, 331
    nas_RFBandInfoExtTlv, 332

nas_RFBandInfoTlv, 334
activeBandClass
nas_RFBandInfoElements, 330
nas_RFInfoTlv, 340
RFBandInfoElements, 652
activeChannel
nas_RFBandInfoElements, 330
nas_RfBandInfoExtFormatElements, 331
nas_RFBandInfoExtTlv, 333
nas_RFBandInfoTlv, 334
nas_RFInfoTlv, 340
RFBandInfoElements, 652
activeInd
sms_messageWaitingInfoContent, 663
ActiveStatus
voice_CLIPResp, 1225
voice_CLIRResp, 1226
voice_CNAPResp, 1227
voice_COLPResp, 1227
voice_COLRResp, 1228
ActiveTechPref
unpack_nas_GetNetworkPreference_t, 882
ActiveTimer
unpack_dms_PSMCfgChange_ind_t, 769
activeTimer
dms_PSMActiveTimerTlv, 71
ActiveTimerInd
dms_PSMActiveTimerIndTlv, 70
additionalRecord
uim_additionalReadResult, 688
additionalRecordLen
uim_additionalReadResult, 688
addr
unpack_qos_IPv4Addr_t, 989
unpack_qos_IPv6Addr_t, 990
address
loc_IPv4Info, 174
loc_IPv6Info, 175
loc_urlAddr, 183
unpack_wds_GetMobileIPProfile_t, 1155
addressSize
unpack_sms_GetSMSCAddress_t, 1014
aggDlBW
nas_PhyCaAggDlBW, 310
aid
uim_refreshevent, 716
uim_sessionInformation, 719
uim_UIMSessionInformation, 725
aidLength
appStats, 45
uim_appStatus, 690
uim_refreshevent, 717
uim_sessionInformation, 719
uim_UIMSessionInformation, 726
aidVal
appStats, 45
uim_appStatus, 690
aidingIndicatorMask
loc_sensorDataUsage, 180
airTimerValue
voice_airTimer, 1203
alertPitch
voice_signalInfo, 1247
alertingPattern
voice_arrAlertingPattern, 1206
AlertingType
voice_arrAlertingType, 1207
alertmsg
unpack_omaDmConfigTlv_t, 970
alertmsglength
unpack_omaDmConfigTlv_t, 971
allCallsAlphaIDInfoArr
voice_arrAlphaID, 1207
AllCallsUUSInfo
voice_arrUUSInfo, 1213
Alpha_Identifier
voice_VoiceUssdNoWaitAlphaIDTlv, 1251
alphaDcs
voice_alphaIDInfo, 1205
AlphaID
cat_AlPhaIdentifierTlv, 53
AlphaIDInfo
voice_allCallsAlphaIDInfo, 1204
alphaIDLen
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
AlphaIDLength
cat_AlPhaIdentifierTlv, 53
alphaLen
voice_alphaIDInfo, 1205
alphaText
voice_alphaIDInfo, 1205
altSrcInfo_t, 42
coverage, 43
linkage, 43
source, 43
Altitude
unpack_pds_SLQSGetGPSStateInfo_t, 986
altitudeAssumed
unpack_loc_GnssSvInfo_Ind_t, 856
altitudeSrcInfo
pack_loc_SLQSLOCInjectPosition_t, 454
altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 454
altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 454
ambr_dl
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 995
ambr_dl_ext
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 995
ambr_dl_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 995
ambr_ul

unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 995
ambr_ul_ext
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 995
ambr_ul_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 995
amrMode
ims_AMRModeInfo, 98
amrOctAlgn
ims_AMROctAlgnInfo, 98
amrWBEnable
ims_EnabAMRWBInfo, 100
amrWBMode
ims_AMRWBModeInfo, 99
amrWBOctAlgn
ims_AMRWBOctAlgnInfo, 99
amssSize
unpack_dms_GetFirmwareRevision_t, 754
unpack_dms_GetFirmwareRevisions_t, 756
apdoxypages.c, 1277
Apn_avail_sys_info_len
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
apn_avail_sys_len
DsdApnAvailableSystemInfo_t, 81
Apn_info_len
unpack_dsd_SLQSGetDsdUIInfo_t, 811
unpack_dsd_SLQSUIInfo_Ind_t, 815
apn_name
DsdApnAvailableSystemInfo_t, 81
DsdApnCurRecPreferredSystemInfo_t, 83
DsdUIApnAvailableSystemInfo_t, 88
apn_name_len
DsdApnAvailableSystemInfo_t, 81
DsdApnCurRecPreferredSystemInfo_t, 83
DsdUIApnAvailableSystemInfo_t, 88
Apn_pref_sys_len
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
apn_type_mask
DsdUIApnAvailableSystemInfo_t, 88
apnId
pack_qos_SLQSQosSwiReadApnExtraParams_t, 529
pack_qos_SLQSQosSwiReadDataStats_t, 529
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 995
unpack_qos_SLQSQosSwiReadDataStats_t, 997
apnName
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1191
apnname
unpack_wds_GetDefaultProfile_t, 1147
unpack_wds_GetDefaultProfileV2_t, 1151
apnsize
unpack_wds_GetDefaultProfile_t, 1147
unpack_wds_GetDefaultProfileV2_t, 1151
appNameLength
loc_LocApplicationInfo, 176
appProviderLength
loc_LocApplicationInfo, 176
appState
appStats, 45
uim_appStatus, 690
appStats, 43
aidLength, 45
aidVal, 45
appState, 45
appType, 45
persoFeature, 45
persoRetries, 45
persoState, 45
persoUnblockRetries, 45
pin1Retries, 46
pin1State, 46
pin2Retries, 46
pin2State, 46
puk1Retries, 46
puk2Retries, 46
univPin, 46
AppStatus
slotInf, 657
uim_slotInfo, 722
appType
appStats, 45
uim_appStatus, 691
appVersionLength
loc_LocApplicationInfo, 176
appVersionValid
loc_LocApplicationInfo, 176
Application
unpack_nas_GetCDMANetworkParameters_t, 877
appversion_str
unpack_dms_GetFirmwareInfo_t, 753
arfcn
nas_GERANInfo, 231
nas_gsmCellInfo, 234
arrCallInfomation
unpack_voice_allCallStatusCallback_ind_t, 1089
arrfileInfo
uim_refreshevent, 717
uim_registerRefresh, 717
atrValue
uim_physlotInfo, 707
atrValueLen
uim_physlotInfo, 707
audio.h, 1277
pack_audio_SLQSGetAudioPathConfig, 1278
pack_audio_SLQSGetAudioProfile, 1278
pack_audio_SLQSGetAudioVolTLBConfig, 1279
pack_audio_SLQSSetAudioPathConfig, 1279
pack_audio_SLQSSetAudioProfile, 1280

pack_audio_SLQSSetAudioVolTLBConfig, 1280
unpack_audio_SLQSGetAudioPathConfig, 1281
unpack_audio_SLQSGetAudioProfile, 1281
unpack_audio_SLQSGetAudioVolTLBConfig, 1281
unpack_audio_SLQSSetAudioPathConfig, 1282
unpack_audio_SLQSSetAudioProfile, 1282
unpack_audio_SLQSSetAudioVolTLBConfig, 1282
audio_RXAGCList, 46
pRXAIG, 46
pRXComprSlope, 47
pRXComprThres, 47
pRXExpSlope, 47
pRXExpThres, 47
pRXStaticGain, 47
audio_RXAVCList, 47
pAVRXAVCHeadroom, 47
pAVRXAVCSens, 47
audio_RXPCMIIRFltr, 47
pFlag, 49
pStage0Val, 49
pStage1Val, 49
pStage2Val, 49
pStage3Val, 49
pStage4Val, 49
pStageCnt, 49
audio_TXAGCList, 49
pTXAIG, 49
pTXComprSlope, 50
pTXComprThres, 50
pTXExpSlope, 50
pTXExpThres, 50
pTXStaticGain, 50
audio_TXPCMIIRFltr, 50
pFlag, 51
pStage0Val, 51
pStage1Val, 51
pStage2Val, 51
pStage3Val, 51
pStage4Val, 51
pStageCnt, 51
auth
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
authData
pack_uim_SLQSUIMAuthenticate_t, 569
AuthProt
nas_protocolSubtypeElement, 322
Authentication
unpack_wds_SLQSGetRuntimeSettings_t, 1176
authentication
pack_wds_SetDefaultProfile_t, 617
AutoConnect
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 550
unpack_swiavms_SLQSAVMSGetSettings_t, 1039
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
AutoReboot
pack_swiavms_SLQSAVMSSetSettings_t, 549
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
autoUimSwitchMode
unpack_swidms_SLQSSwiDmsGetUimSlot_t, 1049
Autosdm
unpack_swioma_SLQSOMADMGetSettings_t, 1063
Avail_sys_len
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
avgPeriod
nas_LTESigRptCfg, 273
nas_LTESigRptConfig, 274
azimuth
loc_satelliteInfo, 179

bAPNLength
PackSwiAvmsSetSettingsAPNInfo, 638
PackSwiAVMSSettingsAPNInfo, 640
bAuto
pack_pds_SetPortAutomaticTracking_t, 520
pack_pds_SetServiceAutomaticTracking_t, 520
bBinaryType
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1194
bEnable
pack_dms_UIMSetPINProtection_t, 432
pack_nas_SLQSSetSignalStrengthsCallback_t, 500
bEnabled
pack_pds_SetXTRAAutomaticDownload_t, 521
bForceDownload
pack_fms_SetImagesPreference_t, 437
bICCID
slot_t, 655
bICCIDLength
slot_t, 655
bLogicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 581
slot_t, 655
bNotification
UnpackSwiAvmsEventReportNotification, 1199
bNumberOfPhySlots
unpack_uim_SetUimSlotStatusChangeCallback_-ind_t, 1072
BOOL
SwiDataTypes.h, 1656
BPTlv
NASQmiCbkNasSystemSelPrefInd, 402
bPWDLength
PackSwiAvmsSetSettingsAPNInfo, 638
PackSwiAVMSSettingsAPNInfo, 640
bPackageID
UnpackSwiAvmsEventReportPackageID, 1200
BROADCAST_CLIENT_ID
QmuxInternal.h, 1592
bRegStatus

UnpackSwiAvmsEventReportRegStatus, 1200
bResetStatistics
rmTrasnferStaticsReq, 653
bSerity
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1194
bState
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1194
UnpackSwiAvmsEventReportConfig, 1196
bType
UnpackSwiAvmsEventReportDataSessionStatus, 1197
UnpackSwiAvmsEventReportSessionType, 1201
bUnameLength
PackSwiAvmsSetSettingsAPNInfo, 638
PackSwiAVMSSettingsAPNInfo, 640
bUserInputRequest
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1194
UnpackSwiAvmsEventReportConfig, 1196
UnpackSwiAvmsEventReportConnectionRequest, 1196
BYT_STAT_STAT_MASK
wds.h, 1766
BYTE
SwiDataTypes.h, 1656
band
nas_LTEInfo, 257
NASPhyCaAggScellArray, 396
band1900
nas_gsmCellInfo, 234
band_pref
NASBandPreferenceTlv, 387
BandCapability
unpack_dms_GetBandCapability_t, 739
bandCapability
unpack_dms_SLQSGetBandCapability_t, 786
unpack_dms_SLQSGetBandCapabilityExt_t, 790
BandPref
nas_BandPrefTlv, 199
bandPref
pack_nas_SLQSSetBandPreference_t, 499
bandwidth
nas_LTEInfo, 257
nas_RfBandwidthInfoElements, 336
nas_RFBandwidthInfoTlv, 337
baseId
nas_CDMAInfo, 202
nas_CDMASysInfo, 206
baseLat
nas_CDMAInfo, 202
nas_CDMASysInfo, 206
baseLong
nas_CDMAInfo, 202
nas_CDMASysInfo, 206
BasestationID
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
BasestationLatitude
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
BasestationLongitude
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
BasicConnectDeviceService.h, 1283
BasicConnectDeviceService_DeviceService-SubscribeParse, 1283
BasicConnectDeviceService_DeviceService-SubscribeSetSend, 1284
BasicConnectDeviceService_Uuid, 1284
MbimEventEntry, 1283
BasicConnectDeviceService_DeviceServiceSubscribe-Parse
BasicConnectDeviceService.h, 1283
BasicConnectDeviceService_DeviceServiceSubscribe-SetSend
BasicConnectDeviceService.h, 1284
BasicConnectDeviceService_Uuid
BasicConnectDeviceService.h, 1284
batchPerSec
loc_accelAcceptReady, 162
loc_accelTempAcceptReady, 163
loc_gyroAcceptReady, 171
loc_gyroTempAcceptReady, 171
BearerID
unpack_qos_QosFlowInfo_t, 992
bearerID
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 1180
bearerId
unpack_QosFlowStat_t, 1010
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1191
bits_129_192
nas_BandPrefInfoTlv, 197
nas_lteBandPrefExt, 250
nas_nr5gBandPref, 296
bits_193_256
nas_BandPrefInfoTlv, 197
nas_lteBandPrefExt, 250
nas_nr5gBandPref, 296
bits_1_64
nas_BandPrefInfoTlv, 197
nas_lteBandPrefExt, 250
nas_nr5gBandPref, 296
bits_65_128
nas_BandPrefInfoTlv, 197
nas_lteBandPrefExt, 250
nas_nr5gBandPref, 296
bootSize
unpack_dms_GetFirmwareRevisions_t, 756
BootString
unpack_dms_GetFirmwareRevisions_t, 756
bootversion_str
unpack_dms_GetFirmwareInfo_t, 753

Broadcast
unpack_nas_GetCDMANetworkParameters_t, 877
broadcastActivate
pack_sms_SLQSSetSmsBroadcastActivation_t, 538
broadcastConfig
sms_qaQmi3GPPBroadcastCfgInfo, 665
bsInfoValid
nas_CDMASysInfo, 206
bsPRev
nas_CDMASysInfo, 206
bsPRevValid
nas_CDMASysInfo, 206
bsic
nas_GERANInfo, 231
bsicId
nas_gsmCellInfo, 234
bucketSz
unpack_qos_tokenBucket_t, 1008
buffer
QmiSyncObject, 645
bufferSize
QmiSyncObject, 645
buildID
FMSImageIdElement, 92
image_info_t, 97
buildIDLen
image_info_t, 97
buildIDLength
FMSImageIdElement, 92
buildId
FMSImageElement, 91
buildIdLength
FMSImageElement, 91
BurstDTMFInfo
pack_voice_SLQSVoiceBurstDTMF_t, 589
ByteLoopbackMode
unpack_wds_SLQSSGetLoopback_t, 1188
ByteLoopbackMultiplier
unpack_wds_SLQSSGetLoopback_t, 1188

CCETlv
unpack_cat_SetCatEventCallback_ind_t, 734
CDMA_P_Rev
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
CDMABroadcastConfig
sms_qaQmi3GPP2BroadcastCfgInfo, 665
CDMAECIOThreshListLen
nas_CDMAECIOThresh, 201
CDMARSSIThreshListLen
nas_CDMARSSIThresh, 203
CDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 946
CDMASystemInfoExt
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
CHAR
SwiDataTypes.h, 1656
CID
MBIM_COMMAND_DONE_MSG, 185
MBIM_COMMAND_MSG, 186
MBIM_INDICATE_STATUS_MSG, 187
CK_MAX_SIZE
dms.h, 1306
CQIValueCW0
unpack_nas_SLQSSwiGetLteCQI_t, 963
CQIValueCW1
unpack_nas_SLQSSwiGetLteCQI_t, 963
CRTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
CSDomain
unpack_nas_GetServingNetwork_t, 885
CSGInfo
unpack_nas_SLQSNasPerformIncNetworkScan-Callback_Ind_t, 951
CSTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
CUGIndex
voice_CUGInfo, 1230
CallBarStatus
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
callDuration
unpack_wds_GetSessionDuration_t, 1160
unpack_wds_GetSessionDurationV2_t, 1161
CallEndReason
wds_LastMdmCallEndRsnTlv, 1266
callEndReason
unpack_wds_SLQSGetDUNCallInfo_t, 1172
voice_arrCallEndReason, 1209
CallFWExtInfo
voice_getCallFWExtInfo, 1237
CallFWInfo
voice_getCallFWInfo, 1237
callID
pack_voice_SLQSVoiceGetCallInfo_t, 594
pack_voice_SLQSVoiceStopContDTMF_t, 605
unpack_voice_OTASPStatusCallback_ind_t, 1092
unpack_voice_SLQSVoiceStopContDTMF_t, 1124
unpack_voice_SUPSNotificationCallback_ind_t, 1129
unpack_voice_VoiceInfoRecCallback_ind_t, 1132
unpack_voice_voicePrivacyChangeCallback_ind_t, 1134
voice_allCallsAlphaIDInfo, 1204
voice_allCallsDiagInfo, 1204
voice_allCallsUUSInfo, 1205
voice_arrAlertingPattern, 1206
voice_arrAlertingType, 1207
voice_arrCallEndReason, 1209
voice_arrSvcOption, 1212
voice_callInfo, 1222
voice_DTMFInfo, 1232
voice_getAllCallRmtPtyName, 1236

voice_getAllCallRmtPtyNum, 1236
voice_peerNumberInfo, 1240
callNumber
pack_voice_SLQSVoiceDialCall_t, 591
callState
voice_callInfo, 1222
callType
voice_callInfo, 1222
CalledPartyNum
voice_arrCalledPartyNum, 1208
callerID
voice_callerIDInfo, 1217
voice_connectNumInfo, 1229
callerIDLen
voice_callerIDInfo, 1217
voice_connectNumInfo, 1229
callerName
voice_remotePartyName, 1245
Callinfo
voice_getAllCallInformation, 1235
campedCiotLteOpMode
nas_LteCiotOpModeTlv, 252
capMask
pack_uim_SLQSUIMGetServiceStatus_t, 574
cardProtocol
uim_physlotInfo, 707
cardState
slotInf, 657
uim_slotInfo, 722
carrier
unpack_dms_SLQSSwiGetFirmwareCurr_t, 796
carrier_str
unpack_dms_GetFirmwareInfo_t, 753
CarrierImage_t, 51
m_FwBuildId, 52
m_FwImageId, 52
m_PriBuildId, 52
m_PriImageId, 53
m_nCarrierId, 52
m_nFolderId, 52
m_nStorage, 52
carrierStr
_litefw_FirmwareFileInfo, 40
cat.h, 1284
pack_cat_CATSendEnvelopeCommand, 1285
pack_cat_CATSendTerminalResponse, 1286
pack_cat_SetCATEventCallback, 1286
unpack_cat_CATSendEnvelopeCommand, 1287
unpack_cat_CATSendEnvelopeCommand_t, 1285
unpack_cat_CATSendTerminalResponse, 1287
unpack_cat_CATSendTerminalResponse_t, 1285
unpack_cat_SetCATEventCallback, 1287
unpack_cat_SetCatEventCallback_ind, 1288
cat_AlPhaIdentifierTlv, 53
AlphaID, 53
AlphaIDLength, 53
ReferenceID, 53
cat_EndProactiveSessionTlv, 55
EndProactiveSession, 55
cat_EventIDDataTlv, 55
Data, 55
DataLength, 55
ReferenceID, 55
cat_EventListTlv, 55
SetupEventList, 56
cat_RefreshTlv, 56
RefreshMode, 56
RefreshStage, 56
cat_commonEventTlv, 53
CatEvent, 54
EventID, 54
EventLength, 54
TlvPresent, 54
cat_currentCatEvent, 54
CatAlphaIdtfr, 54
CatEndPS, 54
CatEvIDData, 54
CatEventLst, 54
CatRefresh, 54
CatAlphaIdtfr
cat_currentCatEvent, 54
CatEndPS
cat_currentCatEvent, 54
CatEvIDData
cat_currentCatEvent, 54
CatEvent
cat_commonEventTlv, 54
CatEventLst
cat_currentCatEvent, 54
CatRefresh
cat_currentCatEvent, 54
causeCode
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
ccsSupported
nas_CDMASysInfo, 206
ccsSupportedValid
nas_CDMASysInfo, 206
cdmaSSInfo, 56
ecio, 57
rssi, 57
cdmaSysIdValid
nas_CDMASysInfo, 206
cell_resel_priority
nas_infoInterFreq, 248
cellBroadcastCap
nas_AddSysInfo, 196
CellID
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
cellID
nas_GERANInfo, 231
nas_QmisNasSlqsNasPCICellInfo, 326
nas_UMTSInfo, 372
cellId
nas_GSMSysInfo, 238

nas_LTESysInfo, 277
nas_NR5GSysInfoTlv, 302
nas_NR5GSystemInfoTlv, 306
nas_UMTSExtInfo, 370
nas_WCDMASysInfo, 385
cellIdValid
nas_gsmCellInfo, 234
nas_GSMSysInfo, 238
nas_LTESysInfo, 278
nas_NR5GSysInfoTlv, 302
nas_NR5GSystemInfoTlv, 306
nas_WCDMASysInfo, 385
cellInterFreqParams
nas_infoInterFreq, 248
cellIsTDD
nas_umtsLTENbrCell, 375
CellParams
nas_LTEInfoIntrafreq, 260
cellReselPriority
nas_lteGsmCellInfo, 255
nas_LTEInfoIntrafreq, 260
nas_lteWcdmaCellInfo, 279
cells_len
nas_infoInterFreq, 248
nas_lteGsmCellInfo, 255
cellsLen
nas_LTEInfoIntrafreq, 260
nas_lteWcdmaCellInfo, 279
CfgValue
pack_swidms_SLQSSwiDmsSetUsbComp_t, 555
swidms_intfaceCfgTlv, 679
chaddr
wds_DHCPv4HWConfig, 1260
wdsDhcpv4HwConfig, 1275
chaddrLen
wds_DHCPv4HWConfig, 1260
wdsDhcpv4HwConfig, 1275
changePIN
pack_uim_ChangePin_t, 565
ChannelRate
wds_ChannelRateTlv, 1254
channelRate
unpack_wds_SLQSGetDUNCallInfo_t, 1172
Chipset
nas_DeviceConfigDetail, 222
CidCount
MbimEventEntry, 188
Cids
MbimEventEntry, 188
CiotAOPTlv
NASQmiCbkNasSystemSelPrefInd, 402
CiotAcqOrderLen
NASCiotAcqOrderPrefTlv, 387
ciotAcqOrderLen
nas_ciotAcqOrderPref, 209
nas_CiotAcqOrderPrefTlv, 210
CiotAcqOrderPref
NASCiotAcqOrderPrefTlv, 387
ciotLteOpMode
nas_LTEOperationalModeTlv, 266
CiotLteOpModePref
NASCiotLteOpModePrefTlv, 388
ciotLteOpModePref
nas_CiotLteOpModePrefTlv, 210
CiotOpMPTlv
NASQmiCbkNasSystemSelPrefInd, 402
ckLen
uim_depersonalizationInformation, 697
ckVal
uim_depersonalizationInformation, 697
clear
pack_dms_SLQSSwiGetCrashInfo_t, 425
ClientId
QMUX_HDR, 648
clientId
GetClientIdRspMsg, 95
pack_ctl_ReleaseClientId_t, 415
QmiService, 643
ReleaseClientIdReqMsg, 651
ReleaseClientIdRspMsg, 651
unpack_ctl_GetClientId_t, 736
unpack_ctl_ReleaseClientId_t, 736
cmdID
pack_cat_CATSendEnvelopeCommand_t, 413
codingScheme
nas_PLMNNetworkNameData, 319
voice_remotePartyName, 1245
CommandType
MBIM_COMMAND_MSG, 186
common.h
eAUDIO, 1291
eCAT, 1291
eCTL, 1291
eDMS, 1291
eDSD, 1291
eIMS, 1291
eIMSA, 1291
eIND, 1292
eLOC, 1291
eLOG_DEBUG, 1291
eLOG_FATAL, 1291
eLOG_INFO, 1291
eLOG_WARN, 1291
eNAS, 1291
ePDS, 1291
eQOS, 1291
eREQ, 1292
eRMS, 1291
eRSP, 1292
eSAR, 1291
eSMS, 1291
eSWIAUDIO, 1291
eSWIDMS, 1291
eSWILOC, 1291
eSWIM2MCMD, 1291
eSWIM2MCMD_AVC2, 1291

eSWIOMA, 1291
eSWIOMAEXT, 1291
eTIMEOUT_10_S, 1292
eTIMEOUT_20_S, 1292
eTIMEOUT_2_S, 1292
eTIMEOUT_300_S, 1292
eTIMEOUT_30_S, 1292
eTIMEOUT_5_S, 1292
eTIMEOUT_60_S, 1292
eTIMEOUT_8_S, 1292
eTIMEOUT_DEFAULT, 1292
eTMD, 1291
eUIM, 1291
eVOICE, 1291
eWDA, 1291
eWDS, 1291
common.h, 1288
eLOG_LEVEL, 1291
eQMI_SVC, 1291
eTimeout, 1291
fill_pack_ctx, 1292
fill_sdu_hdr, 1292
get_version, 1292
glog, 1294
gloglvl, 1294
helper_get_error_code, 1292
helper_get_error_reason, 1292
helper_get_req_str, 1292
helper_get_resp_ctx, 1293
helper_get_xid, 1293
helper_isBootLoader_DebugEnabled, 1293
helper_set_log_func, 1293
helper_set_log_lvl, 1293
liteqmi_GetVersion, 1293
liteqmi_helper_decode7bitAsciiEncString, 1293
liteqmi_log, 1294
MINREQBKLEN, 1290
MSGID_AND_LEN, 1290
MSGID_DONT_CARE, 1290
msgtype, 1292
SDU_HDR_LEN, 1290
UNUSEDPARAM, 1291
unpack_result_code_only, 1294
commonInfo
unpack_nas_SLQSNasSwiModemStatus_t, 955
ConcSvcInfo
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
condition
MbimSyncObject, 190
QmiSyncObject, 645
confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 458
conn_status
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 1180
ConnectedPartyNum
voice_arrConnectPartyNum, 1210
connectionStatus, 57
MDMCallDuration, 57
MDMConnStatus, 57
unpack_wds_GetSessionState_t, 1161
unpack_wds_SLQSGetDUNCallInfo_t, 1172
content
uim_authenticateResult, 691
uim_readResult, 714
uim_readResultInfo, 715
contentLen
uim_authenticateResult, 691
uim_readResult, 714
uim_readResultInfo, 715
context
uim_authenticationData, 692
contextId
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 636
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 1191
contextType
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 636
ControlMac
nas_protocolSubtypeElement, 322
count
swidms_SwiDmsGetHWWatchdog, 682
count0
nas_RankIndicatorTlv, 328
count1
nas_RankIndicatorTlv, 328
countryInitials
nas_PLMNNetworkNameData, 319
coverage
altSrcInfo_t, 43
cphy_ca_dl_bandwidth
NASPhyCaAggScellArray, 396
cphy_scell_info_list_len
NASPhyCaAggScellArray, 396
cpich_ecno
nas_wcdmaCellInfo, 379
cpich_rscp
nas_wcdmaCellInfo, 379
cradleMountConfigStatus
unpack_loc_CradleMountCallback_Ind_t, 847
crashAction
pack_dms_SetCrashAction_t, 418
crashData
crashInformation, 59
crashId
crashInformation, 59
crashInfo
crashInfoParams, 58
crashInfoParam
unpack_dms_SLQSSwiGetCrashInfo_t, 794
crashInfoParams, 58
crashInfo, 58
crashStatus, 58

crashInformation, 58
crashData, 59
crashId, 59
crashString, 59
crashStrlen, 59
gcdumpString, 59
gcdumpStrlen, 59
numCrashes, 59
crashStatus
crashInfoParams, 58
crashString
crashInformation, 59
crashStrlen
crashInformation, 59
csAttachState
nas_servSystem, 351
NASServingSystemInfo, 405
csBarStatus
nas_CallBarringSysInfo, 199
nas_callBarStatus, 200
cscfPortName
ims_CSCFPortNameInfo, 100
csg_info_len
nas_CSGInfoTlv, 215
csg_list_cat
nas_CSGInfo, 214
csgId
nas_CsgId, 212
csgInfo
nas_CSGInfoTlv, 215
ctl.h, 1294
helper_get_ctl_resp_ctx, 1294
pack_ctl_GetClientId, 1294
pack_ctl_ReleaseClientId, 1295
unpack_ctl_GetClientId, 1295
unpack_ctl_ReleaseClientId, 1295
CtlFlags
QMUX_HDR, 648
CtlService, 59
CtlService.h, 1296
service, 59
CtlService.h, 1296
CtlService, 1296
CtlService_GetClientId, 1296
CtlService_Initialize, 1297
CtlService_InitializeRegularService, 1297
CtlService_ReleaseClientId, 1297
CtlService_ShutDown, 1298
CtlService_ShutDownRegularService, 1298
CtlService_GetClientId
CtlService.h, 1296
CtlService_Initialize
CtlService.h, 1297
CtlService_InitializeRegularService
CtlService.h, 1297
CtlService_ReleaseClientId
CtlService.h, 1297
CtlService_ShutDown
CtlService.h, 1298
CtlService_ShutDownRegularService
CtlService.h, 1298
CtlTransactionHdr, 60
flags, 60
QmuxInternal.h, 1593
transactionId, 60
cur_carr_name
unpack_dms_GetFirmwareInfo_t, 753
cur_carr_rev
unpack_dms_GetFirmwareInfo_t, 753
curDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 1169
CurIndex
dms_devMaxCfgListCaps, 64
curProfile
pack_wds_SLQSModifyProfile_t, 627
UnPackGetProfileSettingOut, 1192
UnPackGetProfileSettingOutV2, 1193
CurRecGlobalPrefTlvPresent
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
CurSubsCapsLen
dms_devCurSubsCaps, 63
CurrChanRxRate
dunchannelRate, 89
wds_channelRate, 1253
CurrChanTxRate
dunchannelRate, 89
wds_channelRate, 1253
currDBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1185
currNWInfo
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1185
currNetworkInfo, 60
NetworkType, 60
RATMask, 60
SOMask, 60
unpack_wds_SLQSGetCurrDataSystemStat_t, 1167
current_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 1169
current_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 1169
CurrentCfgType
swidms_intfaceCfgTlv, 679
currentChannelRXRate
unpack_wds_GetConnectionRate_t, 1144
currentChannelTXRate
unpack_wds_GetConnectionRate_t, 1144
CurrentDDTMStatus
nas_DDTMSettings, 220
currentDataBearer

pack_wds_SLQSSetWdsEventCallback_t, 630
CurrentFragment
MBIM_FRAGMENT_HEADER, 186
currentNetwork
qmiWDSDataBearerTechnology, 647
CurrentPLMN
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
CurrentmitigationLvl
unpack_tmd_SLQSTmdGetMitigationLvl_t, 1066
cust_attr
DMScustSettingInfo, 79
cust_id
DMScustSettingInfo, 79
DMSgetCustomInput, 81
pack_dms_GetCustFeaturesV2_t, 416
pack_dms_SetCustFeaturesV2_t, 421
cust_value
DMScustSettingInfo, 79
pack_dms_SetCustFeaturesV2_t, 421
custSetting
DMScustSettingList, 79
CustomSCP
unpack_nas_GetCDMANetworkParameters_t, 877
CwtMute
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1033
unpack_swiaudio_SLQSGetM2MAVMute_t, 1035
cycleLength
nas_EdrxCycleLength, 226

DBTTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
dBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1185
dBTechExtRatValue
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
dBTechExtSoMask
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
dBTechnology
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
dBtechExtAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1185
dBtechnologyExt
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
DDTMAction
nas_DDTMSettings, 220
pack_nas_SLQSSetDDTMPref_t, 500
DDTMPrefSetting
nas_DDTMSettings, 220
pack_nas_SLQSSetDDTMPref_t, 500
DHCPRelayEnabled
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
DHCPv4LeaseOptTlv
unpack_wds_DHCPv4ClientLease_ind_t, 1142
DHCPv4LeaseStateTlv
unpack_wds_DHCPv4ClientLease_ind_t, 1142
dLatitude
unpack_pds_SetEventReport_Ind_t, 980
dLongitude
unpack_pds_SetEventReport_Ind_t, 980
DMS_IMGDETAILS_LEN
dms.h, 1306
DMS_PM_FACTORY
dms.h, 1307
DMS_PM_LOW
dms.h, 1307
DMS_PM_OFFLINE
dms.h, 1307
DMS_PM_ONLINE
dms.h, 1307
DMS_PM_RESET
dms.h, 1307
DMS_PM_SHUT_DOWN
dms.h, 1307
DMS_VALID_FSN_LEN
dms.h, 1308
DMScustSettingInfo, 78
cust_attr, 79
cust_id, 79
cust_value, 79
id_length, 79
value_length, 79
DMScustSettingList, 79
custSetting, 79
list_type, 79
num_instances, 79
DMSgetCustomFeatureV2, 80
pCustSettingInfo, 80
pCustSettingList, 80
pGetCustomInput, 80
DMSgetCustomInput, 80
cust_id, 81
list_type, 81
DRCCover
nas_DRCParams, 223
DRCValue
nas_DRCParams, 223
DSTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
DTMFEvent
voice_DTMFInfo, 1232
DTMFInformation
unpack_voice_DTMFEventCallback_ind_t, 1090
DTMFInterdigitInterval
voice_DTMFLengths, 1233
DTMFPulseWidth
voice_DTMFLengths, 1233

DTMFdigit
pack_voice_SLQSVoiceStartContDTMF_t, 604
DTMInd
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
Data
cat_EventIDDataTlv, 55
data
sMSCAddressInfo, 671
sMSEtwsMessageInfo, 672
sMSTransferRouteMTMessageInfo, 676
uim_authenticationData, 692
uim_UIMNickName, 724
uim_UIMProfileCode, 725
uim_writeRecordInfo, 728
uim_writeTransparentInfo, 729
data_buf
NASOTAMessageTlv, 394
data_len
NASOTAMessageTlv, 394
dataBearer
pack_wds_SLQSSetWdsEventCallback_t, 630
dataBearerMask
unpack_wds_SLQSGetDataBearerTechnology_t, 1169
DataBearerTech
wds_DataBearTechTlv, 1256
dataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 1172
dataBearerTechExt
pack_wds_SLQSSetWdsEventCallback_t, 630
dataCapabilities
nas_dataSrvCapabilities, 217
dataCapabilitiesLen
nas_dataSrvCapabilities, 217
DataCaps
unpack_nas_GetServingNetwork_t, 885
unpack_nas_GetServingNetworkCapabilities_t, 887
dataCaps
unpack_nas_SetDataCapabilitiesCallback_ind_t, 890
DataCapsLen
unpack_nas_GetServingNetwork_t, 885
unpack_nas_GetServingNetworkCapabilities_t, 887
dataCapsSize
unpack_nas_SetDataCapabilitiesCallback_ind_t, 890
dataLen
pack_cat_CATSendEnvelopeCommand_t, 413
pack_cat_CATSendTerminalResponse_t, 414
uim_authenticationData, 692
uim_UIMNickName, 724
uim_UIMProfileCode, 725
uim_writeRecordInfo, 728
uim_writeTransparentInfo, 729
DataLength
cat_EventIDDataTlv, 55
DataRate
unpack_qos_swiQosFlow_t, 1007
dataRateMax
unpack_qos_dataRate_t, 989
dataServiceCaCapability
unpack_dms_GetDeviceCapabilities_t, 745
DataServiceCapability
dms_devCaps, 62
unpack_dms_GetDeviceCap_t, 744
DataSrvCapabilities
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
dataSysStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1185
dataSystemStatus
pack_wds_SLQSSetWdsEventCallback_t, 630
Date
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
DateLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
day
nas_timeInfo, 367
nas_UniversalTime, 377
dayLtSavingAdj
nas_timeInfo, 367
dayOfWeek
nas_timeInfo, 367
nas_UniversalTime, 377
daylightSavings
nas_qaQmi3Gpp2TimeZone, 323
ddtm_Settings
unpack_nas_SLQSDDTMPrefIndicationCallback_-Ind_t, 895
dedicatedBand
nas_RfDedicatedBandInfoElements, 339
nas_RFDedicatedBandInfoTlv, 339
defaultPDNEnabled
unpack_wds_SLQSGet3GPPConfigItem_t, 1166
DefaultRoamInd
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 911
delayClass
LibPackGPRSRequestedQoS, 119
wds_GPRSQoS, 1264
deliveryErrSDU
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
depersonilisationInfo
pack_uim_SLQSUIMDepersonalization_t, 570
description
unpack_omaDmFotaTlv_t, 972
descriptionlength
unpack_omaDmFotaTlv_t, 972
Desription

nas_QmiNas3GppNetworkInfo, 324
destPortRangeEnd
LibPackTFTIDParams, 157
destPortRangeStart
LibPackTFTIDParams, 157
DetailedSvcInfo
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
DevCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
DevCfgListLen
dms_devMaxCfgListCaps, 64
DevCrashState
unpack_dms_GetCrashAction_t, 739
Device
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 543
DeviceError
qmTlvResult, 647
deviceFd
MbimTransport, 192
QmiService, 644
QmuxTransport, 649
deviceID
unpack_tmd_SLQSTmdMitigationLvlRptCallback_-ind_t, 1066
deviceIdLen
unpack_tmd_SLQSTmdMitigationLvlRptCallback_-ind_t, 1066
devicePath
QcqmiService, 642
DeviceResult
qmTlvResult, 647
DeviceServiceId
MBIM_COMMAND_DONE_MSG, 185
MBIM_COMMAND_MSG, 186
MBIM_INDICATE_STATUS_MSG, 187
MbimEventEntry, 188
DiagInfo
voice_allCallsDiagInfo, 1204
voice_arrDiagInfo, 1210
diagInfoLen
voice_diagInfo, 1231
diagnosticInfo
voice_diagInfo, 1231
digitBuff
voice_DTMFInfo, 1232
digitCnt
voice_burstDTMFInfo, 1214
voice_DTMFInfo, 1232
dirNum
nas_dirNum, 223
dirNumLen
nas_dirNum, 223
DirectLayer.h, 1298
DirectLayer_Create, 1298
DirectLayer_GetClientID, 1298
DirectLayer_GetDeviceFD, 1298
DirectLayer_SendQmuxPacket, 1298
DirectLayer_ShutDown, 1298
DriectLay_GetDummyServiceClientID, 1298
DirectLayer_Create
DirectLayer.h, 1298
DirectLayer_GetClientID
DirectLayer.h, 1298
DirectLayer_GetDeviceFD
DirectLayer.h, 1298
DirectLayer_SendQmuxPacket
DirectLayer.h, 1298
DirectLayer_ShutDown
DirectLayer.h, 1298
direction
voice_callInfo, 1222
DisableIMSI
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
dispType
voice_extDispRecInfo, 1234
displayCondition
nas_serviceProviderName, 349
dl_bw_value
nas_PhyCaAggPcellInfo, 311
nas_PhyCaAggScellDlBw, 311
nas_PhyCaAggScellInfo, 315
NASPhyCaAggPcellInfo, 395
NASPhyCaAggScellDlBw, 397
NASPhyCaAggScellInfo, 399
Dl_data_aggregation_max_datagrams
unpack_wda_GetDataFormat_t, 1136
Dl_data_aggregation_max_size
unpack_wda_GetDataFormat_t, 1136
Dl_data_aggregation_protocol
unpack_wda_GetDataFormat_t, 1136
Dl_minimum_padding
unpack_wda_GetDataFormat_t, 1136
dms.h, 1299
ACT_CODE_MAX_SIZE, 1306
CK_MAX_SIZE, 1306
DMS_IMGDETAILS_LEN, 1306
DMS_PM_FACTORY, 1307
DMS_PM_LOW, 1307
DMS_PM_OFFLINE, 1307
DMS_PM_ONLINE, 1307
DMS_PM_RESET, 1307
DMS_PM_SHUT_DOWN, 1307
DMS_VALID_FSN_LEN, 1308
ERI_DATA_MAX_SIZE, 1308
MAX_BUILD_ID_LEN, 1308
MEID_MAX_SIZE, 1308
pack_dms_ActivateAutomatic, 1308
pack_dms_GetActivationState, 1308
pack_dms_GetBandCapability, 1309
pack_dms_GetCrashAction, 1309
pack_dms_GetCustFeature, 1309
pack_dms_GetCustFeaturesV2, 1310
pack_dms_GetDeviceCap, 1310
pack_dms_GetDeviceCapabilities, 1311

pack_dms_GetDeviceCapabilitiesV2, 1311
pack_dms_GetDeviceHardwareRev, 1311
pack_dms_GetDeviceMfr, 1312
pack_dms_GetDeviceSerialNumbers, 1312
pack_dms_GetFSN, 1313
pack_dms_GetFirmwareInfo, 1312
pack_dms_GetFirmwareRevision, 1313
pack_dms_GetFirmwareRevisions, 1313
pack_dms_GetHardwareRevision, 1314
pack_dms_GetIMSI, 1314
pack_dms_GetManufacturer, 1314
pack_dms_GetModelID, 1315
pack_dms_GetNetworkTime, 1315
pack_dms_GetNetworkTimeV2, 1315
pack_dms_GetOfflineReason, 1316
pack_dms_GetPRLVersion, 1316
pack_dms_GetPower, 1316
pack_dms_GetSerialNumbers, 1317
pack_dms_GetUSBComp, 1317
pack_dms_GetVoiceNumber, 1317
pack_dms_ResetToFactoryDefaults, 1318
pack_dms_SLQSDmsSwiGetPCInfo, 1322
pack_dms_SLQSDmsSwiGetResetInfo, 1322
pack_dms_SLQSDmsSwiGetUimSelection, 1322
pack_dms_SLQSDmsSwiIndicationRegister, 1323
pack_dms_SLQSGetBandCapability, 1323
pack_dms_SLQSGetERIFile, 1323
pack_dms_SLQSGetPowerSaveModeConfig, 1324
pack_dms_SLQSSetPowerSaveModeConfig, 1324
pack_dms_SLQSSwiClearDyingGaspStatistics, 1324
pack_dms_SLQSSwiGetCrashInfo, 1325
pack_dms_SLQSSwiGetDyingGaspCfg, 1325
pack_dms_SLQSSwiGetDyingGaspStatistics, 1325
pack_dms_SLQSSwiGetFirmwareCurr, 1326
pack_dms_SLQSSwiGetFwUpdateStatus, 1326
pack_dms_SLQSSwiGetHostDevInfo, 1326
pack_dms_SLQSSwiGetOSInfo, 1327
pack_dms_SLQSSwiGetSerialNoExt, 1327
pack_dms_SLQSSwiSetDyingGaspCfg, 1327
pack_dms_SLQSSwiSetHostDevInfo, 1328
pack_dms_SLQSSwiSetOSInfo, 1328
pack_dms_SLQSUIMGetState, 1329
pack_dms_SetActivationStatusCallback, 1318
pack_dms_SetCrashAction, 1319
pack_dms_SetCustFeature, 1319
pack_dms_SetCustFeaturesV2, 1319
pack_dms_SetEventReport, 1320
pack_dms_SetFirmwarePreference, 1320
pack_dms_SetIndicationRegister, 1320
pack_dms_SetPower, 1321
pack_dms_SetUSBComp, 1321
pack_dms_SwiSetEventReport, 1329
pack_dms_SwiUimSelect, 1329
pack_dms_UIMChangePIN, 1330
pack_dms_UIMGetControlKeyStatus, 1330
pack_dms_UIMGetICCID, 1330
pack_dms_UIMGetPINStatus, 1331
pack_dms_UIMSetControlKeyProtection, 1331
pack_dms_UIMSetPINProtection, 1332
pack_dms_UIMUnblockControlKey, 1332
pack_dms_UIMUnblockPIN, 1332
pack_dms_UIMVerifyPIN, 1333
pack_dms_ValidateSPC, 1333
SPC_SIZE, 1308
UNIQUE_ID_LEN, 1308
unpack_dms_ActivateAutomatic, 1333
unpack_dms_GetActivationState, 1334
unpack_dms_GetBandCapability, 1334
unpack_dms_GetCrashAction, 1335
unpack_dms_GetCustFeature, 1335
unpack_dms_GetCustFeaturesV2, 1335
unpack_dms_GetDeviceCap, 1336
unpack_dms_GetDeviceCapabilities, 1336
unpack_dms_GetDeviceCapabilitiesV2, 1336
unpack_dms_GetDeviceHardwareRev, 1337
unpack_dms_GetDeviceMfr, 1337
unpack_dms_GetDeviceSerialNumbers, 1337
unpack_dms_GetFSN, 1339
unpack_dms_GetFirmwareInfo, 1338
unpack_dms_GetFirmwareRevision, 1338
unpack_dms_GetFirmwareRevisions, 1338
unpack_dms_GetHardwareRevision, 1339
unpack_dms_GetIMSI, 1339
unpack_dms_GetManufacturer, 1340
unpack_dms_GetModelID, 1340
unpack_dms_GetNetworkTime, 1341
unpack_dms_GetNetworkTimeV2, 1341
unpack_dms_GetOfflineReason, 1341
unpack_dms_GetPRLVersion, 1342
unpack_dms_GetPower, 1342
unpack_dms_GetSerialNumbers, 1342
unpack_dms_GetUSBComp, 1343
unpack_dms_GetVoiceNumber, 1343
unpack_dms_PSMCfgChange_ind, 1343
unpack_dms_ResetToFactoryDefaults, 1344
unpack_dms_SLQSDmsSwiGetPCInfo, 1348
unpack_dms_SLQSDmsSwiGetResetInfo, 1348
unpack_dms_SLQSDmsSwiGetResetInfo_Ind, 1348
unpack_dms_SLQSDmsSwiGetUimSelection, 1349
unpack_dms_SLQSDmsSwiIndicationRegister, 1349
unpack_dms_SLQSGetBandCapability, 1349
unpack_dms_SLQSGetBandCapabilityExt, 1350
unpack_dms_SLQSGetERIFile, 1350
unpack_dms_SLQSGetPowerSaveModeConfig, 1351
unpack_dms_SLQSSetPowerSaveModeConfig, 1351
unpack_dms_SLQSSwiClearDyingGaspStatistics, 1351
unpack_dms_SLQSSwiGetCrashInfo, 1352
unpack_dms_SLQSSwiGetDyingGaspCfg, 1352

unpack_dms_SLQSSwiGetDyingGaspStatistics, 1352
unpack_dms_SLQSSwiGetFirmwareCurr, 1353
unpack_dms_SLQSSwiGetFwUpdateStatus, 1353
unpack_dms_SLQSSwiGetHostDevInfo, 1353
unpack_dms_SLQSSwiGetOSInfo, 1354
unpack_dms_SLQSSwiGetSerialNoExt, 1354
unpack_dms_SLQSSwiSetDyingGaspCfg, 1354
unpack_dms_SLQSSwiSetHostDevInfo, 1355
unpack_dms_SLQSSwiSetOSInfo, 1355
unpack_dms_SLQSUIMGetState, 1355
unpack_dms_SetActivationStatusCallback, 1344
unpack_dms_SetCrashAction, 1344
unpack_dms_SetCustFeature, 1345
unpack_dms_SetCustFeaturesV2, 1345
unpack_dms_SetEventReport, 1345
unpack_dms_SetEventReport_ind, 1346
unpack_dms_SetFirmwarePreference, 1346
unpack_dms_SetIndicationRegister, 1346
unpack_dms_SetPower, 1347
unpack_dms_SetUSBComp, 1347
unpack_dms_SwiEventReportCallBack_ind, 1356
unpack_dms_SwiSetEventReport, 1356
unpack_dms_SwiUimSelect, 1356
unpack_dms_UIMChangePIN, 1357
unpack_dms_UIMGetControlKeyStatus, 1357
unpack_dms_UIMGetICCID, 1357
unpack_dms_UIMGetPINStatus, 1358
unpack_dms_UIMSetControlKeyProtection, 1358
unpack_dms_UIMSetPINProtection, 1358
unpack_dms_UIMUnblockControlKey, 1359
unpack_dms_UIMUnblockPIN, 1359
unpack_dms_UIMVerifyPIN, 1360
unpack_dms_ValidateSPC, 1360
dms_ActivationStatusTlv, 60
activationStatus, 61
TlvPresent, 61
dms_LteBandsSupport, 69
lteBands, 69
supportedLteBandLen, 69
TLVPresent, 69
dms_OperatingModeTlv, 69
operatingMode, 70
TlvPresent, 70
dms_PSMActiveTimerIndTlv, 70
ActiveTimerInd, 70
TlvPresent, 70
dms_PSMActiveTimerTlv, 70
activeTimer, 71
TlvPresent, 71
dms_PSMDurationDueToOOSTlv, 71
durationDueToOOS, 71
TlvPresent, 71
dms_PSMDurationThresholdTlv, 71
durationThreshold, 72
TlvPresent, 72
dms_PSMEarlyWakeupTimeTlv, 72
earlyWakeupTime, 72
TlvPresent, 72
dms_PSMEnableStateIndTlv, 73
EnableStateInd, 73
TlvPresent, 73
dms_PSMEnableStateTlv, 73
enableState, 73
TlvPresent, 74
dms_PSMPeriodicUpdateTimerIndTlv, 74
PeriodicUpdateTimerInd, 74
TlvPresent, 74
dms_PSMPeriodicUpdateTimerTlv, 74
periodicUpdateTimer, 75
TlvPresent, 75
dms_PSMRandomizationWindowTlv, 75
randomizationWindow, 75
TlvPresent, 75
dms_TemperatureTlv, 75
TempStat, 76
Temperature, 76
TlvPresent, 76
dms_UimAutoSwitchActSlotTlv, 76
TlvPresent, 76
uimAutoSwitchActSlot, 76
dms_UimStatusTlv, 77
event, 77
intf, 77
TlvPresent, 77
dms_VoltageTlv, 77
TlvPresent, 78
VoltStat, 78
Voltage, 78
dms_devCaps, 61
DataServiceCapability, 62
MaxRXChannelRate, 62
MaxTXChannelRate, 62
RadioIfaces, 62
RadioIfacesSize, 62
SimCapability, 62
dms_devCurSubsCaps, 62
CurSubsCapsLen, 63
SubsCapList, 63
dms_devMaxCfgListCaps, 63
CurIndex, 64
DevCfgListLen, 64
MaxActive, 64
MaxSubs, 64
SubsDevList, 64
dms_devMaxSubsCaps, 64
MaxSubsCapLen, 64
MaxSubsList, 64
dms_devMultiSimCaps, 64
MaxSubs, 65
SubsCfgList, 65
SubsCfgListLen, 65
dms_devMultiSimVoiceDataCaps, 65
MaxActive, 65
MaxSubs, 65
dms_devSubsCfgList, 66

MaxActive, 66
SubsList, 66
SubsListLen, 66
dms_devSubsFeatureModeCaps, 66
SubsFeatureLen, 67
SubsFeatureList, 67
dms_devSubsList, 67
SubsList, 67
SubsListLen, 67
dms_devSubsVoiceDataCaps, 68
SubsVoiceDataCapLen, 68
SubsVoiceDataList, 68
dms_devSubsVoiceDataList, 68
SimVoiceDataCap, 69
SubsVoiceDataCap, 69
domain
wds_DomainNameList, 1263
domainLen
wds_Domain, 1262
DomainList
unpack_wds_SLQSGetRuntimeSettings_t, 1176
domainName
wds_Domain, 1262
DormancyStat
wds_DormStatTlv, 1263
dormancyStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
dormancyState
unpack_wds_GetDormancyState_t, 1152
dormancyStatus
pack_wds_SLQSSetWdsEventCallback_t, 630
unpack_wds_SLQSGetDUNCallInfo_t, 1172
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
downLink
voice_NSSAudioCtrl, 1239
DriectLay_GetDummyServiceClientID
DirectLayer.h, 1298
dsd.h, 1360
MAX_APN_STRING, 1363
pack_dsd_RegisterUIInfoChange, 1363
pack_dsd_SLQSGetSystemStatus, 1363
pack_dsd_SLQSGetUIInfo, 1364
pack_dsd_SystemStatusChange, 1364
unpack_dsd_RegisterUIInfoChange, 1364
unpack_dsd_RegisterUIInfoChange_t, 1363
unpack_dsd_SLQSGetSystemStatus, 1365
unpack_dsd_SLQSGetUIInfo, 1365
unpack_dsd_SLQSSystemStatusChangeInd, 1365
unpack_dsd_SLQSUIInfoChangeInd, 1366
unpack_dsd_SystemStatusChange, 1366
unpack_dsd_SystemStatusChnage_t, 1363
DsdApnAvailableSystemInfo_t, 81
apn_avail_sys_len, 81
apn_name, 81
apn_name_len, 81
dsdGlobalAvailableSystem, 81
DsdApnCurRecAvailableSystemInfo_t, 81
dsdCurrentPreferdSystem, 82
dsdRecommendedPreferdSystem, 82
dsdApnCurRecPreferredSystemInfo
DsdApnCurRecPreferredSystemInfo_t, 83
DsdApnCurRecPreferredSystemInfo_t, 82
apn_name, 83
apn_name_len, 83
dsdApnCurRecPreferredSystemInfo, 83
dsdAvailableSystem
DsdGlobalUIInformation_t, 83
dsdCurrentPreferdSystem
DsdApnCurRecAvailableSystemInfo_t, 82
dsdGlobalAvailableSystem
DsdApnAvailableSystemInfo_t, 81
DsdGlobalUIInformation_t, 83
dsdAvailableSystem, 83
ui_mask, 83
DsdNullBearerReason_t, 84
null_bearer_reason, 84
dsdRecommendedPreferdSystem
DsdApnCurRecAvailableSystemInfo_t, 82
DsdSystemInfo_t, 84
rat_value, 87
so_mask, 87
technology, 87
DsdUIApnAvailableSystemInfo_t, 87
apn_name, 88
apn_name_len, 88
apn_type_mask, 88
is_internet, 88
rat_value, 89
so_mask, 89
technology, 89
ui_mask, 89
dtmSupp
nas_GSMSysInfo, 238
dtmSuppValid
nas_GSMSysInfo, 238
dunchannelRate, 89
CurrChanRxRate, 89
CurrChanTxRate, 89
MaxChanRxRate, 89
MaxChanTxRate, 89
Duration
pack_nas_SetNetworkPreference_t, 467
unpack_nas_GetNetworkPreference_t, 882
durationDueToOOS
dms_PSMDurationDueToOOSTlv, 71
durationThreshold
dms_PSMDurationThresholdTlv, 72

eAUDIO
common.h, 1291
eCAT
common.h, 1291
eCTL
common.h, 1291
eDMS

common.h, 1291
eDSD
common.h, 1291
eFILE_TYPE_CAR_PRI
lite-fw.h, 1394
eFILE_TYPE_COMPO_PRI
lite-fw.h, 1394
eFILE_TYPE_NONE
lite-fw.h, 1394
eFILE_TYPE_OEM_PRI
lite-fw.h, 1394
eFIREHOSE_ERR_RC7611_RETRY_TEST
lite-fw.h, 1395
eFIREHOSE_ERR_SECBOOT_INVALID_CERT_CHAIN
lite-fw.h, 1395
eFW_TYPE_CWE
lite-fw.h, 1394
eFW_TYPE_CWE_NVU
lite-fw.h, 1394
eFW_TYPE_INVALID
lite-fw.h, 1394
eFW_TYPE_MBN
lite-fw.h, 1394
eFW_TYPE_MBN_GOBI
lite-fw.h, 1394
eFW_TYPE_NVU
lite-fw.h, 1394
eFW_TYPE_SPK
lite-fw.h, 1394
eIMAGE_TYPE_ANY
lite-fw.h, 1395
eIMAGE_TYPE_APPL
lite-fw.h, 1395
eIMAGE_TYPE_APPS
lite-fw.h, 1395
eIMAGE_TYPE_BOOT
lite-fw.h, 1395
eIMAGE_TYPE_FILE
lite-fw.h, 1395
eIMAGE_TYPE_INVALID
lite-fw.h, 1395
eIMAGE_TYPE_KEYS
lite-fw.h, 1395
eIMAGE_TYPE_MAX
lite-fw.h, 1395
eIMAGE_TYPE_MIN
lite-fw.h, 1395
eIMAGE_TYPE_MODM
lite-fw.h, 1395
eIMAGE_TYPE_SPKG
lite-fw.h, 1395
eIMAGE_TYPE_USER
lite-fw.h, 1395
eIMS
common.h, 1291
eIMSA
common.h, 1291
eIND
common.h, 1292
eLITE_FWDWL_APP_ERR_RC7611_ERROR_AFTER_RETRY
lite-fw.h, 1395
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_1
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_10
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_11
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_12
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_125
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_126
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_127
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_13
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_14
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_17
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_18
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_19
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_2
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_20
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_21
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_23
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_24
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_25

nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_250
nas.h, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_26
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_27
nas.h, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_28
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_29
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_3
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_30
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_31
nas.h, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_32
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_33
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_34
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_35
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_36
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_37
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_38
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_39
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_4
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_40
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_41
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_42
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_43
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_46
nas.h, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_47
nas.h, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_48
nas.h, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_5
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_6
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_66
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_7
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_71
nas.h, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_8
nas.h, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_9
nas.h, 1454
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED
nas.h, 1455
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
nas.h, 1455
eLOC

common.h, 1291
eLOG_DEBUG
common.h, 1291
eLOG_FATAL
common.h, 1291
eLOG_INFO
common.h, 1291
eLOG_WARN
common.h, 1291
eModel_9X15
lite-fw.h, 1395
eModel_9X30
lite-fw.h, 1395
eModel_9x05
lite-fw.h, 1395
eModel_9x06
lite-fw.h, 1395
eModel_9x07
lite-fw.h, 1395
eModel_Unknown
lite-fw.h, 1395
eModel_WP9X15
lite-fw.h, 1395
eNAS
common.h, 1291
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100
nas.h, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15
nas.h, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25
nas.h, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50
nas.h, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6
nas.h, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75
nas.h, 1455
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE
nas.h, 1455
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE
nas.h, 1455
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE
nas.h, 1455
ePDS
common.h, 1291
eQCWWAN_ERR_API_MUTEX_TIMEOUT
qmerrno.h, 1579
eQCWWAN_ERR_BUFFER_SZ
qmerrno.h, 1578
eQCWWAN_ERR_CANCEL_OP
qmerrno.h, 1579
eQCWWAN_ERR_DRIVER
qmerrno.h, 1579
eQCWWAN_ERR_ENUM_BEGIN
qmerrno.h, 1578
eQCWWAN_ERR_ENUM_END
qmerrno.h, 1579
eQCWWAN_ERR_FILE_COPY
qmerrno.h, 1578
eQCWWAN_ERR_FILE_OPEN
qmerrno.h, 1578
eQCWWAN_ERR_GENERAL
qmerrno.h, 1578
eQCWWAN_ERR_INTERNAL
qmerrno.h, 1578
eQCWWAN_ERR_INVALID_ARG
qmerrno.h, 1578
eQCWWAN_ERR_INVALID_DEVID
qmerrno.h, 1578
eQCWWAN_ERR_INVALID_FILE
qmerrno.h, 1578
eQCWWAN_ERR_INVALID_QMI_RSP
qmerrno.h, 1578
eQCWWAN_ERR_INVALID_XID
qmerrno.h, 1579
eQCWWAN_ERR_MALFORMED_QMI_RSP
qmerrno.h, 1578
eQCWWAN_ERR_MEMORY
qmerrno.h, 1578
eQCWWAN_ERR_MULTIPLE_DEVICES
qmerrno.h, 1579
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED
qmerrno.h, 1579
eQCWWAN_ERR_NO_CANCELABLE_OP
qmerrno.h, 1579
eQCWWAN_ERR_NO_CONNECTION
qmerrno.h, 1578
eQCWWAN_ERR_NO_DEVICE
qmerrno.h, 1578
eQCWWAN_ERR_NO_SIGNAL
qmerrno.h, 1579
eQCWWAN_ERR_NONE
qmerrno.h, 1578
eQCWWAN_ERR_NULL_TLV
qmerrno.h, 1582
eQCWWAN_ERR_OFFLINE
qmerrno.h, 1579
eQCWWAN_ERR_PDU_GENERATION
qmerrno.h, 1579
eQCWWAN_ERR_QMI_ABORTED
qmerrno.h, 1579
eQCWWAN_ERR_QMI_ACCESS_DENIED
qmerrno.h, 1581
eQCWWAN_ERR_QMI_ACK_NOT_SENT
qmerrno.h, 1581
eQCWWAN_ERR_QMI_ARG_TOO_LONG
qmerrno.h, 1579
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED
qmerrno.h, 1580
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK
qmerrno.h, 1580
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED

qmerrno.h, 1581
eQCWWAN_ERR_QMI_CALL_FAILED
qmerrno.h, 1579
eQCWWAN_ERR_QMI_CARD_BUSY_RSP
qmerrno.h, 1582
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED
qmerrno.h, 1581
eQCWWAN_ERR_QMI_CAT_END
qmerrno.h, 1582
eQCWWAN_ERR_QMI_CAT_START
qmerrno.h, 1582
eQCWWAN_ERR_QMI_CAUSE_CODE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED
qmerrno.h, 1579
eQCWWAN_ERR_QMI_CONNECT
qmerrno.h, 1578
eQCWWAN_ERR_QMI_DEVICE_IN_USE
qmerrno.h, 1579
eQCWWAN_ERR_QMI_DEVICE_NOT_READY
qmerrno.h, 1580
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL
qmerrno.h, 1580
eQCWWAN_ERR_QMI_DISABLED
qmerrno.h, 1580
eQCWWAN_ERR_QMI_ENCODING
qmerrno.h, 1580
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE
qmerrno.h, 1582
eQCWWAN_ERR_QMI_EVENT_REG_FAILED
qmerrno.h, 1582
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL
qmerrno.h, 1581
eQCWWAN_ERR_QMI_FDN_RESTRICT
qmerrno.h, 1581
eQCWWAN_ERR_QMI_FLOW_SUSPENDED
qmerrno.h, 1580
eQCWWAN_ERR_QMI_GENERAL
qmerrno.h, 1580
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED
qmerrno.h, 1581
eQCWWAN_ERR_QMI_IFACE
qmerrno.h, 1578
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE
qmerrno.h, 1581
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INCORRECT_PIN
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE
qmerrno.h, 1581
eQCWWAN_ERR_QMI_INJECT_TIMEOUT
qmerrno.h, 1581
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INTERNAL
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_ARG
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD
qmerrno.h, 1582
eQCWWAN_ERR_QMI_INVALID_HANDLE
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_ID
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_INDEX
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_OPERATION
qmerrno.h, 1581
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_PINID
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_PROFILE
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_QMI_CMD
qmerrno.h, 1581
eQCWWAN_ERR_QMI_INVALID_QOS_ID
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_TECH_PREF
qmerrno.h, 1579
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP
qmerrno.h, 1582
eQCWWAN_ERR_QMI_INVALID_TRANSITION
qmerrno.h, 1580
eQCWWAN_ERR_QMI_INVALID_TX_ID
qmerrno.h, 1579
eQCWWAN_ERR_QMI_MALFORMED_MSG
qmerrno.h, 1579
eQCWWAN_ERR_QMI_MAX
qmerrno.h, 1581

eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT
qmerrno.h, 1580
eQCWWAN_ERR_QMI_MISSING_ARG
qmerrno.h, 1579
eQCWWAN_ERR_QMI_MSG_BLOCKED
qmerrno.h, 1581
eQCWWAN_ERR_QMI_NETWORK_ABORTED
qmerrno.h, 1581
eQCWWAN_ERR_QMI_NETWORK_NOT_READY
qmerrno.h, 1580
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_NO_EFFECT
qmerrno.h, 1579
eQCWWAN_ERR_QMI_NO_ENTRY
qmerrno.h, 1580
eQCWWAN_ERR_QMI_NO_FREE_PROFILE
qmerrno.h, 1579
eQCWWAN_ERR_QMI_NO_MEMORY
qmerrno.h, 1579
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND
qmerrno.h, 1579
eQCWWAN_ERR_QMI_NO_RADIO
qmerrno.h, 1581
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION
qmerrno.h, 1581
eQCWWAN_ERR_QMI_NO_THRESHOLDS
qmerrno.h, 1579
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_NOT_PROVISIONED
qmerrno.h, 1579
eQCWWAN_ERR_QMI_NOT_SUPPORTED
qmerrno.h, 1581
eQCWWAN_ERR_QMI_OFFSET
qmerrno.h, 1579
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED
qmerrno.h, 1579
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED
qmerrno.h, 1579
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE
qmerrno.h, 1581
eQCWWAN_ERR_QMI_OUT_OF_CALL
qmerrno.h, 1579
eQCWWAN_ERR_QMI_PIN_BLOCKED
qmerrno.h, 1580
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED
qmerrno.h, 1580
eQCWWAN_ERR_QMI_POLICY_MISMATCH
qmerrno.h, 1581
eQCWWAN_ERR_QMI_REQ
qmerrno.h, 1578
eQCWWAN_ERR_QMI_REQ_SCH
qmerrno.h, 1578
eQCWWAN_ERR_QMI_REQ_TO
qmerrno.h, 1578
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED
qmerrno.h, 1580
eQCWWAN_ERR_QMI_RSP
qmerrno.h, 1578
eQCWWAN_ERR_QMI_RSP_TO
qmerrno.h, 1578
eQCWWAN_ERR_QMI_SEGMENT_ORDER
qmerrno.h, 1581
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG
qmerrno.h, 1581
eQCWWAN_ERR_QMI_SESSION_INACTIVE
qmerrno.h, 1580
eQCWWAN_ERR_QMI_SESSION_INVALID
qmerrno.h, 1580
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP
qmerrno.h, 1580
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND
qmerrno.h, 1581
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED
qmerrno.h, 1580
eQCWWAN_ERR_QMI_SMSC_ADDR
qmerrno.h, 1581
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE
qmerrno.h, 1581
eQCWWAN_ERR_QMI_TPDU_TYPE
qmerrno.h, 1581
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION
qmerrno.h, 1579
eQCWWAN_ERR_QMI_UNKNOWN
qmerrno.h, 1580
eQCWWAN_ERR_QMI_WIDTH
qmerrno.h, 1582
eQCWWAN_ERR_RESET
qmerrno.h, 1579
eQCWWAN_ERR_SWICM_AM_VERS_ERROR
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_END
qmerrno.h, 1582
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID

qmerrno.h, 1581
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_SOCKET_IN_USE
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_START
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_TIMEOUT
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_V4DWN_V6DWN
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_V4DWN_V6UP
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_V4UP_V6DWN
qmerrno.h, 1581
eQCWWAN_ERR_SWICM_V4UP_V6UP
qmerrno.h, 1581
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED
qmerrno.h, 1582
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND
qmerrno.h, 1582
eQCWWAN_ERR_SWIDCS_END
qmerrno.h, 1582
eQCWWAN_ERR_SWIDCS_FILEIO_ERR
qmerrno.h, 1582
eQCWWAN_ERR_SWIDCS_IOCTL_ERR
qmerrno.h, 1582
eQCWWAN_ERR_SWIDCS_START
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_END
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_SECBOOT_IMAGE_NOT_SIGNED
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_SECBOOT_INVALID_CERT_CHAIN
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_TOO_MANY_FILES
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_INVALID_PATH
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_OPENING_DIR
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_OPENING_FILE
qmerrno.h, 1582
eQCWWAN_ERR_SWIIM_START
qmerrno.h, 1582
eQCWWAN_ERR_SWISM_END
qmerrno.h, 1582
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND
qmerrno.h, 1582
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED
qmerrno.h, 1582
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG
qmerrno.h, 1582
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED
qmerrno.h, 1582
eQCWWAN_ERR_SWISMS_START
qmerrno.h, 1582
eQDL_HW_FLOW_DISABLE
lite-fw.h, 1395
eQDL_HW_FLOW_ENABLE
lite-fw.h, 1395
eQDL_HW_FLOW_INIT
lite-fw.h, 1395
eQDL_HW_FLOW_Unknown
lite-fw.h, 1395
eQDL_MODE_INIT
lite-fw.h, 1396
eQDL_MODE_TTYUSB
lite-fw.h, 1396
eQDL_MODE_UART0

lite-fw.h, 1396
eQDL_MODE_Unknown
lite-fw.h, 1396
eQMI_LOC_SESS_STATUS_FAILURE
loc.h, 1409
eQMI_LOC_SESS_STATUS_IN_PROGRESS
loc.h, 1409
eQMI_LOC_SESS_STATUS_SUCCESS
loc.h, 1409
eQMI_LOC_SESS_STATUS_TIMEOUT
loc.h, 1409
eQOS
common.h, 1291
eREQ
common.h, 1292
eRMS
common.h, 1291
eRSP
common.h, 1292
eSAR
common.h, 1291
eSDP_FWDWL_ERR_END
lite-fw.h, 1395
eSDP_FWDWL_ERR_FAIL
lite-fw.h, 1394
eSDP_FWDWL_ERR_FW_UPGRADE
lite-fw.h, 1394
eSDP_FWDWL_ERR_FW_VERSION_FAIL
lite-fw.h, 1395
eSDP_FWDWL_ERR_GENERAL
lite-fw.h, 1394
eSDP_FWDWL_ERR_INVALID_DEV
lite-fw.h, 1394
eSDP_FWDWL_ERR_INVALID_PATH
lite-fw.h, 1394
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED
lite-fw.h, 1394
eSDP_FWDWL_ERR_PATH_TOO_LONG
lite-fw.h, 1394
eSDP_FWDWL_ERR_PRI_FAIL
lite-fw.h, 1394
eSDP_FWDWL_ERR_SDK
lite-fw.h, 1394
eSDP_FWDWL_ERR_SDP_TIMEOUT
lite-fw.h, 1395
eSDP_FWDWL_ERR_SET_CBK
lite-fw.h, 1394
eSDP_FWDWL_ERR_TIMEOUT
lite-fw.h, 1394
eSDP_FWDWL_SUCCESS
lite-fw.h, 1394
eSMS
common.h, 1291
eSWIAUDIO
common.h, 1291
eSWIDMS
common.h, 1291
eSWILOC
common.h, 1291
eSWIM2MCMD
common.h, 1291
eSWIM2MCMD_AVC2
common.h, 1291
eSWIOMA
common.h, 1291
eSWIOMAEXT
common.h, 1291
eTIMEOUT_10_S
common.h, 1292
eTIMEOUT_20_S
common.h, 1292
eTIMEOUT_2_S
common.h, 1292
eTIMEOUT_300_S
common.h, 1292
eTIMEOUT_30_S
common.h, 1292
eTIMEOUT_5_S
common.h, 1292
eTIMEOUT_60_S
common.h, 1292
eTIMEOUT_8_S
common.h, 1292
eTIMEOUT_DEFAULT
common.h, 1292
eTMD
common.h, 1291
eUIM
common.h, 1291
eVOICE
common.h, 1291
eWDA
common.h, 1291
eWDS
common.h, 1291
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID-_IDENT_FOR_PROFILE
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT-_DEFINED
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_-PROFILES
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFIL-E_FAMILY
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_F-LAG_SET
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_N-OT_SET
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_END

qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_FAIL
qmerrno.h, 1583
eWDS_ERR_PROFILE_REG_RESULT_LIST_END
qmerrno.h, 1583
EAWTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
ECIOThresListLen
nas_ECIOThresh, 224
ECTCallState
voice_ECTNum, 1234
eCTL_IND
QmuxInternal.h, 1592
eCTL_REQ
QmuxInternal.h, 1592
eCTL_RSP
QmuxInternal.h, 1592
eInvalidService
QmuxInternal.h, 1592
eLOG_LEVEL
common.h, 1291
EMTlv
NASQmiCbkNasSystemSelPrefInd, 402
eQCWWANError
qmerrno.h, 1578
eQMI_SVC
common.h, 1291
ERI_DATA_MAX_SIZE
dms.h, 1308
ESATlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
ESAWTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
ESNString
unpack_dms_GetDeviceSerialNumbers_t, 752
ETWSPLMNInfo
eTWSPLMNInfoTlv, 90
eTWSPLMNInfoTlv, 90
ETWSPLMNInfo, 90
TlvPresent, 90
ETWSPLMNTlv
unpack_sms_SetNewSMSCallback_ind_t, 1016
ETWSTlv
unpack_sms_SetNewSMSCallback_ind_t, 1017
eTimeout
common.h, 1291
eValid
LibPackTFTIDParams, 157
EarMute
pack_audio_SLQSSetAudioProfile_t, 411
pack_swiaudio_SLQSSetM2MAVMute_t, 546
unpack_audio_SLQSGetAudioProfile_t, 732
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1033
unpack_swiaudio_SLQSGetM2MAVMute_t, 1035
earfcn
nas_infoInterFreq, 248
nas_LTEInfoIntrafreq, 260
nas_ltePCI, 269
nas_umtsLTENbrCell, 375
earfcn0
nas_lteEARFCN, 253
earfcn1
nas_lteEARFCN, 253
earlyWakeupTime
dms_PSMEarlyWakeupTimeTlv, 72
ecio
cdmaSSInfo, 57
hdrSSInfo, 96
nas_ecioListElement, 224
nas_rxInfo, 346
nas_UMTSExtInfo, 370
nas_UMTSInfo, 372
tdscdmaSigInfoExt, 684
ecioDelta
nas_SLQSSignalStrengthsIndReq, 355
ecioInfo
nas_SLQSSignalStrengthsInformation, 357
ecioList
unpack_nas_SLQSGetSignalStrength_t, 914
ecioListLen
unpack_nas_SLQSGetSignalStrength_t, 914
ecioThresholdList
nas_SLQSSignalStrengthsIndReq, 356
ecioThresholdListLen
nas_SLQSSignalStrengthsIndReq, 356
edrxCiotLteModeTlv

unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 935
edrxCycleLengthTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 935
edrxEnableTypeTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 935
edrxEnabled
nas_EdrxEnableType, 226
edrxPagingTimeWindowTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 935
edrxPtw
nas_EdrxPagingTimeWindow, 227
edrxRatType
nas_EdrxRatType, 227
edrxRatTypeTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 935
egprsSupp
nas_GSMSysInfo, 238
egprsSuppValid
nas_GSMSysInfo, 238
ehrpdMTUSize
swidms_ehrpdMTUSizeTlv, 678
eidVal
uim_eidInfo, 697
eidValLen
uim_eidInfo, 697
elevation
loc_satelliteInfo, 179
EmerMode
nas_EmerModeTlv, 228
NASEmergencyModeTlv, 388
emmConnState
nas_LTEInfo, 258
emmState
nas_LTEInfo, 258
emmSubState
nas_LTEInfo, 258
Enable
pack_swiaudio_SLQSSetM2MAudioLPBK_t, 543
enable
pack_pds_SetPDSState_t, 519
pack_qos_SLQSSetQosEventCallback_t, 530
pack_swidms_SLQSSwiDmsSetHWWatchdog_t, 552
swidms_SwiDmsGetHWWatchdog, 682
EnableState
unpack_dms_PSMCfgChange_ind_t, 769
enableState
dms_PSMEnableStateTlv, 73
EnableStateInd
dms_PSMEnableStateIndTlv, 73
enabled
pack_rms_SetSMSWake_t, 530
unpack_rms_GetSMSWake_t, 1011
unpack_wds_GetMobileIPProfile_t, 1155
EnabledStatus
unpack_pds_SetPdsState_Ind_t, 981
EncryptProt
nas_protocolSubtypeElement, 322
EncryptedPIN1
pack_uim_ChangePin_t, 565
pack_uim_SetPinProtection_t, 567
pack_uim_UnblockPin_t, 584
EndPointType
qos_BindDataPortPeripheralEndPointID_t, 650
EndProactiveSession
cat_EndProactiveSessionTlv, 55
EngineState
unpack_pds_SLQSGetGPSStateInfo_t, 986
engineState
unpack_loc_EngineState_Ind_t, 850
ep_type
pack_wda_GetDataFormat_t, 606
eqmiCbkSetStatus
sms.h, 1613
eriData
eriDataparams, 90
eriDataLen
eriDataparams, 90
eriDataparams, 90
eriData, 90
eriDataLen, 90
eriFile
unpack_dms_SLQSGetERIFile_t, 791
Error
voice_VoiceUssdNoWaitErrorTlv, 1251
error
GetClientIdRspMsg, 95
ReleaseClientIdRspMsg, 651
ResultTlv, 652
unpack_wds_GetLastMobileIPError_t, 1152
errorClass
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
ErrorCode
imsa_IMSRegStatusErrorCodeInfo, 111
ErrorCodeData
imsa_RatHandoverStatusInfo, 113
ErrorCodeLen
imsa_RatHandoverStatusInfo, 113
errorMask
unpack_cat_SetCATEventCallback_t, 735
errorRate
nas_errorRateListElement, 229
errorRateInfo
nas_SLQSSignalStrengthsInformation, 357
errorRateList
unpack_nas_SLQSGetSignalStrength_t, 914
errorRateListLen
unpack_nas_SLQSGetSignalStrength_t, 914
errorState
slotInf, 657

uim_slotInfo, 722
esn
unpack_dms_GetSerialNumbers_t, 765
esnSize
unpack_dms_GetDeviceSerialNumbers_t, 751
EspSpi
unpack_qos_swiQosFilter_t, 1003
EtwsMessageInfo
sMSEtwsMessageTlv, 673
event
dms_UimStatusTlv, 77
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 999
unpack_qos_SLQSSetQosStatusCallback_ind_t, 1000
event_Index
unpack_cat_SetCatEventCallback_ind_t, 734
EventID
cat_commonEventTlv, 54
EventLength
cat_commonEventTlv, 54
eventMask
pack_cat_SetCATEventCallback_t, 414
pack_uim_SLQSUIMEventRegister_t, 571
unpack_uim_SLQSUIMEventRegister_t, 1074
eventRegister
pack_loc_EventRegister_t, 448
eventType
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 1055
evrcCapability
voice_prefVoiceSO, 1243
executingImage
FMSImageIDEntries, 93
exponent
unpack_qos_pktErrRate_t, 991
extBit
voice_calledPartySubAdd, 1216
extDispInfo
voice_extDispRecInfo, 1234
extDispInfoLen
voice_extDispRecInfo, 1234
ExtErrorCode
PackCreateProfileOut, 636
extPowerState
pack_loc_SetExtPowerState_t, 448
extendedErrorCode
unpack_wds_SLQSDeleteProfile_t, 1163

FAILURE
QmuxInternal.h, 1592
FCTlv
unpack_voice_USSDNoWaitIndicationCallback_ind_t, 1130
FDNStatus
uim_UIMGetFDNStatus, 723
FIRMWARE_BCHDATESIZE
lite-fw.h, 1391
FLOAT
SwiDataTypes.h, 1656
FMSImageElement, 91
buildId, 91
buildIdLength, 91
imageId, 91
imageType, 91
FMSImageIDEntries, 92
executingImage, 93
imageIDElement, 93
imageIDSize, 93
imageType, 93
maxImages, 93
FMSImageIdElement, 91
buildID, 92
buildIDLength, 92
failureCount, 92
imageID, 92
storageIndex, 92
FMSImageList, 93
imageIDEntries, 93
listSize, 93
FMSPrefImageList, 94
listEntries, 94
listSize, 94
FOTAUpdate
pack_swioma_SLQSOMADMSetSettings_t, 561
pack_swioma_SLQSOMADMSetSettingsExt_t, 562
unpack_swioma_SLQSOMADMGetSettings_t, 1063
FOTAdownload
pack_swioma_SLQSOMADMSetSettings_t, 561
pack_swioma_SLQSOMADMSetSettingsExt_t, 562
unpack_swioma_SLQSOMADMGetSettings_t, 1063
FUMOEnable
pack_swioma_SLQSOMADMSetSettingsExt_t, 562
facility
pack_dms_UIMGetControlKeyStatus_t, 430
pack_dms_UIMSetControlKeyProtection_t, 431
pack_dms_UIMUnblockControlKey_t, 432
facilityCk
pack_dms_UIMSetControlKeyProtection_t, 431
pack_dms_UIMUnblockControlKey_t, 432
facilityState
pack_dms_UIMSetControlKeyProtection_t, 431
unpack_dms_UIMGetControlKeyStatus_t, 806
FailErrCode
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 837
Failure_Cause
voice_VoiceUssdNoWaitFailCauseTlv, 1252
failureCount
FMSImageIdElement, 92
family
pack_wds_GetDefaultProfileNum_t, 612

pack_wds_SetDefaultProfileNum_t, 618
feature
uim_depersonalizationInformation, 697
uim_personalizationStatus, 706
fileID
uim_fileAttributes, 701
uim_fileInfo, 702
fileIndex
pack_uim_ReadTransparent_t, 566
pack_uim_SLQSUIMGetFileAttributes_t, 573
pack_uim_SLQSUIMReadRecord_t, 577
pack_uim_SLQSUIMWriteRecord_t, 583
pack_uim_SLQSUIMWriteTransparent_t, 584
fileSize
uim_fileAttributes, 701
fileType
uim_fileAttributes, 701
fill_pack_ctx
common.h, 1292
fill_sdu_hdr
common.h, 1292
filterId
LibPackTFTIDParams, 157
fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 557
unpack_swiloc_SwiLocGetAutoStart_t, 1054
fix_rate_reported
unpack_swiloc_SwiLocGetAutoStart_t, 1054
fix_type
pack_swiloc_SwiLocSetAutoStart_t, 557
unpack_swiloc_SwiLocGetAutoStart_t, 1054
fix_type_reported
unpack_swiloc_SwiLocGetAutoStart_t, 1054
fixAccuracy
pack_pds_StartPDSTrackingSessionExt_t, 527
fixCount
pack_pds_StartPDSTrackingSessionExt_t, 527
fixInterval
pack_pds_StartPDSTrackingSessionExt_t, 527
fixTimeout
pack_pds_StartPDSTrackingSessionExt_t, 527
flags
CtlTransactionHdr, 60
QmiTransactionHdr, 646
sensorData_t, 654
Flow_control
unpack_wda_GetDataFormat_t, 1136
flowLabel
LibPackTFTIDParams, 157
fms.h, 1367
GetValidFwPriCombinations, 1368
pack_fms_GetImagesPreference, 1369
pack_fms_GetStoredImages, 1369
pack_fms_SetImagesPreference, 1369
unpack_fms_GetImagesPreference, 1369
unpack_fms_GetStoredImages, 1370
unpack_fms_SetImagesPreference, 1370
Forbidden
nas_QmiNas3GppNetworkInfo, 324
forbiddenNwInstLen
nas_ForbiddenNetworks3GPP, 230
forceFlag
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 522
ForceRev0
unpack_nas_GetCDMANetworkParameters_t, 877
format
sMSTransferRouteMTMessageInfo, 676
ForwardMac
nas_protocolSubtypeElement, 322
fqdnAddr
wds_PCSCFFQDNAddress, 1267
fqdnLen
wds_PCSCFFQDNAddress, 1267
FragmentHeader
MBIM_COMMAND_DONE_MSG, 185
MBIM_COMMAND_MSG, 186
MBIM_FRAGMENT_MSG, 186
MBIM_INDICATE_STATUS_MSG, 187
freeSlots
sms_maxStorageSizeResp, 663
freq
nas_PhyCaAggPcellInfo, 311
nas_PhyCaAggScellIndType, 312
nas_PhyCaAggScellInfo, 315
nas_QmisNasSlqsNasPCICellInfo, 326
NASPhyCaAggPcellInfo, 395
NASPhyCaAggScellArray, 396
NASPhyCaAggScellIndType, 398
NASPhyCaAggScellInfo, 399
freqsLen
nas_LTEInfoInterfreq, 258
nas_LTEInfoNeighboringGSM, 261
nas_LTEInfoNeighboringWCDMA, 262
fromServiceId
sms_BroadcastConfig, 658
fullPath
_litefw_FirmwareFileInfo, 40
fumoResultCode
unpack_swioma_SLQSOMADMGetSessionInfoExt_t, 1061
fumoState
unpack_swioma_SLQSOMADMGetSessionInfoExt_t, 1061
function
pack_swiloc_SwiLocSetAutoStart_t, 557
unpack_swiloc_SwiLocGetAutoStart_t, 1054
function_reported
unpack_swiloc_SwiLocGetAutoStart_t, 1054
FwAutoCheck
unpack_swioma_SLQSOMADMGetSettings_t, 1063
FwAutoSDM
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042

FwAvailability
unpack_swioma_SLQSOMADMStartSession_t, 1064
FwPromptUpdate
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
FwPromptdownload
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
fwdloadsize
unpack_omaDmFotaTlv_t, 972
fwloadComplete
unpack_omaDmFotaTlv_t, 972
fwvers
unpack_dms_SLQSSwiGetFirmwareCurr_t, 796

gDlBitRate
LibPackQosClassID, 156
GET_CLIENT_ID
QmuxInternal.h, 1592
GPRSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 1176
GPSLPM
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
GPSSel
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
GSMRSSIThreshListLen
nas_GSMRSSIThresh, 235
GSMSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 946
gUlBitRate
LibPackQosClassID, 156
GWAOPTlv
NASQmiCbkNasSystemSelPrefInd, 402
GWAcqOrderPref
nas_GWAcqOrderPrefTlv, 239
NASGWAcqOrderPrefTlv, 390
GWAddressV4
unpack_wds_SLQSGetRuntimeSettings_t, 1176
gcdumpString
crashInformation, 59
gcdumpStrlen
crashInformation, 59
Generator
pack_audio_SLQSGetAudioProfile_t, 408
pack_audio_SLQSGetAudioVolTLBConfig_t, 408
pack_audio_SLQSSetAudioProfile_t, 411
pack_audio_SLQSSetAudioVolTLBConfig_t, 412
pack_swiaudio_SLQSGetM2MAudioVolume_t, 541
pack_swiaudio_SLQSSetM2MAudioVolume_t, 545
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1033
geoSysIdx
nas_AddCDMASysInfo, 195
nas_AddSysInfo, 196
geranArfcn
nas_geranInstArr, 232
nas_geranInstInfo, 233
geranBsicBcc
nas_geranInstArr, 232
nas_geranInstInfo, 233
geranBsicNcc
nas_geranInstArr, 232
nas_geranInstInfo, 233
geranInst
nas_UMTSExtInfo, 370
nas_UMTSInfo, 372
geranInstElement
nas_UMTSExtInfo, 371
GeranInstInfo
nas_UMTSInfo, 372
geranRank
nas_geranInstArr, 232
geranRssi
nas_geranInstArr, 233
nas_geranInstInfo, 233
get_version
common.h, 1292
getAllCallInfo
voice_arrCallInfo, 1209
GetAllCallRmtPtyName
voice_arrRemotePartyName, 1211
GetClientIdReqMsg, 94
length, 94
messageId, 94
QmuxInternal.h, 1593
serviceType, 94
tlvLength, 94
tlvType, 94
GetClientIdRspMsg, 94
clientId, 95
error, 95
length, 95
messageId, 95
QmuxInternal.h, 1593
result, 95
serviceType, 95
tlv2Length, 95
tlv2Type, 95
tlvLength, 95
tlvType, 95
GetCustomFeatureV2
unpack_dms_GetCustFeaturesV2_t, 743
GetFlashSectorSize
lite-fw.h, 1396
GetValidFwPriCombinations
fms.h, 1368
glo_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
glo_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
glo_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987

glo_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
GlobalCellID
nas_QmisNasSlqsNasPCICellInfo, 326
globalCellId
nas_LTEInfoIntrafreq, 260
GlobalUIInfoTlvPresent
unpack_dsd_SLQSUIInfo_Ind_t, 815
glog
common.h, 1294
gloglvl
common.h, 1294
gnssSvId
loc_satelliteInfo, 179
gnssSvUsedList
loc_svUsedforFix, 182
gnssSvUsedList_len
loc_svUsedforFix, 182
Gpp2TimeZone
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
GppNetworkDSTAdjustment
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
GppTimeZone
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
gps_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
gps_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
gps_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
gps_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
GpsEnable
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
gpsTimeOfWeekMs
loc_gpsTime, 170
gpsWeek
loc_gpsTime, 170
grntDownlinkBitrate
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
grntUplinkBitrate
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
gsmAmrStat
voice_curAMRConfig, 1231
GsmCellInfo
nas_lteGsmCellInfo, 256
gsmUmtsDl
pack_nas_SLQSNasSwiIndicationRegister_t, 491
gsmUmtsUl
pack_nas_SLQSNasSwiIndicationRegister_t, 491
guaranteedRate
unpack_qos_dataRate_t, 989
gwAddressV6
wds_IPV6GWAddressInfo, 1266
gwV6PrefixLen
wds_IPV6GWAddressInfo, 1266
gyroData
pack_loc_SLQSLOCInjectSensorData_t, 456
gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 456
gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 457

HASPI
unpack_wds_GetMobileIPProfile_t, 1155
HAState
unpack_wds_GetMobileIPProfile_t, 1155
HDOP
loc_precisionDilution, 177
HDRECIOThreshListLen
nas_HDRECIOThresh, 240
HDRIOThreshListLen
nas_HDRIOThresh, 240
HDRRSSIThreshListLen
nas_HDRRSSIThresh, 242
HDRSINRThresListLen
nas_HDRSINRThresh, 242
HDRSINRThreshListLen
nas_HDRSINRThreshold, 243
HDRSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 946
HWVersion
nas_DeviceConfigDetail, 222
HardwareControlledMode
unpack_dms_GetPower_t, 763
has_LpmFlag
unpack_dms_SLQSDmsSwiGetPCInfo_t, 777
has_PdpStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
has_PersistentLpm
unpack_dms_SLQSDmsSwiGetPCInfo_t, 777
has_PositionDataNMEA
unpack_pds_SetEventReport_Ind_t, 980
has_PowerOffMode
unpack_dms_SLQSDmsSwiGetPCInfo_t, 777
has_RatHandoverStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
has_RegStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
has_ServiceStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
has_SessionStatus
unpack_pds_SetEventReport_Ind_t, 980
has_Wdisable
unpack_dms_SLQSDmsSwiGetPCInfo_t, 777
has_accelTemp
pack_loc_SLQSLOCInjectSensorData_t, 457
has_acceleroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 457
has_accleroData
pack_loc_SLQSLOCInjectSensorData_t, 457

has_altitudeSrcInfo
pack_loc_SLQSLOCInjectPosition_t, 454
has_altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 454
has_altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 454
has_confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 458
has_dLatitude
unpack_pds_SetEventReport_Ind_t, 980
has_dLongitude
unpack_pds_SetEventReport_Ind_t, 980
has_gyroData
pack_loc_SLQSLOCInjectSensorData_t, 457
has_gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 457
has_gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 457
has_horConfidence
pack_loc_SLQSLOCInjectPosition_t, 454
has_horReliability
pack_loc_SLQSLOCInjectPosition_t, 454
has_horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 454
has_latitude
pack_loc_SLQSLOCInjectPosition_t, 454
has_longitude
pack_loc_SLQSLOCInjectPosition_t, 454
has_opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 457
has_posSrc
unpack_pds_SetEventReport_Ind_t, 980
has_positionSrc
pack_loc_SLQSLOCInjectPosition_t, 454
has_rawHorConfidence
pack_loc_SLQSLOCInjectPosition_t, 454
has_rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 454
has_serverAddrTypeMask
pack_loc_SLQSLOCGetServer_t, 450
has_timestampAge
pack_loc_SLQSLOCInjectPosition_t, 454
has_timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 454
has_vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 454
has_vertRelicability
pack_loc_SLQSLOCInjectPosition_t, 454
has_vertUnc
pack_loc_SLQSLOCInjectPosition_t, 455
hdrActiveProt
nas_HDRSysInfo, 245
hdrActiveProtValid
nas_HDRSysInfo, 245
hdrHybrid
nas_detailSvcInfo, 221
HdrPersonality
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
hdrPersonality
nas_HDRSysInfo, 245
NASServingSystemInfo, 405
hdrPersonalityValid
nas_HDRSysInfo, 245
hdrSSInfo, 95
ecio, 96
io, 96
rssi, 96
sinr, 96
hdrSrvStatus
nas_detailSvcInfo, 221
headerType
_litefw_FirmwareFileInfo, 40
healthStatus
loc_satelliteInfo, 179
helper_get_ctl_resp_ctx
ctl.h, 1294
helper_get_error_code
common.h, 1292
helper_get_error_reason
common.h, 1292
helper_get_req_str
common.h, 1292
helper_get_resp_ctx
common.h, 1293
helper_get_wda_resp_ctx
wda.h, 1755
helper_get_xid
common.h, 1293
helper_isBootLoader_DebugEnabled
common.h, 1293
helper_set_log_func
common.h, 1293
helper_set_log_lvl
common.h, 1293
hfaMaxRetry
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
hfaRetryIndex
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
hfaRetryInterval
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
hfaStatus
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
hiddenKey
uim_UIMGetHiddenKeyStatus, 723
homeOrigVoiceSO
voice_prefVoiceSO, 1243
homePageVoiceSO
voice_prefVoiceSO, 1243
horConfidence
pack_loc_SLQSLOCInjectPosition_t, 455
horReliability

pack_loc_SLQSLOCInjectPosition_t, 455
horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 455
HorizontalUncertainty
unpack_pds_SLQSGetGPSStateInfo_t, 987
hostID
pack_dms_SLQSSwiSetHostDevInfo_t, 427
unpack_dms_SLQSSwiGetHostDevInfo_t, 799
hotSwap
uim_hotSwapStatus, 705
hotSwapLength
uim_hotSwapStatus, 705
hour
nas_timeInfo, 367
nas_UniversalTime, 377
hrpdMTUSize
swidms_hrpdMTUSizeTlv, 678
hsCallStatus
nas_WCDMASysInfo, 385
hsCallStatusValid
nas_WCDMASysInfo, 385
hsInd
nas_WCDMASysInfo, 385
hsIndValid
nas_WCDMASysInfo, 385
hwType
wds_DHCPv4HWConfig, 1260
wdsDhcpv4HwConfig, 1275
hwVer
unpack_dms_GetHardwareRevision_t, 757

IDTlv
unpack_ims_SLQSUserCfgCallBack_ind_t, 831
IF_TYPE_SIZE
QmuxInternal.h, 1592
iFaceTab
swiaudio_PCMparams, 677
iFaceTabLen
swiaudio_PCMparams, 677
iLTEbandValue
nas_PhyCaAggPcellInfo, 311
nas_PhyCaAggScellInfo, 315
NASPhyCaAggPcellInfo, 395
NASPhyCaAggScellInfo, 399
IMCNflag
unpack_wds_SLQSGetRuntimeSettings_t, 1177
IMEIString
unpack_dms_GetDeviceSerialNumbers_t, 752
IMG_MASK_CLEAR
lite-fw.h, 1391
IMG_MASK_GENERIC
lite-fw.h, 1391
IMG_MASK_MDM
lite-fw.h, 1391
IMG_MASK_PRI
lite-fw.h, 1391
IMSInfo
sMSOnIMSTlv, 675
IMSRegistration
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 839
IMSRegistrationError
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 839
IMSTlv
unpack_sms_SetNewSMSCallback_ind_t, 1017
INT32
SwiDataTypes.h, 1656
INT8
SwiDataTypes.h, 1656
INVALID_CLIENT_ID
QmuxTransport.h, 1594
IOThresListLen
nas_IOThresh, 249
IPAddress
nas_DataStatusDetail, 218
IPAddressV6
ipv6AddressInfo, 118
wds_IPV6AddressInfo, 1265
IPFamSupport
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
IPFamilyPreference
pack_wds_SLQSSetIPFamilyPreference_t, 629
unpack_wds_SLQSGetRuntimeSettings_t, 1177
IPSECSPI
LibPackTFTIDParams, 157
IPV6AddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 1177
IPV6GWAddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 1177
IPV6PrefixLen
ipv6AddressInfo, 118
wds_IPV6AddressInfo, 1265
IPv4
unpack_wds_SLQSGetRuntimeSettings_t, 1177
IPv4Addr
loc_IPv4Config, 173
wds_IPv4AdTlv, 1264
IPv4AddrTlv
unpack_wds_DHCPv4ClientLease_ind_t, 1142
IPv4DstAddr
unpack_qos_swiQosFilter_t, 1003
IPv4Port
loc_IPv4Config, 173
IPv4SrcAddr
unpack_qos_swiQosFilter_t, 1003
IPv4Tos
unpack_qos_swiQosFilter_t, 1003
IPv6Addr
loc_IPv6Config, 174
IPv6DstAddr
unpack_qos_swiQosFilter_t, 1003
IPv6Label
unpack_qos_swiQosFilter_t, 1003
IPv6Port
loc_IPv6Config, 174

IPv6SrcAddr
unpack_qos_swiQosFilter_t, 1003
IPv6TrafCls
unpack_qos_swiQosFilter_t, 1003
Id
unpack_qos_swiQosFilter_t, 1003
id
loc_BdsSV, 164
loc_SV, 181
nas_CSGID, 213
nas_CSGInfo, 214
pack_dms_UIMChangePIN_t, 429
pack_dms_UIMSetPINProtection_t, 432
pack_dms_UIMUnblockPIN_t, 433
pack_dms_UIMVerifyPIN_t, 433
unpack_qos_QosFlowInfoState_t, 993
unpack_qos_SLQSSetQosStatusCallback_ind_t, 1000
id_length
DMScustSettingInfo, 79
IdleState
nas_protocolSubtypeElement, 322
IfType
QMUX_HDR, 648
iface_id
pack_wda_GetDataFormat_t, 606
IfaceID
qos_BindDataPortPeripheralEndPointID_t, 650
image_info_t, 96
buildID, 97
buildIDLen, 97
imageType, 97
uniqueID, 97
imageID
FMSImageIdElement, 92
imageIDElement
FMSImageIDEntries, 93
imageIDEntries
FMSImageList, 93
imageIDSize
FMSImageIDEntries, 93
imageId
FMSImageElement, 91
imageList
unpack_fms_GetStoredImages_t, 817
ImageListSize
unpack_fms_GetImagesPreference_t, 816
imageListSize
pack_fms_SetImagesPreference_t, 437
imageMask
_litefw_FirmwareFileInfo, 40
imageType
_litefw_FirmwareFileInfo, 40
FMSImageElement, 91
FMSImageIDEntries, 93
image_info_t, 97
ImageTypes
unpack_fms_SetImagesPreference_t, 817
ImageTypesSize
unpack_fms_SetImagesPreference_t, 817
imagelistSize
unpack_fms_GetStoredImages_t, 817
imei_no
unpack_dms_GetSerialNumbers_t, 765
imeiSize
unpack_dms_GetDeviceSerialNumbers_t, 752
imeiSvnSize
unpack_dms_GetDeviceSerialNumbers_t, 752
ImeiSvnString
unpack_dms_GetDeviceSerialNumbers_t, 752
imeisv_svn
unpack_dms_GetSerialNumbers_t, 765
imgType
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 798
ims.h, 1370
MAX_NAME_LEN, 1372
pack_ims_SLQSGetIMSSMSConfig, 1372
pack_ims_SLQSGetIMSUserConfig, 1373
pack_ims_SLQSGetIMSVoIPConfig, 1373
pack_ims_SLQSGetRegMgrConfig, 1373
pack_ims_SLQSGetSIPConfig, 1374
pack_ims_SLQSImsConfigIndicationRegister, 1374
pack_ims_SLQSSetIMSSMSConfig, 1375
pack_ims_SLQSSetIMSUserConfig, 1375
pack_ims_SLQSSetIMSVoIPConfig, 1376
pack_ims_SLQSSetRegMgrConfig, 1376
pack_ims_SLQSSetSIPConfig, 1377
unpack_ims_SLQSGetIMSSMSConfig, 1377
unpack_ims_SLQSGetIMSUserConfig, 1377
unpack_ims_SLQSGetIMSVoIPConfig, 1378
unpack_ims_SLQSGetRegMgrConfig, 1378
unpack_ims_SLQSGetSIPConfig, 1378
unpack_ims_SLQSImsConfigIndicationRegister, 1379
unpack_ims_SLQSImsConfigIndicationRegister_t, 1372
unpack_ims_SLQSRegMgrCfgCallBack_ind, 1379
unpack_ims_SLQSSIPCfgCallBack_ind, 1381
unpack_ims_SLQSSMSCfgCallBack_ind, 1382
unpack_ims_SLQSSetIMSSMSConfig, 1380
unpack_ims_SLQSSetIMSUserConfig, 1380
unpack_ims_SLQSSetIMSVoIPConfig, 1380
unpack_ims_SLQSSetRegMgrConfig, 1381
unpack_ims_SLQSSetSIPConfig, 1381
unpack_ims_SLQSUserCfgCallBack_ind, 1382
unpack_ims_SLQSVoIPCfgCallBack_ind, 1382
ims_AMRModeInfo, 97
amrMode, 98
TlvPresent, 98
ims_AMROctAlgnInfo, 98
amrOctAlgn, 98
TlvPresent, 98
ims_AMRWBModeInfo, 98
amrWBMode, 99
TlvPresent, 99

ims_AMRWBOctAlgnInfo, 99
amrWBOctAlgn, 99
TlvPresent, 99
ims_CSCFPortNameInfo, 99
cscfPortName, 100
TlvPresent, 100
ims_EnabAMRWBInfo, 100
amrWBEnable, 100
TlvPresent, 100
ims_EnabSCRAMRInfo, 100
scrAmrEnable, 101
TlvPresent, 101
ims_EnabSCRAMRWBInfo, 101
scrAmrWBEnable, 101
TlvPresent, 101
ims_IMSDomainInfo, 101
imsDomainName, 102
TlvPresent, 102
ims_IMSTestModeInfo, 102
imsTestMode, 102
TlvPresent, 102
ims_MinSessExpInfo, 102
minSessExp, 103
TlvPresent, 103
ims_PCSCFPortInfo, 103
priCSCFPort, 103
TlvPresent, 103
ims_PhCtxtURIInfo, 103
PhCtxtURI, 104
TlvPresent, 104
ims_RTPRTCPInactTmrDurInfo, 105
InactTmr, 105
TlvPresent, 105
ims_RngBkTmrInfo, 104
RingBkTmr, 104
TlvPresent, 104
ims_RngTmrInfo, 104
RingTmr, 105
TlvPresent, 105
ims_SIPPortInfo, 106
SIPLocalPort, 107
TlvPresent, 107
ims_SIPRegnTmrInfo, 107
TlvPresent, 107
tmrSIPRegn, 107
ims_SMSFmtInfo, 107
smsFormat, 107
TlvPresent, 108
ims_SMSoIPNwInfo, 108
smsoIPNW, 108
TlvPresent, 108
ims_SessDurInfo, 105
sessExp, 106
TlvPresent, 106
ims_SigCompEnInfo, 106
SigCompEn, 106
TlvPresent, 106
ims_SubscrTmrInfo, 108
subscrTmr, 108
TlvPresent, 108
ims_TmrT1Info, 109
TlvPresent, 109
tmrT1, 109
ims_TmrT2Info, 109
TlvPresent, 109
tmrT2, 109
ims_TmrTfInfo, 109
TlvPresent, 110
tmrTf, 110
imsDomainName
ims_IMSDomainInfo, 102
ImsFailErrCode
imsa_IMSFailErrCodeTlv, 110
ImsRegErrCode
unpack_imsa_SLQSGetIMSARegStatus_t, 834
imsRegState
nas_CommInfo, 212
ImsRegStatus
imsa_NewIMSRegStatusInfo, 112
unpack_imsa_SLQSGetIMSARegStatus_t, 834
ImsRegistered
imsa_IMSRegStatusInfo, 111
imsTestMode
ims_IMSTestModeInfo, 102
ImsVoiceSupportLte
nas_ImsVoiceSupportLteTlv, 247
imsa.h, 1383
MAX_ERROR_CODE_LEN, 1384
pack_imsa_SLQSGetIMSARegStatus, 1384
pack_imsa_SLQSGetIMSAServiceStatus, 1384
pack_imsa_SLQSRegisterIMSAIndication, 1385
unpack_imsa_SLQSGetIMSARegStatus, 1385
unpack_imsa_SLQSGetIMSAServiceStatus, 1386
unpack_imsa_SLQSImsaPdpStatusCallBack_ind, 1386
unpack_imsa_SLQSImsaRatStatusCallBack_ind, 1387
unpack_imsa_SLQSImsaRegStatusCallBack_ind, 1387
unpack_imsa_SLQSImsaSvcStatusCallBack_ind, 1387
unpack_imsa_SLQSRegisterIMSAIndication, 1388
unpack_imsa_SLQSRegisterIMSAIndication_t, 1384
imsa_IMSFailErrCodeTlv, 110
ImsFailErrCode, 110
TlvPresent, 110
imsa_IMSRegStatusErrorCodeInfo, 110
ErrorCode, 111
TlvPresent, 111
imsa_IMSRegStatusInfo, 111
ImsRegistered, 111
TlvPresent, 112
imsa_NewIMSRegStatusInfo, 112
ImsRegStatus, 112
TlvPresent, 112

imsa_RatHandoverStatusInfo, 112
ErrorCodeData, 113
ErrorCodeLen, 113
RatHandoverStatus, 113
SourceRAT, 113
TargetRAT, 113
TlvPresent, 113
imsa_SmsRatInfo, 113
SmsRatVal, 114
TlvPresent, 114
imsa_SmsSvcStatusInfo, 114
SmsSvcStatus, 114
TlvPresent, 114
imsa_UtRatInfo, 114
TlvPresent, 115
UtRatVal, 115
imsa_UtSvcStatusInfo, 115
TlvPresent, 115
UtSvcStatus, 115
imsa_VoipRatInfo, 115
TlvPresent, 116
VoipRatVal, 116
imsa_VoipSvcStatusInfo, 116
TlvPresent, 116
VoipSvcStatus, 116
imsa_VtRatInfo, 116
TlvPresent, 117
VtRatVal, 117
imsa_VtSvcStatusInfo, 117
TlvPresent, 117
VtSvcStatus, 117
imsi
unpack_dms_GetIMSI_t, 757
imsi_11_12
nas_CDMASysInfoExt, 207
imsiM1112
nas_minBasedIMSI, 281
imsiMS1
nas_minBasedIMSI, 281
imsiMS2
nas_minBasedIMSI, 281
imsiT1112
nas_trueIMSI, 368
imsiTS1
nas_trueIMSI, 368
imsiTS2
nas_trueIMSI, 368
imsiTaddrNum
nas_trueIMSI, 368
InUse
nas_QmiNas3GppNetworkInfo, 324
inUse
QmiTransaction, 646
InactTmr
ims_RTPRTCPInactTmrDurInfo, 105
includes_pcs_digit
nas_QmisNasPcsDigit, 325
index
pack_wds_GetMobileIPProfile_t, 613
pack_wds_SetDefaultProfileNum_t, 618
pack_wds_SetMobileIPProfile_t, 621
uim_UIMGetIndex, 724
unpack_qos_swiQosFilter_t, 1003
unpack_qos_swiQosFlow_t, 1007
unpack_wds_GetDefaultProfileNum_t, 1148
index1xPri
uim_cardStatus, 695
index1xSec
uim_cardStatus, 695
indexGwPri
uim_cardStatus, 695
indexGwSec
uim_cardStatus, 695
indicatorList
MbimTransport, 192
indicatorListLock
MbimTransport, 192
Info
unpack_nas_SLQSNasSwiOTAMessageCallback-_ind_t, 956
unpack_nas_SLQSSetSysSelectionPrefCallBack-_ind_t, 959
InfoInterfreq
nas_LTEInfoInterfreq, 258
informationBuffer
MbimSyncObject, 190
InformationBufferLength
MBIM_COMMAND_DONE_MSG, 185
MBIM_COMMAND_MSG, 186
MBIM_INDICATE_STATUS_MSG, 187
informationBufferLength
MbimSyncObject, 190
injectEnable
loc_accelAcceptReady, 162
loc_accelTempAcceptReady, 163
loc_gyroAcceptReady, 171
loc_gyroTempAcceptReady, 171
injectSensorDataStatus
unpack_loc_InjectSensorDataCallback_Ind_t, 859
injectTimeSyncStatus
unpack_loc_InjectTimeSyncDataCallback_Ind_t, 860
insNmrCellInfo
nas_GERANInfo, 231
instancesSize
unpack_nas_GetRFInfo_t, 883
intermediate
loc_IntermediateRptStateTlv, 173
interval
loc_MinIntervalTlv, 177
pack_pds_SetPDSDefaults_t, 519
pack_pds_SetXTRAAutomaticDownload_t, 521
pack_wds_SLQSSetWdsEventCallback_t, 630
intf
dms_UimStatusTlv, 77
Io

unpack_nas_SLQSGetSignalStrength_t, 914
io
hdrSSInfo, 96
nas_SLQSSignalStrengthsInformation, 357
ioDelta
nas_SLQSSignalStrengthsIndReq, 356
Iono_valid
unpack_pds_SLQSGetGPSStateInfo_t, 987
ipAddress
pack_wds_SetDefaultProfile_t, 617
ipFamily
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 1180
ipVersion
LibPackTFTIDParams, 158
ipaddr
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
ipaddrv6
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
ipv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 1191
ipv4GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 1191
ipv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 1191
ipv6AddressInfo, 117
IPAddressV6, 118
IPV6PrefixLen, 118
ipv6GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 1191
is3GppNw
nas_homeNwMNC3GppTlv, 246
is856SysId
nas_HDRSysInfo, 245
is856SysIdValid
nas_HDRSysInfo, 245
is_DataRate_Available
unpack_qos_swiQosFlow_t, 1007
is_EspSpi_Available
unpack_qos_swiQosFilter_t, 1003
is_IPv4DstAddr_Available
unpack_qos_swiQosFilter_t, 1003
is_IPv4SrcAddr_Available
unpack_qos_swiQosFilter_t, 1003
is_IPv4Tos_Available
unpack_qos_swiQosFilter_t, 1003
is_IPv6DstAddr_Available
unpack_qos_swiQosFilter_t, 1003
is_IPv6Label_Available
unpack_qos_swiQosFilter_t, 1003
is_IPv6SrcAddr_Available
unpack_qos_swiQosFilter_t, 1003
is_IPv6TrafCls_Available
unpack_qos_swiQosFilter_t, 1003
is_Id_Available
unpack_qos_swiQosFilter_t, 1003
is_Jitter_Available
unpack_qos_swiQosFlow_t, 1007
is_Latency_Available
unpack_qos_swiQosFlow_t, 1007
is_LteBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 786
unpack_dms_SLQSGetBandCapabilityExt_t, 790
is_LteQci_Available
unpack_qos_swiQosFlow_t, 1007
is_MaxAllowedPktSz_Available
unpack_qos_swiQosFlow_t, 1007
is_MinPolicedPktSz_Available
unpack_qos_swiQosFlow_t, 1007
is_NxtHdrProto_Available
unpack_qos_swiQosFilter_t, 1003
is_PktErrRate_Available
unpack_qos_swiQosFlow_t, 1007
is_Precedence_Available
unpack_qos_swiQosFilter_t, 1003
is_ProfileId3GPP2_Available
unpack_qos_swiQosFlow_t, 1007
is_RxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 992
is_TCPDstPort_Available
unpack_qos_swiQosFilter_t, 1003
is_TCPSrcPort_Available
unpack_qos_swiQosFilter_t, 1003
is_TdsBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 786
unpack_dms_SLQSGetBandCapabilityExt_t, 790
is_TokenBucket_Available
unpack_qos_swiQosFlow_t, 1007
is_TrafficClass_Available
unpack_qos_swiQosFlow_t, 1007
is_TranDstPort_Available
unpack_qos_swiQosFilter_t, 1004
is_TranSrcPort_Available
unpack_qos_swiQosFilter_t, 1004
is_TxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 992
is_UDPDstPort_Available
unpack_qos_swiQosFilter_t, 1004
is_UDPSrcPort_Available
unpack_qos_swiQosFilter_t, 1004
is_internet
DsdUIApnAvailableSystemInfo_t, 88
is_val_3GPP2Pri_Available
unpack_qos_swiQosFlow_t, 1007
is_val_3GPPImCn_Available
unpack_qos_swiQosFlow_t, 1007
is_val_3GPPResResidualBER_Available
unpack_qos_swiQosFlow_t, 1007
is_val_3GPPSigInd_Available
unpack_qos_swiQosFlow_t, 1007

is_val_3GPPTraHdlPri_Available
unpack_qos_swiQosFlow_t, 1007
isEmpty
voice_getAllCallInformation, 1235
isInTraffic
nas_txInfo, 369
isModByCC
voice_SUPSInfo, 1248
isNewFlow
unpack_qos_QosFlowInfoState_t, 993
isPrefDataPath
nas_GSMSrvStatusInfo, 236
nas_NR5GSerStatTlv, 298
nas_NR5GSrvStatusTlv, 299
nas_SrvStatusInfo, 360
isRadioTuned
nas_rxInfo, 346
nas_RxSigInfo, 347
isSysForbidden
nas_detailSvcInfo, 221
nas_NR5GSysInfoTlv, 302
nas_sysInfoCommon, 362
isSysForbiddenValid
nas_NR5GSysInfoTlv, 302
nas_sysInfoCommon, 362
isSysPrlMatch
nas_CDMASysInfo, 206
nas_HDRSysInfo, 245
isSysPrlMatchValid
nas_CDMASysInfo, 206
nas_HDRSysInfo, 245
IsVoiceEnabled
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
iseUICC
uim_physlotInfo, 707
Item
pack_audio_SLQSGetAudioPathConfig_t, 407
pack_audio_SLQSGetAudioVolTLBConfig_t, 408
pack_audio_SLQSSetAudioVolTLBConfig_t, 412

Jitter
unpack_qos_swiQosFlow_t, 1007
jtagAccessAllowed
unpack_swidms_SLQSSwiDmsGetSecureInfo_t, 1048

KeyExchange
nas_protocolSubtypeElement, 322

LITE_SYS_SRV_DOMAIN_CAMPED
nas.h, 1453
LITE_SYS_SRV_DOMAIN_CS_ONLY
nas.h, 1453
LITE_SYS_SRV_DOMAIN_CS_PS
nas.h, 1453
LITE_SYS_SRV_DOMAIN_NO_SRV
nas.h, 1453
LITE_SYS_SRV_DOMAIN_PS_ONLY
nas.h, 1453
LITE_VOICE_SUPS_SRV_CLASS_DATA
voice.h, 1728
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC
voice.h, 1728
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC
voice.h, 1728
LITE_VOICE_SUPS_SRV_CLASS_FAX
voice.h, 1728
LITE_VOICE_SUPS_SRV_CLASS_NONE
voice.h, 1728
LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS
voice.h, 1728
LITE_VOICE_SUPS_SRV_CLASS_PADACCESS
voice.h, 1729
LITE_VOICE_SUPS_SRV_CLASS_SMS
voice.h, 1728
LITE_VOICE_SUPS_SRV_CLASS_VOICE
voice.h, 1728
LITEQMI_QMI_CBK_PARAM_NOCHANGE
sms.h, 1613
LITEQMI_QMI_CBK_PARAM_RESET
sms.h, 1613
LITEQMI_QMI_CBK_PARAM_SET
sms.h, 1613
LBPTlv
NASQmiCbkNasSystemSelPrefInd, 402
LITE_MAX_ARFCN_NUM
nas.h, 1452
LITE_MAX_PCOID_LIST
wds.h, 1766
LMCERTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
LOCEVENTMASKGNSSSVINFO
loc.h, 1407
LOCEVENTMASKNMEA
loc.h, 1408
LOCEVENTMASKWIFIREQ
loc.h, 1408
LPCSTR
SwiDataTypes.h, 1656
LTEAttachProfile
unpack_wds_SLQSGet3GPPConfigItem_t, 1167
LTEAttachProfileList
unpack_wds_SLQSGet3GPPConfigItem_t, 1167
LTEAttachProfileListLen
pack_wds_SLQSSet3GPPConfigItem_t, 629
unpack_wds_SLQSGet3GPPConfigItem_t, 1167
LTEBandPref
nas_LTEBandPrefTlv, 251
NASLTEBandPreferenceTlv, 390
LTECphyCaInfo
unpack_nas_SlqsGetLTECphyCAInfo_t, 898
LTERSRPThreshListLen
nas_LTERSRPThresh, 271

LTERSRQThreshListLen
nas_LTERSRQThresh, 271
LTERSSIThreshListLen
nas_LTERSSIThresh, 272
LTESNRThresListLen
nas_LTESNRThresh, 275
LTESNRThreshListLen
nas_LTESNRThreshold, 275
LTESSInfo
unpack_nas_SLQSNasGetSigInfo_t, 946
LTEWCDMACellInfo
nas_LTEInfoNeighboringWCDMA, 262
Lac
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
lac
nas_GERANInfo, 232
nas_GSMSysInfo, 238
nas_LTESysInfo, 278
nas_NR5GSysInfoTlv, 302
nas_NR5GSystemInfoTlv, 306
nas_UMTSExtInfo, 371
nas_UMTSInfo, 372
nas_WCDMASysInfo, 385
lac1
nas_OperatorPLMNData, 309
lac2
nas_OperatorPLMNData, 309
lacValid
nas_GSMSysInfo, 239
nas_LTESysInfo, 278
nas_NR5GSysInfoTlv, 302
nas_NR5GSystemInfoTlv, 306
nas_WCDMASysInfo, 385
language
sms_CDMABroadcastConfig, 659
lastCallDataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 1172
lastCallDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 1170
lastCallRXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 1172
lastCallTXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 1172
LastErrCode
nas_DataStatusDetail, 218
Latency
unpack_qos_swiQosFlow_t, 1007
Latitude
unpack_pds_SLQSGetGPSStateInfo_t, 987
latitude
pack_loc_SLQSLOCInjectPosition_t, 455
leapSeconds
nas_qaQmi3Gpp2TimeZone, 323
leaseState
wds_DHCPLeaseStateTlv, 1258
len
loc_BdsSVInfo, 165
loc_SVInfo, 182
unpack_nas_GetSignalStrengths_t, 888
Length
QMUX_HDR, 648
length
GetClientIdReqMsg, 94
GetClientIdRspMsg, 95
QmiHdr, 642
ReleaseClientIdReqMsg, 651
ReleaseClientIdRspMsg, 651
sMSCAddressInfo, 671
sMSEtwsMessageInfo, 672
sMSTransferRouteMTMessageInfo, 676
uim_readRecordInfo, 713
uim_readTransparentInfo, 715
Level
pack_swiaudio_SLQSSetM2MAudioVolume_t, 545
unpack_swiaudio_SLQSGetM2MAudioVolume_t, 1034
LibPackGPRSRequestedQoS, 118
delayClass, 119
meanThroughputClass, 119
peakThroughputClass, 119
precedenceClass, 119
reliabilityClass, 119
LibPackPCOIDList, 119
PcoList, 119
LibPackPDNThrottleTimer, 119
ThrottleTimer, 119
LibPackProfileMnc, 154
MNC, 155
PCSFlag, 155
LibPackQosClassID, 155
gDlBitRate, 156
gUlBitRate, 156
maxDlBitRate, 156
maxUlBitRate, 156
QCI, 156
LibPackTFTIDParams, 156
destPortRangeEnd, 157
destPortRangeStart, 157
eValid, 157
filterId, 157
flowLabel, 157
IPSECSPI, 157
ipVersion, 158
nextHeader, 158
pSourceIP, 158
sourceIPMask, 158
srcPortRangeEnd, 158
srcPortRangeStart, 158
tosMask, 158
LibPackUMTSQoS, 158
deliveryErrSDU, 160
grntDownlinkBitrate, 160
grntUplinkBitrate, 160
maxDownlinkBitrate, 160

maxSDUSize, 160
maxUplinkBitrate, 160
qosDeliveryOrder, 160
resBerRatio, 160
sduErrorRatio, 160
trafficClass, 160
trafficPriority, 160
transferDelay, 160
LibPackUMTSReqQoSSigInd, 161
SigInd, 161
UMTSReqQoS, 161
LibPackprofile_3GPP, 143
pAPNClass, 148
pAPNDisabledFlag, 148
pAPNName, 148
pAPNnameSize, 148
pAddrAllocPref, 148
pAuthenticationPref, 148
pGPRSMinimumQoS, 148
pGPRSRequestedQos, 148
pIPv4AddrPref, 148
pIPv6AddPref, 148
pImCnFlag, 148
pPDNInactivTimeout, 148
pPDPtype, 149
pPassword, 148
pPasswordSize, 148
pPcscfAddrUsingDhcp, 148
pPcscfAddrUsingPCO, 148
pPdpAccessConFlag, 148
pPdpContext, 148
pPdpDataCompType, 148
pPdpHdrCompType, 148
pPriDNSIPv4AddPref, 149
pPriDNSIPv6addpref, 149
pPrimaryID, 149
pProfilename, 149
pProfilenameSize, 149
pQosClassID, 149
pSecDNSIPv4AddPref, 149
pSecDNSIPv6addpref, 149
pSecondaryFlag, 149
pSupportEmergencyCalls, 149
pTFTID1Params, 149
pTFTID2Params, 149
pUMTSMinQoS, 149
pUMTSMinQosSigInd, 149
pUMTSReqQoS, 149
pUMTSReqQoSSigInd, 149
pUsername, 149
pUsernameSize, 149
LibPackprofile_3GPP2, 149
pAPNClass3GPP2, 153
pAPNEnabled3GPP2, 153
pAllowLinger, 153
pApnString, 153
pApnStringSize, 153
pAppPriority, 154
pAppType, 154
pAuthPassword, 154
pAuthPassword_tSize, 154
pAuthProtocol, 154
pAuthRetryCount, 154
pAuthTimeout, 154
pDataMode, 154
pDataRate, 154
pIpcpAckTimeout, 154
pIpcpCreqRetryCount, 154
pIsPcscfAddressNedded, 154
pLcpAckTimeout, 154
pLcpCreqRetryCount, 154
pNegoDnsSrvrPref, 154
pPDNInactivTimeout3GPP2, 154
pPdnType, 154
pPppSessCloseTimer1x, 154
pPppSessCloseTimerDO, 154
pPriV6DnsAddress, 154
pPrimaryV4DnsAddress, 154
pRATType, 154
pSecV6DnsAddress, 154
pSecondaryV4DnsAddress, 154
pUserId, 154
pUserIdSize, 154
libSDP_BuildImagesPreferenceRequest
lite-fw.h, 1391
libSDP_CalculateImageMask
lite-fw.h, 1391
libSDP_CheckValidFirmwareInfo
lite-fw.h, 1391
libSDP_DownloadFW
lite-fw.h, 1391
libSDP_ExtractFirmwareParametersByPath
lite-fw.h, 1391
libSDP_FirmwareInfo
lite-fw.h, 1392
libSDP_GetModelFamily
lite-fw.h, 1391
libSDP_GetVersion
lite-fw.h, 1391
libSDP_getFileType
lite-fw.h, 1391
LibpackProfile3GPP, 119
pAPNClass, 125
pAPNDisabledFlag, 125
pAPNName, 125
pAPNnameSize, 125
pAddrAllocPref, 125
pAuthenticationPref, 125
pGPRSMinimumQoS, 125
pGPRSRequestedQos, 125
pIPv4AddrPref, 126
pIPv6AddPref, 126
pImCnFlag, 126
pPDNInactivTimeout, 126
pPDPtype, 126
pPassword, 126

pPasswordSize, 126
pPcscfAddrUsingDhcp, 126
pPcscfAddrUsingPCO, 126
pPdpAccessConFlag, 126
pPdpContext, 126
pPdpDataCompType, 126
pPdpHdrCompType, 126
pPriDNSIPv4AddPref, 126
pPriDNSIPv6addpref, 126
pPrimaryID, 126
pProfilename, 126
pProfilenameSize, 126
pQosClassID, 126
pSecDNSIPv4AddPref, 126
pSecDNSIPv6addpref, 126
pSecondaryFlag, 126
pSupportEmergencyCalls, 126
pTFTID1Params, 126
pTFTID2Params, 126
pUMTSMinQoS, 126
pUMTSMinQosSigInd, 126
pUMTSReqQoS, 126
pUMTSReqQoSSigInd, 127
pUsername, 127
pUsernameSize, 127
LibpackProfile3GPP2, 127
pAPNClass3GPP2, 131
pAPNEnabled3GPP2, 132
pAllowLinger, 131
pApnString, 132
pApnStringSize, 132
pAppPriority, 132
pAppType, 132
pAuthPassword, 132
pAuthPasswordSize, 132
pAuthProtocol, 132
pAuthRetryCount, 132
pAuthTimeout, 132
pDataMode, 132
pDataRate, 132
pIpcpAckTimeout, 132
pIpcpCreqRetryCount, 132
pIsPcscfAddressNedded, 132
pLcpAckTimeout, 132
pLcpCreqRetryCount, 132
pNegoDnsSrvrPref, 132
pPDNInactivTimeout3GPP2, 132
pPdnType, 132
pPppSessCloseTimer1x, 132
pPppSessCloseTimerDO, 132
pPriV6DnsAddress, 132
pPrimaryV4DnsAddress, 132
pRATType, 132
pSecV6DnsAddress, 132
pSecondaryV4DnsAddress, 132
pUserId, 132
pUserIdSize, 133
LibpackProfile3GPPV2, 133
pAPNBearer, 141
pAPNClass, 141
pAPNDisabledFlag, 141
pAPNName, 141
pAPNnameSize, 141
pAddrAllocPref, 141
pAppUserData, 141
pAuthenticationPref, 141
pClatFlag, 141
pDnsWithDHCPFlag, 141
pGPRSMinimumQoS, 141
pGPRSRequestedQos, 141
pIPV6DelegFlag, 142
pIPv4AddrPref, 142
pIPv6AddPref, 142
pIWLANtoLTEHandoverFlag, 142
pImCnFlag, 142
pLTEtoIWLANHandoverFlag, 142
pLteRoamPDPType, 142
pMaxPDN, 142
pMaxPDNTimer, 142
pMcc, 142
pMnc, 142
pMsisdnFlag, 142
pOperatorPCOID, 142
pOverridePDPType, 142
pPCOIDList, 142
pPDNDisconnectWaitTimer, 142
pPDNInactivTimeout, 142
pPDNThrottleTimer, 142
pPDNWaitTimer, 142
pPDPtype, 142
pPassword, 142
pPasswordSize, 142
pPcscfAddrUsingDhcp, 142
pPcscfAddrUsingPCO, 142
pPdpAccessConFlag, 142
pPdpContext, 142
pPdpDataCompType, 142
pPdpHdrCompType, 142
pPersistFlag, 143
pPriDNSIPv4AddPref, 143
pPriDNSIPv6addpref, 143
pPrimaryID, 143
pProfilename, 143
pProfilenameSize, 143
pQosClassID, 143
pRoamDisallowFlag, 143
pSecDNSIPv4AddPref, 143
pSecDNSIPv6addpref, 143
pSecondaryFlag, 143
pSupportEmergencyCalls, 143
pTFTID1Params, 143
pTFTID2Params, 143
pUMTSMinQoS, 143
pUMTSMinQosSigInd, 143
pUMTSReqQoS, 143
pUMTSReqQoSSigInd, 143

pUmtsRoamPDPType, 143
pUsername, 143
pUsernameSize, 143
libsdp_SetReadBlockSize
lite-fw.h, 1391
libsdp_set_log_func
lite-fw.h, 1391
libsdplogger
lite-fw.h, 1392
lineValue
pack_voice_SLQSVoiceALSSelectLine_t, 588
linkage
altSrcInfo_t, 43
list_type
DMScustSettingList, 79
DMSgetCustomInput, 81
pack_dms_GetCustFeaturesV2_t, 416
listEntries
FMSPrefImageList, 94
listSize
FMSImageList, 93
FMSPrefImageList, 94
lite-fw.h
eFILE_TYPE_CAR_PRI, 1394
eFILE_TYPE_COMPO_PRI, 1394
eFILE_TYPE_NONE, 1394
eFILE_TYPE_OEM_PRI, 1394
eFIREHOSE_ERR_RC7611_RETRY_TEST, 1395
eFIREHOSE_ERR_SECBOOT_INVALID_CERT_-CHAIN, 1395
eFW_TYPE_CWE, 1394
eFW_TYPE_CWE_NVU, 1394
eFW_TYPE_INVALID, 1394
eFW_TYPE_MBN, 1394
eFW_TYPE_MBN_GOBI, 1394
eFW_TYPE_NVU, 1394
eFW_TYPE_SPK, 1394
eIMAGE_TYPE_ANY, 1395
eIMAGE_TYPE_APPL, 1395
eIMAGE_TYPE_APPS, 1395
eIMAGE_TYPE_BOOT, 1395
eIMAGE_TYPE_FILE, 1395
eIMAGE_TYPE_INVALID, 1395
eIMAGE_TYPE_KEYS, 1395
eIMAGE_TYPE_MAX, 1395
eIMAGE_TYPE_MIN, 1395
eIMAGE_TYPE_MODM, 1395
eIMAGE_TYPE_SPKG, 1395
eIMAGE_TYPE_USER, 1395
eLITE_FWDWL_APP_ERR_RC7611_ERROR_A-FTER_RETRY, 1395
eModel_9X15, 1395
eModel_9X30, 1395
eModel_9x05, 1395
eModel_9x06, 1395
eModel_9x07, 1395
eModel_Unknown, 1395
eModel_WP9X15, 1395
eQDL_HW_FLOW_DISABLE, 1395
eQDL_HW_FLOW_ENABLE, 1395
eQDL_HW_FLOW_INIT, 1395
eQDL_HW_FLOW_Unknown, 1395
eQDL_MODE_INIT, 1396
eQDL_MODE_TTYUSB, 1396
eQDL_MODE_UART0, 1396
eQDL_MODE_Unknown, 1396
eSDP_FWDWL_ERR_END, 1395
eSDP_FWDWL_ERR_FAIL, 1394
eSDP_FWDWL_ERR_FW_UPGRADE, 1394
eSDP_FWDWL_ERR_FW_VERSION_FAIL, 1395
eSDP_FWDWL_ERR_GENERAL, 1394
eSDP_FWDWL_ERR_INVALID_DEV, 1394
eSDP_FWDWL_ERR_INVALID_PATH, 1394
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED, 1394
eSDP_FWDWL_ERR_PATH_TOO_LONG, 1394
eSDP_FWDWL_ERR_PRI_FAIL, 1394
eSDP_FWDWL_ERR_SDK, 1394
eSDP_FWDWL_ERR_SDP_TIMEOUT, 1395
eSDP_FWDWL_ERR_SET_CBK, 1394
eSDP_FWDWL_ERR_TIMEOUT, 1394
eSDP_FWDWL_SUCCESS, 1394
lite-fw.h, 1388
FIRMWARE_BCHDATESIZE, 1391
GetFlashSectorSize, 1396
IMG_MASK_CLEAR, 1391
IMG_MASK_GENERIC, 1391
IMG_MASK_MDM, 1391
IMG_MASK_PRI, 1391
libSDP_BuildImagesPreferenceRequest, 1391
libSDP_CalculateImageMask, 1391
libSDP_CheckValidFirmwareInfo, 1391
libSDP_DownloadFW, 1391
libSDP_ExtractFirmwareParametersByPath, 1391
libSDP_FirmwareInfo, 1392
libSDP_GetModelFamily, 1391
libSDP_GetVersion, 1391
libSDP_getFileType, 1391
libsdp_SetReadBlockSize, 1391
libsdp_set_log_func, 1391
libsdplogger, 1392
litefw_BuildImagesPreferenceRequest, 1396
litefw_CalculateImageMask, 1396
litefw_CheckValidFirmwareInfo, 1397
litefw_DownloadFW, 1397
litefw_ExtractFirmwareParametersByPath, 1397
litefw_ExtractFirmwarePartNoByPath, 1398
litefw_FirmwareFileInfo, 1392
litefw_FirmwareInfo, 1393
litefw_FirmwarePartNo, 1393
litefw_Fw_Type, 1394
litefw_GetModelFamily, 1399
litefw_GetQTLDownloadMode, 1399
litefw_GetQTLHWFlowControl, 1399
litefw_GetRetryFirehoseSwitch, 1399
litefw_GetVersion, 1399

litefw_Models, 1395
litefw_QDL_FLOW_CONTROLs, 1395
litefw_QDL_MODEs, 1395
litefw_SLQSGetFirmwareFileInfo, 1401
litefw_SetQTLDownloadMode, 1400
litefw_SetQTLHWFlowControl, 1400
litefw_SetReadBlockSize, 1400
litefw_SetRetryFirehoseSwitch, 1401
litefw_fileimgtype, 1394
litefw_fwdwl_error_codes, 1394
litefw_getFileType, 1398
litefw_imagetype, 1395
litefw_logsenable, 1399
litefw_set_log_func, 1400
litefw_switch_9x07_to_downloadmode, 1401
litefw_switch_to_BootHoldMode, 1401
litefwlogger, 1393
SetFlashSectorSize, 1402
liteQmiDataBearerMasks
wds.h, 1768
liteServiceClassInformation
voice.h, 1728
litefw_BuildImagesPreferenceRequest
lite-fw.h, 1396
litefw_CalculateImageMask
lite-fw.h, 1396
litefw_CheckValidFirmwareInfo
lite-fw.h, 1397
litefw_DownloadFW
lite-fw.h, 1397
litefw_ExtractFirmwareParametersByPath
lite-fw.h, 1397
litefw_ExtractFirmwarePartNoByPath
lite-fw.h, 1398
litefw_FirmwareFileInfo
lite-fw.h, 1392
litefw_FirmwareInfo
lite-fw.h, 1393
litefw_FirmwarePartNo
lite-fw.h, 1393
litefw_Fw_Type
lite-fw.h, 1394
litefw_GetModelFamily
lite-fw.h, 1399
litefw_GetQTLDownloadMode
lite-fw.h, 1399
litefw_GetQTLHWFlowControl
lite-fw.h, 1399
litefw_GetRetryFirehoseSwitch
lite-fw.h, 1399
litefw_GetVersion
lite-fw.h, 1399
litefw_Models
lite-fw.h, 1395
litefw_QDL_FLOW_CONTROLs
lite-fw.h, 1395
litefw_QDL_MODEs
lite-fw.h, 1395
litefw_SLQSGetFirmwareFileInfo
lite-fw.h, 1401
litefw_SetQTLDownloadMode
lite-fw.h, 1400
litefw_SetQTLHWFlowControl
lite-fw.h, 1400
litefw_SetReadBlockSize
lite-fw.h, 1400
litefw_SetRetryFirehoseSwitch
lite-fw.h, 1401
litefw_fileimgtype
lite-fw.h, 1394
litefw_fwdwl_error_codes
lite-fw.h, 1394
litefw_getFileType
lite-fw.h, 1398
litefw_imagetype
lite-fw.h, 1395
litefw_logsenable
lite-fw.h, 1399
litefw_set_log_func
lite-fw.h, 1400
litefw_switch_9x07_to_downloadmode
lite-fw.h, 1401
litefw_switch_to_BootHoldMode
lite-fw.h, 1401
litefwlogger
lite-fw.h, 1393
liteqmi_GetVersion
common.h, 1293
liteqmi_helper_decode7bitAsciiEncString
common.h, 1293
liteqmi_log
common.h, 1294
loc.h
eQMI_LOC_SESS_STATUS_FAILURE, 1409
eQMI_LOC_SESS_STATUS_IN_PROGRESS, 1409
eQMI_LOC_SESS_STATUS_SUCCESS, 1409
eQMI_LOC_SESS_STATUS_TIMEOUT, 1409
loc.h, 1402
LOCEVENTMASKNMEA, 1408
LOCEVENTMASKWIFIREQ, 1408
MAX_TEMP_DATA_LEN, 1409
pack_loc_DeleteAssistData, 1409
pack_loc_EventRegister, 1409
pack_loc_GetFixCriteria, 1410
pack_loc_SLQSLOCGetBestAvailPos, 1411
pack_loc_SLQSLOCGetOpMode, 1411
pack_loc_SLQSLOCGetServer, 1412
pack_loc_SLQSLOCInjectPosition, 1412
pack_loc_SLQSLOCInjectSensorData, 1412
pack_loc_SLQSLOCInjectUTCTime, 1413
pack_loc_SLQSLOCSetCradleMountConfig, 1413
pack_loc_SLQSLOCSetServer, 1414
pack_loc_SetExtPowerState, 1410
pack_loc_SetOperationMode, 1411
pack_loc_Start, 1414

pack_loc_Stop, 1414
unpack_loc_BestAvailPos_Ind, 1415
unpack_loc_CradleMountCallback_Ind, 1415
unpack_loc_DeleteAssistData, 1416
unpack_loc_DeleteAssistData_Ind, 1416
unpack_loc_DeleteAssistData_t, 1409
unpack_loc_EngineState_Ind, 1416
unpack_loc_EventNMEA_Ind, 1417
unpack_loc_EventRegister, 1417
unpack_loc_EventTimeSyncCallback_Ind, 1417
unpack_loc_FixCriteria_Ind, 1418
unpack_loc_GetFixCriteria, 1418
unpack_loc_GetFixCriteria_t, 1409
unpack_loc_GetOpMode_Ind, 1418
unpack_loc_GetServer_Ind, 1419
unpack_loc_GnssSvInfo_Ind, 1419
unpack_loc_InjectPositionCallback_Ind, 1419
unpack_loc_InjectSensorDataCallback_Ind, 1420
unpack_loc_InjectTimeSyncDataCallback_Ind, 1420
unpack_loc_InjectUTCTimeCallback_Ind, 1420
unpack_loc_PositionRpt_Ind, 1421
unpack_loc_SLQSLOCGetBestAvailPos, 1423
unpack_loc_SLQSLOCGetOpMode, 1423
unpack_loc_SLQSLOCGetServer, 1424
unpack_loc_SLQSLOCGetServer_t, 1409
unpack_loc_SLQSLOCInjectPosition, 1424
unpack_loc_SLQSLOCInjectPosition_t, 1409
unpack_loc_SLQSLOCInjectSensorData, 1424
unpack_loc_SLQSLOCInjectSensorData_t, 1409
unpack_loc_SLQSLOCInjectUTCTime, 1425
unpack_loc_SLQSLOCInjectUTCTime_t, 1409
unpack_loc_SLQSLOCSetCradleMountConfig, 1425
unpack_loc_SLQSLOCSetCradleMountConfig_t, 1409
unpack_loc_SLQSLOCSetServer, 1425
unpack_loc_SensorStreamingCallback_Ind, 1421
unpack_loc_SetExtPowerConfig_Ind, 1421
unpack_loc_SetExtPowerState, 1422
unpack_loc_SetOperationMode, 1422
unpack_loc_SetOperationMode_Ind, 1422
unpack_loc_SetServer_Ind, 1423
unpack_loc_Start, 1426
unpack_loc_Stop, 1426
loc_AppProviderInfoTlv, 163
name, 164
nameLen, 164
provider, 164
providerLen, 164
TlvPresent, 164
verValid, 164
version, 164
versionLen, 164
loc_BdsSV, 164
id, 164
mask, 164
loc_BdsSVInfo, 165
len, 165
pSV, 165
loc_CellDb, 165
mask, 166
loc_ClkInfo, 166
mask, 167
loc_FixCriteriaStatusTlv, 167
status, 168
TlvPresent, 168
loc_GnssData, 168
mask, 169
loc_HorAccuracyLvlTlv, 172
accuracy, 172
TlvPresent, 172
loc_IPv4Config, 173
IPv4Addr, 173
IPv4Port, 173
loc_IPv4Info, 173
address, 174
port, 174
TlvPresent, 174
loc_IPv6Config, 174
IPv6Addr, 174
IPv6Port, 174
loc_IPv6Info, 174
address, 175
port, 175
TlvPresent, 175
loc_IntermediateRptStateTlv, 172
intermediate, 173
TlvPresent, 173
loc_LocApplicationInfo, 175
appNameLength, 176
appProviderLength, 176
appVersionLength, 176
appVersionValid, 176
pAppName, 176
pAppProvider, 176
pAppVersion, 176
loc_MinIntervalTlv, 176
interval, 177
TlvPresent, 177
loc_SV, 180
id, 181
mask, 181
system, 181
loc_SVInfo, 181
len, 182
pSV, 182
loc_URLAddrInfo, 183
urlAddr, 183
loc_accelAcceptReady, 161
batchPerSec, 162
injectEnable, 162
samplesPerBatch, 162
loc_accelTempAcceptReady, 162
batchPerSec, 163
injectEnable, 163

samplesPerBatch, 163
loc_appProviderInfo
unpack_loc_FixCriteria_Ind_t, 852
loc_fixCriteriaStatus
unpack_loc_FixCriteria_Ind_t, 853
loc_gpsTime, 169
gpsTimeOfWeekMs, 170
gpsWeek, 170
loc_gyroAcceptReady, 170
batchPerSec, 171
injectEnable, 171
samplesPerBatch, 171
loc_gyroTempAcceptReady, 171
batchPerSec, 171
injectEnable, 171
samplesPerBatch, 171
loc_horAccuracy
unpack_loc_FixCriteria_Ind_t, 853
loc_intermediateRptState
unpack_loc_FixCriteria_Ind_t, 853
loc_minInterval
unpack_loc_FixCriteria_Ind_t, 853
loc_precisionDilution, 177
HDOP, 177
PDOP, 177
VDOP, 177
loc_satelliteInfo, 177
azimuth, 179
elevation, 179
gnssSvId, 179
healthStatus, 179
snr, 179
svInfoMask, 179
svListLen, 180
svStatus, 180
system, 180
validMask, 180
loc_sensorDataUsage, 180
aidingIndicatorMask, 180
usageMask, 180
loc_svUsedforFix, 182
gnssSvUsedList, 182
gnssSvUsedList_len, 182
loc_urlAddr, 182
address, 183
TlvPresent, 183
localTimeOffset
nas_qaQmi3Gpp2TimeZone, 323
logString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 798
logger
switype_256bit.h, 1685
longName
nas_PLMNNetworkNameData, 319
unpack_nas_SLQSGetPLMNName_t, 903
longNameCI
unpack_nas_SLQSGetPLMNName_t, 903
longNameEn
unpack_nas_SLQSGetPLMNName_t, 903
longNameLen
nas_PLMNNetworkNameData, 319
unpack_nas_SLQSGetPLMNName_t, 903
longNameSB
unpack_nas_SLQSGetPLMNName_t, 903
longNameSpareBits
nas_PLMNNetworkNameData, 319
Longitude
unpack_pds_SLQSGetGPSStateInfo_t, 987
longitude
pack_loc_SLQSLOCInjectPosition_t, 455
loopbackMode
pack_wds_SLQSSSetLoopback_t, 632
loopbackMultiplier
pack_wds_SLQSSSetLoopback_t, 632
LpmFlag
unpack_dms_SLQSDmsSwiGetPCInfo_t, 778
lteArfcn
nas_LTEARFCNTlv, 249
LteBandCapability
unpack_dms_SLQSGetBandCapability_t, 786
unpack_dms_SLQSGetBandCapabilityExt_t, 790
lteBands
dms_LteBandsSupport, 69
LteBandsSupport
unpack_dms_SLQSGetBandCapabilityExt_t, 790
lteEarfcn
nas_LteEarfcnInfo, 253
LteEmbmsCoverage
nas_LteEmbmsCoverageTlv, 254
LteEmbmsTraceId
nas_LteEmbmsTraceIdTlv, 254
lteEmmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 491
lteEmmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 491
lteEsmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 491
lteEsmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 491
LteGsmCellInfo
nas_LTEInfoNeighboringGSM, 261
lteInterEarfcnlen
nas_LteEarfcnInfo, 253
LteM1BandPref
NASLteM1BandPrefTlv, 390
lteM1BandPref
nas_LteM1BandPrefTlv, 263
LteNB1BandPref
NASLteNB1BandPrefTlv, 392
lteNb1BandPref
nas_LteNb1BandPrefTlv, 265
lteOpMode
nas_EdrxCiotLteMode, 225
nas_LteOpMode, 267
nas_lteOpModeTlv, 268
lteOpModeLen

nas_lteOpModeTlv, 268
lteOperationalMode
unpack_nas_SLQSNasRFBandInfoCallback_Ind_-t, 952
LteQci
unpack_qos_swiQosFlow_t, 1008
LteRegDomain
nas_LteRegDomainTlv, 270
lteRsrpDelta
nas_SLQSSignalStrengthsIndReq, 356
lteRsrpinfo
nas_SLQSSignalStrengthsInformation, 357
lteSSInfo, 183
rsrp, 184
rsrq, 184
rssi, 184
snr, 184
lteSnrDelta
nas_SLQSSignalStrengthsIndReq, 356
lteSnrinfo
nas_SLQSSignalStrengthsInformation, 357
LteVoiceDomain
nas_LteVoiceDomainTlv, 278
ltersrp
unpack_nas_SLQSGetSignalStrength_t, 914
ltesnr
unpack_nas_SLQSGetSignalStrength_t, 914
lvl
unpack_tmd_SLQSTmdMitigationLvlRptCallback_-ind_t, 1066

M1BandPTlv
NASQmiCbkNasSystemSelPrefInd, 402
m_FwBuildId
CarrierImage_t, 52
m_FwImageId
CarrierImage_t, 52
m_PriBuildId
CarrierImage_t, 52
m_PriImageId
CarrierImage_t, 53
m_nCarrierId
CarrierImage_t, 52
m_nFolderId
CarrierImage_t, 52
m_nStorage
CarrierImage_t, 52
MACIndex
nas_NetworkStatEVDO, 294
MAX
QmuxInternal.h, 1592
MAX_APN_STRING
dsd.h, 1363
MAX_ATR_LENGTH
uim.h, 1695
MAX_BUILD_ID_LEN
dms.h, 1308
MAX_ERROR_CODE_LEN
imsa.h, 1384
MAX_ICCID_LENGTH
uim.h, 1695
MAX_MSE_TWS_MSG
sms.h, 1613
MAX_NAME_LEN
ims.h, 1372
MAX_NO_OF_SLOTS
uim.h, 1695
MAX_PHY_SLOTS_INFO
uim.h, 1695
MAX_QMI_CLIENT
QmuxTransport.h, 1594
MAX_QMI_CLIENT_ID
QmuxTransport.h, 1594
MAX_SENSOR_DATA_LEN
loc.h, 1409
MAX_SLOTS_STATUS
uim.h, 1695
MAX_SMS_LIST_SIZE
sms.h, 1613
MAX_TEMP_DATA_LEN
loc.h, 1409
MAXVOICEUSSDLENGTH
voice.h, 1728
MBIM layer (lite-mbim), 36
MBIM_CLOSE
MbimInternal.h, 1429
MBIM_CLOSE_DONE
MbimInternal.h, 1429
MBIM_CLOSE_DONE_MSG, 184
Status, 184
MBIM_CLOSE_MSG, 185
MbimInternal.h, 1429
MessageHeader, 185
MBIM_COMMAND
MbimInternal.h, 1429
MBIM_COMMAND_DONE
MbimInternal.h, 1429
MBIM_COMMAND_MSG, 185
CID, 186
CommandType, 186
DeviceServiceId, 186
FragmentHeader, 186
InformationBufferLength, 186
MbimInternal.h, 1429
MessageHeader, 186
MBIM_DONE_CALLBACK
MbimTransaction.h, 1433
MBIM_FRAGMENT_HEADER, 186
MBIM_FRAGMENT_MSG, 186
FragmentHeader, 186
MbimInternal.h, 1429
MessageHeader, 186
MBIM_FUNCTION_ERROR
MbimInternal.h, 1429
MBIM_HOST_ERROR
MbimInternal.h, 1429
MBIM_MAX_CIDS

MbimTransport.h, 1435
MBIM_MESSAGE_HEADER, 187
MbimInternal.h, 1429
MessageLength, 187
MessageType, 187
TransactionId, 187
MBIM_OPEN
MbimInternal.h, 1429
MBIM_OPEN_DONE
MbimInternal.h, 1429
MBIM_OPEN_DONE_MSG, 187
MbimInternal.h, 1429
MessageHeader, 187
Status, 187
MBIM_OPEN_MSG, 187
MaxControlTransfer, 187
MbimInternal.h, 1429
MessageHeader, 187
MBIM_STATUS_FAILURE
MbimTransport.h, 1435
MBIM_STATUS_SUCCESS
MbimTransport.h, 1435
MBIM_UUID_SIZE
MbimTransport.h, 1435
MCC
nas_CDMASysInfo, 206
nas_CDMASysInfoExt, 207
nas_currentPLMN, 216
nas_ForbiddenNetworks3GPP, 230
nas_GSMSysInfo, 239
nas_lteOpModeTlv, 268
nas_LTESysInfo, 278
nas_MNCPCSDigitStatus, 282
nas_NR5GSystemInfoTlv, 306
nas_QmiNas3GppNetworkInfo, 324
nas_QmiNas3GppNetworkRAT, 325
nas_QmisNasPcsDigit, 326
nas_WCDMASysInfo, 385
unpack_nas_GetServingNetwork_t, 885
MDMCallDuration
connectionStatus, 57
MDMConnStatus
connectionStatus, 57
wds_ConnStatusTlv, 1254
MEID_MAX_SIZE
dms.h, 1308
MEIDString
unpack_dms_GetDeviceSerialNumbers_t, 752
MIN
unpack_dms_GetVoiceNumber_t, 768
MINREQBKLEN
common.h, 1290
MMTlv
unpack_sms_SetNewSMSCallback_ind_t, 1017
MNC
LibPackProfileMnc, 155
nas_CDMASysInfo, 206
nas_currentPLMN, 216
nas_ForbiddenNetworks3GPP, 230
nas_GSMSysInfo, 239
nas_lteOpModeTlv, 268
nas_LTESysInfo, 278
nas_MNCPCSDigitStatus, 282
nas_NR5GSystemInfoTlv, 306
nas_QmiNas3GppNetworkInfo, 324
nas_QmiNas3GppNetworkRAT, 325
nas_QmisNasPcsDigit, 326
nas_WCDMASysInfo, 385
unpack_nas_GetServingNetwork_t, 886
MNCIncPCSDigit
nas_MNCPCSDigitStatus, 282
MNCPCSDigitStatTlv
unpack_nas_SLQSGetServingSystemV2_t, 912
MPTlv
NASQmiCbkNasSystemSelPrefInd, 402
MSETlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
MSGID_AND_LEN
common.h, 1290
MSGID_DONT_CARE
common.h, 1290
MTMessageInfo
newMTMessageTlv, 406
MTUSize
pack_swidms_SLQSSwiDmsSetMTU_t, 553
MTUSize3gpp
swidms_mtuSize3gppTlv, 680
manString
pack_dms_SLQSSwiSetHostDevInfo_t, 427
unpack_dms_SLQSSwiGetHostDevInfo_t, 799
manufacturer
unpack_dms_GetManufacturer_t, 758
Mask
pack_wds_SLQSGetDUNCallInfo_t, 624
mask
loc_BdsSV, 164
loc_CellDb, 166
loc_ClkInfo, 167
loc_GnssData, 169
loc_SV, 181
unpack_qos_IPv6TrafCls_t, 990
unpack_qos_Tos_t, 1009
max
PackSwiAVMSSettingsPeriodsInfo, 641
max_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 1169
max_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 1169
max_dist
pack_swiloc_SwiLocSetAutoStart_t, 557
unpack_swiloc_SwiLocGetAutoStart_t, 1054
max_dist_reported
unpack_swiloc_SwiLocGetAutoStart_t, 1054
max_time

pack_swiloc_SwiLocSetAutoStart_t, 557
unpack_swiloc_SwiLocGetAutoStart_t, 1054
max_time_reported
unpack_swiloc_SwiLocGetAutoStart_t, 1054
MaxActive
dms_devMaxCfgListCaps, 64
dms_devMultiSimVoiceDataCaps, 65
dms_devSubsCfgList, 66
MaxAllowedPktSz
unpack_qos_swiQosFlow_t, 1008
MaxChanRxRate
dunchannelRate, 89
MaxChanTxRate
dunchannelRate, 89
maxChannelRXRate
unpack_wds_GetConnectionRate_t, 1144
maxChannelTXRate
unpack_wds_GetConnectionRate_t, 1144
MaxControlTransfer
MBIM_OPEN_MSG, 187
maxDlBitRate
LibPackQosClassID, 156
maxDownlinkBitrate
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
maxImages
FMSImageIDEntries, 93
maxMitigationLevel
tmd_mitigationDevList, 686
MaxRXChannelRate
dms_devCaps, 62
unpack_dms_GetDeviceCap_t, 744
maxRxChannelRate
unpack_dms_GetDeviceCapabilities_t, 746
maxSDUSize
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
maxStorageSize
sms_maxStorageSizeResp, 663
MaxSubs
dms_devMaxCfgListCaps, 64
dms_devMultiSimCaps, 65
dms_devMultiSimVoiceDataCaps, 65
MaxSubsCapLen
dms_devMaxSubsCaps, 64
MaxSubsList
dms_devMaxSubsCaps, 64
MaxTXChannelRate
dms_devCaps, 62
unpack_dms_GetDeviceCap_t, 744
maxTxChannelRate
unpack_dms_GetDeviceCapabilities_t, 746
maxUlBitRate
LibPackQosClassID, 156
maxUplinkBitrate
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
MbimEventEntry, 188
BasicConnectDeviceService.h, 1283
CidCount, 188
Cids, 188
DeviceServiceId, 188
MbimIndicator, 188
MbimIndicator.h, 1427
pIndicationCallback, 189
pIndicationCallbackContext, 189
pNext, 189
pPrev, 189
MbimIndicator.h, 1426
MbimIndicator, 1427
MbimIndicator_Initialize, 1427
MbimIndicator_Initialize
MbimIndicator.h, 1427
MbimInternal.h, 1428
MBIM_CLOSE, 1429
MBIM_CLOSE_DONE, 1429
MBIM_CLOSE_MSG, 1429
MBIM_COMMAND, 1429
MBIM_COMMAND_DONE, 1429
MBIM_COMMAND_MSG, 1429
MBIM_FRAGMENT_MSG, 1429
MBIM_HOST_ERROR, 1429
MBIM_OPEN, 1429
MBIM_OPEN_DONE, 1429
MBIM_OPEN_MSG, 1429
MbimLayer.h, 1429
MbimLayer_Create, 1429
MbimLayer_SendQmuxPacket, 1429
MbimLayer_ShutDown, 1430
MbimLayer_Create
MbimLayer.h, 1429
MbimLayer_SendQmuxPacket
MbimLayer.h, 1429
MbimLayer_ShutDown
MbimLayer.h, 1430
MbimSyncObject, 189
condition, 190
informationBuffer, 190
informationBufferLength, 190
MbimSyncObject.h, 1430
mutex, 190
status, 190
MbimSyncObject.h, 1430
MbimSyncObject, 1430
MbimSyncObject_Destroy, 1431
MbimSyncObject_DoneCallback, 1431
MbimSyncObject_Initialize, 1431
MbimSyncObject_Lock, 1431
MbimSyncObject_Signal, 1432
MbimSyncObject_TimedWait, 1432
MbimSyncObject_Unlock, 1432
MbimSyncObject_Wait, 1432
MbimSyncObject_Destroy
MbimSyncObject.h, 1431
MbimSyncObject_DoneCallback
MbimSyncObject.h, 1431

MbimSyncObject_Initialize
MbimSyncObject.h, 1431
MbimSyncObject_Lock
MbimSyncObject.h, 1431
MbimSyncObject_Signal
MbimSyncObject.h, 1432
MbimSyncObject_TimedWait
MbimSyncObject.h, 1432
MbimSyncObject_Unlock
MbimSyncObject.h, 1432
MbimSyncObject_Wait
MbimSyncObject.h, 1432
MbimTransaction, 190
MbimTransaction.h, 1433
pDoneCallback, 191
pDoneCallbackContext, 191
pNext, 191
pPrev, 191
status, 191
transactionId, 191
MbimTransaction.h, 1433
MBIM_DONE_CALLBACK, 1433
MbimTransaction, 1433
MbimTransaction_Initialize, 1434
MbimTransaction_Initialize
MbimTransaction.h, 1434
MbimTransport, 191
deviceFd, 192
indicatorList, 192
indicatorListLock, 192
MbimTransport.h, 1435
pIncomingMessage, 192
readThread, 192
shutdownFd, 192
transactionId, 192
transactionIdLock, 192
transactionList, 192
transactionListLock, 192
writeLock, 192
MbimTransport.h, 1434
MBIM_MAX_CIDS, 1435
MBIM_UUID_SIZE, 1435
MbimTransport, 1435
MbimTransport_AttachIndicator, 1436
MbimTransport_CancelTransaction, 1436
MbimTransport_DetachIndicator, 1436
MbimTransport_GetNextTransactionId, 1437
MbimTransport_Initialize, 1437
MbimTransport_SendCommand, 1437
MbimTransport_ShutDown, 1438
MbimTransport_AttachIndicator
MbimTransport.h, 1436
MbimTransport_CancelTransaction
MbimTransport.h, 1436
MbimTransport_DetachIndicator
MbimTransport.h, 1436
MbimTransport_GetNextTransactionId
MbimTransport.h, 1437
MbimTransport_Initialize
MbimTransport.h, 1437
MbimTransport_SendCommand
MbimTransport.h, 1437
MbimTransport_ShutDown
MbimTransport.h, 1438
mcc
nas_CSGID, 213
nas_CSGInfo, 214
nas_MNRInfo, 282
nas_netSelectionPref, 284
nas_NetworkScan3GPPInfo, 288
nas_NR5GSysInfoTlv, 302
nas_OperatorPLMNData, 309
nas_PlmnID, 317
nas_PrefPLMNListTlv, 320
pack_nas_SLQSGetPLMNName_t, 470
unpack_nas_GetHomeNetwork_t, 881
unpack_nas_SLQSGetHomeNetwork_t, 897
mccIncPcsDigit
nas_homeNwMNC3GppTlv, 246
mccM
nas_minBasedIMSI, 281
mccT
nas_trueIMSI, 368
mdmCallDurationActive
unpack_wds_SLQSGetDUNCallInfo_t, 1173
mdn
nas_Mdn, 280
mdnLen
nas_Mdn, 280
meanThroughputClass
LibPackGPRSRequestedQoS, 119
wds_GPRSQoS, 1264
meid
unpack_dms_GetSerialNumbers_t, 765
meidSize
unpack_dms_GetDeviceSerialNumbers_t, 752
meidString
unpack_dms_SLQSSwiGetSerialNoExt_t, 801
memoryDumpAllowed
unpack_swidms_SLQSSwiDmsGetSecureInfo_t, 1048
message
unpack_sms_SLQSGetSMS_t, 1020
message_type
NASOTAMessageTlv, 394
messageClass
sms_routeEntry, 666
messageFailureCode
unpack_sms_SendSMS_t, 1015
messageFormat
pack_sms_SaveSMS_t, 532
pack_sms_SendSMS_t, 532
sms_sendAsyncsmsParams, 668
unpack_sms_SLQSGetSMS_t, 1020
MessageHeader
MBIM_CLOSE_DONE_MSG, 184

MBIM_CLOSE_MSG, 185
MBIM_COMMAND_DONE_MSG, 185
MBIM_COMMAND_MSG, 186
MBIM_FRAGMENT_MSG, 186
MBIM_INDICATE_STATUS_MSG, 187
MBIM_OPEN_DONE_MSG, 187
MBIM_OPEN_MSG, 187
messageID
unpack_sms_SendSMS_t, 1015
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
messageId
GetClientIdReqMsg, 94
GetClientIdRspMsg, 95
QmiHdr, 642
QmiTransaction, 646
ReleaseClientIdReqMsg, 651
ReleaseClientIdRspMsg, 651
messageIndex
pack_sms_SLQSGetSMS_t, 535
pack_sms_SLQSModifySMSStatus_t, 537
qmiSmsMessageList, 644
sMSMTMessageInfo, 674
MessageLength
MBIM_MESSAGE_HEADER, 187
messageList
unpack_sms_SLQSGetSMSList_t, 1022
messageListSize
unpack_sms_SLQSGetSMSList_t, 1022
messageMode
sMSMessageModeInfo, 674
unpack_sms_SLQSWmsMemoryFullCallBack_ind-_t, 1032
MessageModeInfo
messageModeTlv, 193
messageModeTlv, 192
MessageModeInfo, 193
TlvPresent, 193
messageSize
pack_sms_SaveSMS_t, 532
pack_sms_SendSMS_t, 532
sms_sendAsyncsmsParams, 668
unpack_sms_SLQSGetSMS_t, 1020
messageTag
pack_sms_SLQSModifySMSStatus_t, 537
qmiSmsMessageList, 644
unpack_sms_SLQSGetSMS_t, 1020
MessageType
MBIM_MESSAGE_HEADER, 187
messageType
sms_routeEntry, 666
MicMute
pack_audio_SLQSSetAudioProfile_t, 411
pack_swiaudio_SLQSSetM2MAVMute_t, 546
unpack_audio_SLQSGetAudioProfile_t, 732
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1034
unpack_swiaudio_SLQSGetM2MAVMute_t, 1035
min
PackSwiAVMSSettingsPeriodsInfo, 641
MinPolicedPktSz
unpack_qos_swiQosFlow_t, 1008
minSessExp
ims_MinSessExpInfo, 103
minSize
unpack_dms_GetVoiceNumber_t, 768
minute
nas_timeInfo, 367
nas_UniversalTime, 377
mipMode
unpack_wds_GetMobileIP_t, 1153
mipStatus
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
mipstatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
mitigationDevID
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 563
pack_tmd_SLQSTmdGetMitigationLvl_t, 564
pack_tmd_SLQSTmdRegNotMitigationLvl_t, 564
mitigationDevIDLen
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 563
pack_tmd_SLQSTmdGetMitigationLvl_t, 564
pack_tmd_SLQSTmdRegNotMitigationLvl_t, 564
mitigationDevId
tmd_mitigationDevList, 686
mitigationDevIdLen
tmd_mitigationDevList, 686
MitigationDevList
unpack_tmd_SLQSTmdGetMitigationDevList_t, 1065
MitigationDevListLen
unpack_tmd_SLQSTmdGetMitigationDevList_t, 1065
mnc
nas_CSGID, 213
nas_CSGInfo, 214
nas_MNRInfo, 282
nas_netSelectionPref, 284
nas_NetworkScan3GPPInfo, 288
nas_NR5GSysInfoTlv, 302
nas_OperatorPLMNData, 309
nas_PlmnID, 317
nas_PrefPLMNListTlv, 320
pack_nas_SLQSGetPLMNName_t, 470
unpack_nas_GetHomeNetwork_t, 881
unpack_nas_SLQSGetHomeNetwork_t, 897
mnc_includes_pcs_digit
nas_NetworkScan3GPPInfo, 288
mnc_pcs_digit
nas_PrefPLMNListTlv, 320
mncPcsDigits
nas_CSGID, 213

mobileCountryCode
sMSEtwsPlmnInfo, 673
mobileIP
pack_wds_SLQSSetWdsEventCallback_t, 630
mobileNetworkCode
sMSEtwsPlmnInfo, 673
mode
pack_dms_SetEventReport_t, 421
pack_dms_SetPower_t, 423
pack_loc_SetOperationMode_t, 449
pack_sms_SLQSGetSmsBroadcastConfig_t, 535
pack_sms_SLQSSetSmsBroadcastActivation_t, 538
pack_sms_SLQSSetSmsBroadcastConfig_t, 539
pack_wds_SetMobileIP_t, 618
swidms_MultiplexedSimModeTlv, 681
swidms_SetMultiplexedModeTlv, 681
uim_refreshevent, 717
voice_callInfo, 1222
ModePref
nas_ModePrefTlv, 283
NASModePreferenceTlv, 392
modelIdStr
_litefw_FirmwareFileInfo, 40
modelString
pack_dms_SLQSSwiSetHostDevInfo_t, 427
unpack_dms_SLQSSwiGetHostDevInfo_t, 799
modelid
unpack_dms_GetModelID_t, 759
modelid_str
unpack_dms_GetFirmwareInfo_t, 753
modemMode
nas_CommInfo, 212
modemindex
pack_fms_SetImagesPreference_t, 437
month
nas_timeInfo, 367
nas_UniversalTime, 377
msgCount
sms_messageWaitingInfoContent, 663
msgDelFailureCause
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
msgDelFailureType
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
msgProtocol
sms_msgProtocolResp, 664
msgType
sms_messageWaitingInfoContent, 663
msgWaitInfo
sms_getMsgWaitingInfo, 660
unpack_sms_SLQSWmsMessageWaitingCall-Back_ind_t, 1032
msgid
pack_qmi_t, 528
unpack_qmi_t, 988
msgtype
common.h, 1292
Mtu
unpack_wds_SLQSGetRuntimeSettings_t, 1177
MultDisc
nas_protocolSubtypeElement, 322
multiplexedSimMode
unpack_swidms_SLQSSwiDmsGetUimSlot_t, 1049
multiplier
unpack_qos_pktErrRate_t, 991
mutex
MbimSyncObject, 190
QmiSyncObject, 645
MuxID
qos_BindDataPortMuxID_t, 649

NAI
unpack_wds_GetMobileIPProfile_t, 1155
NAS_NAM_NAME_LENGTH
nas.h, 1452
NAS_PLMN_LENGTH
nas.h, 1452
NASAcqOrderPrefTlv, 386
AcqOrderLen, 386
AcqOrderPref, 386
TlvPresent, 386
NASBandPreferenceTlv, 386
band_pref, 387
TlvPresent, 387
NASCiotAcqOrderPrefTlv, 387
CiotAcqOrderLen, 387
CiotAcqOrderPref, 387
TlvPresent, 387
NASCiotLteOpModePrefTlv, 387
CiotLteOpModePref, 388
TlvPresent, 388
NASEmergencyModeTlv, 388
EmerMode, 388
TlvPresent, 388
NASGWAcqOrderPrefTlv, 389
GWAcqOrderPref, 390
TlvPresent, 390
NASLTEBandPreferenceTlv, 390
LTEBandPref, 390
TlvPresent, 390
NASLteM1BandPrefTlv, 390
LteM1BandPref, 390
TlvPresent, 391
NASLteNB1BandPrefTlv, 391
LteNB1BandPref, 392
TlvPresent, 392
NASLteNasReleaseInfoTlv, 391
nas_major, 391
nas_minor, 391
nas_release, 391
TlvPresent, 391
NASModePreferenceTlv, 392
ModePref, 392
TlvPresent, 392

NASNetSelPreferenceTlv, 392
NetSelPref, 392
TlvPresent, 392
NASNr5gBandPrefTlv, 393
Nr5gBandPrefbits1_64, 393
Nr5gBandPrefbits_129_192, 393
Nr5gBandPrefbits_193_256, 393
Nr5gBandPrefbits_65_128, 393
TlvPresent, 393
NASOTAMessageTlv, 393
data_buf, 394
data_len, 394
message_type, 394
TlvPresent, 394
NASPRLPreferenceTlv, 399
PRLPref, 399
TlvPresent, 400
NASPhyCaAggPcellInfo, 394
dl_bw_value, 395
freq, 395
iLTEbandValue, 395
pci, 395
TlvPresent, 395
NASPhyCaAggScellArray, 395
band, 396
cphy_ca_dl_bandwidth, 396
cphy_scell_info_list_len, 396
freq, 396
pci, 396
scell_idx, 396
scell_state, 396
TlvPresent, 396
NASPhyCaAggScellDlBw, 396
dl_bw_value, 397
TlvPresent, 397
NASPhyCaAggScellIndType, 397
freq, 398
pci, 398
scell_state, 398
TlvPresent, 398
NASPhyCaAggScellIndex, 397
scell_idx, 397
TlvPresent, 397
NASPhyCaAggScellInfo, 398
dl_bw_value, 399
freq, 399
iLTEbandValue, 399
pci, 399
scell_state, 399
TlvPresent, 399
NASQmiCbkNasSwiOTAMessageInd, 400
nasRelInfoTlv, 400
otaMsgTlv, 400
timeTlv, 400
NASQmiCbkNasSystemSelPrefInd, 400
AOPTlv, 402
BPTlv, 402
CiotAOPTlv, 402
CiotOpMPTlv, 402
EMTlv, 402
GWAOPTlv, 402
LBPTlv, 402
M1BandPTlv, 402
MPTlv, 402
NB1BandPTlv, 402
NR5gBandPTlv, 402
NSPTlv, 402
PRLPTlv, 402
RPTlv, 403
RatDMTlv, 403
SDPTlv, 403
NASRatDisabledMaskTlv, 403
RatDisabledMask, 403
TlvPresent, 403
NASRoamPreferenceTlv, 403
RoamPref, 403
TlvPresent, 403
NASServDomainPrefTlv, 404
SrvDomainPref, 404
TlvPresent, 404
NASServingSystemInfo, 404
csAttachState, 405
hdrPersonality, 405
psAttachState, 405
radioInterfaceList, 405
radioInterfaceNo, 405
registrationState, 405
selectedNetwork, 405
NASTimeInfoTlv, 405
time, 406
TlvPresent, 406
NB1BandPTlv
NASQmiCbkNasSystemSelPrefInd, 402
NMEAData
unpack_loc_EventNMEA_Ind_t, 850
NR5gBandPTlv
NASQmiCbkNasSystemSelPrefInd, 402
NSPTlv
NASQmiCbkNasSystemSelPrefInd, 402
nUsbNetNum
pack_swidms_SLQSSwiDmsSetUsbNetNum_t, 555
NWQoSStatus
unpack_qos_SLQSQosGetNetworkStatus_t, 994
NWRegStat
unpack_sms_SLQSNWRegInfoCallback_ind_t, 1024
naiSize
unpack_wds_GetMobileIPProfile_t, 1155
namID
pack_nas_SLQSNasGet3GPP2Subscription_t, 476
pack_nas_SLQSSet3GPP2SubscriptionInfo_t, 498
voice_airTimer, 1203
voice_prefVoiceSO, 1243
voice_roamTimer, 1247

namName
nas_namName, 284
namNameLen
nas_namName, 284
Name
unpack_nas_GetServingNetwork_t, 886
name
loc_AppProviderInfoTlv, 164
nas_CSGInfo, 214
unpack_nas_GetHomeNetwork_t, 881
unpack_nas_SLQSGetHomeNetwork_t, 897
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
name_len
nas_CSGInfo, 214
nameLen
loc_AppProviderInfoTlv, 164
voice_remotePartyName, 1245
namePI
voice_remotePartyName, 1245
nameSize
unpack_nas_GetHomeNetwork3GPP2_t, 880
unpack_nas_GetServingNetwork_t, 886
nameString
pack_dms_SLQSSwiSetOSInfo_t, 427
unpack_dms_SLQSSwiGetOSInfo_t, 800
namelength
unpack_omaDmFotaTlv_t, 972
namesize
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
nas.h
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_1, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_10, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_11, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_12, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_125, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_126, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_127, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_13, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_14, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_17, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_18, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_19, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_2, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_20, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_21, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_23, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_24, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_25, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_250, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_26, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_27, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_28, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_29, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_3, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_30, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_31, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_32, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_33, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_34, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_35, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_36, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_37, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_38, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_39, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_4, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_40, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_41, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_42, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_43, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_46, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_47, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_48, 1455

eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_5, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_6, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_66, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_7, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_71, 1455
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_8, 1454
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_9, 1454
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6, 1455
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75, 1455
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED, 1455
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED, 1455
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6, 1455
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75, 1455
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE, 1455
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE, 1455
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE, 1455
LITE_SYS_SRV_DOMAIN_CAMPED, 1453
LITE_SYS_SRV_DOMAIN_CS_ONLY, 1453
LITE_SYS_SRV_DOMAIN_CS_PS, 1453
LITE_SYS_SRV_DOMAIN_NO_SRV, 1453
LITE_SYS_SRV_DOMAIN_PS_ONLY, 1453
nas.h, 1438
LITE_MAX_ARFCN_NUM, 1452
NAS_PLMN_LENGTH, 1452
nas_LTEBandPrefExtTlv, 1452
nas_NR5gBandPrefTlv, 1452
pack_nas_GetACCOLC, 1456
pack_nas_GetANAAAAuthenticationStatus, 1456
pack_nas_GetCDMANetworkParameters, 1456
pack_nas_GetHomeNetwork, 1456
pack_nas_GetHomeNetwork3GPP2, 1457
pack_nas_GetNetworkPreference, 1457
pack_nas_GetRFInfo, 1457
pack_nas_GetServingNetwork, 1458
pack_nas_GetServingNetworkCapabilities, 1458
pack_nas_GetSignalStrengths, 1458
pack_nas_InitiateDomainAttach, 1459
pack_nas_PerformNetworkScan, 1459
pack_nas_PerformNetworkScanPCI, 1459
pack_nas_SLQSConfigSigInfo, 1461
pack_nas_SLQSGetErrorRate, 1462
pack_nas_SLQSGetHomeNetwork, 1462
pack_nas_SLQSGetNetworkTime, 1463
pack_nas_SLQSGetOperatorNameData, 1463
pack_nas_SLQSGetPLMNName, 1463
pack_nas_SLQSGetServingSystem, 1464
pack_nas_SLQSGetServingSystemV2, 1464
pack_nas_SLQSGetSignalStrength, 1464
pack_nas_SLQSGetSysInfo, 1465
pack_nas_SLQSGetSysInfoV2, 1465
pack_nas_SLQSGetSysSelectionPref, 1466
pack_nas_SLQSGetSysSelectionPrefExt, 1466
pack_nas_SLQSGetSysSelectionPrefExtV2, 1466
pack_nas_SLQSInitiateNetworkRegistration, 1467
pack_nas_SLQSNASGetForbiddenNetworks, 1469
pack_nas_SLQSNASGeteDRXParams, 1468
pack_nas_SLQSNASGeteDRXParamsExt, 1469
pack_nas_SLQSNASSeteDRXParams, 1472
pack_nas_SLQSNASSwiGetChannelLock, 1472
pack_nas_SLQSNASSwiSetChannelLock, 1473
pack_nas_SLQSNasConfigSigInfo2, 1467
pack_nas_SLQSNasGet3GPP2Subscription, 1467
pack_nas_SLQSNasGetCellLocationInfo, 1468
pack_nas_SLQSNasGetCellLocationInfoV2, 1468
pack_nas_SLQSNasGetHDRColorCode, 1469
pack_nas_SLQSNasGetRFInfo, 1470
pack_nas_SLQSNasGetSigInfo, 1470
pack_nas_SLQSNasGetTxRxInfo, 1470
pack_nas_SLQSNasIndicationRegisterExt, 1471
pack_nas_SLQSNasIndicationRegisterV2, 1471
pack_nas_SLQSNasSwiIndicationRegister, 1472
pack_nas_SLQSNasSwiModemStatus, 1473
pack_nas_SLQSPerformIncNetworkScanReq, 1473
pack_nas_SLQSPerformNetworkScanV2, 1474
pack_nas_SLQSSet3GPP2SubscriptionInfo, 1474
pack_nas_SLQSSetBandPreference, 1475
pack_nas_SLQSSetDDTMPref, 1475
pack_nas_SLQSSetSignalStrengthsCallback, 1475
pack_nas_SLQSSetSysSelectionPref, 1476
pack_nas_SLQSSetSysSelectionPrefExt, 1476
pack_nas_SLQSSwiGetHDRPersonality, 1476
pack_nas_SLQSSwiGetHDRProtSubtype, 1477
pack_nas_SLQSSwiGetHRPDStats, 1477
pack_nas_SLQSSwiGetLteCQI, 1477

pack_nas_SLQSSwiGetLteSccRxInfo, 1478
pack_nas_SLQSSwiNetworkDebug, 1478
pack_nas_SLQSSwiPSDetach, 1478
pack_nas_SetACCOLC, 1460
pack_nas_SetCDMANetworkParameters, 1460
pack_nas_SetLURejectCallback, 1461
pack_nas_SetNetworkPreference, 1461
pack_nas_SetRFInfoCallback, 1461
pack_nas_SlqsGetLTECphyCAInfo, 1462
unpack_nas_GetACCOLC, 1479
unpack_nas_GetANAAAAuthenticationStatus, 1479
unpack_nas_GetCDMANetworkParameters, 1479
unpack_nas_GetHomeNetwork, 1480
unpack_nas_GetHomeNetwork3GPP2, 1480
unpack_nas_GetNetworkPreference, 1481
unpack_nas_GetRFInfo, 1481
unpack_nas_GetServingNetwork, 1481
unpack_nas_GetServingNetworkCapabilities, 1482
unpack_nas_GetSignalStrengths, 1482
unpack_nas_InitiateDomainAttach, 1482
unpack_nas_InitiateDomainAttach_t, 1452
unpack_nas_PerformNetworkScan, 1483
unpack_nas_SLQSConfigSigInfo, 1487
unpack_nas_SLQSConfigSigInfo_t, 1453
unpack_nas_SLQSDDTMPrefIndicationCallback_Ind, 1487
unpack_nas_SLQSGetErrorRate, 1487
unpack_nas_SLQSGetHomeNetwork, 1488
unpack_nas_SLQSGetNetworkTime, 1488
unpack_nas_SLQSGetOperatorNameData, 1489
unpack_nas_SLQSGetPLMNName, 1489
unpack_nas_SLQSGetServingSystem, 1489
unpack_nas_SLQSGetServingSystemV2, 1490
unpack_nas_SLQSGetSignalStrength, 1490
unpack_nas_SLQSGetSysInfo, 1490
unpack_nas_SLQSGetSysInfoV2, 1491
unpack_nas_SLQSGetSysSelectionPref, 1491
unpack_nas_SLQSGetSysSelectionPrefExt, 1492
unpack_nas_SLQSGetSysSelectionPrefExtV2, 1492
unpack_nas_SLQSInitiateNetworkRegistration, 1492
unpack_nas_SLQSInitiateNetworkRegistration_t, 1453
unpack_nas_SLQSNASGetForbiddenNetworks, 1495
unpack_nas_SLQSNASGeteDRXParams, 1495
unpack_nas_SLQSNASGeteDRXParamsExt, 1495
unpack_nas_SLQSNASSeteDRXParams, 1499
unpack_nas_SLQSNASSeteDRXParams_t, 1453
unpack_nas_SLQSNASSwiGetChannelLock, 1500
unpack_nas_SLQSNASSwiSetChannelLock, 1501
unpack_nas_SLQSNASSwiSetChannelLock_t, 1453
unpack_nas_SLQSNasConfigSigInfo2, 1493
unpack_nas_SLQSNasConfigSigInfo2_t, 1453
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind, 1493
unpack_nas_SLQSNasGet3GPP2Subscription, 1494
unpack_nas_SLQSNasGetCellLocationInfo, 1494
unpack_nas_SLQSNasGetCellLocationInfoV2, 1494
unpack_nas_SLQSNasGetHDRColorCode, 1496
unpack_nas_SLQSNasGetRFInfo, 1496
unpack_nas_SLQSNasGetSigInfo, 1496
unpack_nas_SLQSNasGetTxRxInfo, 1497
unpack_nas_SLQSNasIndicationRegisterExt, 1497
unpack_nas_SLQSNasIndicationRegisterExt_t, 1453
unpack_nas_SLQSNasIndicationRegisterV2, 1497
unpack_nas_SLQSNasIndicationRegisterV2_t, 1453
unpack_nas_SLQSNasNetworkRejectCallback_Ind, 1498
unpack_nas_SLQSNasNetworkTimeCallBack_ind, 1498
unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind, 1499
unpack_nas_SLQSNasRFBandInfoCallback_Ind, 1499
unpack_nas_SLQSNasSigInfoCallback_ind, 1500
unpack_nas_SLQSNasSwiIndicationRegister, 1500
unpack_nas_SLQSNasSwiIndicationRegister_t, 1453
unpack_nas_SLQSNasSwiModemStatus, 1501
unpack_nas_SLQSNasSwiOTAMessageCallback_ind, 1501
unpack_nas_SLQSNasSysInfoCallback_ind, 1502
unpack_nas_SLQSNasTimerCallback_ind, 1502
unpack_nas_SLQSPerformIncNetworkScan, 1502
unpack_nas_SLQSPerformIncNetworkScan_t, 1453
unpack_nas_SLQSPerformNetworkScanV2, 1503
unpack_nas_SLQSSet3GPP2SubscriptionInfo, 1503
unpack_nas_SLQSSet3GPP2SubscriptionInfo_t, 1453
unpack_nas_SLQSSetBandPreference, 1503
unpack_nas_SLQSSetBandPreference_t, 1453
unpack_nas_SLQSSetDDTMPref, 1504
unpack_nas_SLQSSetDDTMPref_t, 1453
unpack_nas_SLQSSetSignalStrengthsCallback, 1504
unpack_nas_SLQSSetSignalStrengthsCallback_t, 1453
unpack_nas_SLQSSetSysSelectionPref, 1504
unpack_nas_SLQSSetSysSelectionPref_t, 1453
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind, 1505
unpack_nas_SLQSSetSysSelectionPrefExt, 1505

unpack_nas_SLQSSetSysSelectionPrefExt_t, 1453
unpack_nas_SLQSSwiGetHDRPersonality, 1505
unpack_nas_SLQSSwiGetHDRProtSubtype, 1506
unpack_nas_SLQSSwiGetHRPDStats, 1506
unpack_nas_SLQSSwiGetLteCQI, 1506
unpack_nas_SLQSSwiGetLteSccRxInfo, 1507
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind, 1507
unpack_nas_SLQSSwiNetworkDebug, 1507
unpack_nas_SLQSSwiPSDetach, 1508
unpack_nas_SLQSSwiPSDetach_t, 1453
unpack_nas_SLQSSwiRandIndicatorCallback_Ind, 1508
unpack_nas_SetACCOLC, 1483
unpack_nas_SetACCOLC_t, 1453
unpack_nas_SetCDMANetworkParameters, 1483
unpack_nas_SetCDMANetworkParameters_t, 1453
unpack_nas_SetDataCapabilitiesCallback_ind, 1484
unpack_nas_SetEventReportInd, 1484
unpack_nas_SetLURejectCallback, 1484
unpack_nas_SetLURejectCallback_t, 1453
unpack_nas_SetNasLTECphyCaIndCallback_ind, 1485
unpack_nas_SetNetworkPreference, 1485
unpack_nas_SetRFInfoCallback, 1486
unpack_nas_SetRFInfoCallback_t, 1453
unpack_nas_SetRoamingIndicatorCallback_ind, 1486
unpack_nas_SetServingSystemCallback_ind, 1486
unpack_nas_SlqsGetLTECphyCAInfo, 1488
unpack_valid_nas_GetCDMANetworkParameters, 1508
unpack_valid_nas_SLQSGetServingSystem, 1509
unpack_valid_nas_SLQSGetSignalStrength, 1510
unpack_valid_nas_SLQSNasGetSigInfo, 1510
nas_AcqOrderPrefTlv, 194
acqOrdeLen, 194
pAcqOrder, 194
TlvPresent, 194
nas_ActPilotPNElement, 194
ActSetPilotPN, 195
ActSetPilotPNStrength, 195
nas_AddCDMASysInfo, 195
geoSysIdx, 195
regPrd, 195
nas_AddSysInfo, 195
cellBroadcastCap, 196
geoSysIdx, 196
nas_BandPrefInfoTlv, 196
bits_129_192, 197
bits_193_256, 197
bits_1_64, 197
bits_65_128, 197
TlvPresent, 197
nas_BandPrefTlv, 197
BandPref, 199
TlvPresent, 199
nas_CDMAChannel, 200
priChA, 201
priChB, 201
secChA, 201
secChB, 201
nas_CDMAECIOThresh, 201
CDMAECIOThreshListLen, 201
pCDMAECIOThreshList, 201
nas_CDMAInfo, 202
baseId, 202
baseLat, 202
baseLong, 202
nid, 202
refpn, 203
sid, 203
nas_CDMARSSIThresh, 203
CDMARSSIThreshListLen, 203
pCDMARSSIThreshList, 203
nas_CDMASysInfo, 203
baseId, 206
baseLat, 206
baseLong, 206
bsInfoValid, 206
bsPRev, 206
bsPRevValid, 206
ccsSupported, 206
ccsSupportedValid, 206
cdmaSysIdValid, 206
isSysPrlMatch, 206
isSysPrlMatchValid, 206
MCC, 206
MNC, 206
networkID, 206
networkIdValid, 206
pRevInUse, 207
pRevInUseValid, 207
packetZone, 207
packetZoneValid, 207
sysInfoCDMA, 207
systemID, 207
nas_CDMASysInfoExt, 207
imsi_11_12, 207
MCC, 207
nas_CSGID, 212
id, 213
mcc, 213
mnc, 213
mncPcsDigits, 213
rat, 213
nas_CSGInfo, 213
csg_list_cat, 214
id, 214
mcc, 214
mnc, 214
name, 214

name_len, 214
nas_CSGInfoTlv, 214
csg_info_len, 215
csgInfo, 215
TlvPresent, 215
nas_CallBarringSysInfo, 199
csBarStatus, 199
psBarStatus, 199
nas_CiotAcqOrderPrefTlv, 209
ciotAcqOrderLen, 210
pCiotAcqOrder, 210
TlvPresent, 210
nas_CiotLteOpModePrefTlv, 210
ciotLteOpModePref, 210
TlvPresent, 210
nas_CommInfo, 210
imsRegState, 212
modemMode, 212
psState, 212
systemMode, 212
temperature, 212
nas_CsgId, 212
csgId, 212
TlvPresent, 212
nas_DDTMSettings, 219
CurrentDDTMStatus, 220
DDTMAction, 220
DDTMPrefSetting, 220
SOLen, 220
SOListAction, 220
ServiceOption, 220
nas_DRCParams, 223
DRCCover, 223
DRCValue, 223
nas_DataStatusDetail, 217
IPAddress, 218
LastErrCode, 218
nas_DeviceConfigDetail, 221
Chipset, 222
HWVersion, 222
QLIC, 222
Technology, 222
nas_ECIOThresh, 224
ECIOThresListLen, 224
pECIOThresList, 224
nas_EdrxCiotLteMode, 225
lteOpMode, 225
TlvPresent, 225
nas_EdrxCycleLength, 225
cycleLength, 226
TlvPresent, 226
nas_EdrxEnableType, 226
edrxEnabled, 226
TlvPresent, 226
nas_EdrxPagingTimeWindow, 226
edrxPtw, 227
TlvPresent, 227
nas_EdrxRatType, 227
edrxRatType, 227
TlvPresent, 227
nas_EmerModeTlv, 227
EmerMode, 228
TlvPresent, 228
nas_ForbiddenNetworks3GPP, 229
forbiddenNwInstLen, 230
MCC, 230
MNC, 230
TlvPresent, 230
nas_GERANInfo, 230
arfcn, 231
bsic, 231
cellID, 231
insNmrCellInfo, 231
lac, 232
nmrInst, 232
plmn, 232
rxLev, 232
timingAdvance, 232
nas_GSMRSSIThresh, 234
GSMRSSIThreshListLen, 235
pGSMRSSIThreshList, 235
nas_GSMSrvStatusInfo, 235
isPrefDataPath, 236
srvStatus, 236
trueSrvStatus, 236
nas_GSMSysInfo, 236
cellId, 238
cellIdValid, 238
dtmSupp, 238
dtmSuppValid, 238
egprsSupp, 238
egprsSuppValid, 238
lac, 238
lacValid, 239
MCC, 239
MNC, 239
networkIdValid, 239
regRejectInfoValid, 239
rejCause, 239
rejectSrvDomain, 239
sysInfoGSM, 239
nas_GWAcqOrderPrefTlv, 239
GWAcqOrderPref, 239
TlvPresent, 239
nas_HDRECIOThresh, 239
HDRECIOThreshListLen, 240
pHDRECIOThreshList, 240
nas_HDRIOThresh, 240
HDRIOThreshListLen, 240
pHDRIOThreshList, 240
nas_HDRPersonality_Ind_Data, 241
pCurrentPersonality, 241
pPersonalityListLength, 241
pProtocolSubtypeElement, 241
nas_HDRRSSIThresh, 241
HDRRSSIThreshListLen, 242

pHDRRSSIThreshList, 242
nas_HDRSINRThresh, 242
HDRSINRThresListLen, 242
pHDRSINRThresList, 242
nas_HDRSINRThreshold, 242
HDRSINRThreshListLen, 243
pHDRSINRThreshList, 243
nas_HDRSysInfo, 243
hdrActiveProt, 245
hdrActiveProtValid, 245
hdrPersonality, 245
hdrPersonalityValid, 245
is856SysId, 245
is856SysIdValid, 245
isSysPrlMatch, 245
isSysPrlMatchValid, 245
sysInfoHDR, 245
nas_IOThresh, 248
IOThresListLen, 249
pIOThresList, 249
nas_ImsVoiceSupportLteTlv, 246
ImsVoiceSupportLte, 247
TLVPresent, 247
nas_LTEARFCNTlv, 249
lteArfcn, 249
numlteArfcn, 249
nas_LTEBandPrefExtTlv
nas.h, 1452
nas_LTEBandPrefTlv, 250
LTEBandPref, 251
TlvPresent, 252
nas_LTEInfo, 256
band, 257
bandwidth, 257
emmConnState, 258
emmState, 258
emmSubState, 258
RXChan, 258
TXChan, 258
nas_LTEInfoInterfreq, 258
freqsLen, 258
InfoInterfreq, 258
ueInIdle, 258
nas_LTEInfoIntrafreq, 258
CellParams, 260
cellReselPriority, 260
cellsLen, 260
earfcn, 260
globalCellId, 260
plmn, 260
sIntraSearch, 260
sNonIntraSearch, 260
servingCellId, 260
tac, 260
threshServingLow, 260
ueInIdle, 260
nas_LTEInfoNeighboringGSM, 260
freqsLen, 261
LteGsmCellInfo, 261
ueInIdle, 261
nas_LTEInfoNeighboringWCDMA, 261
freqsLen, 262
ueInIdle, 262
nas_LTEOperationMode, 266
pLTEOperationMode, 267
TlvPresent, 267
nas_LTEOperationalModeTlv, 266
ciotLteOpMode, 266
TlvPresent, 266
nas_LTERSRPThresh, 270
LTERSRPThreshListLen, 271
pLTERSRPThreshList, 271
nas_LTERSRQThresh, 271
LTERSRQThreshListLen, 271
pLTERSRQThreshList, 271
nas_LTERSSIThresh, 271
LTERSSIThreshListLen, 272
pLTERSSIThreshList, 272
nas_LTESNRThresh, 274
LTESNRThresListLen, 275
pLTESNRThresList, 275
nas_LTESNRThreshold, 275
LTESNRThreshListLen, 275
pLTESNRThreshList, 275
nas_LTESigRptCfg, 272
avgPeriod, 273
rptRate, 273
nas_LTESigRptConfig, 273
avgPeriod, 274
rptRate, 274
nas_LTESysInfo, 276
cellId, 277
cellIdValid, 278
lac, 278
lacValid, 278
MCC, 278
MNC, 278
networkIdValid, 278
regRejectInfoValid, 278
rejCause, 278
rejectSrvDomain, 278
sysInfoLTE, 278
tac, 278
tacValid, 278
nas_LteCiotOpModeTlv, 252
campedCiotLteOpMode, 252
TlvPresent, 252
nas_LteEarfcnInfo, 253
lteEarfcn, 253
lteInterEarfcnlen, 253
TlvPresent, 253
nas_LteEmbmsCoverageTlv, 253
LteEmbmsCoverage, 254
TLVPresent, 254
nas_LteEmbmsTraceIdTlv, 254
LteEmbmsTraceId, 254

TLVPresent, 254
nas_LteM1BandPrefTlv, 262
lteM1BandPref, 263
TlvPresent, 264
nas_LteNb1BandPrefTlv, 264
lteNb1BandPref, 265
TlvPresent, 266
nas_LteOpMode, 267
lteOpMode, 267
TlvPresent, 267
nas_LteRegDomainTlv, 269
LteRegDomain, 270
TLVPresent, 270
nas_LteVoiceDomainTlv, 278
LteVoiceDomain, 278
TLVPresent, 279
nas_MNCPCSDigitStatus, 281
MCC, 282
MNC, 282
MNCIncPCSDigit, 282
TLVPresent, 282
nas_MNRInfo, 282
mcc, 282
mnc, 282
rat, 282
nas_Mdn, 280
mdn, 280
mdnLen, 280
nas_ModePrefTlv, 282
ModePref, 283
TlvPresent, 283
nas_NR5GCellStatusInfoTlv, 296
nr5gCellStatus, 296
TLVPresent, 296
nas_NR5GCellStatusTlv, 296
nr5gCellStatus, 297
TlvPresent, 297
nas_NR5GSerStatTlv, 297
isPrefDataPath, 298
srvStatus, 298
TlvPresent, 298
trueSrvStatus, 298
nas_NR5GSrvStatusTlv, 298
isPrefDataPath, 299
srvStatus, 299
TLVPresent, 299
trueSrvStatus, 299
nas_NR5GSysInfoTlv, 299
cellId, 302
cellIdValid, 302
isSysForbidden, 302
isSysForbiddenValid, 302
lac, 302
lacValid, 302
mcc, 302
mnc, 302
nwIdValid, 302
regRejectInfoValid, 302
rejectCause, 302
rejectSrvDomain, 302
roamStatus, 303
roamStatusValid, 303
srvCapability, 303
srvCapabilityValid, 303
srvDomain, 303
srvDomainValid, 303
TLVPresent, 303
tac, 303
tacValid, 303
nas_NR5GSystemInfoTlv, 303
cellId, 306
cellIdValid, 306
lac, 306
lacValid, 306
MCC, 306
MNC, 306
nwIdValid, 306
regRejectInfoValid, 306
rejCause, 306
rejectSrvDomain, 306
roamStatus, 306
roamStatusValid, 306
srvCapValid, 306
srvDomain, 306
srvDomainValid, 306
srvcapability, 306
sysForbidden, 306
sysForbiddenValid, 306
tac, 306
tacValid, 306
TlvPresent, 306
nas_NR5gBandPrefTlv
nas.h, 1452
nas_NetSelPrefTlv, 284
NetSelPref, 285
TlvPresent, 285
nas_NetworkNameSrcTlv, 285
nw_name_src, 286
nw_name_src_len, 286
TlvPresent, 286
nas_NetworkScan3GPPInfo, 287
mcc, 288
mnc, 288
mnc_includes_pcs_digit, 288
network_desc, 288
network_desc_len, 288
network_status, 288
rat, 288
nas_NetworkScan3GPPInfoTlv, 288
netScan3GPPInfo, 289
num_inst, 289
TlvPresent, 289
nas_NetworkScanStatusTlv, 289
scanStatus, 290
TlvPresent, 290
nas_NetworkStat1x, 290

ActSetCnt, 292
NeighborSetCnt, 292
pActPilotPNElements, 292
pNeighborSetPilotPN, 292
RX_EC_IO, 292
RX_PWR, 292
SO, 292
State, 292
TX_PWR, 292
nas_NetworkStatEVDO, 292
MACIndex, 294
PER, 294
pSectorID, 294
PilotEnergy, 294
RX_PWR, 294
SNR, 294
SectorIDLen, 294
State, 294
nas_NumScellsConfig, 306
numScellsCfg, 307
TlvPresent, 307
nas_OperatorPLMNData, 308
lac1, 309
lac2, 309
mcc, 309
mnc, 309
PLMNRecID, 309
nas_PLMNNetworkName, 317
numInstance, 317
PLMNNetName, 317
nas_PLMNNetworkNameData, 317
codingScheme, 319
countryInitials, 319
longName, 319
longNameLen, 319
longNameSpareBits, 319
shortName, 319
shortNameLen, 319
shortNameSpareBits, 319
nas_PRLPrefTlv, 320
PRLPref, 321
TlvPresent, 321
nas_PhyCaAggDlBW, 309
aggDlBW, 310
TlvPresent, 310
nas_PhyCaAggPcellInfo, 310
dl_bw_value, 311
freq, 311
iLTEbandValue, 311
pci, 311
TlvPresent, 311
nas_PhyCaAggScellDlBw, 311
dl_bw_value, 311
TlvPresent, 311
nas_PhyCaAggScellIndType, 312
freq, 312
pci, 312
scell_state, 312
TlvPresent, 312
nas_PhyCaAggScellIndex, 311
scell_idx, 312
TlvPresent, 312
nas_PhyCaAggScellInfo, 313
dl_bw_value, 315
freq, 315
iLTEbandValue, 315
pci, 315
scell_state, 315
TlvPresent, 315
nas_PilotSetData, 315
NumPilots, 315
pPilotSetInfo, 315
nas_PilotSetParams, 316
PilotPN, 316
PilotStrength, 316
PilotType, 316
nas_PlmnID, 316
mcc, 317
mnc, 317
pcsDigit, 317
TlvPresent, 317
nas_PrefPLMNListTlv, 319
mcc, 320
mnc, 320
mnc_pcs_digit, 320
plmnListLen, 320
nas_QmiNas3GppNetworkInfo, 323
Desription, 324
Forbidden, 324
InUse, 324
MCC, 324
MNC, 324
Preferred, 324
Roaming, 324
nas_QmiNas3GppNetworkRAT, 324
MCC, 325
MNC, 325
RAT, 325
nas_QmisNasPcsDigit, 325
includes_pcs_digit, 325
MCC, 326
MNC, 326
nas_QmisNasSlqsNasPCICellInfo, 326
cellID, 326
freq, 326
GlobalCellID, 326
nasQmisNasPcsDigit, 326
PlmnLen, 326
nas_QmisNasSlqsNasPCIInfo, 326
nasQmisNasSlqsNasPCICellInfo, 327
PCICellInfoLen, 327
rsrp, 327
rsrpRx0, 327
rsrpRx1, 327
rsrq, 327
rsrqRx0, 328

rsrqRx1, 328
nas_RFBandInfoElements, 329
activeBandClass, 330
activeChannel, 330
radioInterface, 330
nas_RFBandInfoExtTlv, 332
activeBand, 332
activeChannel, 333
radioIf, 333
TlvPresent, 333
nas_RFBandInfoTlv, 333
activeBand, 334
activeChannel, 334
radioIf, 334
TlvPresent, 334
nas_RFBandwidthInfoTlv, 336
bandwidth, 337
radioIf, 337
TlvPresent, 337
nas_RFDedicatedBandInfoTlv, 339
dedicatedBand, 339
radioIf, 339
TlvPresent, 340
nas_RFInfoTlv, 340
activeBandClass, 340
activeChannel, 340
radioInterface, 340
radioInterfaceSize, 341
TlvPresent, 341
nas_RSRPThresh, 342
pRSRPThresList, 343
RSRPThresListLen, 343
nas_RSRQThresh, 343
pRSRQThresList, 344
RSRQThresListLen, 344
nas_RSSIThresh, 344
pRSSIThresList, 345
RSSIThresListLen, 345
nas_RankIndicatorTlv, 328
count0, 328
count1, 328
TlvPresent, 328
nas_RatDisabledMaskTlv, 328
ratDisabledMask, 329
TlvPresent, 329
nas_RejectReasonTlv, 329
rejectCause, 329
serviceDomain, 329
TlvPresent, 329
nas_RfBandInfoExtFormat, 330
pInstancesSize, 331
pRfBandInfoExtFormatParam, 331
TlvPresent, 331
nas_RfBandInfoExtFormatElements, 331
activeBand, 331
activeChannel, 331
radioInterface, 331
nas_RfBandInfoList, 333
pInstanceSize, 333
pRFBandInfoParam, 333
nas_RfBandwidthInfo, 334
pInstancesSize, 335
pRfBandwidthInfoParam, 335
TlvPresent, 335
nas_RfBandwidthInfoElements, 335
bandwidth, 336
radioInterface, 336
nas_RfDedicatedBandInfo, 337
pInstancesSize, 338
pRfDedicatedBandInfoParam, 338
TlvPresent, 338
nas_RfDedicatedBandInfoElements, 338
dedicatedBand, 339
radioInterface, 339
nas_RoamPrefTlv, 342
RoamPref, 342
TlvPresent, 342
nas_RxSigInfo, 346
isRadioTuned, 347
rsrp, 347
rxChainIndex, 347
rxPower, 347
nas_SLQSSignalStrengthsIndReq, 354
ecioDelta, 355
ecioThresholdList, 356
ecioThresholdListLen, 356
ioDelta, 356
lteRsrpDelta, 356
lteSnrDelta, 356
rsrqDelta, 356
rxSignalStrengthDelta, 356
sinrDelta, 356
sinrThresholdList, 356
sinrThresholdListLen, 356
nas_SLQSSignalStrengthsInformation, 356
ecioInfo, 357
errorRateInfo, 357
io, 357
lteRsrpinfo, 357
lteSnrinfo, 357
rsrqInfo, 357
rxSignalStrengthInfo, 357
sinr, 357
nas_SLQSSignalStrengthsTlv, 357
sSLQSSignalStrengthsInfo, 358
TlvPresent, 358
nas_SccRxInfo, 348
numInstances, 349
rsrq, 349
sigInfo, 349
snr, 349
TlvPresent, 349
nas_SignalQualityTlv, 352
signal_quality, 352
signal_quality_len, 352
TlvPresent, 352

nas_SignalStrengthTlv, 352
radioInterface, 353
signalStrength, 353
TlvPresent, 353
nas_SignalStrngthTlv, 353
signal_strength, 353
signal_strength_len, 353
TlvPresent, 353
nas_SimRejInfoTlv, 353
SimRejInfo, 354
TLVPresent, 354
nas_SrvDomainPrefTlv, 358
SrvDomainPref, 358
TlvPresent, 358
nas_SrvRegRestricTlv, 359
SrvRegRestriction, 359
TlvPresent, 359
nas_SrvRegRestrictionTlv, 358
SrvRegRestriction, 359
TLVPresent, 359
nas_SrvStatusInfo, 360
isPrefDataPath, 360
srvStatus, 360
nas_TDSCDMABandPrefTlv, 363
TdscdmaBandPref, 363
TlvPresent, 363
nas_TDSCDMAECIOThresh, 363
nas_TDSCDMARSCPThresh, 364
nas_TDSCDMARSSIThresh, 364
nas_TDSCDMASINRCONFThresh, 365
nas_TDSCDMASINRThresh, 365
nas_UMTSExtInfo, 369
cellId, 370
ecio, 370
geranInst, 370
geranInstElement, 371
lac, 371
plmn, 371
psc, 371
rscp, 371
squal, 371
srxLvl, 371
TlvPresent, 371
uarfcn, 371
umtsInst, 371
umtsInstElement, 371
nas_UMTSInfo, 371
cellID, 372
ecio, 372
geranInst, 372
GeranInstInfo, 372
lac, 372
plmn, 372
psc, 372
rscp, 373
UMTSInstInfo, 373
uarfcn, 373
umtsInst, 373
nas_UMTSinstInfo, 374
umtsEcio, 375
umtsPsc, 375
umtsRscp, 375
umtsUarfcn, 375
nas_UniversalTime, 376
day, 377
dayOfWeek, 377
hour, 377
minute, 377
month, 377
second, 377
year, 377
nas_UsageSettingTlv, 377
TlvPresent, 377
UsageSetting, 377
nas_VoiceDomainPrefTlv, 377
TlvPresent, 378
VoiceDomainPref, 378
nas_WCDMAARFCNTlv, 378
numWcdmaArfcn, 378
wcdmaArfcn, 378
nas_WCDMACellInfoExt, 379
TlvPresent, 380
wAgc, 380
wDlBler, 380
wTxAgc, 380
nas_WCDMAECIOThresh, 380
nas_WCDMAInfoLTENeighborCell, 381
umtsLTENbrCellLen, 381
wcdmaRRCState, 381
nas_WCDMARSSIThresh, 381
nas_WCDMASysInfo, 382
cellId, 385
cellIdValid, 385
hsCallStatus, 385
hsCallStatusValid, 385
hsInd, 385
hsIndValid, 385
lac, 385
lacValid, 385
MCC, 385
MNC, 385
networkIdValid, 385
psc, 385
pscValid, 385
regRejectInfoValid, 385
rejCause, 385
rejectSrvDomain, 385
sysInfoWCDMA, 385
nas_acqOrderPref, 193
acqOrdeLen, 193
pAcqOrder, 194
nas_callBarStatus, 199
csBarStatus, 200
psBarStatus, 200
nas_cellParams, 207
pci, 208

rsrp, 208
rsrq, 208
rssi, 208
srxlev, 208
nas_ciotAcqOrderPref, 208
ciotAcqOrderLen, 209
pCiotAcqOrder, 209
nas_currentPLMN, 215
MCC, 216
MNC, 216
netDescr, 216
netDescrLength, 216
nas_dataSrvCapabilities, 216
dataCapabilities, 217
dataCapabilitiesLen, 217
nas_detailSvcInfo, 220
hdrHybrid, 221
hdrSrvStatus, 221
isSysForbidden, 221
srvCapability, 221
srvStatus, 221
nas_dirNum, 222
dirNum, 223
dirNumLen, 223
nas_ecioListElement, 223
ecio, 224
radioIf, 224
nas_errorRateListElement, 228
errorRate, 229
radioIf, 229
nas_geranInstArr, 232
geranArfcn, 232
geranBsicBcc, 232
geranBsicNcc, 232
geranRank, 232
geranRssi, 233
nas_geranInstInfo, 233
geranArfcn, 233
geranBsicBcc, 233
geranBsicNcc, 233
geranRssi, 233
nas_gsmCellInfo, 233
arfcn, 234
band1900, 234
bsicId, 234
cellIdValid, 234
rssi, 234
srxlev, 234
nas_homeNwMNC3GppTlv, 245
is3GppNw, 246
mccIncPcsDigit, 246
TlvPresent, 246
nas_homeSIDNID, 246
numInstances, 246
SidNid, 246
nas_infoInterFreq, 247
cell_resel_priority, 248
cellInterFreqParams, 248
cells_len, 248
earfcn, 248
threshXHigh, 248
threshXLow, 248
nas_lteBandPrefExt, 249
bits_129_192, 250
bits_193_256, 250
bits_1_64, 250
bits_65_128, 250
nas_lteEARFCN, 252
earfcn0, 253
earfcn1, 253
status, 253
nas_lteGsmCellInfo, 255
cellReselPriority, 255
cells_len, 255
GsmCellInfo, 256
nccPermitted, 256
threshGsmHigh, 256
threshGsmLow, 256
nas_lteOpModeTlv, 267
lteOpMode, 268
lteOpModeLen, 268
MCC, 268
MNC, 268
TlvPresent, 268
nas_ltePCI, 268
earfcn, 269
pci, 269
status, 269
nas_lteRsrpinformation, 270
rsrplevel, 270
nas_lteSnrinformation, 274
snrlevel, 274
nas_lteWcdmaCellInfo, 279
cellReselPriority, 279
cellsLen, 279
threshXhigh, 279
threshXlow, 280
uarfcn, 280
WCDMACellInfo, 280
nas_major
NASLteNasReleaseInfoTlv, 391
nas_minBasedIMSI, 280
imsiM1112, 281
imsiMS1, 281
imsiMS2, 281
mccM, 281
nas_minor
NASLteNasReleaseInfoTlv, 391
nas_namName, 283
namName, 284
namNameLen, 284
nas_netSelectionPref, 284
mcc, 284
mnc, 284
netReg, 284
nas_networkNameSrcTlv, 286

nwNameSrc, 286
nwNameSrcLen, 286
TlvPresent, 286
nas_nmrCellInfo, 294
nmrArfcn, 295
nmrBsic, 295
nmrCellID, 295
nmrLac, 295
nmrPlmn, 295
nmrRxLev, 295
nas_nr5gBandPref, 295
bits_129_192, 296
bits_193_256, 296
bits_1_64, 296
bits_65_128, 296
nas_nwNameSrc3GppTlv, 307
NwNameSrc3Gpp, 307
TlvPresent, 307
nas_operatorNameString, 308
PLMNName, 308
nas_operatorPLMNList, 309
numInstance, 309
PLMNData, 309
nas_protocolSubtypeElement, 321
AccessMac, 322
AuthProt, 322
ControlMac, 322
EncryptProt, 322
ForwardMac, 322
IdleState, 322
KeyExchange, 322
MultDisc, 322
PhysicalLayer, 322
ReverseMac, 322
SecProt, 322
VirtStream, 322
nas_qaQmi3Gpp2TimeZone, 322
daylightSavings, 323
leapSeconds, 323
localTimeOffset, 323
nas_release
NASLteNasReleaseInfoTlv, 391
nas_roamIndList, 341
numInstances, 341
radioInterface, 341
roamIndicator, 341
nas_rsrqInformation, 343
radioIf, 343
rsrq, 343
nas_rxInfo, 345
ecio, 346
isRadioTuned, 346
phase, 346
rscp, 346
rsrp, 346
rxPower, 346
nas_rxSignalStrengthListElement, 347
radioIf, 348
rxSignalStrength, 348
nas_servSystem, 350
csAttachState, 351
numRadioInterfaces, 351
psAttachState, 351
radioInterface, 351
regState, 351
selNetwork, 351
nas_serviceProviderName, 349
displayCondition, 349
spn, 349
spnLength, 349
nas_sidNid, 351
nid, 351
sid, 351
nas_sysInfoCommon, 360
isSysForbidden, 362
isSysForbiddenValid, 362
roamStatus, 362
roamStatusValid, 362
srvCapability, 362
srvCapabilityValid, 362
srvDomain, 362
srvDomainValid, 362
nas_timeInfo, 366
day, 367
dayLtSavingAdj, 367
dayOfWeek, 367
hour, 367
minute, 367
month, 367
radioInterface, 367
second, 367
timeZone, 368
TlvPresent, 368
year, 368
nas_trueIMSI, 368
imsiT1112, 368
imsiTS1, 368
imsiTS2, 368
imsiTaddrNum, 368
mccT, 368
nas_txInfo, 369
isInTraffic, 369
txPower, 369
nas_umtsInstArr, 373
umtsEcio, 374
umtsPsc, 374
umtsRank, 374
umtsRscp, 374
umtsSet, 374
umtsSqual, 374
umtsSrxLvl, 374
umtsUarfcn, 374
nas_umtsLTENbrCell, 375
cellIsTDD, 375
earfcn, 375
pci, 376

rsrp, 376
rsrq, 376
srxlev, 376
nas_wcdmaCellInfo, 379
cpich_ecno, 379
cpich_rscp, 379
psc, 379
srxlev, 379
nas_wcdmaUARFCN, 385
status, 386
uarfcn, 386
NasGetLTECphyCaInfo, 388
PhyCaAggPcellInfo, 389
PhyCaAggScellArray, 389
PhyCaAggScellDlBw, 389
PhyCaAggScellIndType, 389
PhyCaAggScellIndex, 389
PhyCaAggScellInfo, 389
nasQmisNasPcsDigit
nas_QmisNasSlqsNasPCICellInfo, 326
nasQmisNasSlqsNasPCICellInfo
nas_QmisNasSlqsNasPCIInfo, 327
nasRelInfoTlv
NASQmiCbkNasSwiOTAMessageInd, 400
nccPermitted
nas_lteGsmCellInfo, 256
Ndp_signature
unpack_wda_GetDataFormat_t, 1136
NeighborSetCnt
nas_NetworkStat1x, 292
netDescr
nas_currentPLMN, 216
netDescrLength
nas_currentPLMN, 216
netInfoLen
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
netReg
nas_netSelectionPref, 284
netScan3GPPInfo
nas_NetworkScan3GPPInfoTlv, 289
NetSelPref
nas_NetSelPrefTlv, 285
NASNetSelPreferenceTlv, 392
network3GPPScanInfo
unpack_nas_SLQSNasPerformIncNetworkScan-Callback_Ind_t, 951
network_desc
nas_NetworkScan3GPPInfo, 288
network_desc_len
nas_NetworkScan3GPPInfo, 288
network_status
nas_NetworkScan3GPPInfo, 288
NetworkID
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
networkID
nas_CDMASysInfo, 206
networkIdValid
nas_CDMASysInfo, 206
nas_GSMSysInfo, 239
nas_LTESysInfo, 278
nas_WCDMASysInfo, 385
NetworkInfo
voice_VoiceUssdNoWaitUSSINFOTlv, 1253
networkInfo
voice_USSDNotificationNetworkInfo, 1249
networkInfoLen
unpack_wds_SLQSGetCurrDataSystemStat_t, 1167
networkNameSrc
unpack_nas_SLQSNasPerformIncNetworkScan-Callback_Ind_t, 951
networkScanStatus
unpack_nas_SLQSNasPerformIncNetworkScan-Callback_Ind_t, 951
NetworkType
currNetworkInfo, 60
wds_currNetworkInfo, 1255
NewIMSRegistration
unpack_imsa_SLQSImsaRegStatusCallBack_ind-_t, 839
NewImsRegStatus
unpack_imsa_SLQSGetIMSARegStatus_t, 834
NewMMTlv
unpack_sms_SetNewSMSCallback_ind_t, 1017
newMTMessageTlv, 406
MTMessageInfo, 406
TlvPresent, 406
newPINLen
uim_unblockUIMPIN, 726
newPINVal
uim_unblockUIMPIN, 726
newPasswd
pack_voice_SLQSVoiceSetCallBarringPassword_t, 600
newPasswdAgain
pack_voice_SLQSVoiceSetCallBarringPassword_t, 600
newPin
pack_dms_UIMUnblockPIN_t, 433
newPwd
voice_newPwdData, 1239
newPwdAgain
voice_newPwdData, 1239
newValue
pack_dms_UIMChangePIN_t, 429
nextHeader
LibPackTFTIDParams, 158
nid
nas_CDMAInfo, 202
nas_sidNid, 351
unpack_nas_GetHomeNetwork_t, 881
unpack_nas_SLQSGetHomeNetwork_t, 897
nmrArfcn
nas_nmrCellInfo, 295

nmrBsic
nas_nmrCellInfo, 295
nmrCellID
nas_nmrCellInfo, 295
nmrInst
nas_GERANInfo, 232
nmrLac
nas_nmrCellInfo, 295
nmrPlmn
nas_nmrCellInfo, 295
nmrRxLev
nas_nmrCellInfo, 295
noReplyTimer
voice_callFWExtInfo, 1219
voice_callFWInfo, 1220
notifType
unpack_voice_SUPSNotificationCallback_ind_t, 1129
notification
unpack_omaDmNotificationsTlv_t, 974
notification_Type
unpack_voice_USSDNotificationCallback_ind_t, 1129
notificationType
sMSEtwsMessageInfo, 672
notused
unpack_dms_SetCrashAction_t, 771
Nr5gBandPrefbits1_64
NASNr5gBandPrefTlv, 393
Nr5gBandPrefbits_129_192
NASNr5gBandPrefTlv, 393
Nr5gBandPrefbits_193_256
NASNr5gBandPrefTlv, 393
Nr5gBandPrefbits_65_128
NASNr5gBandPrefTlv, 393
nr5gCellStatus
nas_NR5GCellStatusInfoTlv, 296
nas_NR5GCellStatusTlv, 297
null_bearer_reason
DsdNullBearerReason_t, 84
NullBearerReasonTlvPresent
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
unpack_dsd_SLQSUIInfo_Ind_t, 815
num_inst
nas_NetworkScan3GPPInfoTlv, 289
num_instances
DMScustSettingList, 79
sms_qaQmi3GPP2BroadcastCfgInfo, 665
sms_qaQmi3GPPBroadcastCfgInfo, 665
numApp
slotInf, 657
uim_physlotInfo, 707
uim_slotInfo, 722
numCrashes
crashInformation, 59
numEntries
unpack_dms_SLQSSwiGetFirmwareCurr_t, 796
numFeatures
uim_personalizationStatus, 706
numFiles
uim_registerRefresh, 718
NumFlows
unpack_qos_SLQSSetQosEventCallback_ind_t, 997
numInstance
nas_operatorPLMNList, 309
nas_PLMNNetworkName, 317
numInstances
nas_homeSIDNID, 246
nas_roamIndList, 341
nas_SccRxInfo, 349
sms_getMsgWaitingInfo, 660
unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t, 1032
voice_arrAlertingPattern, 1206
voice_arrAlertingType, 1207
voice_arrAlphaID, 1207
voice_arrCalledPartyNum, 1208
voice_arrCallEndReason, 1209
voice_arrCallInfo, 1209
voice_arrConnectPartyNum, 1210
voice_arrDiagInfo, 1210
voice_arrRedirPartyNum, 1211
voice_arrRemotePartyName, 1211
voice_arrRemotePartyNum, 1212
voice_arrSvcOption, 1212
voice_arrUUSInfo, 1213
voice_getCallFWExtInfo, 1237
voice_getCallFWInfo, 1237
wds_DomainNameList, 1263
wds_PCSCFFQDNAddressList, 1267
wds_PCSCFIPv4ServerAddressList, 1268
numLen
voice_calledPartyInfo, 1215
voice_callFWExtInfo, 1219
voice_callFWInfo, 1221
voice_callingPartyInfo, 1224
voice_peerNumberInfo, 1241
voice_redirNumInfo, 1244
voice_remotePartyNum, 1246
numOfFiles
uim_refreshevent, 717
numOfRoutes
sms_setRoutesReq, 670
numOpt
wds_DHCPLeaseOptTlv, 1257
wds_DHCPv4OptionList, 1261
wdsDhcpv4OptionList, 1276
numPI
voice_peerNumberInfo, 1241
NumPilots
nas_PilotSetData, 315
numPlan
voice_calledPartyInfo, 1215
voice_callFWExtInfo, 1220

voice_callingPartyInfo, 1224
voice_connectNumInfo, 1229
voice_peerNumberInfo, 1241
voice_redirNumInfo, 1244
numPresInd
voice_connectNumInfo, 1229
numQosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 997
numRadioInterfaces
nas_servSystem, 351
NumRxFilters
unpack_qos_QosFlowInfo_t, 992
numSI
voice_peerNumberInfo, 1241
numScellsCfg
nas_NumScellsConfig, 307
numSlot
uim_cardStatus, 695
NumSupUSBComps
unpack_dms_GetUSBComp_t, 767
NumTxFilters
unpack_qos_QosFlowInfo_t, 992
numType
voice_calledPartyInfo, 1215
voice_callFWExtInfo, 1220
voice_callingPartyInfo, 1224
voice_connectNumInfo, 1230
voice_peerNumberInfo, 1241
voice_redirNumInfo, 1244
numWcdmaArfcn
nas_WCDMAARFCNTlv, 378
number
voice_calledPartyInfo, 1215
voice_callFWExtInfo, 1219
voice_callFWInfo, 1220
voice_callingPartyInfo, 1224
voice_ECTNum, 1234
voice_peerNumberInfo, 1241
voice_redirNumInfo, 1244
NumberOfPhySlot
uim_GetSlotsStatusTlv, 704
NumberOfPhySlotInfo
uim_GetSlotsInfoTlv, 703
numberPlan
voice_callFwdTypeAndPlan, 1218
numberType
voice_callFwdTypeAndPlan, 1218
numIteArfcn
nas_LTEARFCNTlv, 249
nw_name_src
nas_NetworkNameSrcTlv, 286
nw_name_src_len
nas_NetworkNameSrcTlv, 286
NwETlv
unpack_voice_USSDNoWaitIndicationCallback_-ind_t, 1130
nwIdValid
nas_NR5GSysInfoTlv, 302
nas_NR5GSystemInfoTlv, 306
nwNameSrc
nas_networkNameSrcTlv, 286
NwNameSrc3Gpp
nas_nwNameSrc3GppTlv, 307
nwNameSrcLen
nas_networkNameSrcTlv, 286
NxtHdrProto
unpack_qos_swiQosFilter_t, 1004

OKtoRefresh
pack_uim_SLQSUIMRefreshOK_t, 578
OMADMEnable
pack_swioma_SLQSOMADMSetSettingsExt_t, 562
OMADMEnabled
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
unpack_swioma_SLQSOMADMGetSettings_t, 1063
OMADMLogEnable
pack_swioma_SLQSOMADMSetSettingsExt_t, 562
OTASPStatus
unpack_voice_OTASPStatusCallback_ind_t, 1092
oddEvenInd
voice_calledPartySubAdd, 1216
OfflineReason
unpack_dms_GetPower_t, 763
offset
uim_readTransparentInfo, 715
uim_writeTransparentInfo, 729
oldPINLen
uim_changeUIMPIN, 695
oldPINVal
uim_changeUIMPIN, 696
oldPasswd
pack_voice_SLQSVoiceSetCallBarringPassword_t, 600
oldValue
pack_dms_UIMChangePIN_t, 429
opMode
unpack_dms_SLQSDmsSwiGetPCInfo_t, 778
opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 457
operatingMode
dms_OperatingModeTlv, 70
OperatingModeTlv
unpack_dms_SetEventReport_ind_t, 772
operation
pack_pds_SetPDSDefaults_t, 519
uim_depersonalizationInformation, 697
OperationMode
unpack_dms_GetPower_t, 763
optCode
wds_DHCPOpt, 1259
wds_DHCPv4Option, 1260
wdsDhcpv4Option, 1275

optList
wds_DHCPLeaseOptTlv, 1257
optListData
wds_DHCPLeaseOptTlv, 1257
optVal
wds_DHCPv4Option, 1260
wdsDhcpv4Option, 1275
optValLen
wds_DHCPOpt, 1259
wds_DHCPv4Option, 1261
wdsDhcpv4Option, 1275
otaMsgTlv
NASQmiCbkNasSwiOTAMessageInd, 400

p1Status
unpack_dms_UIMGetPINStatus_t, 808
p1UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 808
p1VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 808
p2Status
unpack_dms_UIMGetPINStatus_t, 808
p2UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 808
p2VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 808
p3GPP2TimeInfo
unpack_nas_SLQSGetNetworkTime_t, 899
p3GPPTimeInfo
unpack_nas_SLQSGetNetworkTime_t, 899
p3GppNetworkInfoInstances
unpack_nas_PerformNetworkScan_t, 889
unpack_nas_SLQSPerformNetworkScanV2_t, 958
p3GppNetworkInstanceSize
unpack_nas_PerformNetworkScan_t, 889
unpack_nas_SLQSPerformNetworkScanV2_t, 958
p3gppRelease
pack_wds_SLQSSet3GPPConfigItem_t, 629
pAAASPI
pack_wds_SetMobileIPProfile_t, 621
PACK_WDS_IPV4
wds.h, 1767
PACK_WDS_IPV6
wds.h, 1767
pAMRStatus
pack_voice_SLQSVoiceGetConfig_t, 595
pAPN
PackSwiAVMSSettingsAPNInfo, 640
pAPNBearer
LibpackProfile3GPPV2, 141
pAPNClass
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pAPNClass3GPP2
LibpackProfile3GPP2, 131
LibPackprofile_3GPP2, 153
pAPNDisabledFlag
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pAPNEnabled3GPP2
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 153
pAPNInfo
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
pAPNName
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pAPNnameSize
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pAVRXAVCHeadroom
audio_RXAVCList, 47
pAVRXAVCSens
audio_RXAVCList, 47
pAccelAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 868
pAccelSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 859
pAccelTempAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 868
pAccelTempSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 859
pAccessClassBarringInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pAccoIc
unpack_nas_GetACCOLC_t, 875
pAccuracy
unpack_pds_GetPDSDefaults_t, 974
pAcqOrder
nas_acqOrderPref, 194
nas_AcqOrderPrefTlv, 194
pAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPrefExt_t, 930
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pActPilotPNElements
nas_NetworkStat1x, 292
pActiveProfileInfo
unpack_uim_SLQSUIMGetSIMProfile_t, 1079
pActiveTimer
pack_dms_SLQSSetPowerSaveModeConfig_t, 425
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
pAddCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 917
unpack_nas_SLQSGetSysInfoV2_t, 922

unpack_nas_SLQSSysInfoCallback_ind_t, 969
pAddGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pAddHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pAddLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pAddWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pAdditionalReadResult
unpack_uim_SLQSUIMReadRecord_t, 1082
pAddrAllocPref
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pAddress
pack_wds_SetMobileIPProfile_t, 621
pAirTimer
pack_voice_SLQSVoiceGetConfig_t, 595
pAirTimerCnt
unpack_voice_SLQSVoiceGetConfig_t, 1117
pAirTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 601
pAirTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 1121
pAlertType
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pAlertingPattern
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pAllowLinger
LibpackProfile3GPP2, 131
LibPackprofile_3GPP2, 153
pAlphaID
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
pAlphaIDInfo
unpack_voice_SLQSOriginateUSSD_t, 1094
unpack_voice_SLQSVoiceDialCall_t, 1096
unpack_voice_SLQSVoiceGetCallBarring_t, 1102
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 1103
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
unpack_voice_SLQSVoiceGetCallWaiting_t, 1108
unpack_voice_SLQSVoiceGetCLIP_t, 1109
unpack_voice_SLQSVoiceGetCLIR_t, 1111
unpack_voice_SLQSVoiceGetCNAP_t, 1112
unpack_voice_SLQSVoiceGetCOLP_t, 1114
unpack_voice_SLQSVoiceGetCOLR_t, 1115
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 1120
unpack_voice_SLQSVoiceSetSUPSService_t, 1123
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1126
pAltitudeAssumed
unpack_loc_PositionRpt_Ind_t, 866
pAltitudeWrtEllipsoid
pack_pds_SLQSPDSInjectPositionData_t, 524
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pAltitudeWrtMeanSeaLevel
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pAltitudeWrtSealevel
pack_pds_SLQSPDSInjectPositionData_t, 524
pAmrMode
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pAmrOctetAligned
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pAmrWBMode
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pAmrWBOctetAligned
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pAmrWbEnable
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pApnString
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 153
pApnStringSize
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 153
pApnname
pack_wds_SetDefaultProfile_t, 617
pAppName
loc_LocApplicationInfo, 176
pAppPriority
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pAppProvider
loc_LocApplicationInfo, 176
pAppSubType
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 961
pAppType
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pAppUserData
LibpackProfile3GPPV2, 141
pAppVersion
loc_LocApplicationInfo, 176
pApplication
pack_nas_SetCDMANetworkParameters_t, 466
pApplicationInfo
pack_loc_Start_t, 460

pArrAlertingPattern
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrAlertingType
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrAlphaID
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrCallEndReason
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrCallInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrCalledPartyNum
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrConnectPartyNum
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrDiagInfo
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrRedirPartyNum
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrRemotePartyName
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrRemotePartyNum
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrSvcOption
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pArrUUSInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
pAuth
pack_wds_SLQSStartDataSession_t, 633
pAuthPassword
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pAuthPassword_tSize
LibPackprofile_3GPP2, 154
pAuthPasswordSize
LibpackProfile3GPP2, 132
pAuthProtocol
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pAuthRetryCount
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pAuthStatus
unpack_nas_GetANAAAAuthenticationStatus_t, 875
pAuthTimeout
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pAuthenticateResult
unpack_uim_SLQSUIMAuthenticate_t, 1073
pAuthenticationPref
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pAutoAnsStatus
unpack_voice_SLQSVoiceSetConfig_t, 1121
pAutoAnswer
pack_voice_SLQSVoiceGetConfig_t, 596
pack_voice_SLQSVoiceSetConfig_t, 601
pAutoAnswerStat
unpack_voice_SLQSVoiceGetConfig_t, 1117
pAutoReboot
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
pAutoSelection
unpack_uim_SLQSUIMGetConfiguration_t, 1075
pAutoSwitchMode
pack_swidms_SLQSSwiDmsSetUimSlot_t, 554
pAutosdm
pack_swioma_SLQSOMADMSetSettings_t, 561
pBandPref
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 930
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pBdsSVInfo
pack_loc_Delete_Assist_Data_t, 446
pBinaryUpdateSessionInfo
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 1043
pBroadcast
pack_nas_SetCDMANetworkParameters_t, 466
pBroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 539
unpack_sms_SLQSGetSmsBroadcastConfig_t, 1021
pBurstDTMFLengths
pack_voice_SLQSVoiceBurstDTMF_t, 590
pCCResType
unpack_voice_SLQSVoiceGetCallBarring_t, 1102
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 1103
unpack_voice_SLQSVoiceGetCallWaiting_t, 1108
unpack_voice_SLQSVoiceGetCLIP_t, 1109
unpack_voice_SLQSVoiceGetCLIR_t, 1111
unpack_voice_SLQSVoiceGetCNAP_t, 1112
unpack_voice_SLQSVoiceGetCOLP_t, 1114
unpack_voice_SLQSVoiceGetCOLR_t, 1115

unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 1120
pCCResultType
unpack_voice_SLQSVoiceDialCall_t, 1096
unpack_voice_SLQSVoiceSetSUPSService_t, 1123
pCCSUPSType
unpack_voice_SLQSVoiceDialCall_t, 1096
unpack_voice_SLQSVoiceGetCallBarring_t, 1102
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 1103
unpack_voice_SLQSVoiceGetCallWaiting_t, 1108
unpack_voice_SLQSVoiceGetCLIP_t, 1109
unpack_voice_SLQSVoiceGetCLIR_t, 1111
unpack_voice_SLQSVoiceGetCNAP_t, 1112
unpack_voice_SLQSVoiceGetCOLP_t, 1114
unpack_voice_SLQSVoiceGetCOLR_t, 1115
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 1120
unpack_voice_SLQSVoiceSetSUPSService_t, 1123
pCCSuppsType
unpack_voice_SLQSOriginateUSSD_t, 1094
pCDMABroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 539
unpack_sms_SLQSGetSmsBroadcastConfig_t, 1021
pCDMAChannel
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
pCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 474
pCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 474
pCDMAECIOThreshList
nas_CDMAECIOThresh, 201
pCDMAFrameErrRate
unpack_nas_SLQSGetErrorRate_t, 896
pCDMAInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 474
pCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 474
pCDMARSSIThreshList
nas_CDMARSSIThresh, 203
pCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
pCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
PCICellInfoLen
nas_QmisNasSlqsNasPCIInfo, 327
pCLIPResp
unpack_voice_SLQSVoiceGetCLIP_t, 1109
pCLIPstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pCLIRCause
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pCLIRResp
unpack_voice_SLQSVoiceGetCLIR_t, 1111
pCLIRType
pack_voice_SLQSVoiceDialCall_t, 591
pCLIRstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pCNAPResp
unpack_voice_SLQSVoiceGetCNAP_t, 1112
pCNAPstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pCOLPResp
unpack_voice_SLQSVoiceGetCOLP_t, 1114
pCOLPstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pCOLRResp
unpack_voice_SLQSVoiceGetCOLR_t, 1115
pCOLRstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
PCSCFAddrPCO
unpack_wds_SLQSGetRuntimeSettings_t, 1177
PCSCFFQDNAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 1177
pCSCFPortName
pack_ims_SLQSSetRegMgrConfig_t, 443
pCSCFPortNameLen
pack_ims_SLQSSetRegMgrConfig_t, 443
PCSFlag
LibPackProfileMnc, 155
pCSGID
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
pCSPPLMNModeInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
PCTlv
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 825
pCUGIndex
unpack_voice_SUPSNotificationCallback_ind_t, 1129
pCUGInfo
pack_voice_SLQSVoiceDialCall_t, 591
PCURTlv
unpack_ims_SLQSSMSCfgCallBack_ind_t, 830
pCallActiveDuration
unpack_wds_GetSessionDurationV2_t, 1161
pCallBarPasswd

unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1126
pCallBarringPasswd
pack_voice_SLQSVoiceSetSUPSService_t, 604
pCallFWNum
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pCallFWTimerVal
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pCallForwardingNumber
pack_voice_SLQSVoiceSetSUPSService_t, 604
pCallFwdInfo
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1126
pCallFwdTypeAndPlan
pack_voice_SLQSVoiceSetSUPSService_t, 604
pCallID
pack_voice_SLQSVoiceManageCalls_t, 598
pack_voice_SLQSVoiceSendFlash_t, 599
pack_voice_SLQSVoiceStartContDTMF_t, 604
unpack_voice_SLQSVoiceBurstDTMF_t, 1095
unpack_voice_SLQSVoiceDialCall_t, 1096
unpack_voice_SLQSVoiceGetCallBarring_t, 1102
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 1103
unpack_voice_SLQSVoiceGetCallWaiting_t, 1108
unpack_voice_SLQSVoiceGetCLIP_t, 1109
unpack_voice_SLQSVoiceGetCLIR_t, 1111
unpack_voice_SLQSVoiceGetCNAP_t, 1112
unpack_voice_SLQSVoiceGetCOLP_t, 1114
unpack_voice_SLQSVoiceGetCOLR_t, 1115
unpack_voice_SLQSVoiceSendFlash_t, 1118
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 1120
unpack_voice_SLQSVoiceSetSUPSService_t, 1123
unpack_voice_SLQSVoiceStartContDTMF_t, 1123
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
voice_burstDTMFInfo, 1214
pCallId
pack_voice_SLQSVoiceAnswerCall_t, 589
pack_voice_SLQSVoiceEndCall_t, 592
unpack_voice_SLQSOriginateUSSD_t, 1094
unpack_voice_SLQSVoiceAnswerCall_t, 1095
unpack_voice_SLQSVoiceEndCall_t, 1097
pCallInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pCallModeStatusInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pCallPartySubAdd
pack_voice_SLQSVoiceDialCall_t, 591
pCallType
pack_voice_SLQSVoiceDialCall_t, 591
pCallWaitInd
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pCalledPartyInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pCallerIDInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pCallerNameInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pCallingPartyInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pCampedCiotLteOpMode
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
pCardResult
unpack_uim_ReadTransparent_t, 1070
unpack_uim_SLQSUIMAuthenticate_t, 1073
unpack_uim_SLQSUIMGetFileAttributes_t, 1076
unpack_uim_SLQSUIMReadRecord_t, 1082
unpack_uim_SLQSUIMWriteRecord_t, 1084
unpack_uim_SLQSUIMWriteTransparent_t, 1085
unpack_uim_UnblockPinV2_t, 1086
pCardStatus
unpack_uim_GetCardStatus_t, 1068
unpack_uim_GetCardStatusV2_t, 1069
unpack_uim_SLQSUIMSetStatusChangeCall-Back_ind_t, 1083
pCcResultType
unpack_voice_SLQSOriginateUSSD_t, 1094
pCellDataMask
pack_pds_ResetPDSData_t, 517
pCellDb
pack_loc_Delete_Assist_Data_t, 446
pChangeDuration
pack_nas_SLQSInitiateNetworkRegistration_t, 471
pChgDuration
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
pCiotAcqOrder
nas_ciotAcqOrderPref, 209
nas_CiotAcqOrderPrefTlv, 210
pCiotAcqOrderPref
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPrefExt_t, 930
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pCiotLteOpMode
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
pCiotLteOpModePref
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPrefExt_t, 930
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pCiotOpModePref
pack_nas_PerformNetworkScanPCI_t, 464
pClatFlag
LibpackProfile3GPPV2, 141
pClientPerformOperationFlag
pack_swiavms_SLQSAVMSSendSelection_t, 547
pClkInfo
pack_loc_Delete_Assist_Data_t, 446
pCodecSTGain

pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pColorCode
unpack_nas_SLQSNasGetHDRColorCode_t, 944
pConfig
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 1043
pConfigAltitudeAssumed
pack_loc_Start_t, 460
pConfigurationMask
pack_uim_SLQSUIMGetConfiguration_t, 572
pConnectNumInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pConnectionRequest
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
pConnectionRetryTimers
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
pCreateProfileOut
unpack_wds_SLQSCreateProfile_t, 1163
pCsgId
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 949
pCurAMRConfig
unpack_voice_SLQSVoiceGetConfig_t, 1117
pCurPrefVoiceSO
unpack_voice_SLQSVoiceGetConfig_t, 1117
pCurProfile
pack_wds_SLQSCreateProfile_t, 622
pCurVoiceDomainPref
unpack_voice_SLQSVoiceGetConfig_t, 1117
pCurVoicePrivacyPref
unpack_voice_SLQSVoiceGetConfig_t, 1117
pCurrChannelRateInd
pack_wds_SLQSWdsSetEventReport_t, 635
pCurrDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 635
pCurrImgInfo
unpack_dms_SLQSSwiGetFirmwareCurr_t, 796
pCurrPrefDataSysInd
pack_wds_SLQSWdsSetEventReport_t, 635
pCurrTTYMode
unpack_voice_SLQSVoiceGetConfig_t, 1117
pCurrentPLMNNameInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pCurrentPersonality
nas_HDRPersonality_Ind_Data, 241
unpack_nas_SLQSSwiGetHDRPersonality_t, 960
pCurrentPrsnlty
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 961
pCustSettingInfo
DMSgetCustomFeatureV2, 80
pCustSettingList
DMSgetCustomFeatureV2, 80
pCustomSCP
pack_nas_SetCDMANetworkParameters_t, 466
pCwtMute
pack_swiaudio_SLQSSetM2MAudioProfile_t, 544
pack_swiaudio_SLQSSetM2MAVMute_t, 546
pCycleLen
pack_nas_SLQSNASSeteDRXParams_t, 490
unpack_nas_SLQSNASGeteDRXParams_t, 941
unpack_nas_SLQSNASGeteDRXParamsExt_t, 943
pDDTMInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 487
PDOP
loc_precisionDilution, 177
PDPType
unpack_wds_SLQSGetRuntimeSettings_t, 1177
pDRCParams
unpack_nas_SLQSSwiGetHRPDStats_t, 962
pDTMFTXGain
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pData
pack_cat_CATSendEnvelopeCommand_t, 413
pack_cat_CATSendTerminalResponse_t, 414
pDataBearer
unpack_wds_GetDataBearerTechnology_t, 1145
pDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 635
pDataCallStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 635
pDataLayer
QmuxTransport, 649
pDataMode
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pDataRate
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pDataSessionStatus
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
pDataSrc
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pDataStatusDetail
unpack_nas_SLQSSwiNetworkDebug_t, 965
pDataSubscriptionPriorityInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pDataSystemStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 635
pDayltSavAdj
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 950
pDefaultPDNEnabled

pack_wds_SLQSSet3GPPConfigItem_t, 629
pDeferTime
pack_swiavms_SLQSAVMSSendSelection_t, 547
pack_swioma_SLQSOMADMSendSelection_t, 559
pDestSMSContent
pack_dms_SLQSSwiSetDyingGaspCfg_t, 426
packgetDyingGaspCfg, 637
pDestSMSNum
pack_dms_SLQSSwiSetDyingGaspCfg_t, 426
packgetDyingGaspCfg, 637
pDetachAction
pack_nas_SLQSSwiPSDetach_t, 516
pDevCurSubsCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevExplicitCfgIndex
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevMaxActDataSubsCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevMaxCfgListCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevMaxSubsCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevMultiSimCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevMultiSimVoiceDataCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevSrvCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevSubsFeatureModeCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevSubsVoiceDataCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevVoiceCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDevVoiceDataCaps
unpack_dms_GetDeviceCapabilitiesV2_t, 748
pDeviceConfigDetail
unpack_nas_SLQSSwiNetworkDebug_t, 965
pDiagInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pDigitBuff
voice_burstDTMFInfo, 1214
pDirNum
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
pDispInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pDl_data_aggregation_max_datagrams
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1140
pDl_data_aggregation_max_size
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1140
pDl_data_aggregation_protocol
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1140
pDl_minimum_padding
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1140
pDnsWithDHCPFlag
LibpackProfile3GPPV2, 141
pDoneCallback
MbimTransaction, 191
pDoneCallbackContext
MbimTransaction, 191
pDormancyStatusInd
pack_wds_SLQSWdsSetEventReport_t, 635
pDsdApnAvailableSystemInfo
unpack_dsd_SLQSGetDsdUIInfo_t, 811
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
unpack_dsd_SLQSUIInfo_Ind_t, 815
pDsdApnCurRecGlobalPrefSystem
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
pDsdAvailableSystem
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
pDsdCurRecGlobalPrefSystem
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
pDsdGlobalUIInformation
unpack_dsd_SLQSGetDsdUIInfo_t, 811
unpack_dsd_SLQSUIInfo_Ind_t, 815
pDsdNullBearerReason
unpack_dsd_SLQSGetDsdUIInfo_t, 811
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
unpack_dsd_SLQSUIInfo_Ind_t, 815
pDualStandByPrefInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pDuration
unpack_pds_GetXTRAValidity_t, 979
pDurationDueToOOS
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
pDurationThreshold
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
pE911StateReadyStatusInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pECIOThresList
nas_ECIOThresh, 224
pECIOThresh
pack_nas_SLQSConfigSigInfo_t, 469
pECMode
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pECTNum

unpack_voice_SUPSNotificationCallback_ind_t, 1129
pEMBMSStatusInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
PER
nas_NetworkStatEVDO, 294
pEVDOPageMonPerChangeInd
pack_wds_SLQSWdsSetEventReport_t, 635
pEarMute
pack_swiaudio_SLQSSetM2MAudioProfile_t, 544
pEarlyWakeupTime
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
pEdrxChangeInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pEdrxCiotLteMode
pack_nas_SLQSNASSeteDRXParams_t, 490
pEdrxEnable
pack_nas_SLQSNASSeteDRXParams_t, 490
unpack_nas_SLQSNASGeteDRXParams_t, 941
unpack_nas_SLQSNASGeteDRXParamsExt_t, 943
pEdrxRAT
pack_nas_SLQSNASGeteDRXParamsExt_t, 477
unpack_nas_SLQSNASGeteDRXParamsExt_t, 943
pEdrxRatType
pack_nas_SLQSNASSeteDRXParams_t, 490
pEhrpdMTUSize
unpack_swidms_SLQSSwiDmsGetMTU_t, 1047
pEidInfo
unpack_uim_SLQSUIMGetEID_t, 1075
pEmerMode
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 930
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pEmergencyCategory
pack_voice_SLQSVoiceDialCall_t, 591
pEmergencyModeStatusInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pEnableNotification
pack_wds_DHCPv4ClientLeaseChange_t, 611
pEnabled
pack_wds_SetMobileIPProfile_t, 621
pEnabledStatus
unpack_pds_GetPDSState_t, 975
pEncryptData
pack_uim_ReadTransparent_t, 566
pEncryptedData
unpack_uim_ReadTransparent_t, 1070
pEncryptedPIN1
pack_uim_VerifyPin_t, 586
unpack_uim_ChangePin_t, 1068
unpack_uim_SetPinProtection_t, 1071
unpack_uim_UnblockPin_t, 1086
unpack_uim_UnblockPinV2_t, 1087
unpack_uim_VerifyPin_t, 1087
pEp_type
pack_wda_GetQmapSettings_t, 606
pack_wda_SetDataFormat_t, 609
pack_wda_SetQmapSettings_t, 610
pErrorRateInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pExtDispInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pExtDispRecInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pExtErrCode
UnPackGetProfileSettingOut, 1192
UnPackGetProfileSettingOutV2, 1193
pExtErrorCode
unpack_wds_SLQSModifyProfile_t, 1178
pFDNStatus
pack_uim_SLQSUIMSetServiceStatus_t, 580
unpack_uim_SLQSUIMGetServiceStatus_t, 1079
pFailCause
unpack_voice_SLQSVoiceGetCallBarring_t, 1102
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 1103
unpack_voice_SLQSVoiceGetCallWaiting_t, 1108
unpack_voice_SLQSVoiceGetCLIP_t, 1110
unpack_voice_SLQSVoiceGetCLIR_t, 1111
unpack_voice_SLQSVoiceGetCNAP_t, 1112
unpack_voice_SLQSVoiceGetCOLP_t, 1114
unpack_voice_SLQSVoiceGetCOLR_t, 1115
unpack_voice_SLQSVoiceManageCalls_t, 1118
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 1120
unpack_voice_SLQSVoiceSetSUPSService_t, 1123
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pFailureReason
unpack_wds_SLQSStartDataSession_t, 1188
pFileAttributes
unpack_uim_SLQSUIMGetFileAttributes_t, 1076
pFixId
unpack_loc_PositionRpt_Ind_t, 866
pFlag
audio_RXPCMIIRFltr, 49
audio_TXPCMIIRFltr, 51
pFlashPayLd
pack_voice_SLQSVoiceSendFlash_t, 599
pFlashType
pack_voice_SLQSVoiceSendFlash_t, 599
pFlow_control
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1140
pFollowOnDC
sms_sendAsyncsmsParams, 668
pForbiddenNetworks3GPP

unpack_nas_SLQSNASGetForbiddenNetworks_t, 943
pForceOnDC
sms_sendAsyncsmsParams, 668
pForceRev0
pack_nas_SetCDMANetworkParameters_t, 467
pFwAutoCheck
pack_swioma_SLQSOMADMSetSettings_t, 561
pFwAutoSDM
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
pGERANInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pGPRSMinimumQoS
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pGPRSRequestedQos
LibpackProfile3GPP, 125
LibpackProfile3GPPV2, 141
LibPackprofile_3GPP, 148
pGPSDataMask
pack_pds_ResetPDSData_t, 518
pGPSWeek
unpack_pds_GetXTRAValidity_t, 979
pGPSWeekOffset
unpack_pds_GetXTRAValidity_t, 979
pGSMBER
unpack_nas_SLQSGetErrorRate_t, 896
pGSMCallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pGSMCipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pGSMRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 474
pGSMRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 474
pGSMRSSIThreshList
nas_GSMRSSIThresh, 235
pGSMSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
pGSMSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pGWAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 930
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pGcellInfoInd
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pGenerator
pack_swiaudio_SLQSGetM2MAudioProfile_t, 541
pack_swiaudio_SLQSSetM2MAudioProfile_t, 544
pGetCallFWExtInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 1104
pGetCallFWInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 1104
pGetCustomInput
DMSgetCustomFeatureV2, 80
pGetDyingGaspCfg
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 795
pGetDyingGaspStatistics
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 795
pGetIndicationRegInfo
unpack_sms_SLQSGetIndicationRegister_t, 1019
pGetMsgWaitingInfoResp
unpack_sms_SLQSGetMessageWaiting_t, 1019
pGetSlotsInfoTlv
unpack_uim_SLQSUIMGetSlotsStatusV2_t, 1081
pGetSlotsStatusTlv
unpack_uim_SLQSUIMGetSlotsStatusV2_t, 1081
pGetTransLayerInfo
unpack_sms_SLQSGetTransLayerInfo_t, 1022
pGetTransNWRegInfo
unpack_sms_SLQSGetTransNWRegInfo_t, 1023
pGnssData
pack_loc_Delete_Assist_Data_t, 446
pGpsTime
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pGyroAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 868
pGyroSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 859
pGyroTempAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 868
pGyroTempSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 859
pHA2002bis
pack_wds_SetMobileIPParameters_t, 620
pHAAuthenticator
pack_wds_SetMobileIPParameters_t, 620
pHASPI
pack_wds_SetMobileIPProfile_t, 621
pHDRECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 474
pHDRECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 474
pHDRECIOThreshList

nas_HDRECIOThresh, 240
pHDRIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pHDRIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pHDRIOThreshList
nas_HDRIOThresh, 240
pHDRNewUATIAssInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 487
pHDRPackErrRate
unpack_nas_SLQSGetErrorRate_t, 896
pHDRPersInd
unpack_nas_SLQSSwiHDRPersonalityCallback_-Ind_t, 964
pHDRRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pHDRRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pHDRRSSIThreshList
nas_HDRRSSIThresh, 242
pHDRSINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pHDRSINRThresList
nas_HDRSINRThresh, 242
pHDRSINRThresh
pack_nas_SLQSConfigSigInfo_t, 469
pack_nas_SLQSNasConfigSigInfo2_t, 475
pHDRSINRThreshList
nas_HDRSINRThreshold, 243
pHDRSessionCloseInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pHDRSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
pHDRSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pHTTPStatus
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
pHWWatchdog
unpack_swidms_SLQSSwiDmsGetHWWatchdog-_t, 1046
pHaltSubscription
unpack_uim_SLQSUIMGetConfiguration_t, 1075
pHeading
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pHeadingUnc
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pHiddenKeyStatus
unpack_uim_SLQSUIMGetServiceStatus_t, 1079
pHomeNwMNC3Gpp
unpack_nas_SLQSGetHomeNetwork_t, 898
pHomeSIDNID
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
pHorCirConf
unpack_loc_BestAvailPos_Ind_t, 846
pHorConfidence
unpack_loc_PositionRpt_Ind_t, 866
pHorEllpConf
unpack_loc_BestAvailPos_Ind_t, 846
pHorReliability
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pHorUncCircular
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pHorUncEllipseOrientAzimuth
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pHorUncEllipseSemiMajor
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pHorUncEllipseSemiMinor
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 866
pHorizontalAccuracyLvl
pack_loc_Start_t, 460
pHorizontalConfidence
pack_pds_SLQSPDSInjectPositionData_t, 524
pHorizontalUncCircular
pack_pds_SLQSPDSInjectPositionData_t, 524
pHotSwapStatus
unpack_uim_GetCardStatus_t, 1068
unpack_uim_GetCardStatusV2_t, 1069
pHrpdMTUSize
unpack_swidms_SLQSSwiDmsGetMTU_t, 1047
pHwConfig
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 631
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 1187
PI
voice_calledPartyInfo, 1215
voice_callerIDInfo, 1217
voice_callFWExtInfo, 1220
voice_callingPartyInfo, 1224
voice_redirNumInfo, 1244
PIFACEId
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 543
pIMSDomain
pack_ims_SLQSSetIMSUserConfig_t, 439
unpack_ims_SLQSGetIMSUserConfig_t, 819
pIMSDomainLen
pack_ims_SLQSSetIMSUserConfig_t, 439
unpack_ims_SLQSGetIMSUserConfig_t, 819
pIMSPrefStatusInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488

pIMSTestMode
pack_ims_SLQSSetRegMgrConfig_t, 443
unpack_ims_SLQSGetRegMgrConfig_t, 823
pIOThresList
nas_IOThresh, 249
pIOThresh
pack_nas_SLQSConfigSigInfo_t, 469
pIPV6DelegFlag
LibpackProfile3GPPV2, 142
pIPv4AddrInfo
unpack_loc_GetServer_Ind_t, 855
pIPv4AddrPref
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pIPv4Config
pack_loc_SLQSLOCSetServer_t, 459
pIPv6AddPref
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pIPv6AddrInfo
unpack_loc_GetServer_Ind_t, 855
pIPv6Config
pack_loc_SLQSLOCSetServer_t, 459
pIWLANtoLTEHandoverFlag
LibpackProfile3GPPV2, 142
pIface_id
pack_wda_GetQmapSettings_t, 606
pack_wda_SetDataFormat_t, 609
pack_wda_SetQmapSettings_t, 610
pIgnoreHotSwapSwitch
pack_uim_SLQSUIMPowerUp_t, 576
pImCnFlag
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pImageList
pack_fms_SetImagesPreference_t, 437
unpack_fms_GetImagesPreference_t, 816
pImsVoiceSupportLte
unpack_nas_SLQSGetSysInfoV2_t, 923
pIncResultPeriodicty
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pIncomingMessage
MbimTransport, 192
pIncrementalResults
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pIndex
unpack_uim_SLQSUIMGetServiceStatus_t, 1079
pIndicationCallback
MbimIndicator, 189
QmiService, 644
pIndicationCallbackContext
MbimIndicator, 189
QmiService, 644
pIndicationToken
pack_uim_ChangePin_t, 565
pack_uim_ReadTransparent_t, 566
pack_uim_SetPinProtection_t, 567
pack_uim_SLQSUIMAuthenticate_t, 569
pack_uim_SLQSUIMGetFileAttributes_t, 573
pack_uim_SLQSUIMReadRecord_t, 577
pack_uim_SLQSUIMWriteRecord_t, 583
pack_uim_SLQSUIMWriteTransparent_t, 584
pack_uim_UnblockPin_t, 585
pack_uim_VerifyPin_t, 586
unpack_uim_ChangePin_t, 1068
unpack_uim_ReadTransparent_t, 1070
unpack_uim_SetPinProtection_t, 1071
unpack_uim_SLQSUIMAuthenticate_t, 1073
unpack_uim_SLQSUIMGetFileAttributes_t, 1076
unpack_uim_SLQSUIMReadRecord_t, 1082
unpack_uim_SLQSUIMWriteRecord_t, 1084
unpack_uim_SLQSUIMWriteTransparent_t, 1085
unpack_uim_UnblockPin_t, 1086
unpack_uim_UnblockPinV2_t, 1087
unpack_uim_VerifyPin_t, 1087
pInfo
pack_voice_AnswerUSSD_t, 586
pack_voice_OriginateUSSD_t, 587
pInstanceSize
nas_RfBandInfoList, 333
pInstancesSize
nas_RfBandInfoExtFormat, 331
nas_RfBandwidthInfo, 335
nas_RfDedicatedBandInfo, 338
pInterfaceCfg
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 1050
pIntermediateReportState
pack_loc_Start_t, 460
pInterval
unpack_pds_GetPDSDefaults_t, 974
unpack_pds_GetXTRAAutomaticDownload_t, 977
pIpcpAckTimeout
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pIpcpCreqRetryCount
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pIsPcscfAddressNedded
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pKeyReferenceID
pack_uim_ChangePin_t, 565
pack_uim_SetPinProtection_t, 567
pack_uim_UnblockPin_t, 585
pack_uim_VerifyPin_t, 586
PLMNData
nas_operatorPLMNList, 309
PLMNName
nas_operatorNameString, 308
PLMNNetName

nas_PLMNNetworkName, 317
PLMNRecID
nas_OperatorPLMNData, 309
pLTEArfcn
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pLTEAttachProfile
pack_wds_SLQSSet3GPPConfigItem_t, 629
pLTEAttachProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 629
pLTEBandPref
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 930
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pLTEBandPrefExt
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pLTECphyCa
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pLTEInfo
unpack_nas_SLQSNasSwiModemStatus_t, 955
pLTEInfoInterfreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pLTEInfoIntrafreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pLTEInfoNeighboringGSM
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pLTEInfoNeighboringWCDMA
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pLTEOperationMode
nas_LTEOperationMode, 267
unpack_nas_SLQSNasGetRFInfo_t, 945
pLTERACHFailInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pLTERRCTxInfoInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pLTERSRPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTERSRPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTERSRPThreshList
nas_LTERSRPThresh, 271
pLTERSRQDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTERSRQThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTERSRQThreshList
nas_LTERSRQThresh, 271
pLTERSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTERSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTERSSIThreshList
nas_LTERSSIThresh, 272
pLTESIB16NetworkTimeInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pLTESNRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTESNRThresList
nas_LTESNRThresh, 275
pLTESNRThresh
pack_nas_SLQSConfigSigInfo_t, 469
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTESNRThreshList
nas_LTESNRThreshold, 275
pLTESigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
pLTESigRptCfg
pack_nas_SLQSConfigSigInfo_t, 469
pLTESigRptConfig
pack_nas_SLQSNasConfigSigInfo2_t, 475
pLTESrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 923
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 923
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pLTEVoiceSupportSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 923
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pLTEtoIWLANHandoverFlag
LibpackProfile3GPPV2, 142
pLastCallActiveDuration
unpack_wds_GetSessionDurationV2_t, 1161
pLastCallDuration
unpack_wds_GetSessionDurationV2_t, 1161
pLastRecord
pack_uim_SLQSUIMReadRecord_t, 577
pLatitude
pack_pds_SLQSPDSInjectPositionData_t, 524
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pLcpAckTimeout
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pLcpCreqRetryCount
LibpackProfile3GPP2, 132

LibPackprofile_3GPP2, 154
pLeapSeconds
unpack_loc_PositionRpt_Ind_t, 867
pLimit_so_mask_change_ind
pack_dsd_system_status_changes_t, 435
pLineCtrlInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pLinktimer
pack_sms_SendSMS_t, 532
sms_sendAsyncsmsParams, 668
pLongitude
pack_pds_SLQSPDSInjectPositionData_t, 524
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pLteCiotOpModeTlv
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pLteEARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 492
unpack_nas_SLQSNASSwiGetChannelLock_t, 955
pLteEarfcnInfo
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pLteEmbmsCoverage
unpack_nas_SLQSGetSysInfoV2_t, 923
pLteEmbmsTraceId
unpack_nas_SLQSGetSysInfoV2_t, 923
pLteM1BandPref
pack_nas_PerformNetworkScanPCI_t, 464
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pLteNB1BandPref
pack_nas_PerformNetworkScanPCI_t, 465
pLteNb1BandPref
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pLteOpMode
pack_nas_SLQSNASGeteDRXParamsExt_t, 477
unpack_nas_SLQSNASGeteDRXParamsExt_t, 943
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 949
pLteOpModeTlv
unpack_nas_PerformNetworkScan_t, 890
unpack_nas_SLQSPerformNetworkScanV2_t, 958
pLtePCI
pack_nas_SLQSNASSwiSetChannelLock_t, 492
unpack_nas_SLQSNASSwiGetChannelLock_t, 955
pLteRegDomain
unpack_nas_SLQSGetSysInfoV2_t, 923
pLteRoamPDPType
LibpackProfile3GPPV2, 142
pLteVoiceDomain
unpack_nas_SLQSGetSysInfoV2_t, 923
pMCC
unpack_nas_GetHomeNetwork3GPP2_t, 880
pMICGainSelect
unpack_audio_SLQSGetAudioPathConfig_t, 731
pMIPStatusInd
pack_wds_SLQSWdsSetEventReport_t, 635
pMNAAA
pack_wds_SetMobileIPProfile_t, 621
pMNC
unpack_nas_GetHomeNetwork3GPP2_t, 880
pMNCIncPCSDigStat
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
pMNHA
pack_wds_SetMobileIPProfile_t, 621
pMNRInfo
pack_nas_SLQSInitiateNetworkRegistration_t, 471
pMTUSize3gpp
unpack_swidms_SLQSSwiDmsGetMTU_t, 1047
pMagneticDeviation
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pManagedRoamingInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pManualNWScanFailureInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pMaxPDN
LibpackProfile3GPPV2, 142
pMaxPDNTimer
LibpackProfile3GPPV2, 142
pMaxSearchTime
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pMaxStorageSizeReq
pack_sms_SLQSSmsGetMaxStorageSize_t, 540
pMaxStorageSizeResp
unpack_sms_SLQSSmsGetMaxStorageSize_t, 1027
pMcc
LibpackProfile3GPPV2, 142
pMdn
pack_nas_SLQSSet3GPP2SubscriptionInfo_t, 498
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
pMessage
pack_sms_SaveSMS_t, 532
pack_sms_SendSMS_t, 533
sms_sendAsyncsmsParams, 668
pMessageIndex
pack_sms_SLQSDeleteSMS_t, 534
unpack_sms_SaveSMS_t, 1015
pMessageMode
pack_sms_SLQSDeleteSMS_t, 534
pack_sms_SLQSGetSMS_t, 535
pack_sms_SLQSGetSMSList_t, 536
pack_sms_SLQSModifySMSStatus_t, 537

sms_maxStorageSizeReq, 662
pMessageProtocol
unpack_sms_SLQSSmsGetMessageProtocol_t, 1027
pMessageTag
pack_sms_SLQSDeleteSMS_t, 534
pMicMute
pack_swiaudio_SLQSSetM2MAudioProfile_t, 544
pMinBasedIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
pMinIntervalTime
pack_loc_Start_t, 460
pMinSessionExpiryTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pMnc
LibpackProfile3GPPV2, 142
pMncPcsDigitStatus
pack_nas_SLQSInitiateNetworkRegistration_t, 471
pMncPcsStatus
pack_nas_SLQSGetPLMNName_t, 470
pMode
pack_wds_SetMobileIPParameters_t, 620
unpack_loc_GetOpMode_Ind_t, 854
pModePref
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pMsisdnFlag
LibpackProfile3GPPV2, 142
pMultiplexedSimMode
pack_swidms_SLQSSwiDmsSetUimSlot_t, 554
pMuxID
pack_qos_BindDataPort_t, 529
pNAI
pack_wds_SetMobileIPProfile_t, 621
pNAMNameInfo
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
pNID
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNITZInformation
unpack_nas_SLQSGetOperatorNameData_t, 900
pNR5GCellStatus
unpack_nas_SLQSSysInfoCallback_ind_t, 969
pNR5GCellStatusInfo
unpack_nas_SLQSGetSysInfoV2_t, 923
pNR5GSerStatTlv
unpack_nas_SLQSSysInfoCallback_ind_t, 970
pNR5GSrvStatusinfo
unpack_nas_SLQSGetSysInfoV2_t, 923
pNR5GSysInfo
unpack_nas_SLQSGetSysInfoV2_t, 923
pNR5GSystemInfoTlv
unpack_nas_SLQSSysInfoCallback_ind_t, 970
pNSEnable
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pNSSAudioCtrl
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pNSSRelease
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
PNTlv
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 825
pNamID
pack_voice_SLQSVoiceGetConfig_t, 596
pName
pack_wds_SetDefaultProfile_t, 617
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNdp_signature
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1140
pNegoDnsSrvrPref
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pNeighborSetPilotPN
nas_NetworkStat1x, 292
pNetSelPref
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pNetworkMode
pack_pds_SLQSGetAGPSConfig_t, 522
pack_pds_SLQSSetAGPSConfig_t, 525
pNetworkNameSrcTlv
unpack_nas_SLQSPerformNetworkScanV2_t, 958
pNetworkRejectInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pNetworkStat1x
unpack_nas_SLQSSwiNetworkDebug_t, 965
pNetworkStatEVDO
unpack_nas_SLQSSwiNetworkDebug_t, 965
pNetworkTimeInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pNetworkType
pack_nas_PerformNetworkScanPCI_t, 465
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pNewPwdData
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pNext
MbimIndicator, 189
MbimTransaction, 191
QmiService, 644
pNickName
pack_uim_SLQSUIMUpdateProfile_t, 582

pNotifStore
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
pNotification
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 1044
pNotificationStore
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
pNr5gBandPref
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pNumberOfPhySlot
unpack_uim_SLQSUIMGetSlotsStatus_t, 1080
pNw2DescDisp
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNw2DescEnc
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNw2DescLen
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNw2MCC
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNw2MNC
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNw2Name
unpack_nas_GetHomeNetwork3GPP2_t, 880
pNwNameSrc3Gpp
unpack_nas_SLQSGetHomeNetwork_t, 898
pOTASPStatus
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pObjectVer
unpack_nas_SLQSSwiNetworkDebug_t, 965
pOffLength
unpack_voice_DTMFEventCallback_ind_t, 1090
pOnLength
unpack_voice_DTMFEventCallback_ind_t, 1090
pOpaqueIdentifier
unpack_loc_InjectSensorDataCallback_Ind_t, 859
pOperation
unpack_pds_GetPDSDefaults_t, 974
pOperatorNameDataInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pOperatorNameString
unpack_nas_SLQSGetOperatorNameData_t, 900
pOperatorPCOID
LibpackProfile3GPPV2, 142
pOperatorPLMNList
unpack_nas_SLQSGetOperatorNameData_t, 900
pOptList
wds_DHCPv4OptionList, 1261
wdsDhcpv4OptionList, 1276
pOptVal
wds_DHCPOpt, 1259
pOverridePDPType
LibpackProfile3GPPV2, 142
pPCIInfo
unpack_nas_PerformNetworkScan_t, 890
unpack_nas_SLQSPerformNetworkScanV2_t, 959
pPCMParams
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 543
pPCOIDList
LibpackProfile3GPPV2, 142
pPCSCFPort
unpack_ims_SLQSGetRegMgrConfig_t, 823
pPCSInstance
unpack_nas_PerformNetworkScan_t, 890
unpack_nas_SLQSPerformNetworkScanV2_t, 959
pPCSInstanceSize
unpack_nas_PerformNetworkScan_t, 890
unpack_nas_SLQSPerformNetworkScanV2_t, 959
pPDNDisconnectWaitTimer
LibpackProfile3GPPV2, 142
pPDNInactivTimeout
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPDNInactivTimeout3GPP2
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pPDNThrottleTimer
LibpackProfile3GPPV2, 142
pPDNWaitTimer
LibpackProfile3GPPV2, 142
pPDPtype
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 149
pPLMNNetworkName
unpack_nas_SLQSGetOperatorNameData_t, 900
pPRLPref
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 515
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pPWD
PackSwiAVMSSettingsAPNInfo, 640
pPackageID
pack_swiavms_SLQSAVMSSendSelection_t, 547
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 1044
pPagingTimeWindow
pack_nas_SLQSNASSeteDRXParams_t, 490
unpack_nas_SLQSNASGeteDRXParams_t, 941
unpack_nas_SLQSNASGeteDRXParamsExt_t, 943
pPass

pack_wds_SLQSStartDataSession_t, 633
pPassword
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pack_wds_SetDefaultProfile_t, 617
pPasswordSize
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPcscfAddrUsingDhcp
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPcscfAddrUsingPCO
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPdnType
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pPdpAccessConFlag
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPdpContext
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPdpDataCompType
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPdpHdrCompType
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 142
LibPackprofile_3GPP, 148
pPeriodInfo
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
pPeriodicUpdateTimer
pack_dms_SLQSSetPowerSaveModeConfig_t, 425
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
pPeripheralEndPointID
pack_qos_BindDataPort_t, 529
pPeroidsInfo
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
pPersistFlag
LibpackProfile3GPPV2, 143
pPersonalityListLength
nas_HDRPersonality_Ind_Data, 241
unpack_nas_SLQSSwiGetHDRPersonality_t, 960
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 961
pPersonalizationStatus
unpack_uim_SLQSUIMGetConfiguration_t, 1075
pPhoneCtxtURI
pack_ims_SLQSSetIMSSMSConfig_t, 439
unpack_ims_SLQSGetIMSSMSConfig_t, 818
pPhoneCtxtURILen
pack_ims_SLQSSetIMSSMSConfig_t, 439
unpack_ims_SLQSGetIMSSMSConfig_t, 818
pPilotSetData
unpack_nas_SLQSSwiGetHRPDStats_t, 962
pPilotSetInfo
nas_PilotSetData, 315
pPlmnId
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 949
pPollingTimer
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
pPositionSource
pack_pds_SLQSPDSInjectPositionData_t, 524
pPppSessCloseTimer1x
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pPppSessCloseTimerDO
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pPrecisionDilution
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pPrefPLMNList
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pPrefVoiceDomain
pack_voice_SLQSVoiceSetConfig_t, 601
pPrefVoicePrivacy
pack_voice_SLQSVoiceGetConfig_t, 596
pPrefVoiceSO
pack_voice_SLQSVoiceGetConfig_t, 596
pack_voice_SLQSVoiceSetConfig_t, 601
pPrefVoiceSOStatus
unpack_voice_SLQSVoiceSetConfig_t, 1121
pPreference
unpack_pds_GetXTRANetwork_t, 978
pPrev
MbimIndicator, 189
MbimTransaction, 191
pPriCSCFPort
pack_ims_SLQSSetRegMgrConfig_t, 443
pPriCSCFPortName
unpack_ims_SLQSGetRegMgrConfig_t, 823
pPriCSCFPortNameLen
unpack_ims_SLQSGetRegMgrConfig_t, 823
pPriDNSIPv4AddPref
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149

pPriDNSIPv6addpref
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pPriV6DnsAddress
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pPrimaryHA
pack_wds_SetMobileIPProfile_t, 621
pPrimaryID
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pPrimaryV4DnsAddress
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pProfileChangeInd
pack_wds_SLQSSwiProfileChangeCallback_t, 633
pProfileClass
unpack_uim_SLQSUIMGetProfileInfo_t, 1077
pProfileConfInfo
pack_uim_SLQSUIMAddProfile_t, 568
pProfileICCID
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfileID
unpack_wds_SLQSCreateProfile_t, 1163
pProfileIcon
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfileIconType
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfileId
pack_wds_SLQSCreateProfile_t, 622
pack_wds_SLQSModifyProfile_t, 627
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 630
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 631
pProfileInfo
unpack_uim_SLQSUIMGetSIMProfile_t, 1079
pProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 629
pProfileName
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfileNickname
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfilePolicyRules
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfileSPN
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfileSettings
unpack_wds_SLQSGetProfileSettings_t, 1173
unpack_wds_SLQSGetProfileSettingsV2_t, 1174
pProfileState
unpack_uim_SLQSUIMGetProfileInfo_t, 1078
pProfileStatus
pack_uim_SLQSUIMSetSIMProfile_t, 581
pProfileType
pack_wds_SLQSCreateProfile_t, 622
pack_wds_SLQSModifyProfile_t, 627
pProfilename
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pProfilenameSize
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pProtoSubTypElmnt
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 961
pProtocol
pack_nas_SetCDMANetworkParameters_t, 467
pProtocolSubtypeElement
nas_HDRPersonality_Ind_Data, 241
unpack_nas_SLQSSwiGetHDRPersonality_t, 960
pPsmEnableState
pack_dms_SLQSSetPowerSaveModeConfig_t, 425
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
pQcqmitransport
QmiService, 644
pQlink_prot
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pQmap_dfc
pack_wda_SetQmapSettings_t, 610
unpack_wda_GetQmapSettings_t, 1137
unpack_wda_SetQmapSettings_t, 1141
pQos_format
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pQos_header_format
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pQosClassID
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pRAT
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 516
pRATINstance
unpack_nas_PerformNetworkScan_t, 890
unpack_nas_SLQSPerformNetworkScanV2_t, 959
pRATInstanceSize
unpack_nas_PerformNetworkScan_t, 890
unpack_nas_SLQSPerformNetworkScanV2_t, 959
pRATType
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pRFBandInfoInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pRFBandInfoParam
nas_RfBandInfoList, 333
PRIString
unpack_dms_GetFirmwareRevision_t, 754
unpack_dms_GetFirmwareRevisions_t, 756
PRLEnable

pack_swioma_SLQSOMADMSetSettingsExt_t, 562
PRLInd
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
PRLPTlv
NASQmiCbkNasSystemSelPrefInd, 402
PRLPref
nas_PRLPrefTlv, 321
NASPRLPreferenceTlv, 399
pRSRPThresList
nas_RSRPThresh, 343
pRSRPThresh
pack_nas_SLQSConfigSigInfo_t, 469
pRSRQThresList
nas_RSRQThresh, 344
pRSRQThresh
pack_nas_SLQSConfigSigInfo_t, 469
pRSSIThresList
nas_RSSIThresh, 345
pRSSIThresh
pack_nas_SLQSConfigSigInfo_t, 469
pRTPRTCPInactTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pRTREConfigurationInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pRXAGCList
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pRXAIG
audio_RXAGCList, 46
pRXAVCAGCSwitch
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pRXAVCList
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pRXChain0Info
unpack_nas_SLQSNasGetTxRxInfo_t, 946
pRXChain1Info
unpack_nas_SLQSNasGetTxRxInfo_t, 946
pRXComprSlope
audio_RXAGCList, 47
pRXComprThres
audio_RXAGCList, 47
pRXDroppedCount
unpack_wds_GetPacketStatistics_t, 1157
pRXExpSlope
audio_RXAGCList, 47
pRXExpThres
audio_RXAGCList, 47
pRXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 1157
pRXOkBytesCount
unpack_wds_GetPacketStatistics_t, 1157
pRXPCMIIRFltr
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pRXPacketErrors
unpack_wds_GetPacketStatistics_t, 1157
pRXPacketOverflows
unpack_wds_GetPacketStatistics_t, 1157
pRXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 1157
pRXStaticGain
audio_RXAGCList, 47
pRXTotalBytes
unpack_wds_GetByteTotals_t, 1143
pRadioInterface
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 950
pRandomizationWindow
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
pRankIndicatorInd
pack_nas_SLQSNasSwiIndicationRegister_t, 491
pRatDisabledMask
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pReRegPeriod
pack_wds_SetMobileIPParameters_t, 620
pReRegTraffic
pack_wds_SetMobileIPParameters_t, 620
pReadResult
unpack_uim_ReadTransparent_t, 1070
unpack_uim_SLQSUIMReadRecord_t, 1082
pReason
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pReasonMask
unpack_dms_GetOfflineReason_t, 762
pRecurrenceType
pack_loc_Start_t, 460
pRedirNumInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pRefreshEvent
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 1083
pRegCallStatInfoEvt
sms_getIndicationReg, 660
sms_setIndicationReg, 669
pRegDTMFEvents
pack_voice_SLQSVoiceIndicationRegister_t, 597
pRegInd
sms_getTransLayerInfo, 661
pRegMgrConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 438
pRegStatus
sms_getTransNWRegInfo, 661
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
pRegTransLayerInfoEvt
sms_getIndicationReg, 660

sms_setIndicationReg, 669
pRegTransNWRegInfoEvt
sms_getIndicationReg, 660
sms_setIndicationReg, 669
pRegVoicePrivacyEvents
pack_voice_SLQSVoiceIndicationRegister_t, 597
pRejectReason
pack_swiavms_SLQSAVMSSendSelection_t, 547
pack_swioma_SLQSOMADMSendSelection_t, 559
pRemainingRetries
unpack_uim_ChangePin_t, 1068
unpack_uim_SetPinProtection_t, 1071
unpack_uim_SLQSUIMDepersonalization_t, 1073
unpack_uim_UnblockPin_t, 1086
unpack_uim_UnblockPinV2_t, 1087
unpack_uim_VerifyPin_t, 1087
pRemotePartyName
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pRemotePartyNum
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pReport_data_system_status_changes
pack_dsd_system_status_changes_t, 435
pReport_null_bearer_reason
pack_dsd_system_status_changes_t, 435
pReport_pref_tech_change_only
pack_dsd_system_status_changes_t, 435
pReport_ui_changes
pack_dsd_register_ui_info_changes_t, 434
pReportChannelRate
pack_wds_SLQSGetDUNCallInfo_t, 625
pReportConnStatus
pack_wds_SLQSGetDUNCallInfo_t, 625
pReportDataBearerTech
pack_wds_SLQSGetDUNCallInfo_t, 625
pReportDormStatus
pack_wds_SLQSGetDUNCallInfo_t, 625
pReqSettings
pack_wds_SLQSGetRuntimeSettings_t, 626
pRequestOptionList
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 631
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 1187
pRequestedTag
pack_sms_SLQSGetSMSList_t, 536
pResetOption
pack_uim_SLQSUIMeUICCMemReset_t, 571
pResponseCallback
QmiTransaction, 646
pResponseCallbackContext
QmiTransaction, 646
pRetryInterval
pack_wds_SetMobileIPParameters_t, 620
pRetryLimit
pack_wds_SetMobileIPParameters_t, 620
pRetryMessage
sms_sendAsyncsmsParams, 668
pRetryMessageId
sms_sendAsyncsmsParams, 668
pRevInUse
nas_CDMASysInfo, 207
pRevInUseValid
nas_CDMASysInfo, 207
pRevTunneling
pack_wds_SetMobileIPProfile_t, 621
pRfBandInfoExtFormat
unpack_nas_SLQSNasGetRFInfo_t, 945
pRfBandInfoExtFormatParam
nas_RfBandInfoExtFormat, 331
pRfBandwidthInfo
unpack_nas_SLQSNasGetRFInfo_t, 945
pRfBandwidthInfoParam
nas_RfBandwidthInfo, 335
pRfDedicatedBandInfo
unpack_nas_SLQSNasGetRFInfo_t, 945
pRfDedicatedBandInfoParam
nas_RfDedicatedBandInfo, 338
pRingBackTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pRingingTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pRoamDisallowFlag
LibpackProfile3GPPV2, 143
pRoamPref
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 516
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pRoamTimer
pack_voice_SLQSVoiceGetConfig_t, 596
pRoamTimerCnt
unpack_voice_SLQSVoiceGetConfig_t, 1117
pRoamTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 601
pRoamTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 1121
pRoaming
pack_nas_SetCDMANetworkParameters_t, 467
pRscp
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
PSDomain
unpack_nas_GetServingNetwork_t, 886
pSID
unpack_nas_GetHomeNetwork3GPP2_t, 880
pSIODataPort
pack_qos_BindDataPort_t, 529
pSIPConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 438
pSIPLocalPort
pack_ims_SLQSSetSIPConfig_t, 444
unpack_ims_SLQSGetSIPConfig_t, 824

PSMCfgChangeInfo
pack_dms_SetIndicationRegister_t, 422
pSMSAttemptedFlag
packgetDyingGaspStatistics, 638
pSMSCAddress
pack_sms_SetSMSCAddress_t, 533
unpack_sms_GetSMSCAddress_t, 1014
pSMSCType
pack_sms_SetSMSCAddress_t, 533
unpack_sms_GetSMSCAddress_t, 1014
pSMSConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 438
pSMSFormat
pack_ims_SLQSSetIMSSMSConfig_t, 439
unpack_ims_SLQSGetIMSSMSConfig_t, 818
pSMSOverIPNwInd
pack_ims_SLQSSetIMSSMSConfig_t, 439
unpack_ims_SLQSGetIMSSMSConfig_t, 818
PSMStatus
pack_dms_SetIndicationRegister_t, 422
pSPC
pack_nas_SetCDMANetworkParameters_t, 467
pack_wds_SetMobileIPParameters_t, 620
pSSACBarringExtInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSSACBarringInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSV
loc_BdsSVInfo, 165
loc_SVInfo, 182
pSVInfo
pack_loc_Delete_Assist_Data_t, 446
pSarRFState
unpack_sar_SLQSGetRfSarState_t, 1013
pSatelliteInfo
unpack_loc_GnssSvInfo_Ind_t, 856
pScanResult
unpack_nas_PerformNetworkScan_t, 890
unpack_nas_SLQSPerformNetworkScanV2_t, 959
pScanType
pack_nas_PerformNetworkScanPCI_t, 465
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pSccRxInfo
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 963
pScrAmrEnable
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pScrAmrWbEnable
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pSearchPeriodicity
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pSecDNSIPv4AddPref
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pSecDNSIPv6addpref
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pSecV6DnsAddress
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pSecondaryFlag
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pSecondaryHA
pack_wds_SetMobileIPProfile_t, 621
pSecondaryV4DnsAddress
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pSectorID
nas_NetworkStatEVDO, 294
pSendSmsParams
pack_sms_SLQSSendAsyncSMS_t, 537
pSensorDataUsage
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pServerAddress
pack_pds_SLQSSetAGPSConfig_t, 525
unpack_pds_SLQSGetAGPSConfig_t, 982
pServerPort
pack_pds_SLQSSetAGPSConfig_t, 525
unpack_pds_SLQSGetAGPSConfig_t, 982
pServerURL
pack_pds_SLQSSetAGPSConfig_t, 525
unpack_pds_SLQSGetAGPSConfig_t, 982
pServerURLLength
pack_pds_SLQSSetAGPSConfig_t, 525
unpack_pds_SLQSGetAGPSConfig_t, 982
pServiceClass
pack_voice_SLQSVoiceSetSUPSService_t, 604
pServiceOption
sms_sendAsyncsmsParams, 668
pServingSystemInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSessionExpiryTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 442
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
pSessionType
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
pSetIndicationRegReq
pack_sms_SLQSSetIndicationRegister_t, 538
pSetRoutesReq
pack_sms_SLQSSmsSetRoutes_t, 540
pSettingResp
unpack_ims_SLQSGetIMSSMSConfig_t, 818
unpack_ims_SLQSGetIMSUserConfig_t, 819
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
unpack_ims_SLQSGetRegMgrConfig_t, 823

unpack_ims_SLQSGetSIPConfig_t, 824
unpack_ims_SLQSSetIMSSMSConfig_t, 826
unpack_ims_SLQSSetIMSUserConfig_t, 826
unpack_ims_SLQSSetIMSVoIPConfig_t, 827
unpack_ims_SLQSSetRegMgrConfig_t, 827
unpack_ims_SLQSSetSIPConfig_t, 828
pSigCompEnabled
pack_ims_SLQSSetSIPConfig_t, 443
unpack_ims_SLQSGetSIPConfig_t, 824
pSigIndReq
pack_nas_SLQSSetSignalStrengthsCallback_t, 500
pSignalInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 1133
pSignalStrengthInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSimBusyStatus
unpack_uim_GetCardStatusV2_t, 1069
pSimRejInfo
unpack_nas_SLQSGetSysInfoV2_t, 923
pSlotSelect
pack_swidms_SLQSSwiDmsSetUimSlot_t, 554
pSmsOnIms
sms_sendAsyncsmsParams, 668
pSmsStorage
unpack_sms_SLQSSwiGetSMSStorage_t, 1028
pSourceIP
LibPackTFTIDParams, 158
pSpeedHorizontal
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pSpeedUnc
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pSpeedVertical
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pSpeedVerticalUnc
unpack_loc_BestAvailPos_Ind_t, 846
pSrvDomainPref
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 516
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExt_t, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pSrvOpt
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
pSrvRegRestric
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pSrvRegRestriction
pack_nas_SLQSSetSysSelectionPref_t, 505
pack_nas_SLQSSetSysSelectionPrefExt_t, 516
unpack_nas_SLQSGetSysInfoV2_t, 923
pSrvcProviderName
unpack_nas_SLQSGetOperatorNameData_t, 900
pStage0Val
audio_RXPCMIIRFltr, 49
audio_TXPCMIIRFltr, 51
pStage1Val
audio_RXPCMIIRFltr, 49
audio_TXPCMIIRFltr, 51
pStage2Val
audio_RXPCMIIRFltr, 49
audio_TXPCMIIRFltr, 51
pStage3Val
audio_RXPCMIIRFltr, 49
audio_TXPCMIIRFltr, 51
pStage4Val
audio_RXPCMIIRFltr, 49
audio_TXPCMIIRFltr, 51
pStageCnt
audio_RXPCMIIRFltr, 49
audio_TXPCMIIRFltr, 51
pStatMask
pack_wds_GetPacketStatistics_t, 614
pSubscribeTimer
pack_ims_SLQSSetSIPConfig_t, 444
unpack_ims_SLQSGetSIPConfig_t, 824
pSubscriptionChangeInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSubscriptionInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSupportEmergencyCalls
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pSupportedBitmasks
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 1050
pSuppress_null_bearer_reason
pack_dsd_register_ui_info_changes_t, 434
pSuppress_so_change
pack_dsd_register_ui_info_changes_t, 434
pSuppressSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSuppsNotifEvents
pack_voice_SLQSVoiceIndicationRegister_t, 597
pSvUsedforFix
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pSvcClass
pack_voice_SLQSVoiceGetCallBarring_t, 592
pack_voice_SLQSVoiceGetCallForwardingStatus_t, 593
pack_voice_SLQSVoiceGetCallWaiting_t, 594
unpack_voice_SLQSVoiceGetCallBarring_t, 1102
unpack_voice_SLQSVoiceGetCallWaiting_t, 1108
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pSvcType
pack_voice_SLQSVoiceDialCall_t, 591

pSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pSysInfoNoChange
unpack_nas_SLQSSysInfoCallback_ind_t, 970
pSysTime
unpack_dms_GetNetworkTimeV2_t, 761
pSystemSelectionInd
pack_nas_SLQSNasIndicationRegisterExt_t, 480
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pT3346TimerStatusChangeInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pT3402TimerValueInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pTDSCDMABandPref
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pTDSCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMAECIOThreshList
nas_TDSCDMAECIOThresh, 364
pTDSCDMARSCPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMARSCPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMARSCPThreshList
nas_TDSCDMARSCPThresh, 364
pTDSCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMARSSIThreshList
nas_TDSCDMARSSIThresh, 365
pTDSCDMASINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMASINRThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pTDSCDMASINRThreshList
nas_TDSCDMASINRThresh, 366
pTDSCDMASigInfoExt
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
pTFTID1Params
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pTFTID2Params
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pTTYConfigStatus
unpack_voice_SLQSVoiceSetConfig_t, 1121
pTTYMode
pack_voice_SLQSVoiceGetConfig_t, 596
pack_voice_SLQSVoiceSetConfig_t, 601
pTXAGCList
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pTXAIG
audio_TXAGCList, 49
pTXAVCSwitch
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pTXComprSlope
audio_TXAGCList, 50
pTXComprThres
audio_TXAGCList, 50
pTXDroppedCount
unpack_wds_GetPacketStatistics_t, 1157
pTXExpSlope
audio_TXAGCList, 50
pTXExpThres
audio_TXAGCList, 50
pTXGain
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pTXInfo
unpack_nas_SLQSNasGetTxRxInfo_t, 946
pTXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 1157
pTXOkBytesCount
unpack_wds_GetPacketStatistics_t, 1157
pTXPCMIIRFltr
pack_audio_SLQSSetAudioPathConfig_t, 410
unpack_audio_SLQSGetAudioPathConfig_t, 731
pTXPacketErrors
unpack_wds_GetPacketStatistics_t, 1157
pTXPacketOverflows
unpack_wds_GetPacketStatistics_t, 1157
pTXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 1157
pTXStaticGain
audio_TXAGCList, 50
pTXTotalBytes
unpack_wds_GetByteTotals_t, 1144
pTcp_coalescing
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pTdscdmaBandPref
pack_nas_SLQSSetSysSelectionPref_t, 506
pack_nas_SLQSSetSysSelectionPrefExt_t, 516
pTech
pack_wds_SLQSStartDataSession_t, 633
pTechnologyMask
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pTempReport
pack_dms_SwiSetEventReport_t, 428
pTimeSrc
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_PositionRpt_Ind_t, 867
pTimeStamp
pack_pds_SLQSPDSInjectPositionData_t, 524

packgetDyingGaspStatistics, 638
pTimeType
pack_pds_SLQSPDSInjectPositionData_t, 524
pTimeUnc
unpack_loc_BestAvailPos_Ind_t, 847
unpack_loc_PositionRpt_Ind_t, 867
pTimeZone
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 950
pTimeout
unpack_pds_GetPDSDefaults_t, 975
pTimer
pack_nas_SLQSNasSwiIndicationRegister_t, 491
pTimerExpiryInd
pack_nas_SLQSNasIndicationRegisterV2_t, 488
pTimerSIPReg
pack_ims_SLQSSetSIPConfig_t, 444
unpack_ims_SLQSGetSIPConfig_t, 825
pTimerT1
pack_ims_SLQSSetSIPConfig_t, 444
unpack_ims_SLQSGetSIPConfig_t, 825
pTimerT2
pack_ims_SLQSSetSIPConfig_t, 444
unpack_ims_SLQSGetSIPConfig_t, 825
pTimerTf
pack_ims_SLQSSetSIPConfig_t, 444
unpack_ims_SLQSGetSIPConfig_t, 825
pTimerVal
pack_voice_SLQSVoiceSetSUPSService_t, 604
pTimestampUtc
unpack_loc_BestAvailPos_Ind_t, 847
unpack_loc_PositionRpt_Ind_t, 867
pTrackingStatus
unpack_pds_GetPDSState_t, 975
pTransLayerInfo
sms_getTransLayerInfo, 661
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 1029
pTransferStatInd
pack_wds_SLQSGetDUNCallInfo_t, 625
pack_wds_SLQSWdsSetEventReport_t, 635
pTransferStatusReport
sms_setRoutesReq, 670
pTransport
QmiService, 644
pTrueIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
pUATI
unpack_nas_SLQSSwiGetHRPDStats_t, 962
pUIMStatusReport
pack_dms_SwiSetEventReport_t, 428
pUMTSCellID
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pUMTSExtInfo
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pUMTSInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pUMTSMinQoS
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pUMTSMinQosSigInd
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pUMTSReqQoS
LibpackProfile3GPP, 126
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pUMTSReqQoSSigInd
LibpackProfile3GPP, 127
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pURL
unpack_loc_GetServer_Ind_t, 856
pURLAddr
pack_loc_SLQSLOCSetServer_t, 459
pUSSDInfo
unpack_voice_SLQSOriginateUSSD_t, 1094
pUSSInfo
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
pUUSInFo
pack_voice_SLQSVoiceDialCall_t, 591
pUUSInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 1107
pUdp_coalescing
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pUimAutoSwitchActSlot
unpack_dms_SLQSDmsSwiGetUimSelection_t, 781
pUimSlotID
pack_uim_SLQSUIMGetEID_t, 572
pUimSlotsStatus
unpack_uim_SLQSUIMGetSlotsStatus_t, 1080
pUl_data_aggregation_max_datagrams
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pUl_data_aggregation_max_size
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pUl_data_aggregation_protocol
pack_wda_SetDataFormat_t, 609
unpack_wda_SetDataFormat_t, 1141
pUmtsRoamPDPType
LibpackProfile3GPPV2, 143
pUname
PackSwiAVMSSettingsAPNInfo, 640

pUsageSetting
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pUsbMTUSize
unpack_swidms_SLQSSwiDmsGetMTU_t, 1047
pUser
pack_wds_SLQSStartDataSession_t, 633
pUserConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 438
pUserConsent
pack_uim_SLQSUIMAddProfile_t, 568
pUserData
sms_sendAsyncsmsParams, 668
pUserId
LibpackProfile3GPP2, 132
LibPackprofile_3GPP2, 154
pUserIdSize
LibpackProfile3GPP2, 133
LibPackprofile_3GPP2, 154
pUsername
LibpackProfile3GPP, 127
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pack_wds_SetDefaultProfile_t, 617
pUsernameSize
LibpackProfile3GPP, 127
LibpackProfile3GPPV2, 143
LibPackprofile_3GPP, 149
pUsrTime
unpack_dms_GetNetworkTimeV2_t, 761
pValidCardStatus
unpack_uim_GetCardStatusV2_t, 1069
pVerboseFailReasonType
unpack_wds_SLQSStartDataSession_t, 1189
pVerboseFailureReason
unpack_wds_SLQSStartDataSession_t, 1189
pVertConfidence
unpack_loc_BestAvailPos_Ind_t, 847
unpack_loc_PositionRpt_Ind_t, 867
pVertReliability
unpack_loc_BestAvailPos_Ind_t, 847
unpack_loc_PositionRpt_Ind_t, 867
pVertUnc
unpack_loc_BestAvailPos_Ind_t, 847
unpack_loc_PositionRpt_Ind_t, 867
pVerticalConfidence
pack_pds_SLQSPDSInjectPositionData_t, 524
pVerticalUnc
pack_pds_SLQSPDSInjectPositionData_t, 524
pVoIPConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 438
pVoiceDomainPref
pack_voice_SLQSVoiceGetConfig_t, 596
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
pVoiceDomainPrefStatus
unpack_voice_SLQSVoiceSetConfig_t, 1121
pVoicePrivacy
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
unpack_voice_SLQSVoiceGetCallInfo_t, 1107
pVoltReport
pack_dms_SwiSetEventReport_t, 428
pVolume
pack_swiaudio_SLQSSetM2MAudioProfile_t, 544
pWAMSParaChanged
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
pWCDMAArfcn
pack_nas_SLQSPerformIncNetworkScanReq_t, 497
pWCDMABER
unpack_nas_SLQSGetErrorRate_t, 896
pWCDMACallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 923
unpack_nas_SLQSSysInfoCallback_ind_t, 970
pWCDMACellInfoExt
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pWCDMACipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 923
unpack_nas_SLQSSysInfoCallback_ind_t, 970
pWCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pWCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pWCDMAECIOThreshList
nas_WCDMAECIOThresh, 380
pWCDMAInfoLTENeighborCell
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
pWCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 475
pWCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 475
pWCDMARSSIThreshList
nas_WCDMARSSIThresh, 382
pWCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
pWCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 923
unpack_nas_SLQSSysInfoCallback_ind_t, 970
pWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 923
unpack_nas_SLQSSysInfoCallback_ind_t, 970
pWcdmaUARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 492
unpack_nas_SLQSNASSwiGetChannelLock_t, 955
pWifiState

pack_pds_SLQSSetPositionMethodState_t, 526
pXid
unpack_loc_BestAvailPos_Ind_t, 847
pXtraDataState
pack_pds_SLQSSetPositionMethodState_t, 526
pXtraTimeState
pack_pds_SLQSSetPositionMethodState_t, 526
pack_audio_SLQSGetAudioPathConfig
audio.h, 1278
pack_audio_SLQSGetAudioPathConfig_t, 406
Item, 407
Profile, 407
pack_audio_SLQSGetAudioProfile
audio.h, 1278
pack_audio_SLQSGetAudioProfile_t, 407
Generator, 408
pack_audio_SLQSGetAudioVolTLBConfig
audio.h, 1279
pack_audio_SLQSGetAudioVolTLBConfig_t, 408
Generator, 408
Item, 408
Profile, 408
Volume, 408
pack_audio_SLQSSetAudioPathConfig
audio.h, 1279
pack_audio_SLQSSetAudioPathConfig_t, 408
pCodecSTGain, 410
pECMode, 410
pNSEnable, 410
pRXAGCList, 410
pRXAVCList, 410
pTXAGCList, 410
pTXAVCSwitch, 410
pTXGain, 410
Profile, 410
pack_audio_SLQSSetAudioProfile
audio.h, 1280
pack_audio_SLQSSetAudioProfile_t, 410
EarMute, 411
Generator, 411
MicMute, 411
Profile, 411
Volume, 411
pack_audio_SLQSSetAudioVolTLBConfig
audio.h, 1280
pack_audio_SLQSSetAudioVolTLBConfig_t, 412
Generator, 412
Item, 412
Profile, 412
VolValue, 412
Volume, 412
pack_cat_CATSendEnvelopeCommand
cat.h, 1285
pack_cat_CATSendEnvelopeCommand_t, 412
cmdID, 413
dataLen, 413
pData, 413
pack_cat_CATSendTerminalResponse
cat.h, 1286
pack_cat_CATSendTerminalResponse_t, 413
dataLen, 414
pData, 414
refID, 414
pack_cat_SetCATEventCallback
cat.h, 1286
pack_cat_SetCATEventCallback_t, 414
eventMask, 414
pack_ctl_GetClientId
ctl.h, 1294
pack_ctl_GetClientId_t, 415
serviceType, 415
pack_ctl_ReleaseClientId
ctl.h, 1295
pack_ctl_ReleaseClientId_t, 415
clientId, 415
serviceType, 415
pack_dms_ActivateAutomatic
dms.h, 1308
pack_dms_ActivateAutomatic_t, 415
actCode, 416
pack_dms_GetActivationState
dms.h, 1308
pack_dms_GetBandCapability
dms.h, 1309
pack_dms_GetCrashAction
dms.h, 1309
pack_dms_GetCustFeature
dms.h, 1309
pack_dms_GetCustFeaturesV2
dms.h, 1310
pack_dms_GetCustFeaturesV2_t, 416
cust_id, 416
list_type, 416
Tlvresult, 416
pack_dms_GetDeviceCap
dms.h, 1310
pack_dms_GetDeviceCapabilities
dms.h, 1311
pack_dms_GetDeviceCapabilitiesV2
dms.h, 1311
pack_dms_GetDeviceHardwareRev
dms.h, 1311
pack_dms_GetDeviceMfr
dms.h, 1312
pack_dms_GetDeviceSerialNumbers
dms.h, 1312
pack_dms_GetFSN
dms.h, 1313
pack_dms_GetFirmwareInfo
dms.h, 1312
pack_dms_GetFirmwareRevision
dms.h, 1313
pack_dms_GetFirmwareRevisions
dms.h, 1313
pack_dms_GetHardwareRevision
dms.h, 1314

pack_dms_GetIMSI
dms.h, 1314
pack_dms_GetManufacturer
dms.h, 1314
pack_dms_GetModelID
dms.h, 1315
pack_dms_GetNetworkTime
dms.h, 1315
pack_dms_GetNetworkTimeV2
dms.h, 1315
pack_dms_GetOfflineReason
dms.h, 1316
pack_dms_GetPRLVersion
dms.h, 1316
pack_dms_GetPower
dms.h, 1316
pack_dms_GetSerialNumbers
dms.h, 1317
pack_dms_GetUSBComp
dms.h, 1317
pack_dms_GetVoiceNumber
dms.h, 1317
pack_dms_ResetToFactoryDefaults
dms.h, 1318
pack_dms_ResetToFactoryDefaults_t, 416
spc, 417
pack_dms_SLQSDmsSwiGetPCInfo
dms.h, 1322
pack_dms_SLQSDmsSwiGetResetInfo
dms.h, 1322
pack_dms_SLQSDmsSwiGetUimSelection
dms.h, 1322
pack_dms_SLQSDmsSwiIndicationRegister
dms.h, 1323
pack_dms_SLQSDmsSwiIndicationRegister_t, 424
resetInfoInd, 424
pack_dms_SLQSGetBandCapability
dms.h, 1323
pack_dms_SLQSGetERIFile
dms.h, 1323
pack_dms_SLQSGetPowerSaveModeConfig
dms.h, 1324
pack_dms_SLQSSetPowerSaveModeConfig
dms.h, 1324
pack_dms_SLQSSetPowerSaveModeConfig_t, 424
pActiveTimer, 425
pPeriodicUpdateTimer, 425
pPsmEnableState, 425
pack_dms_SLQSSwiClearDyingGaspStatistics
dms.h, 1324
pack_dms_SLQSSwiGetCrashInfo
dms.h, 1325
pack_dms_SLQSSwiGetCrashInfo_t, 425
clear, 425
pack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 1325
pack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 1325
pack_dms_SLQSSwiGetFirmwareCurr
dms.h, 1326
pack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 1326
pack_dms_SLQSSwiGetHostDevInfo
dms.h, 1326
pack_dms_SLQSSwiGetOSInfo
dms.h, 1327
pack_dms_SLQSSwiGetSerialNoExt
dms.h, 1327
pack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 1327
pack_dms_SLQSSwiSetDyingGaspCfg_t, 425
pDestSMSContent, 426
pDestSMSNum, 426
pack_dms_SLQSSwiSetHostDevInfo
dms.h, 1328
pack_dms_SLQSSwiSetHostDevInfo_t, 426
hostID, 427
manString, 427
modelString, 427
plasmaIDString, 427
swVerString, 427
pack_dms_SLQSSwiSetOSInfo
dms.h, 1328
pack_dms_SLQSSwiSetOSInfo_t, 427
nameString, 427
versionString, 427
pack_dms_SLQSUIMGetState
dms.h, 1329
pack_dms_SetActivationStatusCallback
dms.h, 1318
pack_dms_SetActivationStatusCallback_t, 417
activationState, 417
pack_dms_SetCrashAction
dms.h, 1319
pack_dms_SetCrashAction_t, 417
crashAction, 418
pack_dms_SetCustFeature
dms.h, 1319
pack_dms_SetCustFeature_t, 418
DHCPRelayEnabled, 420
DisableIMSI, 420
GPSLPM, 420
GPSSel, 420
GpsEnable, 420
IPFamSupport, 420
IsVoiceEnabled, 420
RMAutoConnect, 420
SMSSupport, 420
pack_dms_SetCustFeaturesV2
dms.h, 1319
pack_dms_SetCustFeaturesV2_t, 420
cust_id, 421
cust_value, 421
Tlvresult, 421
value_length, 421
pack_dms_SetEventReport

dms.h, 1320
pack_dms_SetEventReport_t, 421
mode, 421
pack_dms_SetFirmwarePreference
dms.h, 1320
pack_dms_SetIndicationRegister
dms.h, 1320
pack_dms_SetIndicationRegister_t, 421
PSMCfgChangeInfo, 422
PSMStatus, 422
RptIMSCapability, 422
pack_dms_SetPower
dms.h, 1321
pack_dms_SetPower_t, 422
mode, 423
Tlvresult, 423
pack_dms_SetUSBComp
dms.h, 1321
pack_dms_SetUSBComp_t, 423
Tlvresult, 423
USBComp, 423
pack_dms_SwiSetEventReport
dms.h, 1329
pack_dms_SwiSetEventReport_t, 427
pTempReport, 428
pUIMStatusReport, 428
pVoltReport, 428
pack_dms_SwiUimSelect
dms.h, 1329
pack_dms_SwiUimSelect_t, 428
uim_select, 428
pack_dms_UIMChangePIN
dms.h, 1330
pack_dms_UIMChangePIN_t, 429
id, 429
newValue, 429
oldValue, 429
pack_dms_UIMGetControlKeyStatus
dms.h, 1330
pack_dms_UIMGetControlKeyStatus_t, 429
facility, 430
pack_dms_UIMGetICCID
dms.h, 1330
pack_dms_UIMGetICCID_t, 430
ParamPresenceMask, 430
Tlvresult, 430
pack_dms_UIMGetPINStatus
dms.h, 1331
pack_dms_UIMSetControlKeyProtection
dms.h, 1331
pack_dms_UIMSetControlKeyProtection_t, 430
facility, 431
facilityCk, 431
facilityState, 431
pack_dms_UIMSetPINProtection
dms.h, 1332
pack_dms_UIMSetPINProtection_t, 431
bEnable, 432
id, 432
value, 432
pack_dms_UIMUnblockControlKey
dms.h, 1332
pack_dms_UIMUnblockControlKey_t, 432
facility, 432
facilityCk, 432
pack_dms_UIMUnblockPIN
dms.h, 1332
pack_dms_UIMUnblockPIN_t, 432
id, 433
newPin, 433
pukValue, 433
pack_dms_UIMVerifyPIN
dms.h, 1333
pack_dms_UIMVerifyPIN_t, 433
id, 433
value, 433
pack_dms_ValidateSPC
dms.h, 1333
pack_dsd_RegisterUIInfoChange
dsd.h, 1363
pack_dsd_SLQSGetSystemStatus
dsd.h, 1363
pack_dsd_SLQSGetUIInfo
dsd.h, 1364
pack_dsd_SystemStatusChange
dsd.h, 1364
pack_dsd_register_ui_info_changes_t, 433
pReport_ui_changes, 434
pSuppress_null_bearer_reason, 434
pSuppress_so_change, 434
pack_dsd_system_status_changes_t, 434
pLimit_so_mask_change_ind, 435
pReport_data_system_status_changes, 435
pReport_null_bearer_reason, 435
pReport_pref_tech_change_only, 435
pack_fms_GetImagesPreference
fms.h, 1369
pack_fms_GetImagesPreference_t, 435
Tlvresult, 435
pack_fms_GetStoredImages
fms.h, 1369
pack_fms_GetStoredImages_t, 436
Tlvresult, 436
pack_fms_SetImagesPreference
fms.h, 1369
pack_fms_SetImagesPreference_t, 436
bForceDownload, 437
imageListSize, 437
modemindex, 437
pImageList, 437
Tlvresult, 437
pack_ims_SLQSGetIMSSMSConfig
ims.h, 1372
pack_ims_SLQSGetIMSUserConfig
ims.h, 1373
pack_ims_SLQSGetIMSVoIPConfig

ims.h, 1373
pack_ims_SLQSGetRegMgrConfig
ims.h, 1373
pack_ims_SLQSGetSIPConfig
ims.h, 1374
pack_ims_SLQSImsConfigIndicationRegister
ims.h, 1374
pack_ims_SLQSImsConfigIndicationRegister_t, 437
pRegMgrConfigEvents, 438
pSIPConfigEvents, 438
pSMSConfigEvents, 438
pUserConfigEvents, 438
pVoIPConfigEvents, 438
pack_ims_SLQSSetIMSSMSConfig
ims.h, 1375
pack_ims_SLQSSetIMSSMSConfig_t, 438
pack_ims_SLQSSetIMSUserConfig
ims.h, 1375
pack_ims_SLQSSetIMSUserConfig_t, 439
pIMSDomain, 439
pack_ims_SLQSSetIMSVoIPConfig
ims.h, 1376
pack_ims_SLQSSetIMSVoIPConfig_t, 439
pAmrMode, 442
pRingingTimer, 442
pack_ims_SLQSSetRegMgrConfig
ims.h, 1376
pack_ims_SLQSSetRegMgrConfig_t, 442
pCSCFPortName, 443
pIMSTestMode, 443
pPriCSCFPort, 443
pack_ims_SLQSSetSIPConfig
ims.h, 1377
pack_ims_SLQSSetSIPConfig_t, 443
pSIPLocalPort, 444
pSigCompEnabled, 443
pSubscribeTimer, 444
pTimerSIPReg, 444
pTimerT1, 444
pTimerT2, 444
pTimerTf, 444
pack_imsa_SLQSGetIMSARegStatus
imsa.h, 1384
pack_imsa_SLQSGetIMSAServiceStatus
imsa.h, 1384
pack_imsa_SLQSRegisterIMSAIndication
imsa.h, 1385
pack_imsa_SLQSRegisterIMSAIndication_t, 444
PdpStatusConfig, 445
RatHandoverStatusConfig, 445
RegStatusConfig, 445
ServiceStatusConfig, 445
pack_loc_Delete_Assist_Data_t, 445
pBdsSVInfo, 446
pCellDb, 446
pClkInfo, 446
pGnssData, 446
pSVInfo, 446
Tlvresult, 446
pack_loc_DeleteAssistData
loc.h, 1409
pack_loc_EventRegister
loc.h, 1409
pack_loc_EventRegister_t, 446
eventRegister, 448
Tlvresult, 448
pack_loc_GetFixCriteria
loc.h, 1410
pack_loc_SLQSLOCGetBestAvailPos
loc.h, 1411
pack_loc_SLQSLOCGetBestAvailPos_t, 449
Tlvresult, 449
xid, 449
pack_loc_SLQSLOCGetOpMode
loc.h, 1411
pack_loc_SLQSLOCGetServer
loc.h, 1412
pack_loc_SLQSLOCGetServer_t, 450
has_serverAddrTypeMask, 450
serverAddrTypeMask, 450
serverType, 450
pack_loc_SLQSLOCInjectPosition
loc.h, 1412
pack_loc_SLQSLOCInjectPosition_t, 450
altitudeSrcInfo, 454
altitudeWrtEllipsoid, 454
altitudeWrtMeanSeaLevel, 454
has_altitudeSrcInfo, 454
has_altitudeWrtEllipsoid, 454
has_horConfidence, 454
has_horReliability, 454
has_horUncCircular, 454
has_latitude, 454
has_longitude, 454
has_positionSrc, 454
has_rawHorConfidence, 454
has_rawHorUncCircular, 454
has_timestampAge, 454
has_timestampUtc, 454
has_vertConfidence, 454
has_vertRelicability, 454
has_vertUnc, 455
horConfidence, 455
horReliability, 455
horUncCircular, 455
latitude, 455
longitude, 455
positionSrc, 455
rawHorConfidence, 455
rawHorUncCircular, 455
timestampAge, 455
timestampUtc, 455
vertConfidence, 455
vertReliability, 455
vertUnc, 455
pack_loc_SLQSLOCInjectSensorData

loc.h, 1412
pack_loc_SLQSLOCInjectSensorData_t, 455
accelTemp, 456
acceleroData, 456
acceleroTimeSrc, 456
gyroData, 456
gyroTemp, 456
gyroTimeSrc, 457
has_accelTemp, 457
has_acceleroTimeSrc, 457
has_accleroData, 457
has_gyroData, 457
has_gyroTemp, 457
has_gyroTimeSrc, 457
has_opaqueId, 457
opaqueId, 457
pack_loc_SLQSLOCInjectUTCTime
loc.h, 1413
pack_loc_SLQSLOCInjectUTCTime_t, 457
timeMsec, 457
timeUncMsec, 457
pack_loc_SLQSLOCSetCradleMountConfig
loc.h, 1413
pack_loc_SLQSLOCSetCradleMountConfig_t, 457
confidence, 458
has_confidence, 458
state, 458
pack_loc_SLQSLOCSetServer
loc.h, 1414
pack_loc_SLQSLOCSetServer_t, 458
pIPv4Config, 459
pIPv6Config, 459
pURLAddr, 459
serverType, 459
pack_loc_SetExtPowerState
loc.h, 1410
pack_loc_SetExtPowerState_t, 448
extPowerState, 448
Tlvresult, 448
pack_loc_SetOperationMode
loc.h, 1411
pack_loc_SetOperationMode_t, 448
mode, 449
Tlvresult, 449
pack_loc_Start
loc.h, 1414
pack_loc_Start_t, 459
pApplicationInfo, 460
pConfigAltitudeAssumed, 460
pHorizontalAccuracyLvl, 460
pIntermediateReportState, 460
pMinIntervalTime, 460
pRecurrenceType, 460
SessionId, 461
Tlvresult, 461
pack_loc_Stop
loc.h, 1414
pack_loc_Stop_t, 461
SessionId, 461
Tlvresult, 461
pack_nas_GetACCOLC
nas.h, 1456
pack_nas_GetANAAAAuthenticationStatus
nas.h, 1456
pack_nas_GetCDMANetworkParameters
nas.h, 1456
pack_nas_GetHomeNetwork
nas.h, 1456
pack_nas_GetHomeNetwork3GPP2
nas.h, 1457
pack_nas_GetNetworkPreference
nas.h, 1457
pack_nas_GetRFInfo
nas.h, 1457
pack_nas_GetServingNetwork
nas.h, 1458
pack_nas_GetServingNetworkCapabilities
nas.h, 1458
pack_nas_GetSignalStrengths
nas.h, 1458
pack_nas_InitiateDomainAttach
nas.h, 1459
pack_nas_InitiateDomainAttach_t, 461
action, 461
pack_nas_PerformNetworkScan
nas.h, 1459
pack_nas_PerformNetworkScanPCI
nas.h, 1459
pack_nas_PerformNetworkScanPCI_t, 462
pCiotOpModePref, 464
pLteM1BandPref, 464
pLteNB1BandPref, 465
pNetworkType, 465
pScanType, 465
pack_nas_SLQSConfigSigInfo
nas.h, 1461
pack_nas_SLQSConfigSigInfo_t, 468
pECIOThresh, 469
pIOThresh, 469
pLTESNRThresh, 469
pLTESigRptCfg, 469
pRSRPThresh, 469
pRSRQThresh, 469
pRSSIThresh, 469
pack_nas_SLQSGetErrorRate
nas.h, 1462
pack_nas_SLQSGetHomeNetwork
nas.h, 1462
pack_nas_SLQSGetNetworkTime
nas.h, 1463
pack_nas_SLQSGetOperatorNameData
nas.h, 1463
pack_nas_SLQSGetPLMNName
nas.h, 1463
pack_nas_SLQSGetPLMNName_t, 469
mcc, 470

mnc, 470
pMncPcsStatus, 470
pack_nas_SLQSGetServingSystem
nas.h, 1464
pack_nas_SLQSGetServingSystemV2
nas.h, 1464
pack_nas_SLQSGetSignalStrength
nas.h, 1464
pack_nas_SLQSGetSysInfo
nas.h, 1465
pack_nas_SLQSGetSysInfoV2
nas.h, 1465
pack_nas_SLQSGetSysSelectionPref
nas.h, 1466
pack_nas_SLQSGetSysSelectionPrefExt
nas.h, 1466
pack_nas_SLQSGetSysSelectionPrefExtV2
nas.h, 1466
pack_nas_SLQSInitiateNetworkRegistration
nas.h, 1467
pack_nas_SLQSInitiateNetworkRegistration_t, 470
pChangeDuration, 471
pMNRInfo, 471
pMncPcsDigitStatus, 471
regAction, 471
pack_nas_SLQSNASGetForbiddenNetworks
nas.h, 1469
pack_nas_SLQSNASGeteDRXParams
nas.h, 1468
pack_nas_SLQSNASGeteDRXParamsExt
nas.h, 1469
pack_nas_SLQSNASGeteDRXParamsExt_t, 476
pack_nas_SLQSNASSeteDRXParams
nas.h, 1472
pack_nas_SLQSNASSeteDRXParams_t, 488
pCycleLen, 490
pEdrxEnable, 490
pack_nas_SLQSNASSwiGetChannelLock
nas.h, 1472
pack_nas_SLQSNASSwiSetChannelLock
nas.h, 1473
pack_nas_SLQSNASSwiSetChannelLock_t, 491
pack_nas_SLQSNasConfigSigInfo2
nas.h, 1467
pack_nas_SLQSNasConfigSigInfo2_t, 471
pHDRIODelta, 475
pHDRIOThresh, 475
pLTESigRptConfig, 475
pack_nas_SLQSNasGet3GPP2Subscription
nas.h, 1467
pack_nas_SLQSNasGet3GPP2Subscription_t, 476
namID, 476
pack_nas_SLQSNasGetCellLocationInfo
nas.h, 1468
pack_nas_SLQSNasGetCellLocationInfoV2
nas.h, 1468
pack_nas_SLQSNasGetHDRColorCode
nas.h, 1469
pack_nas_SLQSNasGetRFInfo
nas.h, 1470
pack_nas_SLQSNasGetSigInfo
nas.h, 1470
pack_nas_SLQSNasGetTxRxInfo
nas.h, 1470
pack_nas_SLQSNasGetTxRxInfo_t, 477
radio_if, 477
pack_nas_SLQSNasIndicationRegisterExt
nas.h, 1471
pack_nas_SLQSNasIndicationRegisterExt_t, 477
pDDTMInd, 480
pDualStandByPrefInd, 480
pEdrxChangeInfoInd, 480
pErrorRateInd, 480
pHDRSessionCloseInd, 480
pLTECphyCa, 480
pManagedRoamingInd, 480
pNetworkRejectInd, 480
pNetworkTimeInd, 480
pServingSystemInd, 480
pSignalStrengthInd, 480
pSubscriptionInfoInd, 480
pSuppressSysInfoInd, 480
pSysInfoInd, 480
pSystemSelectionInd, 480
pack_nas_SLQSNasIndicationRegisterV2
nas.h, 1471
pack_nas_SLQSNasIndicationRegisterV2_t, 480
pAccessClassBarringInd, 487
pCallModeStatusInd, 487
pDDTMInd, 487
pDataSubscriptionPriorityInd, 487
pDualStandByPrefInd, 487
pE911StateReadyStatusInd, 487
pEdrxChangeInfoInd, 487
pEmergencyModeStatusInd, 487
pErrorRateInd, 487
pGcellInfoInd, 487
pHDRSessionCloseInd, 488
pIMSPrefStatusInd, 488
pLTECphyCa, 488
pManagedRoamingInd, 488
pNetworkRejectInd, 488
pNetworkTimeInd, 488
pOperatorNameDataInd, 488
pRFBandInfoInd, 488
pRTREConfigurationInd, 488
pSSACBarringInd, 488
pServingSystemInd, 488
pSignalStrengthInd, 488
pSubscriptionChangeInd, 488
pSubscriptionInfoInd, 488
pSuppressSysInfoInd, 488
pSysInfoInd, 488
pSystemSelectionInd, 488
pT3346TimerStatusChangeInd, 488
pT3402TimerValueInd, 488

pTimerExpiryInd, 488
pack_nas_SLQSNasSwiIndicationRegister
nas.h, 1472
pack_nas_SLQSNasSwiIndicationRegister_t, 490
gsmUmtsDl, 491
gsmUmtsUl, 491
lteEmmDl, 491
lteEmmUl, 491
lteEsmDl, 491
lteEsmUl, 491
pRankIndicatorInd, 491
pTimer, 491
pack_nas_SLQSNasSwiModemStatus
nas.h, 1473
pack_nas_SLQSPerformIncNetworkScanReq
nas.h, 1473
pack_nas_SLQSPerformIncNetworkScanReq_t, 492
pBandPref, 497
pIncResultPeriodicty, 497
pIncrementalResults, 497
pLTEArfcn, 497
pLTEBandPref, 497
pMaxSearchTime, 497
pNetworkType, 497
pScanType, 497
pSearchPeriodicity, 497
pack_nas_SLQSPerformNetworkScanV2
nas.h, 1474
pack_nas_SLQSSet3GPP2SubscriptionInfo
nas.h, 1474
pack_nas_SLQSSet3GPP2SubscriptionInfo_t, 497
namID, 498
pMdn, 498
pack_nas_SLQSSetBandPreference
nas.h, 1475
pack_nas_SLQSSetBandPreference_t, 498
bandPref, 499
pack_nas_SLQSSetDDTMPref
nas.h, 1475
pack_nas_SLQSSetDDTMPref_t, 500
DDTMAction, 500
SOLen, 500
SOListAction, 500
ServiceOption, 500
pack_nas_SLQSSetSignalStrengthsCallback
nas.h, 1475
pack_nas_SLQSSetSignalStrengthsCallback_t, 500
bEnable, 500
pSigIndReq, 500
pack_nas_SLQSSetSysSelectionPref
nas.h, 1476
pack_nas_SLQSSetSysSelectionPref_t, 500
pAcqOrderPref, 505
pBandPref, 505
pCSGID, 505
pChgDuration, 505
pEmerMode, 505
pGWAcqOrderPref, 505
pLTEBandPref, 505
pModePref, 505
pNetSelPref, 505
pPRLPref, 505
pRAT, 505
pRoamPref, 505
pSrvDomainPref, 505
pSrvRegRestriction, 505
pTdscdmaBandPref, 506
pack_nas_SLQSSetSysSelectionPrefExt
nas.h, 1476
pack_nas_SLQSSetSysSelectionPrefExt_t, 506
pAcqOrderPref, 515
pBandPref, 515
pCSGID, 515
pChgDuration, 515
pCiotAcqOrderPref, 515
pCiotLteOpMode, 515
pCiotLteOpModePref, 515
pEmerMode, 515
pGWAcqOrderPref, 515
pLTEBandPref, 515
pLteM1BandPref, 515
pLteNb1BandPref, 515
pModePref, 515
pNetSelPref, 515
pNr5gBandPref, 515
pPRLPref, 515
pRAT, 516
pRoamPref, 516
pSrvDomainPref, 516
pSrvRegRestriction, 516
pTdscdmaBandPref, 516
pack_nas_SLQSSwiGetHDRPersonality
nas.h, 1476
pack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 1477
pack_nas_SLQSSwiGetHRPDStats
nas.h, 1477
pack_nas_SLQSSwiGetLteCQI
nas.h, 1477
pack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 1478
pack_nas_SLQSSwiNetworkDebug
nas.h, 1478
pack_nas_SLQSSwiPSDetach
nas.h, 1478
pack_nas_SLQSSwiPSDetach_t, 516
pDetachAction, 516
pack_nas_SetACCOLC
nas.h, 1460
pack_nas_SetACCOLC_t, 465
accolc, 465
spc, 465
pack_nas_SetCDMANetworkParameters
nas.h, 1460
pack_nas_SetCDMANetworkParameters_t, 465
pApplication, 466

pBroadcast, 466
pCustomSCP, 466
pForceRev0, 467
pProtocol, 467
pRoaming, 467
pSPC, 467
pack_nas_SetLURejectCallback
nas.h, 1461
pack_nas_SetNetworkPreference
nas.h, 1461
pack_nas_SetNetworkPreference_t, 467
Duration, 467
TechnologyPref, 467
Tlvresult, 468
pack_nas_SetRFInfoCallback
nas.h, 1461
pack_nas_SlqsGetLTECphyCAInfo
nas.h, 1462
pack_pds_ForceXTRADownload
pds.h, 1514
pack_pds_GetPDSDefaults
pds.h, 1514
pack_pds_GetPDSState
pds.h, 1515
pack_pds_GetPortAutomaticTracking
pds.h, 1515
pack_pds_GetServiceAutomaticTracking
pds.h, 1515
pack_pds_GetXTRAAutomaticDownload
pds.h, 1516
pack_pds_GetXTRANetwork
pds.h, 1516
pack_pds_GetXTRAValidity
pds.h, 1516
pack_pds_PDSInjectTimeReference
pds.h, 1517
pack_pds_PDSInjectTimeReference_t, 516
systemDiscontinuities, 517
systemTime, 517
pack_pds_ResetPDSData
pds.h, 1517
pack_pds_ResetPDSData_t, 517
pCellDataMask, 517
pGPSDataMask, 518
pack_pds_SLQSGetAGPSConfig
pds.h, 1520
pack_pds_SLQSGetAGPSConfig_t, 521
pNetworkMode, 522
pack_pds_SLQSGetGPSStateInfo
pds.h, 1520
pack_pds_SLQSPDSDeterminePosition
pds.h, 1521
pack_pds_SLQSPDSInjectAbsoluteTimeReference
pds.h, 1521
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 522
forceFlag, 522
timeBase, 522
timeMsec, 522
timeUncMsec, 523
pack_pds_SLQSPDSInjectPositionData
pds.h, 1521
pack_pds_SLQSPDSInjectPositionData_t, 523
pAltitudeWrtEllipsoid, 524
pAltitudeWrtSealevel, 524
pHorizontalConfidence, 524
pHorizontalUncCircular, 524
pLatitude, 524
pLongitude, 524
pPositionSource, 524
pTimeStamp, 524
pTimeType, 524
pVerticalConfidence, 524
pVerticalUnc, 524
pack_pds_SLQSSetAGPSConfig
pds.h, 1522
pack_pds_SLQSSetAGPSConfig_t, 525
pNetworkMode, 525
pServerAddress, 525
pServerPort, 525
pServerURL, 525
pack_pds_SLQSSetPositionMethodState
pds.h, 1522
pack_pds_SLQSSetPositionMethodState_t, 525
pWifiState, 526
pXtraDataState, 526
pXtraTimeState, 526
pack_pds_SetEventReportCallback
pds.h, 1517
pack_pds_SetEventReportCallback_t, 518
posDataNmea, 518
rptPosData, 518
pack_pds_SetPDSDefaults
pds.h, 1518
pack_pds_SetPDSDefaults_t, 518
accuracy, 519
interval, 519
operation, 519
timeout, 519
pack_pds_SetPDSState
pds.h, 1518
pack_pds_SetPDSState_t, 519
enable, 519
pack_pds_SetPortAutomaticTracking
pds.h, 1518
pack_pds_SetPortAutomaticTracking_t, 519
bAuto, 520
pack_pds_SetServiceAutomaticTracking
pds.h, 1519
pack_pds_SetServiceAutomaticTracking_t, 520
bAuto, 520
pack_pds_SetXTRAAutomaticDownload
pds.h, 1519
pack_pds_SetXTRAAutomaticDownload_t, 520
bEnabled, 521
interval, 521

pack_pds_SetXTRANetwork
pds.h, 1519
pack_pds_SetXTRANetwork_t, 521
preference, 521
pack_pds_StartPDSTrackingSessionExt
pds.h, 1522
pack_pds_StartPDSTrackingSessionExt_t, 526
fixAccuracy, 527
fixCount, 527
fixInterval, 527
fixTimeout, 527
sessionControl, 527
sessionOperation, 527
sessionServerOption, 527
sessionType, 527
pack_pds_StopPDSTrackingSession
pds.h, 1523
pack_qmi_t, 527
msgid, 528
svc, 528
timeout, 528
xid, 528
pack_qos_BindDataPort
qos.h, 1597
pack_qos_BindDataPort_t, 528
pMuxID, 529
pPeripheralEndPointID, 529
pSIODataPort, 529
pack_qos_SLQSQosGetNetworkStatus
qos.h, 1598
pack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 1599
pack_qos_SLQSQosSwiReadApnExtraParams_t, 529
apnId, 529
pack_qos_SLQSQosSwiReadDataStats
qos.h, 1599
pack_qos_SLQSQosSwiReadDataStats_t, 529
apnId, 529
pack_qos_SLQSSetQosEventCallback
qos.h, 1600
pack_qos_SLQSSetQosEventCallback_t, 529
enable, 530
pack_rms_GetSMSWake
rms.h, 1606
pack_rms_SetSMSWake
rms.h, 1606
pack_rms_SetSMSWake_t, 530
enabled, 530
wake_mask, 530
pack_sar_SLQSGetRfSarState
sar.h, 1608
pack_sar_SLQSSetRfSarState
sar.h, 1608
pack_sar_SLQSSetRfSarState_t, 530
RfSarState, 531
pack_sms_GetSMSCAddress
sms.h, 1613
pack_sms_SLQSDeleteSMS
sms.h, 1615
pack_sms_SLQSDeleteSMS_t, 533
pMessageIndex, 534
pMessageMode, 534
pMessageTag, 534
storageType, 534
pack_sms_SLQSGetIndicationRegister
sms.h, 1615
pack_sms_SLQSGetMessageWaiting
sms.h, 1616
pack_sms_SLQSGetSMS
sms.h, 1616
pack_sms_SLQSGetSMS_t, 534
messageIndex, 535
pMessageMode, 535
storageType, 535
pack_sms_SLQSGetSMSList
sms.h, 1617
pack_sms_SLQSGetSMSList_t, 535
pMessageMode, 536
pRequestedTag, 536
storageType, 536
pack_sms_SLQSGetSmsBroadcastConfig
sms.h, 1616
pack_sms_SLQSGetSmsBroadcastConfig_t, 535
mode, 535
pack_sms_SLQSGetTransLayerInfo
sms.h, 1617
pack_sms_SLQSGetTransNWRegInfo
sms.h, 1618
pack_sms_SLQSModifySMSStatus
sms.h, 1618
pack_sms_SLQSModifySMSStatus_t, 536
messageIndex, 537
messageTag, 537
pMessageMode, 537
storageType, 537
pack_sms_SLQSSendAsyncSMS
sms.h, 1618
pack_sms_SLQSSendAsyncSMS_t, 537
pSendSmsParams, 537
pack_sms_SLQSSetIndicationRegister
sms.h, 1619
pack_sms_SLQSSetIndicationRegister_t, 537
pSetIndicationRegReq, 538
pack_sms_SLQSSetSmsBroadcastActivation
sms.h, 1619
pack_sms_SLQSSetSmsBroadcastActivation_t, 538
broadcastActivate, 538
mode, 538
pack_sms_SLQSSetSmsBroadcastConfig
sms.h, 1619
pack_sms_SLQSSetSmsBroadcastConfig_t, 538
mode, 539
pBroadcastConfig, 539
pCDMABroadcastConfig, 539
pack_sms_SLQSSetSmsStorage
sms.h, 1620

pack_sms_SLQSSetSmsStorage_t, 539
smsStorage, 539
pack_sms_SLQSSmsGetMaxStorageSize
sms.h, 1620
pack_sms_SLQSSmsGetMaxStorageSize_t, 540
pMaxStorageSizeReq, 540
pack_sms_SLQSSmsGetMessageProtocol
sms.h, 1620
pack_sms_SLQSSmsSetRoutes
sms.h, 1621
pack_sms_SLQSSmsSetRoutes_t, 540
pSetRoutesReq, 540
pack_sms_SLQSSwiGetSMSStorage
sms.h, 1621
pack_sms_SaveSMS
sms.h, 1614
pack_sms_SaveSMS_t, 531
messageFormat, 532
messageSize, 532
pMessage, 532
storageType, 532
pack_sms_SendSMS
sms.h, 1614
pack_sms_SendSMS_t, 532
messageFormat, 532
messageSize, 532
pLinktimer, 532
pMessage, 533
pack_sms_SetNewSMSCallback
sms.h, 1614
pack_sms_SetNewSMSCallback_t, 533
status, 533
pack_sms_SetSMSCAddress
sms.h, 1615
pack_sms_SetSMSCAddress_t, 533
pSMSCAddress, 533
pSMSCType, 533
pack_swiaudio_SLQSGetM2MAVMute
swiaudio.h, 1634
pack_swiaudio_SLQSGetM2MAVMute_t, 541
Profile, 542
pack_swiaudio_SLQSGetM2MAudioProfile
swiaudio.h, 1633
pack_swiaudio_SLQSGetM2MAudioProfile_t, 540
pGenerator, 541
pack_swiaudio_SLQSGetM2MAudioVolume
swiaudio.h, 1634
pack_swiaudio_SLQSGetM2MAudioVolume_t, 541
Generator, 541
Profile, 541
pack_swiaudio_SLQSGetM2MSpkrGain
swiaudio.h, 1635
pack_swiaudio_SLQSGetM2MSpkrGain_t, 542
Profile, 542
pack_swiaudio_SLQSSetM2MAVMute
swiaudio.h, 1637
pack_swiaudio_SLQSSetM2MAVMute_t, 545
EarMute, 546
MicMute, 546
pCwtMute, 546
Profile, 546
pack_swiaudio_SLQSSetM2MAudioAVCFG
swiaudio.h, 1635
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 542
Device, 543
Profile, 543
pack_swiaudio_SLQSSetM2MAudioLPBK
swiaudio.h, 1635
pack_swiaudio_SLQSSetM2MAudioLPBK_t, 543
Enable, 543
pack_swiaudio_SLQSSetM2MAudioNVDef
swiaudio.h, 1636
pack_swiaudio_SLQSSetM2MAudioProfile
swiaudio.h, 1636
pack_swiaudio_SLQSSetM2MAudioProfile_t, 543
pCwtMute, 544
pEarMute, 544
pGenerator, 544
pMicMute, 544
pVolume, 544
Profile, 544
pack_swiaudio_SLQSSetM2MAudioVolume
swiaudio.h, 1636
pack_swiaudio_SLQSSetM2MAudioVolume_t, 545
Generator, 545
Level, 545
Profile, 545
pack_swiaudio_SLQSSetM2MSpkrGain
swiaudio.h, 1637
pack_swiaudio_SLQSSetM2MSpkrGain_t, 546
Profile, 546
Value, 546
pack_swiavms_SLQSAVMSGetSettings
swiavms.h, 1644
pack_swiavms_SLQSAVMSGetSettings_v2
swiavms.h, 1644
pack_swiavms_SLQSAVMSSendSelection
swiavms.h, 1645
pack_swiavms_SLQSAVMSSendSelection_t, 547
pDeferTime, 547
pPackageID, 547
pRejectReason, 547
selection, 547
pack_swiavms_SLQSAVMSSessionGetInfo
swiavms.h, 1646
pack_swiavms_SLQSAVMSSetSettings
swiavms.h, 1647
pack_swiavms_SLQSAVMSSetSettings_t, 547
AutoConnect, 549
AutoReboot, 549
pAPNInfo, 549
pConnectionRetryTimers, 549
pNotifStore, 549
pPeriodInfo, 549
pPollingTimer, 549
PromptFwDownload, 549

PromptFwUpdate, 549
pack_swiavms_SLQSAVMSSetSettings_v2
swiavms.h, 1647
pack_swiavms_SLQSAVMSSetSettings_v2_t, 549
AutoConnect, 550
pAutoReboot, 551
pNotifStore, 551
pPeriodInfo, 551
pPollingTimer, 551
PromptFwDownload, 551
PromptFwUpdate, 551
pack_swiavms_SLQSAVMSSetSettingsNoAutoReboot-
Field
swiavms.h, 1648
pack_swiavms_SLQSAVMSStartSession
swiavms.h, 1649
pack_swiavms_SLQSAVMSStartSession_t, 551
sessionType, 551
pack_swiavms_SLQSAVMSStopSession
swiavms.h, 1649
pack_swiavms_SLQSAVMSStopSession_t, 551
sessionType, 552
pack_swiavms_SLQSAvmsSetEventReport
swiavms.h, 1646
pack_swidms_SLQSSwiDmsGetHWWatchdog
swidms.h, 1658
pack_swidms_SLQSSwiDmsGetMTU
swidms.h, 1659
pack_swidms_SLQSSwiDmsGetSecureInfo
swidms.h, 1659
pack_swidms_SLQSSwiDmsGetUimSlot
swidms.h, 1659
pack_swidms_SLQSSwiDmsGetUsbComp
swidms.h, 1660
pack_swidms_SLQSSwiDmsGetUsbNetNum
swidms.h, 1660
pack_swidms_SLQSSwiDmsSetHWWatchdog
swidms.h, 1660
pack_swidms_SLQSSwiDmsSetHWWatchdog_t, 552
enable, 552
resetDelay, 552
timeout, 552
pack_swidms_SLQSSwiDmsSetMTU
swidms.h, 1661
pack_swidms_SLQSSwiDmsSetMTU_t, 552
MTUSize, 553
pack_swidms_SLQSSwiDmsSetUimSlot
swidms.h, 1661
pack_swidms_SLQSSwiDmsSetUimSlot_t, 553
pAutoSwitchMode, 554
pMultiplexedSimMode, 554
pSlotSelect, 554
slotSelectaction, 554
pack_swidms_SLQSSwiDmsSetUsbComp
swidms.h, 1661
pack_swidms_SLQSSwiDmsSetUsbComp_t, 554
CfgValue, 555
pack_swidms_SLQSSwiDmsSetUsbNetNum
swidms.h, 1662
pack_swidms_SLQSSwiDmsSetUsbNetNum_t, 555
nUsbNetNum, 555
pack_swiloc_SwiLocGetAutoStart
swiloc.h, 1667
pack_swiloc_SwiLocSetAutoStart
swiloc.h, 1667
pack_swiloc_SwiLocSetAutoStart_t, 556
fix_rate, 557
fix_type, 557
function, 557
max_dist, 557
max_time, 557
set_fix_rate, 557
set_fix_type, 557
set_function, 557
set_max_dist, 557
set_max_time, 557
pack_swioma_SLQSOMADMAlertCallback
swioma.h, 1669
pack_swioma_SLQSOMADMCancelSession
swioma.h, 1670
pack_swioma_SLQSOMADMCancelSession_t, 557
sessionType, 558
pack_swioma_SLQSOMADMCancelSessionExt
swiomaext.h, 1679
pack_swioma_SLQSOMADMCancelSessionExt_t, 558
sessionType, 558
pack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 1671
pack_swioma_SLQSOMADMGetSessionInfo_t, 558
SessionType, 559
pack_swioma_SLQSOMADMGetSessionInfoExt
swiomaext.h, 1679
pack_swioma_SLQSOMADMGetSettings
swioma.h, 1671
pack_swioma_SLQSOMADMSendSelection
swioma.h, 1672
pack_swioma_SLQSOMADMSendSelection_t, 559
pDeferTime, 559
pRejectReason, 559
selection, 559
pack_swioma_SLQSOMADMSendSelectionExt
swiomaext.h, 1680
pack_swioma_SLQSOMADMSendSelectionExt_t, 559
selection, 560
pack_swioma_SLQSOMADMSetSettings
swioma.h, 1672
pack_swioma_SLQSOMADMSetSettings_t, 560
FOTAdownload, 561
pAutosdm, 561
pFwAutoCheck, 561
pack_swioma_SLQSOMADMSetSettingsExt
swiomaext.h, 1680
pack_swioma_SLQSOMADMSetSettingsExt_t, 561
pack_swioma_SLQSOMADMStartSession
swioma.h, 1673
pack_swioma_SLQSOMADMStartSession_t, 562

sessionType, 562
pack_swioma_SLQSOMADMStartSessionExt
swiomaext.h, 1681
pack_swioma_SLQSOMADMStartSessionExt_t, 562
sessionType, 563
pack_tmd_SLQSTmdDeRegNotMitigationLvl
tmd.h, 1687
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 563
mitigationDevID, 563
mitigationDevIDLen, 563
pack_tmd_SLQSTmdGetMitigationDevList
tmd.h, 1687
pack_tmd_SLQSTmdGetMitigationLvl
tmd.h, 1687
pack_tmd_SLQSTmdGetMitigationLvl_t, 563
mitigationDevID, 564
mitigationDevIDLen, 564
pack_tmd_SLQSTmdRegNotMitigationLvl
tmd.h, 1688
pack_tmd_SLQSTmdRegNotMitigationLvl_t, 564
mitigationDevID, 564
mitigationDevIDLen, 564
pack_uim_ChangePin
uim.h, 1696
pack_uim_ChangePin_t, 564
changePIN, 565
EncryptedPIN1, 565
pIndicationToken, 565
pKeyReferenceID, 565
sessionInfo, 565
Tlvresult, 565
pack_uim_GetCardStatus
uim.h, 1696
pack_uim_ReadTransparent
uim.h, 1697
pack_uim_ReadTransparent_t, 565
fileIndex, 566
pEncryptData, 566
pIndicationToken, 566
readTransparent, 566
sessionInfo, 566
Tlvresult, 566
pack_uim_SLQSUIMAddProfile
uim.h, 1698
pack_uim_SLQSUIMAddProfile_t, 568
pProfileConfInfo, 568
pUserConsent, 568
profileActInfo, 568
slot, 568
pack_uim_SLQSUIMAuthenticate
uim.h, 1698
pack_uim_SLQSUIMAuthenticate_t, 568
authData, 569
pIndicationToken, 569
sessionInfo, 569
pack_uim_SLQSUIMDeleteProfile
uim.h, 1698
pack_uim_SLQSUIMDeleteProfile_t, 569
profileId, 569
slot, 569
pack_uim_SLQSUIMDepersonalization
uim.h, 1699
pack_uim_SLQSUIMDepersonalization_t, 570
depersonilisationInfo, 570
pack_uim_SLQSUIMEventRegister
uim.h, 1699
pack_uim_SLQSUIMEventRegister_t, 571
eventMask, 571
pack_uim_SLQSUIMGetConfiguration
uim.h, 1700
pack_uim_SLQSUIMGetConfiguration_t, 571
pConfigurationMask, 572
pack_uim_SLQSUIMGetEID
uim.h, 1700
pack_uim_SLQSUIMGetEID_t, 572
pUimSlotID, 572
pack_uim_SLQSUIMGetFileAttributes
uim.h, 1700
pack_uim_SLQSUIMGetFileAttributes_t, 572
fileIndex, 573
pIndicationToken, 573
sessionInfo, 573
pack_uim_SLQSUIMGetProfileInfo
uim.h, 1701
pack_uim_SLQSUIMGetProfileInfo_t, 573
profileId, 574
slotNo, 574
pack_uim_SLQSUIMGetSIMProfile
uim.h, 1702
pack_uim_SLQSUIMGetSIMProfile_t, 575
slotNo, 575
pack_uim_SLQSUIMGetServiceStatus
uim.h, 1701
pack_uim_SLQSUIMGetServiceStatus_t, 574
capMask, 574
sessionInfo, 575
pack_uim_SLQSUIMGetSlotsStatus
uim.h, 1702
pack_uim_SLQSUIMPowerDown
uim.h, 1702
pack_uim_SLQSUIMPowerDown_t, 575
slot, 575
pack_uim_SLQSUIMPowerUp
uim.h, 1703
pack_uim_SLQSUIMPowerUp_t, 576
pIgnoreHotSwapSwitch, 576
slot, 576
pack_uim_SLQSUIMReadRecord
uim.h, 1703
pack_uim_SLQSUIMReadRecord_t, 576
fileIndex, 577
pIndicationToken, 577
pLastRecord, 577
readRecord, 577
sessionInfo, 577
pack_uim_SLQSUIMRefreshComplete

uim.h, 1703
pack_uim_SLQSUIMRefreshComplete_t, 577
refreshComplete, 577
sessionInfo, 578
pack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 1704
pack_uim_SLQSUIMRefreshGetLastEvent_t, 578
sessionInfo, 578
pack_uim_SLQSUIMRefreshOK
uim.h, 1704
pack_uim_SLQSUIMRefreshOK_t, 578
OKtoRefresh, 578
sessionInfo, 578
pack_uim_SLQSUIMRefreshRegister
uim.h, 1705
pack_uim_SLQSUIMRefreshRegister_t, 579
regRefresh, 579
sessionInfo, 579
pack_uim_SLQSUIMReset
uim.h, 1705
pack_uim_SLQSUIMSetSIMProfile
uim.h, 1706
pack_uim_SLQSUIMSetSIMProfile_t, 580
pProfileStatus, 581
profileId, 581
setSlot, 581
pack_uim_SLQSUIMSetServiceStatus
uim.h, 1705
pack_uim_SLQSUIMSetServiceStatus_t, 579
pFDNStatus, 580
sessionInfo, 580
pack_uim_SLQSUIMSwitchSlot
uim.h, 1706
pack_uim_SLQSUIMSwitchSlot_t, 581
bLogicalSlot, 581
ulPhysicalSlot, 581
pack_uim_SLQSUIMUpdateProfile
uim.h, 1706
pack_uim_SLQSUIMUpdateProfile_t, 581
pNickName, 582
profileId, 582
slot, 582
pack_uim_SLQSUIMWriteRecord
uim.h, 1707
pack_uim_SLQSUIMWriteRecord_t, 582
fileIndex, 583
pIndicationToken, 583
sessionInfo, 583
writeRecord, 583
pack_uim_SLQSUIMWriteTransparent
uim.h, 1707
pack_uim_SLQSUIMWriteTransparent_t, 583
fileIndex, 584
pIndicationToken, 584
sessionInfo, 584
writeTransparent, 584
pack_uim_SLQSUIMeUICCMemReset
uim.h, 1699
pack_uim_SLQSUIMeUICCMemReset_t, 570
uimSlot, 571
pack_uim_SetPinProtection
uim.h, 1697
pack_uim_SetPinProtection_t, 567
EncryptedPIN1, 567
pIndicationToken, 567
pKeyReferenceID, 567
pinProtection, 567
sessionInfo, 567
Tlvresult, 567
pack_uim_UnblockPin
uim.h, 1708
pack_uim_UnblockPin_t, 584
EncryptedPIN1, 584
pIndicationToken, 585
pKeyReferenceID, 585
pinProtection, 585
sessionInfo, 585
Tlvresult, 585
pack_uim_VerifyPin
uim.h, 1708
pack_uim_VerifyPin_t, 585
pEncryptedPIN1, 586
pIndicationToken, 586
pKeyReferenceID, 586
sessionInfo, 586
Tlvresult, 586
verifyPIN, 586
pack_voice_AnswerUSSD
voice.h, 1729
pack_voice_AnswerUSSD_t, 586
pInfo, 586
pack_voice_CancelUSSD
voice.h, 1729
pack_voice_OriginateUSSD
voice.h, 1729
pack_voice_OriginateUSSD_t, 586
pInfo, 587
pack_voice_SLQSOriginateUSSD
voice.h, 1730
pack_voice_SLQSOriginateUSSD_t, 587
ussDCS, 587
ussData, 587
ussLen, 587
pack_voice_SLQSVoiceALSSelectLine
voice.h, 1730
pack_voice_SLQSVoiceALSSelectLine_t, 587
lineValue, 588
pack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 1730
pack_voice_SLQSVoiceALSSetLineSwitching_t, 588
switchOption, 588
pack_voice_SLQSVoiceAnswerCall
voice.h, 1731
pack_voice_SLQSVoiceAnswerCall_t, 588
pCallId, 589
pack_voice_SLQSVoiceBindSubscription

voice.h, 1731
pack_voice_SLQSVoiceBindSubscription_t, 589
subsType, 589
pack_voice_SLQSVoiceBurstDTMF
voice.h, 1731
pack_voice_SLQSVoiceBurstDTMF_t, 589
pack_voice_SLQSVoiceDialCall
voice.h, 1732
pack_voice_SLQSVoiceDialCall_t, 590
callNumber, 591
pCLIRType, 591
pCUGInfo, 591
pCallPartySubAdd, 591
pCallType, 591
pEmergencyCategory, 591
pSvcType, 591
pUUSInFo, 591
pack_voice_SLQSVoiceEndCall
voice.h, 1732
pack_voice_SLQSVoiceEndCall_t, 591
pCallId, 592
pack_voice_SLQSVoiceGetAllCallInfo
voice.h, 1733
pack_voice_SLQSVoiceGetCLIP
voice.h, 1734
pack_voice_SLQSVoiceGetCLIR
voice.h, 1735
pack_voice_SLQSVoiceGetCNAP
voice.h, 1735
pack_voice_SLQSVoiceGetCOLP
voice.h, 1735
pack_voice_SLQSVoiceGetCOLR
voice.h, 1736
pack_voice_SLQSVoiceGetCallBarring
voice.h, 1733
pack_voice_SLQSVoiceGetCallBarring_t, 592
pSvcClass, 592
reason, 592
pack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 1733
pack_voice_SLQSVoiceGetCallForwardingStatus_t, 593
pSvcClass, 593
Reason, 593
pack_voice_SLQSVoiceGetCallInfo
voice.h, 1734
pack_voice_SLQSVoiceGetCallInfo_t, 593
callID, 594
pack_voice_SLQSVoiceGetCallWaiting
voice.h, 1734
pack_voice_SLQSVoiceGetCallWaiting_t, 594
pSvcClass, 594
pack_voice_SLQSVoiceGetConfig
voice.h, 1736
pack_voice_SLQSVoiceGetConfig_t, 594
pAMRStatus, 595
pAirTimer, 595
pAutoAnswer, 596
pNamID, 596
pPrefVoicePrivacy, 596
pPrefVoiceSO, 596
pRoamTimer, 596
pTTYMode, 596
pVoiceDomainPref, 596
pack_voice_SLQSVoiceIndicationRegister
voice.h, 1736
pack_voice_SLQSVoiceIndicationRegister_t, 596
pRegDTMFEvents, 597
pRegVoicePrivacyEvents, 597
pSuppsNotifEvents, 597
pack_voice_SLQSVoiceManageCalls
voice.h, 1737
pack_voice_SLQSVoiceManageCalls_t, 597
pCallID, 598
SUPSType, 598
pack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 1737
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 598
pack_voice_SLQSVoiceSendFlash
voice.h, 1738
pack_voice_SLQSVoiceSendFlash_t, 598
pCallID, 599
pFlashPayLd, 599
pFlashType, 599
pack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 1738
pack_voice_SLQSVoiceSetCallBarringPassword_t, 599
newPasswd, 600
newPasswdAgain, 600
oldPasswd, 600
Reason, 600
pack_voice_SLQSVoiceSetConfig
voice.h, 1738
pack_voice_SLQSVoiceSetConfig_t, 600
pAirTimerConfig, 601
pAutoAnswer, 601
pPrefVoiceDomain, 601
pPrefVoiceSO, 601
pRoamTimerConfig, 601
pTTYMode, 601
pack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 1739
pack_voice_SLQSVoiceSetPreferredPrivacy_t, 601
privacyPref, 601
pack_voice_SLQSVoiceSetSUPSService
voice.h, 1739
pack_voice_SLQSVoiceSetSUPSService_t, 601
pCallBarringPasswd, 604
pCallForwardingNumber, 604
pServiceClass, 604
pTimerVal, 604
reason, 604
voiceSvc, 604
pack_voice_SLQSVoiceStartContDTMF
voice.h, 1739
pack_voice_SLQSVoiceStartContDTMF_t, 604
DTMFdigit, 604

pCallID, 604
pack_voice_SLQSVoiceStopContDTMF
voice.h, 1740
pack_voice_SLQSVoiceStopContDTMF_t, 605
callID, 605
pack_wda_GetDataFormat
wda.h, 1755
pack_wda_GetDataFormat_t, 605
ep_type, 606
iface_id, 606
pack_wda_GetQmapSettings
wda.h, 1755
pack_wda_GetQmapSettings_t, 606
pEp_type, 606
pIface_id, 606
pack_wda_SetDataFormat
wda.h, 1756
pack_wda_SetDataFormat_t, 606
pDl_data_aggregation_max_datagrams, 609
pDl_data_aggregation_max_size, 609
pDl_data_aggregation_protocol, 609
pDl_minimum_padding, 609
pEp_type, 609
pFlow_control, 609
pIface_id, 609
pNdp_signature, 609
pQlink_prot, 609
pQos_format, 609
pQos_header_format, 609
pTcp_coalescing, 609
pUdp_coalescing, 609
pUl_data_aggregation_max_datagrams, 609
pUl_data_aggregation_max_size, 609
pUl_data_aggregation_protocol, 609
pack_wda_SetQmapSettings
wda.h, 1756
pack_wda_SetQmapSettings_t, 609
pEp_type, 610
pIface_id, 610
pQmap_dfc, 610
pack_wds_DHCPv4ClientLeaseChange
wds.h, 1768
pack_wds_DHCPv4ClientLeaseChange_t, 610
pEnableNotification, 611
pack_wds_GetAutoconnect
wds.h, 1769
pack_wds_GetByteTotals
wds.h, 1769
pack_wds_GetConnectionRate
wds.h, 1769
pack_wds_GetDataBearerTechnology
wds.h, 1770
pack_wds_GetDefaultProfile
wds.h, 1770
pack_wds_GetDefaultProfile_t, 611
profiletype, 611
pack_wds_GetDefaultProfileNum
wds.h, 1771
pack_wds_GetDefaultProfileNum_t, 611
family, 612
type, 612
pack_wds_GetDefaultProfileV2
wds.h, 1771
pack_wds_GetDefaultProfileV2_t, 612
profiletype, 612
pack_wds_GetDormancyState
wds.h, 1771
pack_wds_GetDormancyState_t, 612
pack_wds_GetLastMobileIPError
wds.h, 1772
pack_wds_GetLastMobileIPError_t, 612
pack_wds_GetMobileIP
wds.h, 1772
pack_wds_GetMobileIP_t, 613
pack_wds_GetMobileIPProfile
wds.h, 1773
pack_wds_GetMobileIPProfile_t, 613
index, 613
pack_wds_GetPacketStatistics
wds.h, 1773
pack_wds_GetPacketStatistics_t, 613
pStatMask, 614
pack_wds_GetPacketStatus
wds.h, 1774
pack_wds_GetPacketStatus_t, 614
statmask, 614
pack_wds_GetSessionDuration
wds.h, 1774
pack_wds_GetSessionDuration_t, 614
pack_wds_GetSessionDurationV2
wds.h, 1774
pack_wds_GetSessionState
wds.h, 1775
pack_wds_RMSetTransferStatistics
wds.h, 1775
pack_wds_RMSetTransferStatistics_t, 615
RmTrasnferStaticsReq, 615
pack_wds_SLQSCreateProfile
wds.h, 1779
pack_wds_SLQSCreateProfile_t, 622
pCurProfile, 622
pProfileId, 622
pProfileType, 622
pack_wds_SLQSDeleteProfile
wds.h, 1779
pack_wds_SLQSDeleteProfile_t, 622
profileIndex, 623
profileType, 623
pack_wds_SLQSGet3GPPConfigItem
wds.h, 1779
pack_wds_SLQSGetCurrDataSystemStat
wds.h, 1780
pack_wds_SLQSGetCurrDataSystemStat_t, 623
pack_wds_SLQSGetCurrentChannelRate
wds.h, 1780
pack_wds_SLQSGetDUNCallInfo

wds.h, 1781
pack_wds_SLQSGetDUNCallInfo_t, 623
Mask, 624
pReportChannelRate, 625
pReportConnStatus, 625
pReportDataBearerTech, 625
pReportDormStatus, 625
pTransferStatInd, 625
pack_wds_SLQSGetDataBearerTechnology
wds.h, 1781
pack_wds_SLQSGetDataBearerTechnology_t, 623
pack_wds_SLQSGetProfileSettings
wds.h, 1782
pack_wds_SLQSGetProfileSettings_t, 625
ProfileId, 625
ProfileType, 625
pack_wds_SLQSGetProfileSettingsV2
wds.h, 1782
pack_wds_SLQSGetRuntimeSettings
wds.h, 1782
pack_wds_SLQSGetRuntimeSettings_t, 626
pReqSettings, 626
pack_wds_SLQSModifyProfile
wds.h, 1783
pack_wds_SLQSModifyProfile_t, 626
curProfile, 627
pProfileId, 627
pProfileType, 627
pack_wds_SLQSResetPacketStatics
wds.h, 1783
pack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 1785
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 630
pProfileId, 630
pack_wds_SLQSSGetLoopback
wds.h, 1785
pack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 1786
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 630
pHwConfig, 631
pProfileId, 631
pack_wds_SLQSSSetLoopback
wds.h, 1786
pack_wds_SLQSSSetLoopback_t, 631
loopbackMode, 632
loopbackMultiplier, 632
pack_wds_SLQSSet3GPPConfigItem
wds.h, 1784
pack_wds_SLQSSet3GPPConfigItem_t, 627
p3gppRelease, 629
pProfileList, 629
pack_wds_SLQSSetIPFamilyPreference
wds.h, 1784
pack_wds_SLQSSetIPFamilyPreference_t, 629
IPFamilyPreference, 629
pack_wds_SLQSSetWdsEventCallback
wds.h, 1784
pack_wds_SLQSSetWdsEventCallback_t, 629
currentDataBearer, 630
dataBearer, 630
dataBearerTechExt, 630
dataSystemStatus, 630
dormancyStatus, 630
interval, 630
mobileIP, 630
transferStats, 630
pack_wds_SLQSStartDataSession
wds.h, 1786
pack_wds_SLQSStartDataSession_t, 632
pAuth, 633
pPass, 633
pTech, 633
pUser, 633
pprofileid3gpp, 633
pprofileid3gpp2, 633
pack_wds_SLQSStopDataSession
wds.h, 1787
pack_wds_SLQSStopDataSession_t, 633
psid, 633
pack_wds_SLQSSwiProfileChangeCallback
wds.h, 1787
pack_wds_SLQSSwiProfileChangeCallback_t, 633
pProfileChangeInd, 633
pack_wds_SLQSWdsGoActive
wds.h, 1788
pack_wds_SLQSWdsGoDormant
wds.h, 1788
pack_wds_SLQSWdsSetEventReport
wds.h, 1789
pack_wds_SLQSWdsSetEventReport_t, 634
pCurrChannelRateInd, 635
pCurrDataBearerTechInd, 635
pCurrPrefDataSysInd, 635
pDataBearerTechInd, 635
pDataCallStatusChangeInd, 635
pDataSystemStatusChangeInd, 635
pDormancyStatusInd, 635
pMIPStatusInd, 635
pTransferStatInd, 635
pack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 1789
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 635
contextId, 636
contextType, 636
pack_wds_SetAutoconnect
wds.h, 1776
pack_wds_SetAutoconnect_t, 615
acroamsetting, 616
acsetting, 616
pack_wds_SetDefaultProfile
wds.h, 1776
pack_wds_SetDefaultProfile_t, 616
authentication, 617
ipAddress, 617
pApnname, 617
pName, 617

pPassword, 617
pUsername, 617
pdpType, 617
primaryDNS, 617
profileType, 617
secondaryDNS, 617
pack_wds_SetDefaultProfileNum
wds.h, 1776
pack_wds_SetDefaultProfileNum_t, 618
family, 618
index, 618
type, 618
pack_wds_SetMobileIP
wds.h, 1777
pack_wds_SetMobileIP_t, 618
mode, 618
pack_wds_SetMobileIPParameters
wds.h, 1777
pack_wds_SetMobileIPParameters_t, 619
pHA2002bis, 620
pHAAuthenticator, 620
pMode, 620
pReRegPeriod, 620
pReRegTraffic, 620
pRetryInterval, 620
pRetryLimit, 620
pSPC, 620
pack_wds_SetMobileIPProfile
wds.h, 1778
pack_wds_SetMobileIPProfile_t, 620
index, 621
pAAASPI, 621
pAddress, 621
pEnabled, 621
pHASPI, 621
pMNAAA, 621
pMNHA, 621
pNAI, 621
pPrimaryHA, 621
pRevTunneling, 621
pSecondaryHA, 621
spc, 622
pack_wds_SetMuxID
wds.h, 1778
PackCreateProfileOut, 636
ExtErrorCode, 636
ProfileIndex, 636
ProfileType, 636
PackSwiAVMSSettingsAPNInfo, 640
bAPNLength, 640
bPWDLength, 640
bUnameLength, 640
pAPN, 640
pPWD, 640
pUname, 640
PackSwiAVMSSettingsConnectionRetryTimers, 640
Timers, 641
PackSwiAVMSSettingsPeriodsInfo, 641
max, 641
min, 641
PackSwiAvmsSetSettingsAPNInfo, 638
bAPNLength, 638
bPWDLength, 638
bUnameLength, 638
szAPN, 638
szPWD, 638
szUname, 638
PackSwiAvmsSetSettingsConnectionRetryTimers, 638
Timers, 639
PackSwiAvmsSetSettingsPeriodInfo, 639
ulMax, 639
ulMin, 639
package_name
unpack_omaDmFotaTlv_t, 972
packageIdStr
_litefw_FirmwareFileInfo, 40
packageid_str
unpack_dms_GetFirmwareInfo_t, 753
packetZone
nas_CDMASysInfo, 207
packetZoneValid
nas_CDMASysInfo, 207
packgetDyingGaspCfg, 637
pDestSMSContent, 637
pDestSMSNum, 637
packgetDyingGaspStatistics, 637
pSMSAttemptedFlag, 638
pTimeStamp, 638
ParamPresenceMask
pack_dms_UIMGetICCID_t, 430
unpack_audio_SLQSGetAudioPathConfig_t, 731
unpack_audio_SLQSGetAudioProfile_t, 732
unpack_audio_SLQSGetAudioVolTLBConfig_t, 733
unpack_audio_SLQSSetAudioVolTLBConfig_t, 733
unpack_cat_SetCatEventCallback_ind_t, 734
unpack_cat_SetCATEventCallback_t, 735
unpack_ctl_GetClientId_t, 736
unpack_ctl_ReleaseClientId_t, 736
unpack_dms_GetActivationState_t, 737
unpack_dms_GetBandCapability_t, 739
unpack_dms_GetCrashAction_t, 739
unpack_dms_GetCustFeature_t, 742
unpack_dms_GetCustFeaturesV2_t, 743
unpack_dms_GetDeviceCap_t, 744
unpack_dms_GetDeviceCapabilities_t, 746
unpack_dms_GetDeviceCapabilitiesV2_t, 748
unpack_dms_GetDeviceHardwareRev_t, 749
unpack_dms_GetDeviceMfr_t, 750
unpack_dms_GetDeviceSerialNumbers_t, 752
unpack_dms_GetFirmwareInfo_t, 753
unpack_dms_GetFirmwareRevision_t, 754
unpack_dms_GetFirmwareRevisions_t, 756
unpack_dms_GetFSN_t, 756
unpack_dms_GetHardwareRevision_t, 757

unpack_dms_GetIMSI_t, 758
unpack_dms_GetManufacturer_t, 758
unpack_dms_GetModelID_t, 759
unpack_dms_GetNetworkTime_t, 760
unpack_dms_GetNetworkTimeV2_t, 761
unpack_dms_GetOfflineReason_t, 762
unpack_dms_GetPower_t, 763
unpack_dms_GetPRLVersion_t, 764
unpack_dms_GetSerialNumbers_t, 765
unpack_dms_GetUSBComp_t, 767
unpack_dms_GetVoiceNumber_t, 768
unpack_dms_PSMCfgChange_ind_t, 769
unpack_dms_ResetToFactoryDefaults_t, 770
unpack_dms_SetActivationStatusCallback_t, 770
unpack_dms_SetCrashAction_t, 771
unpack_dms_SetCustFeature_t, 771
unpack_dms_SetCustFeaturesV2_t, 772
unpack_dms_SetEventReport_ind_t, 773
unpack_dms_SetEventReport_t, 773
unpack_dms_SetFirmwarePreference_t, 774
unpack_dms_SetIndicationRegister_t, 774
unpack_dms_SetPower_t, 775
unpack_dms_SetUSBComp_t, 775
unpack_dms_SLQSDmsSwiGetPCInfo_t, 778
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 779
unpack_dms_SLQSDmsSwiGetResetInfo_t, 780
unpack_dms_SLQSDmsSwiGetUimSelection_t, 781
unpack_dms_SLQSDmsSwiIndicationRegister_t, 782
unpack_dms_SLQSGetBandCapability_t, 786
unpack_dms_SLQSGetBandCapabilityExt_t, 790
unpack_dms_SLQSGetERIFile_t, 791
unpack_dms_SLQSGetPowerSaveModeConfig_t, 792
unpack_dms_SLQSSetPowerSaveModeConfig_t, 793
unpack_dms_SLQSSwiClearDyingGaspStatistics_t, 793
unpack_dms_SLQSSwiGetCrashInfo_t, 794
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 795
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 795
unpack_dms_SLQSSwiGetFirmwareCurr_t, 796
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 798
unpack_dms_SLQSSwiGetHostDevInfo_t, 799
unpack_dms_SLQSSwiGetOSInfo_t, 800
unpack_dms_SLQSSwiGetSerialNoExt_t, 801
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 801
unpack_dms_SLQSSwiSetHostDevInfo_t, 802
unpack_dms_SLQSSwiSetOSInfo_t, 802
unpack_dms_SLQSUIMGetState_t, 803
unpack_dms_SwiEventReportCallBack_ind_t, 804
unpack_dms_SwiSetEventReport_t, 804
unpack_dms_SwiUimSelect_t, 805
unpack_dms_UIMGetControlKeyStatus_t, 806
unpack_dms_UIMGetICCID_t, 806
unpack_dms_UIMGetPINStatus_t, 808
unpack_dms_UIMSetControlKeyProtection_t, 809
unpack_dms_UIMSetPINProtection_t, 810
unpack_dms_UIMUnblockControlKey_t, 810
unpack_dsd_SLQSGetDsdUIInfo_t, 811
unpack_dsd_SLQSGetSystemStatus_t, 812
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
unpack_dsd_SLQSUIInfo_Ind_t, 815
unpack_fms_GetImagesPreference_t, 816
unpack_fms_GetStoredImages_t, 817
unpack_fms_SetImagesPreference_t, 817
unpack_ims_SLQSGetIMSSMSConfig_t, 818
unpack_ims_SLQSGetIMSUserConfig_t, 819
unpack_ims_SLQSGetIMSVoIPConfig_t, 822
unpack_ims_SLQSGetRegMgrConfig_t, 823
unpack_ims_SLQSGetSIPConfig_t, 824
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 825
unpack_ims_SLQSSetIMSSMSConfig_t, 826
unpack_ims_SLQSSetIMSUserConfig_t, 826
unpack_ims_SLQSSetIMSVoIPConfig_t, 827
unpack_ims_SLQSSetRegMgrConfig_t, 827
unpack_ims_SLQSSetSIPConfig_t, 828
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
unpack_ims_SLQSSMSCfgCallBack_ind_t, 830
unpack_ims_SLQSUserCfgCallBack_ind_t, 831
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
unpack_imsa_SLQSGetIMSARegStatus_t, 834
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 837
unpack_imsa_SLQSImsaRatStatusCallBack_ind_t, 838
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 839
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
unpack_loc_BestAvailPos_Ind_t, 846
unpack_loc_CradleMountCallback_Ind_t, 848
unpack_loc_Delete_Assist_Data_t, 848
unpack_loc_DeleteAssistData_Ind_t, 849
unpack_loc_EngineState_Ind_t, 850
unpack_loc_EventNMEA_Ind_t, 851
unpack_loc_EventRegister_t, 851
unpack_loc_EventTimeSyncCallback_Ind_t, 852
unpack_loc_FixCriteria_Ind_t, 853
unpack_loc_GetOpMode_Ind_t, 854
unpack_loc_GetServer_Ind_t, 855
unpack_loc_GnssSvInfo_Ind_t, 856
unpack_loc_InjectPositionCallback_Ind_t, 857
unpack_loc_InjectSensorDataCallback_Ind_t, 859
unpack_loc_InjectTimeSyncDataCallback_Ind_t, 860
unpack_loc_InjectUTCTimeCallback_Ind_t, 860
unpack_loc_PositionRpt_Ind_t, 866
unpack_loc_SensorStreamingCallback_Ind_t, 868
unpack_loc_SetExtPowerConfig_Ind_t, 869
unpack_loc_SetExtPowerState_t, 870

unpack_loc_SetOperationMode_Ind_t, 870
unpack_loc_SetOperationMode_t, 871
unpack_loc_SetServer_Ind_t, 872
unpack_loc_SLQSLOCGetBestAvailPos_t, 873
unpack_loc_SLQSLOCGetOpMode_t, 873
unpack_loc_Start_t, 874
unpack_loc_Stop_t, 874
unpack_nas_GetACCOLC_t, 875
unpack_nas_GetANAAAAuthenticationStatus_t, 875
unpack_nas_GetCDMANetworkParameters_t, 877
unpack_nas_GetHomeNetwork3GPP2_t, 880
unpack_nas_GetHomeNetwork_t, 881
unpack_nas_GetNetworkPreference_t, 882
unpack_nas_GetRFInfo_t, 883
unpack_nas_GetServingNetwork_t, 886
unpack_nas_GetServingNetworkCapabilities_t, 887
unpack_nas_GetSignalStrengths_t, 888
unpack_nas_PerformNetworkScan_t, 889
unpack_nas_SetDataCapabilitiesCallback_ind_t, 890
unpack_nas_SetEventReportInd_t, 891
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 892
unpack_nas_SetNetworkPreference_t, 893
unpack_nas_SetRoamingIndicatorCallback_ind_t, 894
unpack_nas_SetServingSystemCallback_ind_t, 894
unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t, 895
unpack_nas_SLQSGetErrorRate_t, 896
unpack_nas_SLQSGetHomeNetwork_t, 897
unpack_nas_SlqsGetLTECphyCAInfo_t, 898
unpack_nas_SLQSGetNetworkTime_t, 899
unpack_nas_SLQSGetOperatorNameData_t, 900
unpack_nas_SLQSGetPLMNName_t, 903
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
unpack_nas_SLQSGetSignalStrength_t, 915
unpack_nas_SLQSGetSysInfo_t, 918
unpack_nas_SLQSGetSysInfoV2_t, 922
unpack_nas_SLQSGetSysSelectionPref_t, 928
unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 934
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t, 935
unpack_nas_SLQSNasGet3GPP2Subscription_t, 937
unpack_nas_SLQSNasGetCellLocationInfo_t, 938
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 940
unpack_nas_SLQSNASGeteDRXParams_t, 941
unpack_nas_SLQSNASGeteDRXParamsExt_t, 943
unpack_nas_SLQSNASGetForbiddenNetworks_t, 943
unpack_nas_SLQSNasGetHDRColorCode_t, 944
unpack_nas_SLQSNasGetSigInfo_t, 946
unpack_nas_SLQSNasGetTxRxInfo_t, 946
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t, 949
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 950
unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t, 951
unpack_nas_SLQSNasRFBandInfoCallback_Ind_t, 952
unpack_nas_SLQSNasSigInfoCallback_ind_t, 954
unpack_nas_SLQSNASSwiGetChannelLock_t, 955
unpack_nas_SLQSNasSwiModemStatus_t, 955
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 956
unpack_nas_SLQSNasTimerCallback_ind_t, 957
unpack_nas_SLQSPerformNetworkScanV2_t, 958
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 959
unpack_nas_SLQSSwiGetHDRPersonality_t, 960
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 961
unpack_nas_SLQSSwiGetHRPDStats_t, 962
unpack_nas_SLQSSwiGetLteCQI_t, 963
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 963
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t, 964
unpack_nas_SLQSSwiNetworkDebug_t, 965
unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t, 966
unpack_nas_SLQSSysInfoCallback_ind_t, 969
unpack_pds_GetPDSDefaults_t, 974
unpack_pds_GetPDSState_t, 975
unpack_pds_GetPortAutomaticTracking_t, 976
unpack_pds_GetServiceAutomaticTracking_t, 976
unpack_pds_GetXTRAAutomaticDownload_t, 977
unpack_pds_GetXTRANetwork_t, 978
unpack_pds_GetXTRAValidity_t, 978
unpack_pds_SetEventReport_Ind_t, 980
unpack_pds_SetPdsState_Ind_t, 981
unpack_pds_SLQSGetAGPSConfig_t, 982
unpack_pds_SLQSGetGPSStateInfo_t, 987
unpack_qos_BindDataPort_t, 988
unpack_qos_SLQSQosGetNetworkStatus_t, 994
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 995
unpack_qos_SLQSQosSwiReadDataStats_t, 997
unpack_qos_SLQSSetQosEventCallback_ind_t, 997
unpack_qos_SLQSSetQosEventCallback_t, 998
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 998
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 999
unpack_qos_SLQSSetQosStatusCallback_ind_t, 1000
unpack_result_t, 1010

unpack_rms_GetSMSWake_t, 1011
unpack_rms_SetSMSWake_t, 1011
unpack_RMTransferStatistics_ind_t, 1012
unpack_sar_SLQSGetRfSarState_t, 1013
unpack_sms_GetSMSCAddress_t, 1014
unpack_sms_SaveSMS_t, 1015
unpack_sms_SendSMS_t, 1015
unpack_sms_SetNewSMSCallback_ind_t, 1017
unpack_sms_SetNewSMSCallback_t, 1017
unpack_sms_SetSMSCAddress_t, 1018
unpack_sms_SLQSDeleteSMS_t, 1018
unpack_sms_SLQSGetIndicationRegister_t, 1019
unpack_sms_SLQSGetMessageWaiting_t, 1019
unpack_sms_SLQSGetSMS_t, 1020
unpack_sms_SLQSGetSmsBroadcastConfig_t, 1021
unpack_sms_SLQSGetSMSList_t, 1022
unpack_sms_SLQSGetTransLayerInfo_t, 1022
unpack_sms_SLQSGetTransNWRegInfo_t, 1023
unpack_sms_SLQSModifySMSStatus_t, 1023
unpack_sms_SLQSNWRegInfoCallback_ind_t, 1024
unpack_sms_SLQSSendAsyncSMS_t, 1024
unpack_sms_SLQSSetIndicationRegister_t, 1025
unpack_sms_SLQSSetSmsBroadcastActivation_t, 1025
unpack_sms_SLQSSetSmsBroadcastConfig_t, 1026
unpack_sms_SLQSSetSmsStorage_t, 1026
unpack_sms_SLQSSmsGetMaxStorageSize_t, 1026
unpack_sms_SLQSSmsGetMessageProtocol_t, 1027
unpack_sms_SLQSSmsSetRoutes_t, 1027
unpack_sms_SLQSSwiGetSMSStorage_t, 1028
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 1029
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t, 1031
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 1032
unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t, 1032
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1034
unpack_swiaudio_SLQSGetM2MAudioVolume_t, 1034
unpack_swiaudio_SLQSGetM2MAVMute_t, 1035
unpack_swiaudio_SLQSGetM2MSpkrGain_t, 1036
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
unpack_swiavms_SLQSAVMSSendSelection_t, 1043
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 1043
unpack_swiavms_SLQSAvmsSetEventReport_t, 1044
unpack_swiavms_SLQSAVMSSetSettings_t, 1044
unpack_swiavms_SLQSAVMSSetSettings_v2_t, 1045
unpack_swiavms_SLQSAVMSStartSession_t, 1045
unpack_swiavms_SLQSAVMSStopSession_t, 1046
unpack_swidms_SLQSSwiDmsGetHWWatchdog_t, 1046
unpack_swidms_SLQSSwiDmsGetMTU_t, 1047
unpack_swidms_SLQSSwiDmsGetSecureInfo_t, 1048
unpack_swidms_SLQSSwiDmsGetUimSlot_t, 1049
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 1050
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t, 1050
unpack_swidms_SLQSSwiDmsSetHWWatchdog_t, 1051
unpack_swidms_SLQSSwiDmsSetMTU_t, 1051
unpack_swidms_SLQSSwiDmsSetUimSlot_t, 1052
unpack_swidms_SLQSSwiDmsSetUsbComp_t, 1052
unpack_swiloc_SwiLocGetAutoStart_t, 1054
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 1055
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
unpack_swioma_SLQSOMADMGetSessionInfoExt_t, 1061
unpack_swioma_SLQSOMADMGetSettings_t, 1063
unpack_swioma_SLQSOMADMStartSession_t, 1064
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 1064
unpack_tmd_SLQSTmdGetMitigationDevList_t, 1065
unpack_tmd_SLQSTmdGetMitigationLvl_t, 1066
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t, 1066
unpack_tmd_SLQSTmdRegNotMitigationLvl_t, 1067
unpack_uim_ChangePin_t, 1068
unpack_uim_GetCardStatus_t, 1068
unpack_uim_GetCardStatusV2_t, 1069
unpack_uim_ReadTransparent_t, 1070
unpack_uim_SetPinProtection_t, 1071
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 1072
unpack_uim_SLQSUIMAuthenticate_t, 1073
unpack_uim_SLQSUIMDepersonalization_t, 1073
unpack_uim_SLQSUIMEventRegister_t, 1074
unpack_uim_SLQSUIMGetConfiguration_t, 1075

unpack_uim_SLQSUIMGetEID_t, 1075
unpack_uim_SLQSUIMGetFileAttributes_t, 1076
unpack_uim_SLQSUIMGetProfileInfo_t, 1077
unpack_uim_SLQSUIMGetServiceStatus_t, 1078
unpack_uim_SLQSUIMGetSIMProfile_t, 1079
unpack_uim_SLQSUIMGetSlotsStatus_t, 1080
unpack_uim_SLQSUIMGetSlotsStatusV2_t, 1081
unpack_uim_SLQSUIMReadRecord_t, 1082
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 1082
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 1083
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t, 1083
unpack_uim_SLQSUIMWriteRecord_t, 1084
unpack_uim_SLQSUIMWriteTransparent_t, 1085
unpack_uim_UnblockPin_t, 1085
unpack_uim_UnblockPinV2_t, 1086
unpack_uim_VerifyPin_t, 1087
unpack_voice_allCallStatusCallback_ind_t, 1089
unpack_voice_DTMFEventCallback_ind_t, 1090
unpack_voice_OTASPStatusCallback_ind_t, 1092
unpack_voice_SLQSOriginateUSSD_t, 1094
unpack_voice_SLQSVoiceAnswerCall_t, 1095
unpack_voice_SLQSVoiceBurstDTMF_t, 1095
unpack_voice_SLQSVoiceDialCall_t, 1096
unpack_voice_SLQSVoiceEndCall_t, 1097
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1100
unpack_voice_SLQSVoiceGetCallBarring_t, 1102
unpack_voice_SLQSVoiceGetCallForwardingStatus_t, 1103
unpack_voice_SLQSVoiceGetCallInfo_t, 1106
unpack_voice_SLQSVoiceGetCallWaiting_t, 1108
unpack_voice_SLQSVoiceGetCLIP_t, 1109
unpack_voice_SLQSVoiceGetCLIR_t, 1111
unpack_voice_SLQSVoiceGetCNAP_t, 1112
unpack_voice_SLQSVoiceGetCOLP_t, 1114
unpack_voice_SLQSVoiceGetCOLR_t, 1115
unpack_voice_SLQSVoiceGetConfig_t, 1117
unpack_voice_SLQSVoiceManageCalls_t, 1118
unpack_voice_SLQSVoiceSendFlash_t, 1118
unpack_voice_SLQSVoiceSetCallBarringPassword_t, 1120
unpack_voice_SLQSVoiceSetConfig_t, 1121
unpack_voice_SLQSVoiceSetSUPSService_t, 1123
unpack_voice_SLQSVoiceStartContDTMF_t, 1123
unpack_voice_SLQSVoiceStopContDTMF_t, 1124
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1126
unpack_voice_SUPSNotificationCallback_ind_t, 1129
unpack_voice_USSDNotificationCallback_ind_t, 1129
unpack_voice_USSDNoWaitIndicationCallback_ind_t, 1130
unpack_voice_VoiceInfoRecCallback_ind_t, 1132
unpack_voice_voicePrivacyChangeCallback_ind_t, 1134
unpack_wda_GetDataFormat_t, 1136
unpack_wda_GetQmapSettings_t, 1137
unpack_wda_SetDataFormat_t, 1140
unpack_wda_SetQmapSettings_t, 1141
unpack_wds_DHCPv4ClientLease_ind_t, 1142
unpack_wds_GetAutoconnect_t, 1143
unpack_wds_GetByteTotals_t, 1143
unpack_wds_GetConnectionRate_t, 1144
unpack_wds_GetDataBearerTechnology_t, 1145
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileNum_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
unpack_wds_GetDormancyState_t, 1152
unpack_wds_GetLastMobileIPError_t, 1152
unpack_wds_GetMobileIP_t, 1153
unpack_wds_GetMobileIPProfile_t, 1155
unpack_wds_GetPacketStatistics_t, 1157
unpack_wds_GetPacketStatus_t, 1159
unpack_wds_GetSessionDuration_t, 1160
unpack_wds_GetSessionDurationV2_t, 1161
unpack_wds_GetSessionState_t, 1161
unpack_wds_RMSetTransferStatistics_t, 1162
unpack_wds_SetMobileIPProfile_t, 1162
unpack_wds_SLQSCreateProfile_t, 1163
unpack_wds_SLQSDeleteProfile_t, 1163
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
unpack_wds_SLQSGet3GPPConfigItem_t, 1167
unpack_wds_SLQSGetCurrDataSystemStat_t, 1168
unpack_wds_SLQSGetCurrentChannelRate_t, 1169
unpack_wds_SLQSGetDataBearerTechnology_t, 1170
unpack_wds_SLQSGetDUNCallInfo_t, 1173
unpack_wds_SLQSGetProfileSettings_t, 1173
unpack_wds_SLQSGetProfileSettingsV2_t, 1174
unpack_wds_SLQSGetRuntimeSettings_t, 1177
unpack_wds_SLQSModifyProfile_t, 1178
unpack_wds_SLQSSetIPFamilyPreference_t, 1178
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 1180
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 1187
unpack_wds_SLQSSGetLoopback_t, 1188
unpack_wds_SLQSStartDataSession_t, 1188
unpack_wds_SLQSSwiProfileChangeCallback_Ind_t, 1189
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 1191
partNoStr
_litefw_FirmwareFileInfo, 40
path

uim_fileInfo, 702
pathLen
uim_fileInfo, 702
pbAuto
unpack_pds_GetPortAutomaticTracking_t, 976
unpack_pds_GetServiceAutomaticTracking_t, 976
pbEnabled
unpack_pds_GetXTRAAutomaticDownload_t, 977
pbPlatform
unpack_dms_GetOfflineReason_t, 762
pci
nas_cellParams, 208
nas_ltePCI, 269
nas_PhyCaAggPcellInfo, 311
nas_PhyCaAggScellIndType, 312
nas_PhyCaAggScellInfo, 315
nas_umtsLTENbrCell, 376
NASPhyCaAggPcellInfo, 395
NASPhyCaAggScellArray, 396
NASPhyCaAggScellIndType, 398
NASPhyCaAggScellInfo, 399
PcoList
LibPackPCOIDList, 119
pcsDigit
nas_PlmnID, 317
pcsfFQDNAddress
wds_PCSCFFQDNAddressList, 1267
PdpConnState
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 837
PdpStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
pdpType
pack_wds_SetDefaultProfile_t, 617
pdptype
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
pds.h, 1511
pack_pds_ForceXTRADownload, 1514
pack_pds_GetPDSDefaults, 1514
pack_pds_GetPDSState, 1515
pack_pds_GetPortAutomaticTracking, 1515
pack_pds_GetServiceAutomaticTracking, 1515
pack_pds_GetXTRAAutomaticDownload, 1516
pack_pds_GetXTRANetwork, 1516
pack_pds_GetXTRAValidity, 1516
pack_pds_PDSInjectTimeReference, 1517
pack_pds_ResetPDSData, 1517
pack_pds_SLQSGetAGPSConfig, 1520
pack_pds_SLQSGetGPSStateInfo, 1520
pack_pds_SLQSPDSDeterminePosition, 1521
pack_pds_SLQSPDSInjectAbsoluteTimeReference, 1521
pack_pds_SLQSPDSInjectPositionData, 1521
pack_pds_SLQSSetAGPSConfig, 1522
pack_pds_SLQSSetPositionMethodState, 1522
pack_pds_SetEventReportCallback, 1517
pack_pds_SetPDSDefaults, 1518
pack_pds_SetPDSState, 1518
pack_pds_SetPortAutomaticTracking, 1518
pack_pds_SetServiceAutomaticTracking, 1519
pack_pds_SetXTRAAutomaticDownload, 1519
pack_pds_SetXTRANetwork, 1519
pack_pds_StartPDSTrackingSessionExt, 1522
pack_pds_StopPDSTrackingSession, 1523
unpack_pds_ForceXTRADownload, 1523
unpack_pds_ForceXTRADownload_t, 1513
unpack_pds_GetPDSDefaults, 1523
unpack_pds_GetPDSState, 1524
unpack_pds_GetPortAutomaticTracking, 1524
unpack_pds_GetServiceAutomaticTracking, 1524
unpack_pds_GetXTRAAutomaticDownload, 1525
unpack_pds_GetXTRANetwork, 1525
unpack_pds_GetXTRAValidity, 1526
unpack_pds_PDSInjectTimeReference, 1526
unpack_pds_PDSInjectTimeReference_t, 1513
unpack_pds_ResetPDSData, 1526
unpack_pds_ResetPDSData_t, 1513
unpack_pds_SLQSGetAGPSConfig, 1530
unpack_pds_SLQSGetGPSStateInfo, 1530
unpack_pds_SLQSPDSDeterminePosition, 1530
unpack_pds_SLQSPDSDeterminePosition_t, 1514
unpack_pds_SLQSPDSInjectAbsoluteTimeReference, 1531
unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 1514
unpack_pds_SLQSPDSInjectPositionData, 1531
unpack_pds_SLQSPDSInjectPositionData_t, 1514
unpack_pds_SLQSSetAGPSConfig, 1531
unpack_pds_SLQSSetAGPSConfig_t, 1514
unpack_pds_SLQSSetPositionMethodState, 1532
unpack_pds_SLQSSetPositionMethodState_t, 1514
unpack_pds_SetEventReport_Ind, 1527
unpack_pds_SetEventReportCallback, 1527
unpack_pds_SetEventReportCallback_t, 1513
unpack_pds_SetPDSDefaults, 1527
unpack_pds_SetPDSDefaults_t, 1513
unpack_pds_SetPDSState, 1528
unpack_pds_SetPDSState_t, 1513
unpack_pds_SetPdsState_Ind, 1528
unpack_pds_SetPortAutomaticTracking, 1528
unpack_pds_SetPortAutomaticTracking_t, 1513
unpack_pds_SetServiceAutomaticTracking, 1529
unpack_pds_SetServiceAutomaticTracking_t, 1514
unpack_pds_SetXTRAAutomaticDownload, 1529
unpack_pds_SetXTRAAutomaticDownload_t, 1514
unpack_pds_SetXTRANetwork, 1529
unpack_pds_SetXTRANetwork_t, 1514
unpack_pds_StartPDSTrackingSessionExt, 1532
unpack_pds_StartPDSTrackingSessionExt_t, 1514
unpack_pds_StopPDSTrackingSession, 1532
unpack_pds_StopPDSTrackingSession_t, 1514
peakRate
unpack_qos_tokenBucket_t, 1008
peakThroughputClass

LibPackGPRSRequestedQoS, 119
wds_GPRSQoS, 1264
PeriodicUpdateTimer
unpack_dms_PSMCfgChange_ind_t, 769
periodicUpdateTimer
dms_PSMPeriodicUpdateTimerTlv, 75
PeriodicUpdateTimerInd
dms_PSMPeriodicUpdateTimerIndTlv, 74
PersistentLpm
unpack_dms_SLQSDmsSwiGetPCInfo_t, 778
PersistentTechPref
unpack_nas_GetNetworkPreference_t, 882
persoFeature
appStats, 45
uim_appStatus, 691
persoRetries
appStats, 45
uim_appStatus, 691
persoState
appStats, 45
uim_appStatus, 691
persoUnblockRetries
appStats, 45
uim_appStatus, 691
pfailureCause
unpack_voice_SLQSOriginateUSSD_t, 1094
PhCtxtURI
ims_PhCtxtURIInfo, 104
phase
nas_rxInfo, 346
PhyCaAggPcellInfo
NasGetLTECphyCaInfo, 389
PhyCaAggScellArray
NasGetLTECphyCaInfo, 389
PhyCaAggScellDlBw
NasGetLTECphyCaInfo, 389
PhyCaAggScellIndType
NasGetLTECphyCaInfo, 389
PhyCaAggScellIndex
NasGetLTECphyCaInfo, 389
PhyCaAggScellInfo
NasGetLTECphyCaInfo, 389
PhysicalLayer
nas_protocolSubtypeElement, 322
PilotEnergy
nas_NetworkStatEVDO, 294
PilotPN
nas_PilotSetParams, 316
PilotStrength
nas_PilotSetParams, 316
PilotType
nas_PilotSetParams, 316
pin1Len
uim_encryptedPIN1, 698
pin1Retries
appStats, 46
uim_appStatus, 691
pin1State
appStats, 46
uim_appStatus, 691
pin1Val
uim_encryptedPIN1, 698
pin2Retries
appStats, 46
uim_appStatus, 691
pin2State
appStats, 46
uim_appStatus, 691
pinID
uim_changeUIMPIN, 696
uim_setPINProtection, 720
uim_unblockUIMPIN, 726
uim_verifyUIMPIN, 728
pinLen
uim_changeUIMPIN, 696
uim_verifyUIMPIN, 728
pinLength
uim_setPINProtection, 720
pinOperation
uim_setPINProtection, 720
pinProtection
pack_uim_SetPinProtection_t, 567
pack_uim_UnblockPin_t, 585
pinVal
uim_changeUIMPIN, 696
uim_verifyUIMPIN, 728
pinValue
uim_setPINProtection, 720
pkgDate
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgDateLength
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgDesc
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
PkgDescLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
pkgDescLength
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
PkgDescription
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
pkgInstallTime
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgInstallTimeLength
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
PkgName
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
pkgName

unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
PkgNameLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
pkgNameLength
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgSize
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgVendorName
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgVendorNameLength
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgVersionName
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgVersionNameLength
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
pkgver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 796
PktErrRate
unpack_qos_swiQosFlow_t, 1008
plasmaIDString
pack_dms_SLQSSwiSetHostDevInfo_t, 427
unpack_dms_SLQSSwiGetHostDevInfo_t, 799
plmn
nas_GERANInfo, 232
nas_LTEInfoIntrafreq, 260
nas_UMTSExtInfo, 371
nas_UMTSInfo, 372
PlmnLen
nas_QmisNasSlqsNasPCICellInfo, 326
plmnListLen
nas_PrefPLMNListTlv, 320
polarityIncluded
voice_lineCtrlInfo, 1238
port
loc_IPv4Info, 174
loc_IPv6Info, 175
unpack_qos_Port_t, 991
posDataNmea
pack_pds_SetEventReportCallback_t, 518
posSrc
unpack_pds_SetEventReport_Ind_t, 981
PositionDataNMEA
unpack_pds_SetEventReport_Ind_t, 980
positionSrc
pack_loc_SLQSLOCInjectPosition_t, 455
PowerOffMode
unpack_dms_SLQSDmsSwiGetPCInfo_t, 778
pprofileid3gpp
pack_wds_SLQSStartDataSession_t, 633
pprofileid3gpp2
pack_wds_SLQSStartDataSession_t, 633
prDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1191
prDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1191
prPCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1192
prPCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1192
Precedence
unpack_qos_swiQosFilter_t, 1004
precedenceClass
LibPackGPRSRequestedQoS, 119
wds_GPRSQoS, 1264
prefNetwork
unpack_wds_SLQSGetCurrDataSystemStat_t, 1168
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
preference
pack_pds_SetXTRANetwork_t, 521
Preferred
nas_QmiNas3GppNetworkInfo, 324
prefixLen
unpack_qos_IPv6Addr_t, 990
presentationInd
voice_ECTNum, 1234
voice_remotePartyNum, 1246
priCSCFPort
ims_PCSCFPortInfo, 103
priChA
nas_CDMAChannel, 201
priChB
nas_CDMAChannel, 201
priSize
unpack_dms_GetFirmwareRevisions_t, 756
priVersionStr
_litefw_FirmwareFileInfo, 40
pridns
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
pridnsv6
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
primaryDNS
pack_wds_SetDefaultProfile_t, 617
PrimaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 1177
PrimaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 1177
primaryHA
unpack_wds_GetMobileIPProfile_t, 1155
privacyPref

pack_voice_SLQSVoiceSetPreferredPrivacy_t, 601
priver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 796
priversion_str
unpack_dms_GetFirmwareInfo_t, 753
Profile
pack_audio_SLQSGetAudioPathConfig_t, 407
pack_audio_SLQSGetAudioVolTLBConfig_t, 408
pack_audio_SLQSSetAudioPathConfig_t, 410
pack_audio_SLQSSetAudioProfile_t, 411
pack_audio_SLQSSetAudioVolTLBConfig_t, 412
pack_swiaudio_SLQSGetM2MAudioVolume_t, 541
pack_swiaudio_SLQSGetM2MAVMute_t, 542
pack_swiaudio_SLQSGetM2MSpkrGain_t, 542
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 543
pack_swiaudio_SLQSSetM2MAudioProfile_t, 544
pack_swiaudio_SLQSSetM2MAudioVolume_t, 545
pack_swiaudio_SLQSSetM2MAVMute_t, 546
pack_swiaudio_SLQSSetM2MSpkrGain_t, 546
unpack_audio_SLQSGetAudioProfile_t, 732
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1034
profileActInfo
pack_uim_SLQSUIMAddProfile_t, 568
profileClass
uim_profileClassTlv, 708
profileICCID
uim_profileICCIDTlv, 709
profileICCIDLen
uim_profileICCIDTlv, 709
ProfileID
unpack_wds_SLQSGetRuntimeSettings_t, 1177
profileIconData
uim_profileIconTlv, 709
profileIconDataLen
uim_profileIconTlv, 709
profileIconType
uim_profileIconTypeTlv, 710
ProfileId
pack_wds_SLQSGetProfileSettings_t, 625
profileId
pack_uim_SLQSUIMDeleteProfile_t, 569
pack_uim_SLQSUIMGetProfileInfo_t, 574
pack_uim_SLQSUIMSetSIMProfile_t, 581
pack_uim_SLQSUIMUpdateProfile_t, 582
uim_activeProfileInfo, 687
uim_getSimProfileInfo, 703
wds_DHCPProfileIdTlv, 1259
wds_DHCPv4ProfileId, 1261
wdsDhcpv4ProfileId, 1276
ProfileId3GPP2
unpack_qos_swiQosFlow_t, 1008
ProfileIdTlv
unpack_wds_DHCPv4ClientLease_ind_t, 1142
ProfileIndex
PackCreateProfileOut, 636
profileIndex
pack_wds_SLQSDeleteProfile_t, 623
wds_ProfileIdentifier, 1269
profileInfoLen
uim_getSimProfileInfo, 703
profileInx
wds_profileChange, 1268
profileList
unpack_wds_SLQSGet3GPPConfigItem_t, 1167
ProfileName
unpack_wds_SLQSGetRuntimeSettings_t, 1177
profileName
uim_profileNameTlv, 711
profileNameLen
uim_profileNameTlv, 711
profileNickname
uim_profileNicknameTlv, 711
profileNicknameLen
uim_profileNicknameTlv, 711
profilePolicyRules
uim_profilePolicyRulesTlv, 712
profileSPN
uim_profileSPNTlv, 712
profileSPNLen
uim_profileSPNTlv, 712
profileState
uim_profileStateTlv, 713
ProfileTlv
unpack_wds_SLQSSwiProfileChangeCallback_-Ind_t, 1189
ProfileType
pack_wds_SLQSGetProfileSettings_t, 625
PackCreateProfileOut, 636
unpack_wds_SLQSGetProfileSettings_t, 1173
unpack_wds_SLQSGetProfileSettingsV2_t, 1174
profileType
pack_wds_SetDefaultProfile_t, 617
pack_wds_SLQSDeleteProfile_t, 623
uim_activeProfileInfo, 687
uim_getSimProfileInfo, 703
wds_DHCPProfileIdTlv, 1259
wds_DHCPv4ProfileId, 1262
wds_profileChange, 1268
wds_ProfileIdentifier, 1269
wdsDhcpv4ProfileId, 1276
profiletype
pack_wds_GetDefaultProfile_t, 611
pack_wds_GetDefaultProfileV2_t, 612
PromptFwDownload
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
PromptFwUpdate
pack_swiavms_SLQSAVMSSetSettings_t, 549
pack_swiavms_SLQSAVMSSetSettings_v2_t, 551
Protocol
unpack_nas_GetCDMANetworkParameters_t, 878
provider
loc_AppProviderInfoTlv, 164
providerLen

loc_AppProviderInfoTlv, 164
ProvisionStatus
voice_CLIPResp, 1225
voice_CLIRResp, 1226
voice_CNAPResp, 1227
voice_COLPResp, 1227
voice_COLRResp, 1228
psAttachState
nas_servSystem, 351
NASServingSystemInfo, 405
psBarStatus
nas_CallBarringSysInfo, 199
nas_callBarStatus, 200
psState
nas_CommInfo, 212
psc
nas_UMTSExtInfo, 371
nas_UMTSInfo, 372
nas_wcdmaCellInfo, 379
nas_WCDMASysInfo, 385
pscValid
nas_WCDMASysInfo, 385
pscsfIPv4Addr
wds_PCSCFIPv4ServerAddressList, 1268
psetting
unpack_wds_GetAutoconnect_t, 1143
psid
pack_wds_SLQSStopDataSession_t, 633
unpack_wds_SLQSStartDataSession_t, 1188
puk1Retries
appStats, 46
uim_appStatus, 691
puk2Retries
appStats, 46
uim_appStatus, 691
pukLen
uim_unblockUIMPIN, 726
pukVal
uim_unblockUIMPIN, 726
pukValue
pack_dms_UIMUnblockPIN_t, 433
pwd
unpack_wds_GetDefaultProfileV2_t, 1151
pwdsize
unpack_wds_GetDefaultProfileV2_t, 1151
pwrDenialTime
voice_lineCtrlInfo, 1238

QMI_LITE_WDS_CURRENT_CALL_DB_MASK
wds.h, 1768
QMI_LITE_WDS_LAST_CALL_DB_MASK
wds.h, 1768
QCI
LibPackQosClassID, 156
QFlowState
unpack_qos_QosFlowInfo_t, 992
QLIC
nas_DeviceConfigDetail, 222
QMI pack/unpack (lite-qmi), 33
QMI_ERR_GENERAL
QmuxInternal.h, 1592
QMI_RESULT_FAILURE
QmuxInternal.h, 1592
QMI_RESULT_SUCCESS
QmuxInternal.h, 1592
QMIType
QMUX_HDR, 648
QMUX layer (lite-qmux), 35
QMUX_HDR, 647
ClientId, 648
CtlFlags, 648
IfType, 648
Length, 648
QMIType, 648
QMUX_MSG_FLAG
QmuxInternal.h, 1593
qaGobiApiTableBandClasses.h, 1533
qaGobiApiTableCallControlReturnReasons.h, 1536
qaGobiApiTableCallEndReasons.h, 1537
qaGobiApiTableCarrierCodes.h, 1553
qaGobiApiTableCodingScheme.h, 1555
qaGobiApiTableFwDldErrorCodes.h, 1558
qaGobiApiTableGpsCapabilityCodes.h, 1562
qaGobiApiTablePowerModes.h, 1562
qaGobiApiTableRadioInterfaces.h, 1563
qaGobiApiTableRegionCodes.h, 1564
qaGobiApiTableServiceOptions.h, 1564
qaGobiApiTableSupServiceInfoClasses.h, 1566
qaGobiApiTableSwiAudio.h, 1567
qaGobiApiTableSwiOMADMSessionStatus.h, 1567
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h, 1568
qaGobiApiTableVoiceCallEndReasons.h, 1570
qcqmi
QmuxTransport, 649
QcqmiService, 641
devicePath, 642
readyToRead, 642
useQcqmi, 642
Qlink_prot
unpack_wda_GetDataFormat_t, 1136
qm_wds_ds_profile_extended_err_codes
qmerrno.h, 1582
qmTlvResult, 647
DeviceError, 647
DeviceResult, 647
TlvPresenceMask, 647
TlvResultCode, 647
qmerrno.h
eQCWWAN_ERR_API_MUTEX_TIMEOUT, 1579
eQCWWAN_ERR_BUFFER_SZ, 1578
eQCWWAN_ERR_CANCEL_OP, 1579
eQCWWAN_ERR_DRIVER, 1579
eQCWWAN_ERR_ENUM_BEGIN, 1578
eQCWWAN_ERR_ENUM_END, 1579
eQCWWAN_ERR_FILE_COPY, 1578
eQCWWAN_ERR_FILE_OPEN, 1578

eQCWWAN_ERR_GENERAL, 1578
eQCWWAN_ERR_INTERNAL, 1578
eQCWWAN_ERR_INVALID_ARG, 1578
eQCWWAN_ERR_INVALID_DEVID, 1578
eQCWWAN_ERR_INVALID_FILE, 1578
eQCWWAN_ERR_INVALID_QMI_RSP, 1578
eQCWWAN_ERR_INVALID_XID, 1579
eQCWWAN_ERR_MALFORMED_QMI_RSP, 1578
eQCWWAN_ERR_MEMORY, 1578
eQCWWAN_ERR_MULTIPLE_DEVICES, 1579
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED, 1579
eQCWWAN_ERR_NO_CANCELABLE_OP, 1579
eQCWWAN_ERR_NO_CONNECTION, 1578
eQCWWAN_ERR_NO_DEVICE, 1578
eQCWWAN_ERR_NO_SIGNAL, 1579
eQCWWAN_ERR_NONE, 1578
eQCWWAN_ERR_NULL_TLV, 1582
eQCWWAN_ERR_OFFLINE, 1579
eQCWWAN_ERR_PDU_GENERATION, 1579
eQCWWAN_ERR_QMI_ABORTED, 1579
eQCWWAN_ERR_QMI_ACCESS_DENIED, 1581
eQCWWAN_ERR_QMI_ACK_NOT_SENT, 1581
eQCWWAN_ERR_QMI_ARG_TOO_LONG, 1579
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED, 1580
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK, 1580
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED, 1581
eQCWWAN_ERR_QMI_CALL_FAILED, 1579
eQCWWAN_ERR_QMI_CARD_BUSY_RSP, 1582
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED, 1581
eQCWWAN_ERR_QMI_CAT_END, 1582
eQCWWAN_ERR_QMI_CAT_START, 1582
eQCWWAN_ERR_QMI_CAUSE_CODE, 1580
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED, 1579
eQCWWAN_ERR_QMI_CONNECT, 1578
eQCWWAN_ERR_QMI_DEVICE_IN_USE, 1579
eQCWWAN_ERR_QMI_DEVICE_NOT_READY, 1580
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL, 1580
eQCWWAN_ERR_QMI_DISABLED, 1580
eQCWWAN_ERR_QMI_ENCODING, 1580
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE, 1582
eQCWWAN_ERR_QMI_EVENT_REG_FAILED, 1582
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL, 1581
eQCWWAN_ERR_QMI_FDN_RESTRICT, 1581
eQCWWAN_ERR_QMI_FLOW_SUSPENDED, 1580
eQCWWAN_ERR_QMI_GENERAL, 1580
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED, 1581
eQCWWAN_ERR_QMI_IFACE, 1578
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE, 1581
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER, 1580
eQCWWAN_ERR_QMI_INCORRECT_PIN, 1579
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE, 1581
eQCWWAN_ERR_QMI_INJECT_TIMEOUT, 1581
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES, 1580
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND, 1580
eQCWWAN_ERR_QMI_INTERNAL, 1579
eQCWWAN_ERR_QMI_INVALID_ARG, 1580
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID, 1579
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT, 1580
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD, 1582
eQCWWAN_ERR_QMI_INVALID_HANDLE, 1579
eQCWWAN_ERR_QMI_INVALID_ID, 1580
eQCWWAN_ERR_QMI_INVALID_INDEX, 1580
eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF, 1580
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE, 1580
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID, 1580
eQCWWAN_ERR_QMI_INVALID_OPERATION, 1581
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE, 1579
eQCWWAN_ERR_QMI_INVALID_PINID, 1579
eQCWWAN_ERR_QMI_INVALID_PROFILE, 1579
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE, 1579
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION, 1580
eQCWWAN_ERR_QMI_INVALID_QMI_CMD, 1581
eQCWWAN_ERR_QMI_INVALID_QOS_ID, 1580
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION, 1580
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE, 1579
eQCWWAN_ERR_QMI_INVALID_TECH_PREF, 1579
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP, 1582
eQCWWAN_ERR_QMI_INVALID_TRANSITION, 1580
eQCWWAN_ERR_QMI_INVALID_TX_ID, 1579
eQCWWAN_ERR_QMI_MALFORMED_MSG, 1579

eQCWWAN_ERR_QMI_MAX, 1581
eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE, 1580
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE, 1580
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE, 1580
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT, 1580
eQCWWAN_ERR_QMI_MISSING_ARG, 1579
eQCWWAN_ERR_QMI_MSG_BLOCKED, 1581
eQCWWAN_ERR_QMI_NETWORK_ABORTED, 1581
eQCWWAN_ERR_QMI_NETWORK_NOT_READY, 1580
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE, 1580
eQCWWAN_ERR_QMI_NO_EFFECT, 1579
eQCWWAN_ERR_QMI_NO_ENTRY, 1580
eQCWWAN_ERR_QMI_NO_FREE_PROFILE, 1579
eQCWWAN_ERR_QMI_NO_MEMORY, 1579
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND, 1579
eQCWWAN_ERR_QMI_NO_RADIO, 1581
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION, 1581
eQCWWAN_ERR_QMI_NO_THRESHOLDS, 1579
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE, 1580
eQCWWAN_ERR_QMI_NOT_PROVISIONED, 1579
eQCWWAN_ERR_QMI_NOT_SUPPORTED, 1581
eQCWWAN_ERR_QMI_OFFSET, 1579
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED, 1579
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED, 1579
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE, 1581
eQCWWAN_ERR_QMI_OUT_OF_CALL, 1579
eQCWWAN_ERR_QMI_PIN_BLOCKED, 1580
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED, 1580
eQCWWAN_ERR_QMI_POLICY_MISMATCH, 1581
eQCWWAN_ERR_QMI_REQ, 1578
eQCWWAN_ERR_QMI_REQ_SCH, 1578
eQCWWAN_ERR_QMI_REQ_TO, 1578
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED, 1580
eQCWWAN_ERR_QMI_RSP, 1578
eQCWWAN_ERR_QMI_RSP_TO, 1578
eQCWWAN_ERR_QMI_SEGMENT_ORDER, 1581
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG, 1581
eQCWWAN_ERR_QMI_SESSION_INACTIVE, 1580
eQCWWAN_ERR_QMI_SESSION_INVALID, 1580
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP, 1580
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND, 1581
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED, 1580
eQCWWAN_ERR_QMI_SMSC_ADDR, 1581
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE, 1581
eQCWWAN_ERR_QMI_TPDU_TYPE, 1581
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION, 1579
eQCWWAN_ERR_QMI_UNKNOWN, 1580
eQCWWAN_ERR_QMI_WIDTH, 1582
eQCWWAN_ERR_RESET, 1579
eQCWWAN_ERR_SWICM_AM_VERS_ERROR, 1581
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS, 1581
eQCWWAN_ERR_SWICM_END, 1582
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS, 1581
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID, 1581
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID, 1581
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID, 1581
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED, 1581
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED, 1581
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED, 1581
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS, 1581
eQCWWAN_ERR_SWICM_SOCKET_IN_USE, 1581
eQCWWAN_ERR_SWICM_START, 1581
eQCWWAN_ERR_SWICM_TIMEOUT, 1581
eQCWWAN_ERR_SWICM_V4DWN_V6DWN, 1581
eQCWWAN_ERR_SWICM_V4DWN_V6UP, 1581
eQCWWAN_ERR_SWICM_V4UP_V6DWN, 1581
eQCWWAN_ERR_SWICM_V4UP_V6UP, 1581
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED, 1582
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND, 1582
eQCWWAN_ERR_SWIDCS_END, 1582
eQCWWAN_ERR_SWIDCS_FILEIO_ERR, 1582
eQCWWAN_ERR_SWIDCS_IOCTL_ERR, 1582

eQCWWAN_ERR_SWIDCS_START, 1582
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE, 1582
eQCWWAN_ERR_SWIIM_END, 1582
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND, 1582
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED, 1582
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE, 1582
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE, 1582
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX, 1582
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH, 1582
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE, 1582
eQCWWAN_ERR_SWIIM_FW_SECBOOT_IMAGE_NOT_SIGNED, 1582
eQCWWAN_ERR_SWIIM_FW_SECBOOT_INVALID_CERT_CHAIN, 1582
eQCWWAN_ERR_SWIIM_FW_TOO_MANY_FILES, 1582
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL, 1582
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS, 1582
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT, 1582
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE, 1582
eQCWWAN_ERR_SWIIM_INVALID_PATH, 1582
eQCWWAN_ERR_SWIIM_OPENING_DIR, 1582
eQCWWAN_ERR_SWIIM_OPENING_FILE, 1582
eQCWWAN_ERR_SWIIM_START, 1582
eQCWWAN_ERR_SWISM_END, 1582
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND, 1582
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED, 1582
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG, 1582
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED, 1582
eQCWWAN_ERR_SWISMS_START, 1582
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE, 1583
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR, 1583
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED, 1583
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES, 1583
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY, 1583
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET, 1583
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET, 1583
eWDS_ERR_PROFILE_REG_END, 1583
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID, 1583
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED, 1583
eWDS_ERR_PROFILE_REG_RESULT_FAIL, 1583
eWDS_ERR_PROFILE_REG_RESULT_LIST_END, 1583
qmerrno.h, 1576
eQCWWANError, 1578
qm_wds_ds_profile_extended_err_codes, 1582
QmiHdr, 642
length, 642
messageId, 642
QmuxInternal.h, 1593
QmiService, 642
clientId, 643
deviceFd, 644
pIndicationCallback, 644
pIndicationCallbackContext, 644
pNext, 644
pQcqmitransport, 644
pTransport, 644
QmiService.h, 1584
serviceType, 644
transactionId, 644
transactionIdLock, 644
transactionPool, 644
transactionPoolLock, 644
QmiService.h, 1583
QmiService, 1584
QmiService_CancelTransaction, 1585
QmiService_GetNextTransactionId, 1585
QmiService_HandleQmiPacket, 1586
QmiService_Initialize, 1586
QmiService_SendRequest, 1586
QmiService_ShutDown, 1587
QmiService_CancelTransaction
QmiService.h, 1585

QmiService_GetNextTransactionId
QmiService.h, 1585
QmiService_HandleQmiPacket
QmiService.h, 1586
QmiService_Initialize
QmiService.h, 1586
QmiService_SendRequest
QmiService.h, 1586
QmiService_ShutDown
QmiService.h, 1587
qmiServiceList
QmuxTransport, 649
qmiServiceListLock
QmuxTransport, 649
qmiSmsMessageList, 644
messageIndex, 644
messageTag, 644
QmiSyncObject, 644
buffer, 645
bufferSize, 645
condition, 645
mutex, 645
QmiSyncObject.h, 1587
QmiSyncObject.h, 1587
QmiSyncObject, 1587
QmiSyncObject_Destroy, 1588
QmiSyncObject_Initialize, 1588
QmiSyncObject_Lock, 1588
QmiSyncObject_ResponseCallback, 1589
QmiSyncObject_Signal, 1589
QmiSyncObject_TimedWait, 1589
QmiSyncObject_Unlock, 1589
QmiSyncObject_Wait, 1589
QmiSyncObject_Destroy
QmiSyncObject.h, 1588
QmiSyncObject_Initialize
QmiSyncObject.h, 1588
QmiSyncObject_Lock
QmiSyncObject.h, 1588
QmiSyncObject_ResponseCallback
QmiSyncObject.h, 1589
QmiSyncObject_Signal
QmiSyncObject.h, 1589
QmiSyncObject_TimedWait
QmiSyncObject.h, 1589
QmiSyncObject_Unlock
QmiSyncObject.h, 1589
QmiSyncObject_Wait
QmiSyncObject.h, 1589
QmiTransaction, 645
inUse, 646
messageId, 646
pResponseCallback, 646
pResponseCallbackContext, 646
QmiTransaction.h, 1590
transactionId, 646
QmiTransaction.h, 1590
QmiTransaction, 1590
QmiTransaction_Initialize, 1591
QmiTransaction_Initialize
QmiTransaction.h, 1591
QmiTransactionHdr, 646
flags, 646
QmuxInternal.h, 1593
transactionId, 646
qmiWDSDataBearerTechnology, 646
currentNetwork, 647
ratMask, 647
soMask, 647
qmuint16
SwiDataTypes.h, 1657
qmulong
SwiDataTypes.h, 1657
QmuxInternal.h, 1591
CtlTransactionHdr, 1593
eCTL_IND, 1592
eCTL_REQ, 1592
eCTL_RSP, 1592
eInvalidService, 1592
FAILURE, 1592
GET_CLIENT_ID, 1592
GetClientIdReqMsg, 1593
GetClientIdRspMsg, 1593
IF_TYPE_SIZE, 1592
MAX, 1592
QMI_ERR_GENERAL, 1592
QMI_RESULT_FAILURE, 1592
QMI_RESULT_SUCCESS, 1592
QMUX_MSG_FLAG, 1593
QmiHdr, 1593
QmiTransactionHdr, 1593
RELEASE_CLIENT_ID, 1593
ReleaseClientIdReqMsg, 1593
ReleaseClientIdRspMsg, 1593
ResultTlv, 1593
SUCCESS, 1593
QmuxTransport, 648
deviceFd, 649
pDataLayer, 649
qcqmi, 649
qmiServiceList, 649
qmiServiceListLock, 649
QmuxTransport.h, 1594
readThread, 649
shutdownFd, 649
useMbim, 649
writeLock, 649
QmuxTransport.h, 1593
INVALID_CLIENT_ID, 1594
MAX_QMI_CLIENT, 1594
MAX_QMI_CLIENT_ID, 1594
QmuxTransport, 1594
QmuxTransport_AttachQmiService, 1595
QmuxTransport_DetachQmiService, 1595
QmuxTransport_Initialize, 1595
QmuxTransport_SendQmiPacket, 1595

QmuxTransport_ShutDown, 1596
QmuxTransport_AttachQmiService
QmuxTransport.h, 1595
QmuxTransport_DetachQmiService
QmuxTransport.h, 1595
QmuxTransport_Initialize
QmuxTransport.h, 1595
QmuxTransport_SendQmiPacket
QmuxTransport.h, 1595
QmuxTransport_ShutDown
QmuxTransport.h, 1596
qos.h, 1596
pack_qos_BindDataPort, 1597
pack_qos_SLQSQosGetNetworkStatus, 1598
pack_qos_SLQSQosSwiReadApnExtraParams, 1599
pack_qos_SLQSQosSwiReadDataStats, 1599
pack_qos_SLQSSetQosEventCallback, 1600
unpack_qos_BindDataPort, 1601
unpack_qos_SLQSQosGetNetworkStatus, 1601
unpack_qos_SLQSQosSwiReadApnExtraParams, 1601
unpack_qos_SLQSQosSwiReadDataStats, 1602
unpack_qos_SLQSSetQosEventCallback, 1602
unpack_qos_SLQSSetQosEventCallback_ind, 1603
unpack_qos_SLQSSetQosNWStatusCallback_ind, 1603
unpack_qos_SLQSSetQosPriEventCallback_ind, 1604
unpack_qos_SLQSSetQosStatusCallback_ind, 1605
qos_BindDataPortMuxID_t, 649
MuxID, 649
qos_BindDataPortPeripheralEndPointID_t, 649
EndPointType, 650
IfaceID, 650
qos_BindDataPortSIODataPort_t, 650
SIODataPort, 650
Qos_format
unpack_wda_GetDataFormat_t, 1136
Qos_header_format
unpack_wda_GetDataFormat_t, 1136
qosDeliveryOrder
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
qosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 997
QosFlowInfo
unpack_qos_SLQSSetQosEventCallback_ind_t, 998

RAN
unpack_nas_GetServingNetwork_t, 886
RAT
nas_QmiNas3GppNetworkRAT, 325
RATMask
currNetworkInfo, 60
wds_currNetworkInfo, 1255
RBTTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
RELEASE_CLIENT_ID
QmuxInternal.h, 1593
RFBandInfoElements, 652
activeBandClass, 652
activeChannel, 652
radioInterface, 652
unpack_nas_GetRFInfo_t, 883
RFTlv
unpack_nas_SetEventReportInd_t, 891
RMAutoConnect
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
RPCause
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
RPTlv
NASQmiCbkNasSystemSelPrefInd, 403
RRTlv
unpack_nas_SetEventReportInd_t, 891
RSRPThresListLen
nas_RSRPThresh, 343
RSRQThresListLen
nas_RSRQThresh, 344
RSSIThresListLen
nas_RSSIThresh, 345
RTIDTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
RTTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
RX_EC_IO
nas_NetworkStat1x, 292
RX_PWR
nas_NetworkStat1x, 292
nas_NetworkStatEVDO, 294
RXBOTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
RXChan
nas_LTEInfo, 258
rXDroppedCount
unpack_wds_GetPacketStatus_t, 1159
rXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 1159
rXOkBytesCount
unpack_wds_GetPacketStatus_t, 1159
rXPacketErrors
unpack_wds_GetPacketStatus_t, 1159
rXPacketOverflows
unpack_wds_GetPacketStatus_t, 1159
rXPacketSuccesses
unpack_wds_GetPacketStatus_t, 1159
radio
unpack_nas_GetSignalStrengths_t, 888
radio_if
pack_nas_SLQSNasGetTxRxInfo_t, 477
radioIf

nas_ecioListElement, 224
nas_errorRateListElement, 229
nas_RFBandInfoExtTlv, 333
nas_RFBandInfoTlv, 334
nas_RFBandwidthInfoTlv, 337
nas_RFDedicatedBandInfoTlv, 339
nas_rsrqInformation, 343
nas_rxSignalStrengthListElement, 348
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 949
RadioIfaces
dms_devCaps, 62
unpack_dms_GetDeviceCap_t, 744
unpack_dms_GetDeviceCapabilities_t, 746
unpack_nas_GetServingNetwork_t, 886
RadioIfacesSize
dms_devCaps, 62
unpack_dms_GetDeviceCap_t, 744
unpack_nas_GetServingNetwork_t, 886
radioIfacesSize
unpack_dms_GetDeviceCapabilities_t, 746
radioInterface
nas_RFBandInfoElements, 330
nas_RfBandInfoExtFormatElements, 331
nas_RfBandwidthInfoElements, 336
nas_RfDedicatedBandInfoElements, 339
nas_RFInfoTlv, 340
nas_roamIndList, 341
nas_servSystem, 351
nas_SignalStrengthTlv, 353
nas_timeInfo, 367
RFBandInfoElements, 652
radioInterfaceList
NASServingSystemInfo, 405
radioInterfaceNo
NASServingSystemInfo, 405
radioInterfaceSize
nas_RFInfoTlv, 341
randomizationWindow
dms_PSMRandomizationWindowTlv, 75
range
unpack_qos_Port_t, 991
rankIndicatorTlv
unpack_nas_SLQSSwiRandIndicatorCallback_-Ind_t, 966
rat
nas_CSGID, 213
nas_MNRInfo, 282
nas_NetworkScan3GPPInfo, 288
rat_value
DsdSystemInfo_t, 87
DsdUIApnAvailableSystemInfo_t, 89
RatDMTlv
NASQmiCbkNasSystemSelPrefInd, 403
RatDisabledMask
NASRatDisabledMaskTlv, 403
ratDisabledMask
nas_RatDisabledMaskTlv, 329
RatHandover
unpack_imsa_SLQSImsaRatStatusCallBack_ind_-t, 838
RatHandoverStatus
imsa_RatHandoverStatusInfo, 113
RatHandoverStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
ratMask
qmiWDSDataBearerTechnology, 647
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
rawHorConfidence
pack_loc_SLQSLOCInjectPosition_t, 455
rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 455
rawLen
uim_fileAttributes, 701
rawValue
uim_fileAttributes, 701
readRecord
pack_uim_SLQSUIMReadRecord_t, 577
readThread
MbimTransport, 192
QmuxTransport, 649
readTransparent
pack_uim_ReadTransparent_t, 566
readyToRead
QcqmiService, 642
Reason
pack_voice_SLQSVoiceGetCallForwardingStatus-_t, 593
pack_voice_SLQSVoiceSetCallBarringPassword_t, 600
reason
pack_voice_SLQSVoiceGetCallBarring_t, 592
pack_voice_SLQSVoiceSetSUPSService_t, 604
unpack_qos_SLQSSetQosStatusCallback_ind_t, 1000
voice_ccSUPSType, 1224
voice_redirNumInfo, 1244
receiptAction
sms_routeEntry, 666
reconfigReqd
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 1180
record
uim_readRecordInfo, 713
uim_writeRecordInfo, 728
recordCount
uim_fileAttributes, 701
recordSize
uim_fileAttributes, 701
RedirPartyNum
voice_arrRedirPartyNum, 1211
refData
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 798
refID
pack_cat_CATSendTerminalResponse_t, 414

refString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 798
ReferenceID
cat_AlPhaIdentifierTlv, 53
cat_EventIDDataTlv, 55
refpn
nas_CDMAInfo, 203
refreshComplete
pack_uim_SLQSUIMRefreshComplete_t, 577
refreshEvent
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 1082
RefreshMode
cat_RefreshTlv, 56
RefreshStage
cat_RefreshTlv, 56
regAction
pack_nas_SLQSInitiateNetworkRegistration_t, 471
RegForeignNID
unpack_nas_GetCDMANetworkParameters_t, 878
RegForeignSID
unpack_nas_GetCDMANetworkParameters_t, 878
RegHomeSID
unpack_nas_GetCDMANetworkParameters_t, 878
regInd
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 1029
regPrd
nas_AddCDMASysInfo, 195
regRefresh
pack_uim_SLQSUIMRefreshRegister_t, 579
regRejectInfoValid
nas_GSMSysInfo, 239
nas_LTESysInfo, 278
nas_NR5GSysInfoTlv, 302
nas_NR5GSystemInfoTlv, 306
nas_WCDMASysInfo, 385
regState
nas_servSystem, 351
RegStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
registerFlag
uim_registerRefresh, 718
RegistrationState
unpack_nas_GetServingNetwork_t, 886
registrationState
NASServingSystemInfo, 405
rejCause
nas_GSMSysInfo, 239
nas_LTESysInfo, 278
nas_NR5GSystemInfoTlv, 306
nas_WCDMASysInfo, 385
rejectCause
nas_NR5GSysInfoTlv, 302
nas_RejectReasonTlv, 329
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t, 949
rejectSrvDomain
nas_GSMSysInfo, 239
nas_LTESysInfo, 278
nas_NR5GSysInfoTlv, 302
nas_NR5GSystemInfoTlv, 306
nas_WCDMASysInfo, 385
ReleaseClientIdReqMsg, 650
clientId, 651
length, 651
messageId, 651
QmuxInternal.h, 1593
serviceType, 651
tlvLength, 651
tlvType, 651
ReleaseClientIdRspMsg, 651
clientId, 651
error, 651
length, 651
messageId, 651
QmuxInternal.h, 1593
result, 651
serviceType, 651
tlv2Length, 651
tlv2Type, 651
tlvLength, 651
tlvType, 651
releaseDate
_litefw_FirmwareFileInfo, 40
reliabilityClass
LibPackGPRSRequestedQoS, 119
wds_GPRSQoS, 1264
remPartyNumber
voice_remotePartyNum, 1246
RemotePartyName
voice_getAllCallRmtPtyName, 1236
RemotePartyNum
voice_getAllCallRmtPtyNum, 1236
ReqMitigationLvl
unpack_tmd_SLQSTmdGetMitigationLvl_t, 1066
resBerRatio
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
ResCode
unpack_audio_SLQSGetAudioVolTLBConfig_t, 733
unpack_audio_SLQSSetAudioVolTLBConfig_t, 733
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 798
resetDelay
pack_swidms_SLQSSwiDmsSetHWWatchdog_t, 552
swidms_SwiDmsGetHWWatchdog, 682
resetInfoInd
pack_dms_SLQSDmsSwiIndicationRegister_t, 424
result
GetClientIdRspMsg, 95
ReleaseClientIdRspMsg, 651
ResultTlv, 652
ResultTlv, 652

error, 652
QmuxInternal.h, 1593
result, 652
tlvLength, 652
tlvType, 652
resultcode
unpack_swiavms_SLQSAVMSEventReportInd_t, 1037
unpack_swiavms_SLQSAVMSGetSettings_t, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1042
unpack_swiavms_SLQSAVMSSendSelection_t, 1043
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 1044
unpack_swiavms_SLQSAvmsSetEventReport_t, 1044
unpack_swiavms_SLQSAVMSSetSettings_t, 1044
unpack_swiavms_SLQSAVMSSetSettings_v2_t, 1045
unpack_swiavms_SLQSAVMSStartSession_t, 1045
unpack_swiavms_SLQSAVMSStopSession_t, 1046
RetryCount
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
revPolarity
voice_lineCtrlInfo, 1238
revTunneling
unpack_wds_GetMobileIPProfile_t, 1155
ReverseMac
nas_protocolSubtypeElement, 322
rfBandInfo
unpack_nas_SLQSNasRFBandInfoCallback_Ind_t, 952
rfBandInfoExt
unpack_nas_SLQSNasRFBandInfoCallback_Ind_t, 952
rfBandwidthInfo
unpack_nas_SLQSNasRFBandInfoCallback_Ind_t, 952
rfDedicatedBandInfo
unpack_nas_SLQSNasRFBandInfoCallback_Ind_t, 953
RfSarState
pack_sar_SLQSSetRfSarState_t, 531
rfbandInfoList
unpack_nas_SLQSNasGetRFInfo_t, 945
RingBkTmr
ims_RngBkTmrInfo, 104
RingTmr
ims_RngTmrInfo, 105
RmTrasnferStaticsReq
pack_wds_RMSetTransferStatistics_t, 615
rmTrasnferStaticsReq, 653
bResetStatistics, 653
ulMask, 653
rms.h, 1605
pack_rms_GetSMSWake, 1606
pack_rms_SetSMSWake, 1606
unpack_rms_GetSMSWake, 1606
unpack_rms_SetSMSWake, 1607
RmtPtyNum
voice_arrRemotePartyNum, 1212
roamIndicator
nas_roamIndList, 341
RoamIndicatorVal
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
roamOrigVoiceSO
voice_prefVoiceSO, 1243
RoamPref
nas_RoamPrefTlv, 342
NASRoamPreferenceTlv, 403
roamStatus
nas_NR5GSysInfoTlv, 303
nas_NR5GSystemInfoTlv, 306
nas_sysInfoCommon, 362
roamStatusValid
nas_NR5GSysInfoTlv, 303
nas_NR5GSystemInfoTlv, 306
nas_sysInfoCommon, 362
roamTimerValue
voice_roamTimer, 1247
Roaming
nas_QmiNas3GppNetworkInfo, 324
unpack_nas_GetCDMANetworkParameters_t, 878
unpack_nas_GetServingNetwork_t, 886
roaming
unpack_nas_SetRoamingIndicatorCallback_ind_t, 894
RoamingIndicatorList
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
routeList
sms_setRoutesReq, 670
routeStorage
sms_routeEntry, 666
RptIMSCapability
pack_dms_SetIndicationRegister_t, 422
rptPosData
pack_pds_SetEventReportCallback_t, 518
rptRate
nas_LTESigRptCfg, 273
nas_LTESigRptConfig, 274
rscp
nas_rxInfo, 346
nas_UMTSExtInfo, 371
nas_UMTSInfo, 373
tdscdmaSigInfoExt, 684
rsrp
lteSSInfo, 184
nas_cellParams, 208
nas_QmisNasSlqsNasPCIInfo, 327
nas_rxInfo, 346

nas_RxSigInfo, 347
nas_umtsLTENbrCell, 376
rsrpRx0
nas_QmisNasSlqsNasPCIInfo, 327
rsrpRx1
nas_QmisNasSlqsNasPCIInfo, 327
rsrplevel
nas_lteRsrpinformation, 270
rsrq
lteSSInfo, 184
nas_cellParams, 208
nas_QmisNasSlqsNasPCIInfo, 327
nas_rsrqInformation, 343
nas_SccRxInfo, 349
nas_umtsLTENbrCell, 376
rsrqDelta
nas_SLQSSignalStrengthsIndReq, 356
rsrqInfo
nas_SLQSSignalStrengthsInformation, 357
unpack_nas_SLQSGetSignalStrength_t, 915
rsrqRx0
nas_QmisNasSlqsNasPCIInfo, 328
rsrqRx1
nas_QmisNasSlqsNasPCIInfo, 328
rssi
cdmaSSInfo, 57
hdrSSInfo, 96
lteSSInfo, 184
nas_cellParams, 208
nas_gsmCellInfo, 234
tdscdmaSigInfoExt, 684
unpack_nas_GetSignalStrengths_t, 888
rx_bytes
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
rx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
RxByteOKCnt
wds_RXBytesOKTlv, 1270
rxChainIndex
nas_RxSigInfo, 347
RxDropConutTlv
unpack_RMTransferStatistics_ind_t, 1012
rxLev
nas_GERANInfo, 232
rxOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 1173
RxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 1012
RxOkConutTlv
unpack_RMTransferStatistics_ind_t, 1012
rxPower
nas_rxInfo, 346
nas_RxSigInfo, 347
RxQFilter
unpack_qos_QosFlowInfo_t, 992
RxQFlowGranted
unpack_qos_QosFlowInfo_t, 993
rxSignalStrength
nas_rxSignalStrengthListElement, 348
rxSignalStrengthDelta
nas_SLQSSignalStrengthsIndReq, 356
rxSignalStrengthInfo
nas_SLQSSignalStrengthsInformation, 357
rxSignalStrengthList
unpack_nas_SLQSGetSignalStrength_t, 915
rxSignalStrengthListLen
unpack_nas_SLQSGetSignalStrength_t, 915

SCI
unpack_nas_GetCDMANetworkParameters_t, 878
SCM
unpack_nas_GetCDMANetworkParameters_t, 878
SCTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
SDPTlv
NASQmiCbkNasSystemSelPrefInd, 403
SDTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 833
SDU_HDR_LEN
common.h, 1290
SFTlv
unpack_ims_SLQSSMSCfgCallBack_ind_t, 830
SHORT
SwiDataTypes.h, 1657
SI
voice_calledPartyInfo, 1215
voice_callFWExtInfo, 1220
voice_callingPartyInfo, 1224
voice_redirNumInfo, 1244
SINTlv
unpack_ims_SLQSSMSCfgCallBack_ind_t, 830
SIODataPort
qos_BindDataPortSIODataPort_t, 650
SIPLocalPort
ims_SIPPortInfo, 107
sIntraSearch
nas_LTEInfoIntrafreq, 260
SLQSSSTlv
unpack_nas_SetEventReportInd_t, 891
SMS_CONFIG_LEN
sms.h, 1613
SMS_MAX_SMS_ROUTES
sms.h, 1613
SMS_NUM_OF_SET
sms.h, 1613
SMSC_TYPE_LEN
sms.h, 1613
sMSCAddressInfo, 671
data, 671
length, 671
sMSCAddressTlv, 671
SMSCInfo, 671
TlvPresent, 672
SMSCInfo
sMSCAddressTlv, 671

SMSCTlv
unpack_sms_SetNewSMSCallback_ind_t, 1017
sMSEtwsMessageInfo, 672
data, 672
length, 672
notificationType, 672
sMSEtwsMessageTlv, 672
EtwsMessageInfo, 673
TlvPresent, 673
sMSEtwsPlmnInfo, 673
mobileCountryCode, 673
mobileNetworkCode, 673
sMSMTMessageInfo, 674
messageIndex, 674
storageType, 674
sMSMessageModeInfo, 673
messageMode, 674
sMSOnIMSInfo, 674
smsOnIMS, 674
sMSOnIMSTlv, 675
IMSInfo, 675
TlvPresent, 675
SMSSupport
pack_dms_SetCustFeature_t, 420
unpack_dms_GetCustFeature_t, 742
sMSTransferRouteMTMessageInfo, 675
ackIndicator, 676
data, 676
format, 676
length, 676
transactionID, 676
SNR
nas_NetworkStatEVDO, 294
sNonIntraSearch
nas_LTEInfoIntrafreq, 260
sNumScellsConfig
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 893
SO
nas_NetworkStat1x, 292
SOLen
nas_DDTMSettings, 220
pack_nas_SLQSSetDDTMPref_t, 500
SOListAction
nas_DDTMSettings, 220
pack_nas_SLQSSetDDTMPref_t, 500
SOMask
currNetworkInfo, 60
wds_currNetworkInfo, 1255
SPC_SIZE
dms.h, 1308
SPTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
sPhyCaAggDlBW
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 893
sPhyCaAggPcellInfo
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 893
sPhyCaAggScellDlBw
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 893
sPhyCaAggScellIndType
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 893
sPhyCaAggScellIndex
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 893
sPhyCaAggScellInfo
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 893
SRTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
SSInfo
unpack_nas_SetServingSystemCallback_ind_t, 894
sSLQSSignalStrengthsInfo
nas_SLQSSignalStrengthsTlv, 358
SSTlv
unpack_nas_SetEventReportInd_t, 891
STTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
SUCCESS
QmuxInternal.h, 1593
SUPSInformation
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1127
SUPSType
pack_voice_SLQSVoiceManageCalls_t, 598
SWI_API
SwiDataTypes.h, 1656
SWIWWANCMAPI.h, 1686
samplesPerBatch
loc_accelAcceptReady, 162
loc_accelTempAcceptReady, 163
loc_gyroAcceptReady, 171
loc_gyroTempAcceptReady, 171
sar.h, 1607
pack_sar_SLQSGetRfSarState, 1608
pack_sar_SLQSSetRfSarState, 1608
unpack_sar_SLQSGetRfSarState, 1608
unpack_sar_SLQSSetRfSarState, 1609
unpack_sar_SLQSSetRfSarState_t, 1608
sbas_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
sbas_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
sbas_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
sbas_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 987
scanStatus
nas_NetworkScanStatusTlv, 290
scell_idx
nas_PhyCaAggScellIndex, 312

NASPhyCaAggScellArray, 396
NASPhyCaAggScellIndex, 397
scell_state
nas_PhyCaAggScellIndType, 312
nas_PhyCaAggScellInfo, 315
NASPhyCaAggScellArray, 396
NASPhyCaAggScellIndType, 398
NASPhyCaAggScellInfo, 399
scrAmrEnable
ims_EnabSCRAMRInfo, 101
scrAmrWBEnable
ims_EnabSCRAMRWBInfo, 101
screeningInd
voice_connectNumInfo, 1230
sduErrorRatio
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
seDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1192
seDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1192
sePCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1192
sePCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 1192
secActivate
uim_fileAttributes, 701
secActivateMask
uim_fileAttributes, 701
secChA
nas_CDMAChannel, 201
secChB
nas_CDMAChannel, 201
secDeactivate
uim_fileAttributes, 701
secDeactivateMask
uim_fileAttributes, 701
secIncrease
uim_fileAttributes, 701
secIncreaseMask
uim_fileAttributes, 701
SecProt
nas_protocolSubtypeElement, 322
secRead
uim_fileAttributes, 701
secReadMask
uim_fileAttributes, 701
secWrite
uim_fileAttributes, 701
secWriteMask
uim_fileAttributes, 701
secdns
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
secdnsv6
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
second
nas_timeInfo, 367
nas_UniversalTime, 377
secondaryDNS
pack_wds_SetDefaultProfile_t, 617
SecondaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 1177
SecondaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 1177
secondaryHA
unpack_wds_GetMobileIPProfile_t, 1155
SectorIDLen
nas_NetworkStatEVDO, 294
secureBootEnabled
unpack_swidms_SLQSSwiDmsGetSecureInfo_t, 1048
selNetwork
nas_servSystem, 351
selected
sms_BroadcastConfig, 658
sms_CDMABroadcastConfig, 659
selectedNetwork
NASServingSystemInfo, 405
selection
pack_swiavms_SLQSAVMSSendSelection_t, 547
pack_swioma_SLQSOMADMSendSelection_t, 559
pack_swioma_SLQSOMADMSendSelectionExt_t, 560
sendStatus
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
sensorData_t, 653
flags, 654
sensorDataLen, 654
timeOfFirstSample, 654
timeOffset, 654
xAxis, 654
yAxis, 654
zAxis, 655
sensorDataLen
sensorData_t, 654
ServerAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 1177
serverAddrTypeMask
pack_loc_SLQSLOCGetServer_t, 450
serverStatus
unpack_loc_GetServer_Ind_t, 856
unpack_loc_SetServer_Ind_t, 872
serverType
pack_loc_SLQSLOCGetServer_t, 450
pack_loc_SLQSLOCSetServer_t, 459
unpack_loc_GetServer_Ind_t, 856
service
CtlService, 59

serviceCategory
sms_CDMABroadcastConfig, 659
serviceDomain
nas_RejectReasonTlv, 329
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 949
ServiceOption
nas_DDTMSettings, 220
pack_nas_SLQSSetDDTMPref_t, 500
ServiceStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 445
serviceType
GetClientIdReqMsg, 94
GetClientIdRspMsg, 95
pack_ctl_GetClientId_t, 415
pack_ctl_ReleaseClientId_t, 415
QmiService, 644
ReleaseClientIdReqMsg, 651
ReleaseClientIdRspMsg, 651
unpack_ctl_GetClientId_t, 736
unpack_ctl_ReleaseClientId_t, 736
servingCellId
nas_LTEInfoIntrafreq, 260
ServingSystem
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
sessExp
ims_SessDurInfo, 106
sessionControl
pack_pds_StartPDSTrackingSessionExt_t, 527
sessionEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 1180
SessionId
pack_loc_Start_t, 461
pack_loc_Stop_t, 461
sessionId
unpack_loc_PositionRpt_Ind_t, 867
sessionInfo
pack_uim_ChangePin_t, 565
pack_uim_ReadTransparent_t, 566
pack_uim_SetPinProtection_t, 567
pack_uim_SLQSUIMAuthenticate_t, 569
pack_uim_SLQSUIMGetFileAttributes_t, 573
pack_uim_SLQSUIMGetServiceStatus_t, 575
pack_uim_SLQSUIMReadRecord_t, 577
pack_uim_SLQSUIMRefreshComplete_t, 578
pack_uim_SLQSUIMRefreshGetLastEvent_t, 578
pack_uim_SLQSUIMRefreshOK_t, 578
pack_uim_SLQSUIMRefreshRegister_t, 579
pack_uim_SLQSUIMSetServiceStatus_t, 580
pack_uim_SLQSUIMWriteRecord_t, 583
pack_uim_SLQSUIMWriteTransparent_t, 584
pack_uim_UnblockPin_t, 585
pack_uim_VerifyPin_t, 586
SessionInfoConfig
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 1055
SessionInfoFota
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 1055
SessionInfoNotification
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 1055
sessionOperation
pack_pds_StartPDSTrackingSessionExt_t, 527
sessionResponse
unpack_swiavms_SLQSAVMSStartSession_t, 1046
sessionServerOption
pack_pds_StartPDSTrackingSessionExt_t, 527
SessionState
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
sessionState
unpack_swioma_SLQSOMADMGetSessionInfo-Ext_t, 1061
SessionStatus
unpack_pds_SetEventReport_Ind_t, 981
sessionStatus
unpack_loc_PositionRpt_Ind_t, 867
unpack_omaDmNotificationsTlv_t, 974
SessionType
pack_swioma_SLQSOMADMGetSessionInfo_t, 559
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
sessionType
pack_pds_StartPDSTrackingSessionExt_t, 527
pack_swiavms_SLQSAVMSStartSession_t, 551
pack_swiavms_SLQSAVMSStopSession_t, 552
pack_swioma_SLQSOMADMCancelSession_t, 558
pack_swioma_SLQSOMADMCancelSessionExt_t, 558
pack_swioma_SLQSOMADMStartSession_t, 562
pack_swioma_SLQSOMADMStartSessionExt_t, 563
uim_refreshevent, 717
uim_sessionInformation, 719
uim_UIMSessionInformation, 726
unpack_omaDmFotaTlv_t, 972
set_fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 557
set_fix_type
pack_swiloc_SwiLocSetAutoStart_t, 557
set_function
pack_swiloc_SwiLocSetAutoStart_t, 557
set_max_dist
pack_swiloc_SwiLocSetAutoStart_t, 557
set_max_time
pack_swiloc_SwiLocSetAutoStart_t, 557
SetFlashSectorSize
lite-fw.h, 1402
setSlot
pack_uim_SLQSUIMSetSIMProfile_t, 581

SetupEventList
cat_EventListTlv, 56
Severity
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
severity
unpack_omaDmFotaTlv_t, 972
shortName
nas_PLMNNetworkNameData, 319
unpack_nas_SLQSGetPLMNName_t, 903
shortNameCI
unpack_nas_SLQSGetPLMNName_t, 903
shortNameEn
unpack_nas_SLQSGetPLMNName_t, 903
shortNameLen
nas_PLMNNetworkNameData, 319
unpack_nas_SLQSGetPLMNName_t, 903
shortNameSB
unpack_nas_SLQSGetPLMNName_t, 903
shortNameSpareBits
nas_PLMNNetworkNameData, 319
shutdownFd
MbimTransport, 192
QmuxTransport, 649
sid
nas_CDMAInfo, 203
nas_sidNid, 351
unpack_nas_GetHomeNetwork_t, 881
unpack_nas_SLQSGetHomeNetwork_t, 898
SidNid
nas_homeSIDNID, 246
SigCompEn
ims_SigCompEnInfo, 106
SigInd
LibPackUMTSReqQoSSigInd, 161
sigInfo
nas_SccRxInfo, 349
signal
voice_signalInfo, 1247
signal_quality
nas_SignalQualityTlv, 352
signal_quality_len
nas_SignalQualityTlv, 352
signal_strength
nas_SignalStrngthTlv, 353
signal_strength_len
nas_SignalStrngthTlv, 353
signalQuality
unpack_nas_SLQSNasPerformIncNetworkScan-Callback_Ind_t, 951
signalStrength
nas_SignalStrengthTlv, 353
unpack_nas_SLQSNasPerformIncNetworkScan-Callback_Ind_t, 951
signalStrengthReqMask
unpack_nas_SLQSGetSignalStrength_t, 915
signalType
voice_signalInfo, 1247
simBusy
uim_simBusyStatus, 720
simBusyLength
uim_simBusyStatus, 720
SimCapability
dms_devCaps, 62
unpack_dms_GetDeviceCap_t, 744
simCapability
unpack_dms_GetDeviceCapabilities_t, 746
SimRejInfo
nas_SimRejInfoTlv, 354
SimVoiceDataCap
dms_devSubsVoiceDataList, 69
sinr
hdrSSInfo, 96
nas_SLQSSignalStrengthsInformation, 357
tdscdmaSigInfoExt, 684
unpack_nas_SLQSGetSignalStrength_t, 915
sinrDelta
nas_SLQSSignalStrengthsIndReq, 356
sinrThresholdList
nas_SLQSSignalStrengthsIndReq, 356
sinrThresholdListLen
nas_SLQSSignalStrengthsIndReq, 356
sku_str
unpack_dms_GetFirmwareInfo_t, 753
skuStr
_litefw_FirmwareFileInfo, 40
slot
pack_uim_SLQSUIMAddProfile_t, 568
pack_uim_SLQSUIMDeleteProfile_t, 569
pack_uim_SLQSUIMPowerDown_t, 575
pack_uim_SLQSUIMPowerUp_t, 576
pack_uim_SLQSUIMUpdateProfile_t, 582
slot_t, 655
bICCID, 655
bICCIDLength, 655
bLogicalSlot, 655
uPhyCardStatus, 656
uPhySlotStatus, 656
slotInf, 656
AppStatus, 657
cardState, 657
errorState, 657
numApp, 657
upinRetries, 657
upinState, 657
upukRetries, 657
SlotInfo
uim_cardStatus, 695
slotNo
pack_uim_SLQSUIMGetProfileInfo_t, 574
pack_uim_SLQSUIMGetSIMProfile_t, 575
slotSelectaction
pack_swidms_SLQSSwiDmsSetUimSlot_t, 554
slots_t, 657
uimSlotStatus, 658
slotsstatusChange

unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 1072
SlqsProfile3GPP
unpackWdsProfileParam, 1202
unpackWdsProfileParamV2, 1203
wds_profileInfo, 1269
SlqsProfile3GPP2
unpackWdsProfileParam, 1202
unpackWdsProfileParamV2, 1203
wds_profileInfo, 1269
sms.h
LITEQMI_QMI_CBK_PARAM_NOCHANGE, 1613
LITEQMI_QMI_CBK_PARAM_RESET, 1613
LITEQMI_QMI_CBK_PARAM_SET, 1613
sms.h, 1609
eqmiCbkSetStatus, 1613
MAX_MSE_TWS_MSG, 1613
MAX_SMS_LIST_SIZE, 1613
pack_sms_GetSMSCAddress, 1613
pack_sms_SLQSDeleteSMS, 1615
pack_sms_SLQSGetIndicationRegister, 1615
pack_sms_SLQSGetMessageWaiting, 1616
pack_sms_SLQSGetSMS, 1616
pack_sms_SLQSGetSMSList, 1617
pack_sms_SLQSGetSmsBroadcastConfig, 1616
pack_sms_SLQSGetTransLayerInfo, 1617
pack_sms_SLQSGetTransNWRegInfo, 1618
pack_sms_SLQSModifySMSStatus, 1618
pack_sms_SLQSSendAsyncSMS, 1618
pack_sms_SLQSSetIndicationRegister, 1619
pack_sms_SLQSSetSmsBroadcastActivation, 1619
pack_sms_SLQSSetSmsBroadcastConfig, 1619
pack_sms_SLQSSetSmsStorage, 1620
pack_sms_SLQSSmsGetMaxStorageSize, 1620
pack_sms_SLQSSmsGetMessageProtocol, 1620
pack_sms_SLQSSmsSetRoutes, 1621
pack_sms_SLQSSwiGetSMSStorage, 1621
pack_sms_SaveSMS, 1614
pack_sms_SendSMS, 1614
pack_sms_SetNewSMSCallback, 1614
pack_sms_SetSMSCAddress, 1615
SMS_CONFIG_LEN, 1613
SMS_MAX_SMS_ROUTES, 1613
SMS_NUM_OF_SET, 1613
SMSC_TYPE_LEN, 1613
unpack_sms_GetSMSCAddress, 1622
unpack_sms_SLQSDeleteSMS, 1624
unpack_sms_SLQSGetIndicationRegister, 1624
unpack_sms_SLQSGetMessageWaiting, 1624
unpack_sms_SLQSGetSMS, 1625
unpack_sms_SLQSGetSMSList, 1625
unpack_sms_SLQSGetSmsBroadcastConfig, 1625
unpack_sms_SLQSGetTransLayerInfo, 1626
unpack_sms_SLQSGetTransNWRegInfo, 1626
unpack_sms_SLQSModifySMSStatus, 1626
unpack_sms_SLQSNWRegInfoCallback_ind, 1627
unpack_sms_SLQSSendAsyncSMS, 1627
unpack_sms_SLQSSetIndicationRegister, 1627
unpack_sms_SLQSSetSmsBroadcastActivation, 1628
unpack_sms_SLQSSetSmsBroadcastConfig, 1628
unpack_sms_SLQSSetSmsStorage, 1628
unpack_sms_SLQSSmsGetMaxStorageSize, 1629
unpack_sms_SLQSSmsGetMessageProtocol, 1629
unpack_sms_SLQSSmsSetRoutes, 1629
unpack_sms_SLQSSwiGetSMSStorage, 1630
unpack_sms_SLQSTransLayerInfoCallback_ind, 1630
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind, 1630
unpack_sms_SLQSWmsMemoryFullCallBack_ind, 1631
unpack_sms_SLQSWmsMessageWaitingCallBack_ind, 1631
unpack_sms_SaveSMS, 1622
unpack_sms_SendSMS, 1622
unpack_sms_SetNewSMSCallback, 1623
unpack_sms_SetNewSMSCallback_ind, 1623
unpack_sms_SetSMSCAddress, 1623
sms_BroadcastConfig, 658
fromServiceId, 658
selected, 658
toServiceId, 658
sms_CDMABroadcastConfig, 658
language, 659
selected, 659
serviceCategory, 659
sms_getIndicationReg, 659
pRegCallStatInfoEvt, 660
pRegTransLayerInfoEvt, 660
pRegTransNWRegInfoEvt, 660
sms_getMsgWaitingInfo, 660
msgWaitInfo, 660
numInstances, 660
sms_getTransLayerInfo, 660
pRegInd, 661
pTransLayerInfo, 661
sms_getTransNWRegInfo, 661
pRegStatus, 661
sms_maxStorageSizeReq, 662
pMessageMode, 662
storageType, 662
sms_maxStorageSizeResp, 662
freeSlots, 663
maxStorageSize, 663
sms_messageWaitingInfoContent, 663
activeInd, 663
msgCount, 663
msgType, 663
sms_msgProtocolResp, 664
msgProtocol, 664
sms_qaQmi3GPP2BroadcastCfgInfo, 664
activated_ind, 665

CDMABroadcastConfig, 665
num_instances, 665
sms_qaQmi3GPPBroadcastCfgInfo, 665
activated_ind, 665
broadcastConfig, 665
num_instances, 665
sms_routeEntry, 665
messageClass, 666
messageType, 666
receiptAction, 666
routeStorage, 666
sms_sendAsyncsmsParams, 666
messageFormat, 668
messageSize, 668
pFollowOnDC, 668
pForceOnDC, 668
pLinktimer, 668
pMessage, 668
pRetryMessage, 668
pRetryMessageId, 668
pServiceOption, 668
pSmsOnIms, 668
pUserData, 668
sms_setIndicationReg, 669
pRegCallStatInfoEvt, 669
pRegTransLayerInfoEvt, 669
pRegTransNWRegInfoEvt, 669
sms_setRoutesReq, 669
numOfRoutes, 670
pTransferStatusReport, 670
routeList, 670
sms_transLayerInfo, 670
TransCap, 670
TransType, 671
smsFormat
ims_SMSFmtInfo, 107
smsOnIMS
sMSOnIMSInfo, 674
SmsRat
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
SmsRatVal
imsa_SmsRatInfo, 114
SmsService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
SmsServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
SmsServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
smsStorage
pack_sms_SLQSSetSmsStorage_t, 539
SmsSvcStatus
imsa_SmsSvcStatusInfo, 114
smsoIPNW
ims_SMSoIPNwInfo, 108
snr
loc_satelliteInfo, 179
lteSSInfo, 184
nas_SccRxInfo, 349
snrlevel
nas_lteSnrinformation, 274
so_mask
DsdSystemInfo_t, 87
DsdUlApnAvailableSystemInfo_t, 89
soMask
qmiWDSDataBearerTechnology, 647
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
Source
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
source
altSrcInfo_t, 43
unpack_dms_GetNetworkTime_t, 760
unpack_dms_GetNetworkTimeV2_t, 761
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 779
unpack_dms_SLQSDmsSwiGetResetInfo_t, 780
wds_sourceOfChange, 1270
sourceIPMask
LibPackTFTIDParams, 158
SourceLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
SourceRAT
imsa_RatHandoverStatusInfo, 113
spc
pack_dms_ResetToFactoryDefaults_t, 417
pack_nas_SetACCOLC_t, 465
pack_wds_SetMobileIPProfile_t, 622
spn
nas_serviceProviderName, 349
unpack_nas_SLQSGetPLMNName_t, 903
spnEncoding
unpack_nas_SLQSGetPLMNName_t, 903
spnLength
nas_serviceProviderName, 349
unpack_nas_SLQSGetPLMNName_t, 903
squal
nas_UMTSExtInfo, 371
srcPortRangeEnd
LibPackTFTIDParams, 158
srcPortRangeStart
LibPackTFTIDParams, 158
srcTlv
unpack_wds_SLQSSwiProfileChangeCallback_Ind_t, 1189
srvCapValid
nas_NR5GSystemInfoTlv, 306
srvCapability
nas_detailSvcInfo, 221
nas_NR5GSysInfoTlv, 303
nas_sysInfoCommon, 362
srvCapabilityValid
nas_NR5GSysInfoTlv, 303

nas_sysInfoCommon, 362
srvDomain
nas_NR5GSysInfoTlv, 303
nas_NR5GSystemInfoTlv, 306
nas_sysInfoCommon, 362
SrvDomainPref
nas_SrvDomainPrefTlv, 358
NASServDomainPrefTlv, 404
srvDomainValid
nas_NR5GSysInfoTlv, 303
nas_NR5GSystemInfoTlv, 306
nas_sysInfoCommon, 362
srvOption
voice_arrSvcOption, 1212
SrvRegRestriction
nas_SrvRegRestrictionTlv, 359
nas_SrvRegRestricTlv, 359
srvStatus
nas_detailSvcInfo, 221
nas_GSMSrvStatusInfo, 236
nas_NR5GSerStatTlv, 298
nas_NR5GSrvStatusTlv, 299
nas_SrvStatusInfo, 360
srvcapability
nas_NR5GSystemInfoTlv, 306
srxLvl
nas_UMTSExtInfo, 371
srxlev
nas_cellParams, 208
nas_gsmCellInfo, 234
nas_umtsLTENbrCell, 376
nas_wcdmaCellInfo, 379
stage
uim_refreshevent, 717
State
nas_NetworkStat1x, 292
nas_NetworkStatEVDO, 294
state
pack_loc_SLQSLOCSetCradleMountConfig_t, 458
unpack_dms_GetActivationState_t, 737
unpack_dms_SLQSUIMGetState_t, 803
unpack_omaDmConfigTlv_t, 971
unpack_omaDmFotaTlv_t, 973
unpack_qos_QosFlowInfoState_t, 993
statmask
pack_wds_GetPacketStatus_t, 614
StatsMask
wds_transferStatInd, 1271
statsMask
wds_TrStatInd, 1271
StatsPeriod
wds_transferStatInd, 1271
statsPeriod
wds_TrStatInd, 1271
Status
MBIM_CLOSE_DONE_MSG, 184
MBIM_COMMAND_DONE_MSG, 185
MBIM_OPEN_DONE_MSG, 187
unpack_loc_GetOpMode_Ind_t, 854
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
status
loc_FixCriteriaStatusTlv, 168
MbimSyncObject, 190
MbimTransaction, 191
nas_lteEARFCN, 253
nas_ltePCI, 269
nas_wcdmaUARFCN, 386
pack_sms_SetNewSMSCallback_t, 533
unpack_loc_BestAvailPos_Ind_t, 847
unpack_loc_DeleteAssistData_Ind_t, 849
unpack_loc_InjectPositionCallback_Ind_t, 857
unpack_loc_InjectUTCTimeCallback_Ind_t, 860
unpack_loc_SetExtPowerConfig_Ind_t, 869
unpack_loc_SetOperationMode_Ind_t, 870
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 998
unpack_qos_SLQSSetQosStatusCallback_ind_t, 1000
unpack_swioma_SLQSOMADMGetSessionInfoExt_t, 1061
storageIndex
FMSImageIdElement, 92
storageType
pack_sms_SaveSMS_t, 532
pack_sms_SLQSDeleteSMS_t, 534
pack_sms_SLQSGetSMS_t, 535
pack_sms_SLQSGetSMSList_t, 536
pack_sms_SLQSModifySMSStatus_t, 537
sms_maxStorageSizeReq, 662
sMSMTMessageInfo, 674
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 1032
Streaming Download Protocol (lite-fw), 34
String
unpack_dms_GetDeviceHardwareRev_t, 749
unpack_dms_GetDeviceMfr_t, 750
unpack_dms_GetFSN_t, 756
unpack_dms_UIMGetICCID_t, 806
stringSize
unpack_dms_GetDeviceHardwareRev_t, 749
unpack_dms_GetDeviceMfr_t, 750
unpack_dms_UIMGetICCID_t, 806
subAddr
voice_calledPartySubAdd, 1216
subAddrLen
voice_calledPartySubAdd, 1216
subAddrType
voice_calledPartySubAdd, 1216
subnetMask
unpack_qos_IPv4Addr_t, 989
SubnetMaskV4
unpack_wds_SLQSGetRuntimeSettings_t, 1177
SubsCapList
dms_devCurSubsCaps, 63
SubsCfgList

dms_devMultiSimCaps, 65
SubsCfgListLen
dms_devMultiSimCaps, 65
SubsDevList
dms_devMaxCfgListCaps, 64
SubsFeatureLen
dms_devSubsFeatureModeCaps, 67
SubsFeatureList
dms_devSubsFeatureModeCaps, 67
SubsList
dms_devSubsCfgList, 66
dms_devSubsList, 67
SubsListLen
dms_devSubsCfgList, 66
dms_devSubsList, 67
subsType
pack_voice_SLQSVoiceBindSubscription_t, 589
SubsVoiceDataCap
dms_devSubsVoiceDataList, 69
SubsVoiceDataCapLen
dms_devSubsVoiceDataCaps, 68
SubsVoiceDataList
dms_devSubsVoiceDataCaps, 68
subscrTmr
ims_SubscrTmrInfo, 108
SupUSBComps
unpack_dms_GetUSBComp_t, 767
SuppOA
voice_CUGInfo, 1230
SuppPrefCUG
voice_CUGInfo, 1230
supportedLteBandLen
dms_LteBandsSupport, 69
svInfoMask
loc_satelliteInfo, 179
svListLen
loc_satelliteInfo, 180
svStatus
loc_satelliteInfo, 180
svc
pack_qmi_t, 528
SvcClass
voice_callFWExtInfo, 1220
voice_callFWInfo, 1221
SvcStatus
voice_callFWExtInfo, 1220
voice_callFWInfo, 1221
svcType
voice_ccSUPSType, 1225
voice_SUPSInfo, 1248
sw1
uim_cardResult, 693
uim_cardResultInfo, 693
sw2
uim_cardResult, 693
uim_cardResultInfo, 694
swVerString
pack_dms_SLQSSwiSetHostDevInfo_t, 427
unpack_dms_SLQSSwiGetHostDevInfo_t, 799
swi_uint256_clear_bit
switype_256bit.h, 1685
swi_uint256_get_bit
switype_256bit.h, 1685
swi_uint256_print_mask
switype_256bit.h, 1685
swi_uint256_set_bit
switype_256bit.h, 1685
swi_uint256_t, 676
word, 676
SwiDataTypes.h, 1655
BOOL, 1656
BYTE, 1656
CHAR, 1656
FLOAT, 1656
INT32, 1656
INT8, 1656
LPCSTR, 1656
qmuint16, 1657
qmulong, 1657
SHORT, 1657
SWI_API, 1656
ULONG, 1657
ULONGLONG, 1657
UNUSEDPARAM, 1656
USHORT, 1657
WORD, 1657
swiaudio.h, 1632
pack_swiaudio_SLQSGetM2MAVMute, 1634
pack_swiaudio_SLQSGetM2MAudioProfile, 1633
pack_swiaudio_SLQSGetM2MAudioVolume, 1634
pack_swiaudio_SLQSGetM2MSpkrGain, 1635
pack_swiaudio_SLQSSetM2MAVMute, 1637
pack_swiaudio_SLQSSetM2MAudioAVCFG, 1635
pack_swiaudio_SLQSSetM2MAudioLPBK, 1635
pack_swiaudio_SLQSSetM2MAudioNVDef, 1636
pack_swiaudio_SLQSSetM2MAudioProfile, 1636
pack_swiaudio_SLQSSetM2MAudioVolume, 1636
pack_swiaudio_SLQSSetM2MSpkrGain, 1637
unpack_swiaudio_SLQSGetM2MAVMute, 1638
unpack_swiaudio_SLQSGetM2MAudioProfile, 1638
unpack_swiaudio_SLQSGetM2MAudioVolume, 1638
unpack_swiaudio_SLQSGetM2MSpkrGain, 1639
unpack_swiaudio_SLQSSetM2MAVMute, 1641
unpack_swiaudio_SLQSSetM2MAVMute_t, 1633
unpack_swiaudio_SLQSSetM2MAudioAVCFG, 1639
unpack_swiaudio_SLQSSetM2MAudioAVCFG_t, 1633
unpack_swiaudio_SLQSSetM2MAudioLPBK, 1639
unpack_swiaudio_SLQSSetM2MAudioLPBK_t, 1633
unpack_swiaudio_SLQSSetM2MAudioNVDef, 1640

unpack_swiaudio_SLQSSetM2MAudioNVDef_t, 1633
unpack_swiaudio_SLQSSetM2MAudioProfile, 1640
unpack_swiaudio_SLQSSetM2MAudioProfile_t, 1633
unpack_swiaudio_SLQSSetM2MAudioVolume, 1640
unpack_swiaudio_SLQSSetM2MAudioVolume_t, 1633
unpack_swiaudio_SLQSSetM2MSpkrGain, 1641
unpack_swiaudio_SLQSSetM2MSpkrGain_t, 1633
swiaudio_PCMparams, 676
iFaceTab, 677
iFaceTabLen, 677
swiavms.h, 1641
pack_swiavms_SLQSAVMSGetSettings, 1644
pack_swiavms_SLQSAVMSGetSettings_v2, 1644
pack_swiavms_SLQSAVMSSendSelection, 1645
pack_swiavms_SLQSAVMSSessionGetInfo, 1646
pack_swiavms_SLQSAVMSSetSettings, 1647
pack_swiavms_SLQSAVMSSetSettings_v2, 1647
pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField, 1648
pack_swiavms_SLQSAVMSStartSession, 1649
pack_swiavms_SLQSAVMSStopSession, 1649
pack_swiavms_SLQSAvmsSetEventReport, 1646
unpack_swiavms_SLQSAVMSEventReportInd, 1650
unpack_swiavms_SLQSAVMSGetSettings, 1650
unpack_swiavms_SLQSAVMSGetSettings_v2, 1651
unpack_swiavms_SLQSAVMSSendSelection, 1651
unpack_swiavms_SLQSAVMSSessionGetInfo, 1652
unpack_swiavms_SLQSAVMSSetSettings, 1653
unpack_swiavms_SLQSAVMSSetSettings_v2, 1653
unpack_swiavms_SLQSAVMSStartSession, 1654
unpack_swiavms_SLQSAVMSStopSession, 1654
unpack_swiavms_SLQSAVMSStopSession_avc2, 1655
unpack_swiavms_SLQSAVMSStopSession_avc2_t, 1644
unpack_swiavms_SLQSAvmsSetEventReport, 1652
swidms.h, 1657
pack_swidms_SLQSSwiDmsGetHWWatchdog, 1658
pack_swidms_SLQSSwiDmsGetMTU, 1659
pack_swidms_SLQSSwiDmsGetSecureInfo, 1659
pack_swidms_SLQSSwiDmsGetUimSlot, 1659
pack_swidms_SLQSSwiDmsGetUsbComp, 1660
pack_swidms_SLQSSwiDmsGetUsbNetNum, 1660
pack_swidms_SLQSSwiDmsSetHWWatchdog, 1660
pack_swidms_SLQSSwiDmsSetMTU, 1661
pack_swidms_SLQSSwiDmsSetUimSlot, 1661
pack_swidms_SLQSSwiDmsSetUsbComp, 1661
pack_swidms_SLQSSwiDmsSetUsbNetNum, 1662
unpack_swidms_SLQSSwiDmsGetHWWatchdog, 1662
unpack_swidms_SLQSSwiDmsGetMTU, 1662
unpack_swidms_SLQSSwiDmsGetSecureInfo, 1663
unpack_swidms_SLQSSwiDmsGetUimSlot, 1663
unpack_swidms_SLQSSwiDmsGetUsbComp, 1664
unpack_swidms_SLQSSwiDmsGetUsbNetNum, 1664
unpack_swidms_SLQSSwiDmsSetHWWatchdog, 1664
unpack_swidms_SLQSSwiDmsSetMTU, 1665
unpack_swidms_SLQSSwiDmsSetUimSlot, 1665
unpack_swidms_SLQSSwiDmsSetUsbComp, 1665
unpack_swidms_SLQSSwiDmsSetUsbNetNum, 1666
unpack_swidms_SLQSSwiDmsSetUsbNetNum_t, 1658
swidms_AutoUimSwitchModeTlv, 677
TlvPresent, 677
uimSlot, 677
swidms_MultiplexedSimModeTlv, 680
mode, 681
TlvPresent, 681
uimSlot, 681
swidms_SetMultiplexedModeTlv, 681
mode, 681
uimSlot, 681
swidms_SwiDmsGetHWWatchdog, 682
count, 682
enable, 682
resetDelay, 682
timeout, 682
swidms_UimSlotSelectedTlv, 683
TlvPresent, 683
uimSlot, 683
swidms_ehrpdMTUSizeTlv, 677
ehrpdMTUSize, 678
TlvPresent, 678
swidms_hrpdMTUSizeTlv, 678
hrpdMTUSize, 678
TlvPresent, 678
swidms_intfaceCfgTlv, 678
CfgValue, 679
CurrentCfgType, 679
TlvPresent, 679
swidms_mtuSize3gppTlv, 679
MTUSize3gpp, 680
TlvPresent, 680
swidms_supportedIntBitmaskTlv, 681
TlvPresent, 682

ValidBitmasks, 682
swidms_usbMTUSizeTlv, 683
TlvPresent, 683
UsbMTUSize, 684
swiloc.h, 1666
pack_swiloc_SwiLocGetAutoStart, 1667
pack_swiloc_SwiLocSetAutoStart, 1667
unpack_swiloc_SwiLocGetAutoStart, 1667
unpack_swiloc_SwiLocSetAutoStart, 1668
unpack_swiloc_SwiLocSetAutoStart_t, 1666
swioma.h, 1668
pack_swioma_SLQSOMADMAlertCallback, 1669
pack_swioma_SLQSOMADMCancelSession, 1670
pack_swioma_SLQSOMADMGetSessionInfo, 1671
pack_swioma_SLQSOMADMGetSettings, 1671
pack_swioma_SLQSOMADMSendSelection, 1672
pack_swioma_SLQSOMADMSetSettings, 1672
pack_swioma_SLQSOMADMStartSession, 1673
unpack_swioma_SLQSOMADMAlertCallback, 1674
unpack_swioma_SLQSOMADMAlertCallback_ind, 1674
unpack_swioma_SLQSOMADMAlertCallback_t, 1669
unpack_swioma_SLQSOMADMCancelSession, 1675
unpack_swioma_SLQSOMADMCancelSession_t, 1669
unpack_swioma_SLQSOMADMGetSessionInfo, 1675
unpack_swioma_SLQSOMADMGetSettings, 1676
unpack_swioma_SLQSOMADMSendSelection, 1676
unpack_swioma_SLQSOMADMSendSelection_t, 1669
unpack_swioma_SLQSOMADMSetSettings, 1677
unpack_swioma_SLQSOMADMSetSettings_t, 1669
unpack_swioma_SLQSOMADMStartSession, 1677
swiomaext.h, 1678
pack_swioma_SLQSOMADMCancelSessionExt, 1679
pack_swioma_SLQSOMADMGetSessionInfoExt, 1679
pack_swioma_SLQSOMADMSendSelectionExt, 1680
pack_swioma_SLQSOMADMSetSettingsExt, 1680
pack_swioma_SLQSOMADMStartSessionExt, 1681
unpack_swioma_SLQSOMADMCancelSessionExt, 1682
unpack_swioma_SLQSOMADMCancelSessionExt_t, 1678
unpack_swioma_SLQSOMADMGetSessionInfoExt, 1682
unpack_swioma_SLQSOMADMSendSelectionExt, 1683
unpack_swioma_SLQSOMADMSendSelectionExt_t, 1679
unpack_swioma_SLQSOMADMSetSettingsExt, 1683
unpack_swioma_SLQSOMADMSetSettingsExt_t, 1679
unpack_swioma_SLQSOMADMStartSessionExt, 1684
unpack_swioma_SLQSOMADMStartSessionExt_t, 1679
switchOption
pack_voice_SLQSVoiceALSSetLineSwitching_t, 588
switype_256bit.h, 1684
logger, 1685
swi_uint256_clear_bit, 1685
swi_uint256_get_bit, 1685
swi_uint256_print_mask, 1685
swi_uint256_set_bit, 1685
sysForbidden
nas_NR5GSystemInfoTlv, 306
sysForbiddenValid
nas_NR5GSystemInfoTlv, 306
sysInfoCDMA
nas_CDMASysInfo, 207
sysInfoGSM
nas_GSMSysInfo, 239
sysInfoHDR
nas_HDRSysInfo, 245
sysInfoLTE
nas_LTESysInfo, 278
sysInfoWCDMA
nas_WCDMASysInfo, 385
system
loc_satelliteInfo, 180
loc_SV, 181
systemDiscontinuities
pack_pds_PDSInjectTimeReference_t, 517
SystemID
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912
systemID
nas_CDMASysInfo, 207
systemMode
nas_CommInfo, 212
systemTime
pack_pds_PDSInjectTimeReference_t, 517
szAPN
PackSwiAvmsSetSettingsAPNInfo, 638
szAlertMsg
UnpackSwiAvmsEventReportConfig, 1196
szCarrier_str
_litefw_FirmwareInfo_, 41
szCarrierPriversion_str
_litefw_FirmwareInfo_, 41
szDescription

UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1194
szFwversion_str
_litefw_FirmwareInfo_, 41
szModelid_str
_litefw_FirmwareInfo_, 41
szName
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1194
szPWD
PackSwiAvmsSetSettingsAPNInfo, 638
szPackageid_str
_litefw_FirmwareInfo_, 41
szPartno_str
_litefw_FirmwarePartNo_, 42
szSku_str
_litefw_FirmwareInfo_, 41
szUname
PackSwiAvmsSetSettingsAPNInfo, 638
szVersion
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1195

t3396_apn
unpack_nas_SLQSNasTimerCallback_ind_t, 957
t3396_plmn_id
unpack_nas_SLQSNasTimerCallback_ind_t, 957
t3396_val
unpack_nas_SLQSNasTimerCallback_ind_t, 957
TCPDstPort
unpack_qos_swiQosFilter_t, 1004
TCPSrcPort
unpack_qos_swiQosFilter_t, 1004
TDSCDMAECIOThreshListLen
nas_TDSCDMAECIOThresh, 364
TDSCDMARSCPThreshListLen
nas_TDSCDMARSCPThresh, 364
TDSCDMARSSIThreshListLen
nas_TDSCDMARSSIThresh, 365
TDSCDMASINRThreshListLen
nas_TDSCDMASINRThresh, 366
TLVPresent
dms_LteBandsSupport, 69
nas_ImsVoiceSupportLteTlv, 247
nas_LteEmbmsCoverageTlv, 254
nas_LteEmbmsTraceIdTlv, 254
nas_LteRegDomainTlv, 270
nas_LteVoiceDomainTlv, 279
nas_MNCPCSDigitStatus, 282
nas_NR5GCellStatusInfoTlv, 296
nas_NR5GSrvStatusTlv, 299
nas_NR5GSysInfoTlv, 303
nas_SimRejInfoTlv, 354
nas_SrvRegRestrictionTlv, 359
TMD_MAX_DEV_LIST
tmd.h, 1686
TMTlv
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 825
TPCause
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
TRMessageTlv
unpack_sms_SetNewSMSCallback_ind_t, 1017
TT1Tlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
TT2Tlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
TTfTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 829
TX_PWR
nas_NetworkStat1x, 292
TXBOTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1165
TXChan
nas_LTEInfo, 258
tXDroppedCount
unpack_wds_GetPacketStatus_t, 1159
tXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 1159
tXOkBytesCount
unpack_wds_GetPacketStatus_t, 1159
tXPacketErrors
unpack_wds_GetPacketStatus_t, 1159
tXPacketOverflows
unpack_wds_GetPacketStatus_t, 1159
tXPacketSuccesses
unpack_wds_GetPacketStatus_t, 1159
Tables, 37
tac
nas_LTEInfoIntrafreq, 260
nas_LTESysInfo, 278
nas_NR5GSysInfoTlv, 303
nas_NR5GSystemInfoTlv, 306
tacValid
nas_LTESysInfo, 278
nas_NR5GSysInfoTlv, 303
nas_NR5GSystemInfoTlv, 306
TargetRAT
imsa_RatHandoverStatusInfo, 113
Tcp_coalescing
unpack_wda_GetDataFormat_t, 1136
TdsBandCapability
unpack_dms_SLQSGetBandCapability_t, 786
unpack_dms_SLQSGetBandCapabilityExt_t, 790
TdscdmaBandPref
nas_TDSCDMABandPrefTlv, 363
tdscdmaSigInfoExt, 684
ecio, 684
rscp, 684
rssi, 684
sinr, 684
techName
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 1180
Technology
nas_DeviceConfigDetail, 222

unpack_wds_SLQSGetRuntimeSettings_t, 1177
technology
DsdSystemInfo_t, 87
DsdUIApnAvailableSystemInfo_t, 89
TechnologyPref
pack_nas_SetNetworkPreference_t, 467
tempData_t, 684
temperature, 685
temperatureDataLen, 685
timeOfFirstSample, 685
timeOffset, 685
timeSource, 685
TempStat
dms_TemperatureTlv, 76
TempTlv
unpack_dms_SwiEventReportCallBack_ind_t, 804
Temperature
dms_TemperatureTlv, 76
temperature
nas_CommInfo, 212
tempData_t, 685
temperatureDataLen
tempData_t, 685
threshGsmHigh
nas_lteGsmCellInfo, 256
threshGsmLow
nas_lteGsmCellInfo, 256
threshServingLow
nas_LTEInfoIntrafreq, 260
threshXHigh
nas_infoInterFreq, 248
threshXLow
nas_infoInterFreq, 248
threshXhigh
nas_lteWcdmaCellInfo, 279
threshXlow
nas_lteWcdmaCellInfo, 280
ThrottleTimer
LibPackPDNThrottleTimer, 119
Time
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
time
NASTimeInfoTlv, 406
Time_uncert_ms
unpack_pds_SLQSGetGPSStateInfo_t, 987
timeBase
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 522
TimeLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
timeMsec
pack_loc_SLQSLOCInjectUTCTime_t, 457
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 522
timeOfFirstSample
sensorData_t, 654
tempData_t, 685
timeOffset
sensorData_t, 654
tempData_t, 685
timeSource
tempData_t, 685
TimeStmp_gps_week
unpack_pds_SLQSGetGPSStateInfo_t, 987
TimeStmp_tow_ms
unpack_pds_SLQSGetGPSStateInfo_t, 987
timeSyncRefCounter
unpack_loc_EventTimeSyncCallback_Ind_t, 852
timeTlv
NASQmiCbkNasSwiOTAMessageInd, 400
timeUncMsec
pack_loc_SLQSLOCInjectUTCTime_t, 457
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 523
timeZone
nas_timeInfo, 368
timeout
pack_pds_SetPDSDefaults_t, 519
pack_qmi_t, 528
pack_swidms_SLQSSwiDmsSetHWWatchdog_t, 552
swidms_SwiDmsGetHWWatchdog, 682
Timers
PackSwiAvmsSetSettingsConnectionRetryTimers, 639
PackSwiAVMSSettingsConnectionRetryTimers, 641
timestamp
unpack_dms_GetNetworkTime_t, 760
unpack_dms_GetNetworkTimeV2_t, 761
timestampAge
pack_loc_SLQSLOCInjectPosition_t, 455
timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 455
timingAdvance
nas_GERANInfo, 232
tlv2Length
GetClientIdRspMsg, 95
ReleaseClientIdRspMsg, 651
tlv2Type
GetClientIdRspMsg, 95
ReleaseClientIdRspMsg, 651
tlvLength
GetClientIdReqMsg, 94
GetClientIdRspMsg, 95
ReleaseClientIdReqMsg, 651
ReleaseClientIdRspMsg, 651
ResultTlv, 652
TlvPresenceMask
qmTlvResult, 647
TlvPresent
cat_commonEventTlv, 54
dms_ActivationStatusTlv, 61
dms_OperatingModeTlv, 70

dms_PSMActiveTimerIndTlv, 70
dms_PSMActiveTimerTlv, 71
dms_PSMDurationDueToOOSTlv, 71
dms_PSMDurationThresholdTlv, 72
dms_PSMEarlyWakeupTimeTlv, 72
dms_PSMEnableStateIndTlv, 73
dms_PSMEnableStateTlv, 74
dms_PSMPeriodicUpdateTimerIndTlv, 74
dms_PSMPeriodicUpdateTimerTlv, 75
dms_PSMRandomizationWindowTlv, 75
dms_TemperatureTlv, 76
dms_UimAutoSwitchActSlotTlv, 76
dms_UimStatusTlv, 77
dms_VoltageTlv, 78
eTWSPLMNInfoTlv, 90
ims_AMRModeInfo, 98
ims_AMROctAlgnInfo, 98
ims_AMRWBModeInfo, 99
ims_AMRWBOctAlgnInfo, 99
ims_CSCFPortNameInfo, 100
ims_EnabAMRWBInfo, 100
ims_EnabSCRAMRInfo, 101
ims_EnabSCRAMRWBInfo, 101
ims_IMSDomainInfo, 102
ims_IMSTestModeInfo, 102
ims_MinSessExpInfo, 103
ims_PCSCFPortInfo, 103
ims_PhCtxtURIInfo, 104
ims_RngBkTmrInfo, 104
ims_RngTmrInfo, 105
ims_RTPRTCPInactTmrDurInfo, 105
ims_SessDurInfo, 106
ims_SigCompEnInfo, 106
ims_SIPPortInfo, 107
ims_SIPRegnTmrInfo, 107
ims_SMSFmtInfo, 108
ims_SMSoIPNwInfo, 108
ims_SubscrTmrInfo, 108
ims_TmrT1Info, 109
ims_TmrT2Info, 109
ims_TmrTfInfo, 110
imsa_IMSFailErrCodeTlv, 110
imsa_IMSRegStatusErrorCodeInfo, 111
imsa_IMSRegStatusInfo, 112
imsa_NewIMSRegStatusInfo, 112
imsa_RatHandoverStatusInfo, 113
imsa_SmsRatInfo, 114
imsa_SmsSvcStatusInfo, 114
imsa_UtRatInfo, 115
imsa_UtSvcStatusInfo, 115
imsa_VoipRatInfo, 116
imsa_VoipSvcStatusInfo, 116
imsa_VtRatInfo, 117
imsa_VtSvcStatusInfo, 117
loc_AppProviderInfoTlv, 164
loc_FixCriteriaStatusTlv, 168
loc_HorAccuracyLvlTlv, 172
loc_IntermediateRptStateTlv, 173
loc_IPv4Info, 174
loc_IPv6Info, 175
loc_MinIntervalTlv, 177
loc_urlAddr, 183
messageModeTlv, 193
nas_AcqOrderPrefTlv, 194
nas_BandPrefInfoTlv, 197
nas_BandPrefTlv, 199
nas_CiotAcqOrderPrefTlv, 210
nas_CiotLteOpModePrefTlv, 210
nas_CsgId, 212
nas_CSGInfoTlv, 215
nas_EdrxCiotLteMode, 225
nas_EdrxCycleLength, 226
nas_EdrxEnableType, 226
nas_EdrxPagingTimeWindow, 227
nas_EdrxRatType, 227
nas_EmerModeTlv, 228
nas_ForbiddenNetworks3GPP, 230
nas_GWAcqOrderPrefTlv, 239
nas_homeNwMNC3GppTlv, 246
nas_LTEBandPrefTlv, 252
nas_LteCiotOpModeTlv, 252
nas_LteEarfcnInfo, 253
nas_LteM1BandPrefTlv, 264
nas_LteNb1BandPrefTlv, 266
nas_LTEOperationalModeTlv, 266
nas_LTEOperationMode, 267
nas_LteOpMode, 267
nas_lteOpModeTlv, 268
nas_ModePrefTlv, 283
nas_NetSelPrefTlv, 285
nas_NetworkNameSrcTlv, 286
nas_networkNameSrcTlv, 286
nas_NetworkScan3GPPInfoTlv, 289
nas_NetworkScanStatusTlv, 290
nas_NR5GCellStatusTlv, 297
nas_NR5GSerStatTlv, 298
nas_NR5GSystemInfoTlv, 306
nas_NumScellsConfig, 307
nas_nwNameSrc3GppTlv, 307
nas_PhyCaAggDlBW, 310
nas_PhyCaAggPcellInfo, 311
nas_PhyCaAggScellDlBw, 311
nas_PhyCaAggScellIndex, 312
nas_PhyCaAggScellIndType, 312
nas_PhyCaAggScellInfo, 315
nas_PlmnID, 317
nas_PRLPrefTlv, 321
nas_RankIndicatorTlv, 328
nas_RatDisabledMaskTlv, 329
nas_RejectReasonTlv, 329
nas_RfBandInfoExtFormat, 331
nas_RFBandInfoExtTlv, 333
nas_RFBandInfoTlv, 334
nas_RfBandwidthInfo, 335
nas_RFBandwidthInfoTlv, 337
nas_RfDedicatedBandInfo, 338

nas_RFDedicatedBandInfoTlv, 340
nas_RFInfoTlv, 341
nas_RoamPrefTlv, 342
nas_SccRxInfo, 349
nas_SignalQualityTlv, 352
nas_SignalStrengthTlv, 353
nas_SignalStrngthTlv, 353
nas_SLQSSignalStrengthsTlv, 358
nas_SrvDomainPrefTlv, 358
nas_SrvRegRestricTlv, 359
nas_TDSCDMABandPrefTlv, 363
nas_timeInfo, 368
nas_UMTSExtInfo, 371
nas_UsageSettingTlv, 377
nas_VoiceDomainPrefTlv, 378
nas_WCDMACellInfoExt, 380
NASAcqOrderPrefTlv, 386
NASBandPreferenceTlv, 387
NASCiotAcqOrderPrefTlv, 387
NASCiotLteOpModePrefTlv, 388
NASEmergencyModeTlv, 388
NASGWAcqOrderPrefTlv, 390
NASLTEBandPreferenceTlv, 390
NASLteM1BandPrefTlv, 391
NASLteNasReleaseInfoTlv, 391
NASLteNB1BandPrefTlv, 392
NASModePreferenceTlv, 392
NASNetSelPreferenceTlv, 392
NASNr5gBandPrefTlv, 393
NASOTAMessageTlv, 394
NASPhyCaAggPcellInfo, 395
NASPhyCaAggScellArray, 396
NASPhyCaAggScellDlBw, 397
NASPhyCaAggScellIndex, 397
NASPhyCaAggScellIndType, 398
NASPhyCaAggScellInfo, 399
NASPRLPreferenceTlv, 400
NASRatDisabledMaskTlv, 403
NASRoamPreferenceTlv, 403
NASServDomainPrefTlv, 404
NASTimeInfoTlv, 406
newMTMessageTlv, 406
sMSCAddressTlv, 672
sMSEtwsMessageTlv, 673
sMSOnIMSTlv, 675
swidms_AutoUimSwitchModeTlv, 677
swidms_ehrpdMTUSizeTlv, 678
swidms_hrpdMTUSizeTlv, 678
swidms_intfaceCfgTlv, 679
swidms_mtuSize3gppTlv, 680
swidms_MultiplexedSimModeTlv, 681
swidms_supportedIntBitmaskTlv, 682
swidms_UimSlotSelectedTlv, 683
swidms_usbMTUSizeTlv, 683
transferRouteMessageTlv, 686
uim_activeProfileInfo, 687
uim_additionalReadResult, 688
uim_cardResultInfo, 694
uim_eidInfo, 697
uim_getSimProfileInfo, 703
uim_GetSlotsInfoTlv, 703
uim_GetSlotsStatusTlv, 704
uim_indToken, 705
uim_profileClassTlv, 708
uim_profileICCIDTlv, 709
uim_profileIconTlv, 709
uim_profileIconTypeTlv, 710
uim_profileNameTlv, 711
uim_profileNicknameTlv, 711
uim_profilePolicyRulesTlv, 712
uim_profileSPNTlv, 712
uim_profileStateTlv, 713
uim_readResultInfo, 715
uim_UIMGetFDNStatus, 723
uim_UIMGetHiddenKeyStatus, 723
uim_UIMGetIndex, 724
unpack_pds_SetPdsState_Ind_t, 981
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 1082
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1195
UnpackSwiAvmsEventReportConfig, 1196
UnpackSwiAvmsEventReportConnectionRequest, 1196
UnpackSwiAvmsEventReportDataSessionStatus, 1197
UnpackSwiAvmsEventReportHTTPStatus, 1198
UnpackSwiAvmsEventReportNotification, 1200
UnpackSwiAvmsEventReportPackageID, 1200
UnpackSwiAvmsEventReportRegStatus, 1200
UnpackSwiAvmsEventReportSessionType, 1201
UnpackSwiAvmsEventReportWAMSParam-Change, 1202
voice_VoiceUssdNoWaitAlphaIDTlv, 1251
voice_VoiceUssdNoWaitErrorTlv, 1252
voice_VoiceUssdNoWaitFailCauseTlv, 1252
voice_VoiceUssdNoWaitUSSINFOTlv, 1253
wds_ChannelRateTlv, 1254
wds_ConnStatusTlv, 1254
wds_DataBearTechTlv, 1256
wds_DataULongLongTlv, 1256
wds_DataULongTlv, 1257
wds_DHCPLeaseOptTlv, 1257
wds_DHCPLeaseStateTlv, 1258
wds_DHCPProfileIdTlv, 1259
wds_DormStatTlv, 1263
wds_IPv4AdTlv, 1264
wds_LastMdmCallEndRsnTlv, 1266
wds_RXBytesOKTlv, 1270
wds_TXBytesOKTlv, 1272
tlvPresent
voice_USSDNotificationNetworkInfo, 1249
TlvResult
unpack_swidms_SLQSSwiDmsGetSecureInfo_t, 1048
TlvResultCode

qmTlvResult, 647
tlvType
GetClientIdReqMsg, 94
GetClientIdRspMsg, 95
ReleaseClientIdReqMsg, 651
ReleaseClientIdRspMsg, 651
ResultTlv, 652
Tlvresult
pack_dms_GetCustFeaturesV2_t, 416
pack_dms_SetCustFeaturesV2_t, 421
pack_dms_SetPower_t, 423
pack_dms_SetUSBComp_t, 423
pack_dms_UIMGetICCID_t, 430
pack_fms_GetImagesPreference_t, 435
pack_fms_GetStoredImages_t, 436
pack_fms_SetImagesPreference_t, 437
pack_loc_Delete_Assist_Data_t, 446
pack_loc_EventRegister_t, 448
pack_loc_SetExtPowerState_t, 448
pack_loc_SetOperationMode_t, 449
pack_loc_SLQSLOCGetBestAvailPos_t, 449
pack_loc_Start_t, 461
pack_loc_Stop_t, 461
pack_nas_SetNetworkPreference_t, 468
pack_uim_ChangePin_t, 565
pack_uim_ReadTransparent_t, 566
pack_uim_SetPinProtection_t, 567
pack_uim_UnblockPin_t, 585
pack_uim_VerifyPin_t, 586
unpack_cat_SetCATEventCallback_t, 735
unpack_ctl_GetClientId_t, 736
unpack_ctl_ReleaseClientId_t, 737
unpack_dms_GetBandCapability_t, 739
unpack_dms_GetCrashAction_t, 740
unpack_dms_GetCustFeature_t, 742
unpack_dms_GetCustFeaturesV2_t, 743
unpack_dms_GetDeviceCap_t, 744
unpack_dms_GetDeviceCapabilitiesV2_t, 748
unpack_dms_GetDeviceHardwareRev_t, 749
unpack_dms_GetDeviceMfr_t, 750
unpack_dms_GetDeviceSerialNumbers_t, 752
unpack_dms_GetFirmwareInfo_t, 753
unpack_dms_GetFirmwareRevision_t, 754
unpack_dms_GetFirmwareRevisions_t, 756
unpack_dms_GetFSN_t, 756
unpack_dms_GetIMSI_t, 758
unpack_dms_GetManufacturer_t, 758
unpack_dms_GetModelID_t, 759
unpack_dms_GetNetworkTime_t, 760
unpack_dms_GetNetworkTimeV2_t, 761
unpack_dms_GetOfflineReason_t, 762
unpack_dms_GetPower_t, 763
unpack_dms_GetPRLVersion_t, 764
unpack_dms_GetUSBComp_t, 767
unpack_dms_GetVoiceNumber_t, 768
unpack_dms_PSMCfgChange_ind_t, 769
unpack_dms_ResetToFactoryDefaults_t, 770
unpack_dms_SetActivationStatusCallback_t, 770
unpack_dms_SetCustFeature_t, 771
unpack_dms_SetCustFeaturesV2_t, 772
unpack_dms_SetEventReport_ind_t, 773
unpack_dms_SetEventReport_t, 773
unpack_dms_SetFirmwarePreference_t, 774
unpack_dms_SetIndicationRegister_t, 774
unpack_dms_SetPower_t, 775
unpack_dms_SetUSBComp_t, 775
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 779
unpack_dms_SLQSDmsSwiGetResetInfo_t, 780
unpack_dms_SLQSDmsSwiIndicationRegister_t, 782
unpack_dms_SLQSGetERIFile_t, 791
unpack_dms_SLQSSetPowerSaveModeConfig_t, 793
unpack_dms_SLQSSwiClearDyingGaspStatistics-_t, 793
unpack_dms_SLQSSwiGetCrashInfo_t, 794
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 795
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 795
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 798
unpack_dms_SLQSSwiGetHostDevInfo_t, 799
unpack_dms_SLQSSwiGetOSInfo_t, 800
unpack_dms_SLQSSwiGetSerialNoExt_t, 801
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 801
unpack_dms_SLQSSwiSetHostDevInfo_t, 802
unpack_dms_SLQSSwiSetOSInfo_t, 802
unpack_dms_SLQSUIMGetState_t, 803
unpack_dms_SwiSetEventReport_t, 804
unpack_dms_SwiUimSelect_t, 805
unpack_dms_UIMGetControlKeyStatus_t, 806
unpack_dms_UIMGetICCID_t, 806
unpack_dms_UIMGetPINStatus_t, 808
unpack_dms_UIMSetControlKeyProtection_t, 809
unpack_dms_UIMSetPINProtection_t, 810
unpack_dms_UIMUnblockControlKey_t, 810
unpack_dsd_SLQSSystemStatusChange_Ind_t, 814
unpack_dsd_SLQSUIInfo_Ind_t, 815
unpack_fms_GetImagesPreference_t, 816
unpack_fms_GetStoredImages_t, 817
unpack_fms_SetImagesPreference_t, 817
unpack_loc_BestAvailPos_Ind_t, 847
unpack_loc_Delete_Assist_Data_t, 848
unpack_loc_DeleteAssistData_Ind_t, 849
unpack_loc_EngineState_Ind_t, 850
unpack_loc_EventNMEA_Ind_t, 851
unpack_loc_EventRegister_t, 851
unpack_loc_GetServer_Ind_t, 856
unpack_loc_GnssSvInfo_Ind_t, 856
unpack_loc_PositionRpt_Ind_t, 867
unpack_loc_SetExtPowerConfig_Ind_t, 869
unpack_loc_SetExtPowerState_t, 870
unpack_loc_SetOperationMode_Ind_t, 870
unpack_loc_SetOperationMode_t, 871
unpack_loc_SetServer_Ind_t, 872

unpack_loc_SLQSLOCGetBestAvailPos_t, 873
unpack_loc_SLQSLOCGetOpMode_t, 873
unpack_loc_Start_t, 874
unpack_loc_Stop_t, 874
unpack_nas_GetNetworkPreference_t, 882
unpack_nas_SetNetworkPreference_t, 893
unpack_nas_SetServingSystemCallback_ind_t, 894
unpack_nas_SlqsGetLTECphyCAInfo_t, 898
unpack_nas_SLQSNasGetRFInfo_t, 945
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 956
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 959
unpack_result_t, 1010
unpack_swidms_SLQSSwiDmsSetHWWatchdog_t, 1051
unpack_swidms_SLQSSwiDmsSetMTU_t, 1051
unpack_swidms_SLQSSwiDmsSetUimSlot_t, 1052
unpack_swidms_SLQSSwiDmsSetUsbComp_t, 1052
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 1064
unpack_tmd_SLQSTmdGetMitigationDevList_t, 1065
unpack_tmd_SLQSTmdGetMitigationLvl_t, 1066
unpack_tmd_SLQSTmdRegNotMitigationLvl_t, 1067
unpack_uim_ChangePin_t, 1068
unpack_uim_GetCardStatus_t, 1068
unpack_uim_GetCardStatusV2_t, 1069
unpack_uim_ReadTransparent_t, 1070
unpack_uim_SetPinProtection_t, 1071
unpack_uim_UnblockPin_t, 1086
unpack_uim_UnblockPinV2_t, 1087
unpack_uim_VerifyPin_t, 1087
unpack_wda_GetDataFormat_t, 1136
unpack_wda_GetQmapSettings_t, 1137
unpack_wda_SetDataFormat_t, 1141
unpack_wda_SetQmapSettings_t, 1141
unpack_wds_SLQSCreateProfile_t, 1163
unpack_wds_SLQSGetProfileSettings_t, 1173
unpack_wds_SLQSGetProfileSettingsV2_t, 1174
unpack_wds_SLQSSetIPFamilyPreference_t, 1178

tmd.h, 1686
pack_tmd_SLQSTmdDeRegNotMitigationLvl, 1687
pack_tmd_SLQSTmdGetMitigationDevList, 1687
pack_tmd_SLQSTmdGetMitigationLvl, 1687
pack_tmd_SLQSTmdRegNotMitigationLvl, 1688
TMD_MAX_DEV_LIST, 1686
unpack_tmd_SLQSTmdDeRegNotMitigationLvl, 1688
unpack_tmd_SLQSTmdGetMitigationDevList, 1688
unpack_tmd_SLQSTmdGetMitigationLvl, 1689
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind, 1689
unpack_tmd_SLQSTmdRegNotMitigationLvl, 1689

tmd_mitigationDevList, 685
maxMitigationLevel, 686
mitigationDevId, 686
mitigationDevIdLen, 686

tmrSIPRegn
ims_SIPRegnTmrInfo, 107

tmrT1
ims_TmrT1Info, 109

tmrT2
ims_TmrT2Info, 109

tmrTf
ims_TmrTfInfo, 110

toServiceId
sms_BroadcastConfig, 658

toggleMode
voice_lineCtrlInfo, 1238

token
uim_indToken, 705

TokenBucket
unpack_qos_swiQosFlow_t, 1008

tokenRate
unpack_qos_tokenBucket_t, 1008

tosMask
LibPackTFTIDParams, 158

total_rx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 997

total_rx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 997

total_tx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 997

total_tx_bytes_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 997

total_tx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 997

total_tx_pkt_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 997

TotalFragments
MBIM_FRAGMENT_HEADER, 186

TrackAreaCode
unpack_nas_SLQSGetServingSystem_t, 907
unpack_nas_SLQSGetServingSystemV2_t, 912

TrackingStatus
unpack_pds_SetPdsState_Ind_t, 981

TrafficClass
unpack_qos_swiQosFlow_t, 1008

trafficClass
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274

trafficPriority
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274

TranDstPort
unpack_qos_swiQosFilter_t, 1004

TranSrcPort
unpack_qos_swiQosFilter_t, 1004

TransCap
sms_transLayerInfo, 670
TransType
sms_transLayerInfo, 671
transactionID
sMSTransferRouteMTMessageInfo, 676
TransactionId
MBIM_MESSAGE_HEADER, 187
transactionId
CtlTransactionHdr, 60
MbimTransaction, 191
MbimTransport, 192
QmiService, 644
QmiTransaction, 646
QmiTransactionHdr, 646
transactionIdLock
MbimTransport, 192
QmiService, 644
transactionList
MbimTransport, 192
transactionListLock
MbimTransport, 192
transactionPool
QmiService, 644
transactionPoolLock
QmiService, 644
transferDelay
LibPackUMTSQoS, 160
wds_UMTSMinQoS, 1274
TransferRouteMTMessageInfo
transferRouteMessageTlv, 686
transferRouteMessageTlv, 686
TlvPresent, 686
TransferRouteMTMessageInfo, 686
transferStats
pack_wds_SLQSSetWdsEventCallback_t, 630
trueSrvStatus
nas_GSMSrvStatusInfo, 236
nas_NR5GSerStatTlv, 298
nas_NR5GSrvStatusTlv, 299
tx_bytes
unpack_QosFlowStat_t, 1010
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
tx_bytes_drp
unpack_QosFlowStat_t, 1010
tx_pkt
unpack_QosFlowStat_t, 1010
tx_pkt_drp
unpack_QosFlowStat_t, 1010
tx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
TxByteOKCnt
wds_TXBytesOKTlv, 1272
TxDropConutTlv
unpack_RMTransferStatistics_ind_t, 1012
txOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 1173
TxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 1012
TxOkConutTlv
unpack_RMTransferStatistics_ind_t, 1012
txPower
nas_txInfo, 369
TxQFilter
unpack_qos_QosFlowInfo_t, 993
TxQFlowGranted
unpack_qos_QosFlowInfo_t, 993
type
pack_wds_GetDefaultProfileNum_t, 612
pack_wds_SetDefaultProfileNum_t, 618
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 779
unpack_dms_SLQSDmsSwiGetResetInfo_t, 780
unpack_qmi_t, 988
typeSize
unpack_sms_GetSMSCAddress_t, 1014

u16PRLVersion
unpack_dms_GetPRLVersion_t, 764
u8PRLPreference
unpack_dms_GetPRLVersion_t, 764
UDPDstPort
unpack_qos_swiQosFilter_t, 1004
UDPSrcPort
unpack_qos_swiQosFilter_t, 1004
UInfoTLV
unpack_voice_USSDNoWaitIndicationCallback_ind_t, 1130
ULONG
SwiDataTypes.h, 1657
ULONGLONG
SwiDataTypes.h, 1657
UMTSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 1177
UMTSInstInfo
nas_UMTSInfo, 373
UMTSLTENbrCell
nas_WCDMAInfoLTENeighborCell, 381
UMTSReqQoS
LibPackUMTSReqQoSSigInd, 161
UNIQUE_ID_LEN
dms.h, 1308
UNUSEDPARAM
common.h, 1291
SwiDataTypes.h, 1656
uPhyCardStatus
slot_t, 656
uPhySlotStatus
slot_t, 656
USBComp
pack_dms_SetUSBComp_t, 423
unpack_dms_GetUSBComp_t, 767
USHORT
SwiDataTypes.h, 1657
USSDNotificationNetworkInfo

unpack_voice_USSDNotificationCallback_ind_t, 1129
USSInformation
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 598
UUSData
voice_UUSInfo, 1250
UUSDatalen
voice_UUSInfo, 1250
UUSDcs
voice_UUSInfo, 1250
UUSType
voice_UUSInfo, 1250
uarfcn
nas_lteWcdmaCellInfo, 280
nas_UMTSExtInfo, 371
nas_UMTSInfo, 373
nas_wcdmaUARFCN, 386
Udp_coalescing
unpack_wda_GetDataFormat_t, 1136
ueInIdle
nas_LTEInfoInterfreq, 258
nas_LTEInfoIntrafreq, 260
nas_LTEInfoNeighboringGSM, 261
nas_LTEInfoNeighboringWCDMA, 262
ui_mask
DsdGlobalUIInformation_t, 83
DsdUIApnAvailableSystemInfo_t, 89
uim.h, 1690
MAX_ATR_LENGTH, 1695
MAX_ICCID_LENGTH, 1695
MAX_NO_OF_SLOTS, 1695
MAX_PHY_SLOTS_INFO, 1695
MAX_SLOTS_STATUS, 1695
pack_uim_ChangePin, 1696
pack_uim_GetCardStatus, 1696
pack_uim_ReadTransparent, 1697
pack_uim_SLQSUIMAddProfile, 1698
pack_uim_SLQSUIMAuthenticate, 1698
pack_uim_SLQSUIMDeleteProfile, 1698
pack_uim_SLQSUIMDepersonalization, 1699
pack_uim_SLQSUIMEventRegister, 1699
pack_uim_SLQSUIMGetConfiguration, 1700
pack_uim_SLQSUIMGetEID, 1700
pack_uim_SLQSUIMGetFileAttributes, 1700
pack_uim_SLQSUIMGetProfileInfo, 1701
pack_uim_SLQSUIMGetSIMProfile, 1702
pack_uim_SLQSUIMGetServiceStatus, 1701
pack_uim_SLQSUIMGetSlotsStatus, 1702
pack_uim_SLQSUIMPowerDown, 1702
pack_uim_SLQSUIMPowerUp, 1703
pack_uim_SLQSUIMReadRecord, 1703
pack_uim_SLQSUIMRefreshComplete, 1703
pack_uim_SLQSUIMRefreshGetLastEvent, 1704
pack_uim_SLQSUIMRefreshOK, 1704
pack_uim_SLQSUIMRefreshRegister, 1705
pack_uim_SLQSUIMReset, 1705
pack_uim_SLQSUIMSetSIMProfile, 1706
pack_uim_SLQSUIMSetServiceStatus, 1705
pack_uim_SLQSUIMSwitchSlot, 1706
pack_uim_SLQSUIMUpdateProfile, 1706
pack_uim_SLQSUIMWriteRecord, 1707
pack_uim_SLQSUIMWriteTransparent, 1707
pack_uim_SLQSUIMeUICCMemReset, 1699
pack_uim_SetPinProtection, 1697
pack_uim_UnblockPin, 1708
pack_uim_VerifyPin, 1708
unpack_uim_ChangePin, 1708
unpack_uim_GetCardStatus, 1709
unpack_uim_GetCardStatusV2, 1709
unpack_uim_ReadTransparent, 1709
unpack_uim_SLQSUIMAddProfile, 1711
unpack_uim_SLQSUIMAddProfile_t, 1696
unpack_uim_SLQSUIMAuthenticate, 1711
unpack_uim_SLQSUIMDeleteProfile, 1711
unpack_uim_SLQSUIMDeleteProfile_t, 1696
unpack_uim_SLQSUIMDepersonalization, 1712
unpack_uim_SLQSUIMEventRegister, 1712
unpack_uim_SLQSUIMGetConfiguration, 1713
unpack_uim_SLQSUIMGetEID, 1713
unpack_uim_SLQSUIMGetFileAttributes, 1713
unpack_uim_SLQSUIMGetProfileInfo, 1714
unpack_uim_SLQSUIMGetSIMProfile, 1714
unpack_uim_SLQSUIMGetServiceStatus, 1714
unpack_uim_SLQSUIMGetSlotsStatus, 1715
unpack_uim_SLQSUIMGetSlotsStatusV2, 1715
unpack_uim_SLQSUIMPowerDown, 1716
unpack_uim_SLQSUIMPowerDown_t, 1696
unpack_uim_SLQSUIMPowerUp, 1716
unpack_uim_SLQSUIMPowerUp_t, 1696
unpack_uim_SLQSUIMReadRecord, 1716
unpack_uim_SLQSUIMRefreshCallback_Ind, 1717
unpack_uim_SLQSUIMRefreshComplete, 1717
unpack_uim_SLQSUIMRefreshComplete_t, 1696
unpack_uim_SLQSUIMRefreshGetLastEvent, 1717
unpack_uim_SLQSUIMRefreshOK, 1718
unpack_uim_SLQSUIMRefreshOK_t, 1696
unpack_uim_SLQSUIMRefreshRegister, 1718
unpack_uim_SLQSUIMRefreshRegister_t, 1696
unpack_uim_SLQSUIMReset, 1718
unpack_uim_SLQSUIMReset_t, 1696
unpack_uim_SLQSUIMSetSIMProfile, 1719
unpack_uim_SLQSUIMSetSIMProfile_t, 1696
unpack_uim_SLQSUIMSetServiceStatus, 1719
unpack_uim_SLQSUIMSetServiceStatus_t, 1696
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind, 1719
unpack_uim_SLQSUIMSwitchSlot, 1720
unpack_uim_SLQSUIMSwitchSlot_t, 1696
unpack_uim_SLQSUIMUpdateProfile, 1720
unpack_uim_SLQSUIMUpdateProfile_t, 1696
unpack_uim_SLQSUIMWriteRecord, 1720
unpack_uim_SLQSUIMWriteTransparent, 1721
unpack_uim_SLQSUIMeUICCMemReset, 1712
unpack_uim_SLQSUIMeUICCMemReset_t, 1696
unpack_uim_SetPinProtection, 1710

unpack_uim_SetUimSlotStatusChangeCallback_ind, 1710
unpack_uim_UnblockPin, 1721
unpack_uim_UnblockPinV2, 1721
unpack_uim_VerifyPin, 1722
uim_GetSlotsInfoTlv, 703
NumberOfPhySlotInfo, 703
TlvPresent, 703
uimSlotInfo, 703
uim_GetSlotsStatusTlv, 704
NumberOfPhySlot, 704
TlvPresent, 704
uimSlotStatus, 704
uim_UIMGetFDNStatus, 722
FDNStatus, 723
TlvPresent, 723
uim_UIMGetHiddenKeyStatus, 723
hiddenKey, 723
TlvPresent, 723
uim_UIMGetIndex, 723
index, 724
TlvPresent, 724
uim_UIMNickName, 724
data, 724
dataLen, 724
uim_UIMProfileCode, 724
data, 725
dataLen, 725
uim_UIMSessionInformation, 725
aid, 725
aidLength, 726
sessionType, 726
uim_activeProfileInfo, 687
profileId, 687
profileType, 687
TlvPresent, 687
uim_additionalReadResult, 687
additionalRecord, 688
additionalRecordLen, 688
TlvPresent, 688
uim_appStatus, 688
aidLength, 690
aidVal, 690
appState, 690
appType, 691
persoFeature, 691
persoRetries, 691
persoState, 691
persoUnblockRetries, 691
pin1Retries, 691
pin1State, 691
pin2Retries, 691
pin2State, 691
puk1Retries, 691
puk2Retries, 691
univPin, 691
uim_authenticateResult, 691
content, 691
contentLen, 691
uim_authenticationData, 692
context, 692
data, 692
dataLen, 692
uim_cardResult, 693
sw1, 693
sw2, 693
uim_cardResultInfo, 693
sw1, 693
sw2, 694
TlvPresent, 694
uim_cardStatus, 694
index1xPri, 695
index1xSec, 695
indexGwPri, 695
indexGwSec, 695
numSlot, 695
SlotInfo, 695
uim_changeUIMPIN, 695
oldPINLen, 695
oldPINVal, 696
pinID, 696
pinLen, 696
pinVal, 696
uim_depersonalizationInformation, 696
ckLen, 697
ckVal, 697
feature, 697
operation, 697
uim_eidInfo, 697
eidVal, 697
eidValLen, 697
TlvPresent, 697
uim_encryptedPIN1, 697
pin1Len, 698
pin1Val, 698
uim_fileAttributes, 698
fileID, 701
fileSize, 701
fileType, 701
rawLen, 701
rawValue, 701
recordCount, 701
recordSize, 701
secActivate, 701
secActivateMask, 701
secDeactivate, 701
secDeactivateMask, 701
secIncrease, 701
secIncreaseMask, 701
secRead, 701
secReadMask, 701
secWrite, 701
secWriteMask, 701
uim_fileInfo, 701
fileID, 702
path, 702

pathLen, 702
uim_getSimProfileInfo, 702
profileId, 703
profileInfoLen, 703
profileType, 703
TlvPresent, 703
uim_hotSwapStatus, 704
hotSwap, 705
hotSwapLength, 705
uim_indToken, 705
TlvPresent, 705
token, 705
uim_personalizationStatus, 705
feature, 706
numFeatures, 706
unblockLeft, 706
verifyLeft, 706
uim_physlotInfo, 706
atrValue, 707
atrValueLen, 707
cardProtocol, 707
iseUICC, 707
numApp, 707
uim_physlotsInfo, 707
uimSlotInfo, 708
uim_profileClassTlv, 708
profileClass, 708
TlvPresent, 708
uim_profileICCIDTlv, 708
profileICCID, 709
profileICCIDLen, 709
TlvPresent, 709
uim_profileIconTlv, 709
profileIconData, 709
profileIconDataLen, 709
TlvPresent, 709
uim_profileIconTypeTlv, 710
profileIconType, 710
TlvPresent, 710
uim_profileNameTlv, 710
profileName, 711
profileNameLen, 711
TlvPresent, 711
uim_profileNicknameTlv, 711
profileNickname, 711
profileNicknameLen, 711
TlvPresent, 711
uim_profilePolicyRulesTlv, 711
profilePolicyRules, 712
TlvPresent, 712
uim_profileSPNTlv, 712
profileSPN, 712
profileSPNLen, 712
TlvPresent, 712
uim_profileStateTlv, 712
profileState, 713
TlvPresent, 713
uim_readRecordInfo, 713
length, 713
record, 713
uim_readResult, 714
content, 714
contentLen, 714
uim_readResultInfo, 714
content, 715
contentLen, 715
TlvPresent, 715
uim_readTransparentInfo, 715
length, 715
offset, 715
uim_refreshevent, 715
aid, 716
aidLength, 717
arrfileInfo, 717
mode, 717
numOfFiles, 717
sessionType, 717
stage, 717
uim_registerRefresh, 717
arrfileInfo, 717
numFiles, 718
registerFlag, 718
voteForInit, 718
uim_remainingRetries, 718
unblockLeft, 718
verifyLeft, 718
uim_select
pack_dms_SwiUimSelect_t, 428
uim_sessionInformation, 718
aid, 719
aidLength, 719
sessionType, 719
uim_setPINProtection, 719
pinID, 720
pinLength, 720
pinOperation, 720
pinValue, 720
uim_simBusyStatus, 720
simBusy, 720
simBusyLength, 720
uim_slotInfo, 721
AppStatus, 722
cardState, 722
errorState, 722
numApp, 722
upinRetries, 722
upinState, 722
upukRetries, 722
uim_unblockUIMPIN, 726
newPINLen, 726
newPINVal, 726
pinID, 726
pukLen, 726
pukVal, 726
uim_validCardStatus, 727
validCard, 727

validCardLength, 727
uim_verifyUIMPIN, 727
pinID, 728
pinLen, 728
pinVal, 728
uim_writeRecordInfo, 728
data, 728
dataLen, 728
record, 728
uim_writeTransparentInfo, 728
data, 729
dataLen, 729
offset, 729
uimAutoSwitchActSlot
dms_UimAutoSwitchActSlotTlv, 76
uimSelect
unpack_dms_SLQSDmsSwiGetUimSelection_t, 781
uimSlot
pack_uim_SLQSUIMeUICCMemReset_t, 571
swidms_AutoUimSwitchModeTlv, 677
swidms_MultiplexedSimModeTlv, 681
swidms_SetMultiplexedModeTlv, 681
swidms_UimSlotSelectedTlv, 683
uimSlotInfo
uim_GetSlotsInfoTlv, 703
uim_physlotsInfo, 708
uimSlotSelected
unpack_swidms_SLQSSwiDmsGetUimSlot_t, 1049
uimSlotStatus
slots_t, 658
uim_GetSlotsStatusTlv, 704
UimStatusTlv
unpack_dms_SwiEventReportCallBack_ind_t, 804
Ul_data_aggregation_max_datagrams
unpack_wda_GetDataFormat_t, 1136
Ul_data_aggregation_max_size
unpack_wda_GetDataFormat_t, 1136
Ul_data_aggregation_protocol
unpack_wda_GetDataFormat_t, 1136
ulData
wds_DataULongTlv, 1257
ulMask
rmTrasnferStaticsReq, 653
ulMax
PackSwiAvmsSetSettingsPeriodInfo, 639
ulMin
PackSwiAvmsSetSettingsPeriodInfo, 639
ulPhysicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 581
ulPkgDownloadComplete
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1195
ulPkgDownloadSize
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1195
ullData
wds_DataULongLongTlv, 1256
umtsEcio
nas_umtsInstArr, 374
nas_UMTSinstInfo, 375
umtsInst
nas_UMTSExtInfo, 371
nas_UMTSInfo, 373
umtsInstElement
nas_UMTSExtInfo, 371
umtsLTENbrCellLen
nas_WCDMAInfoLTENeighborCell, 381
umtsPsc
nas_umtsInstArr, 374
nas_UMTSinstInfo, 375
umtsRank
nas_umtsInstArr, 374
umtsRscp
nas_umtsInstArr, 374
nas_UMTSinstInfo, 375
umtsSet
nas_umtsInstArr, 374
umtsSqual
nas_umtsInstArr, 374
umtsSrxLvl
nas_umtsInstArr, 374
umtsUarfcn
nas_umtsInstArr, 374
nas_UMTSinstInfo, 375
UnPackGetProfileSettingOut, 1192
curProfile, 1192
pExtErrCode, 1192
UnPackGetProfileSettingOutV2, 1192
curProfile, 1193
pExtErrCode, 1193
unblockLeft
uim_personalizationStatus, 706
uim_remainingRetries, 718
unblockRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 806
unpack_dms_UIMSetPINProtection_t, 810
unpack_dms_UIMUnblockControlKey_t, 810
uniqueID
image_info_t, 97
univPin
appStats, 46
uim_appStatus, 691
universalTime
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 950
unpack_QosFlowStat_t, 1009
bearerId, 1010
tx_bytes, 1010
tx_bytes_drp, 1010
tx_pkt, 1010
tx_pkt_drp, 1010
unpack_RMTransferStatistics_ind_t, 1011
ParamPresenceMask, 1012
RxDropConutTlv, 1012

RxOkByteCountTlv, 1012
RxOkConutTlv, 1012
TxDropConutTlv, 1012
TxOkByteCountTlv, 1012
TxOkConutTlv, 1012
unpack_audio_SLQSGetAudioPathConfig
audio.h, 1281
unpack_audio_SLQSGetAudioPathConfig_t, 729
pCodecSTGain, 731
pECMode, 731
pMICGainSelect, 731
pNSEnable, 731
pRXAGCList, 731
pRXAVCList, 731
pTXAGCList, 731
pTXAVCSwitch, 731
pTXGain, 731
ParamPresenceMask, 731
unpack_audio_SLQSGetAudioProfile
audio.h, 1281
unpack_audio_SLQSGetAudioProfile_t, 731
EarMute, 732
MicMute, 732
ParamPresenceMask, 732
Profile, 732
Volume, 732
unpack_audio_SLQSGetAudioVolTLBConfig
audio.h, 1281
unpack_audio_SLQSGetAudioVolTLBConfig_t, 732
ParamPresenceMask, 733
ResCode, 733
unpack_audio_SLQSSetAudioPathConfig
audio.h, 1282
unpack_audio_SLQSSetAudioProfile
audio.h, 1282
unpack_audio_SLQSSetAudioVolTLBConfig
audio.h, 1282
unpack_audio_SLQSSetAudioVolTLBConfig_t, 733
ParamPresenceMask, 733
ResCode, 733
unpack_cat_CATSendEnvelopeCommand
cat.h, 1287
unpack_cat_CATSendEnvelopeCommand_t
cat.h, 1285
unpack_cat_CATSendTerminalResponse
cat.h, 1287
unpack_cat_CATSendTerminalResponse_t
cat.h, 1285
unpack_cat_SetCATEventCallback
cat.h, 1287
unpack_cat_SetCATEventCallback_t, 734
errorMask, 735
ParamPresenceMask, 735
Tlvresult, 735
unpack_cat_SetCatEventCallback_ind
cat.h, 1288
unpack_cat_SetCatEventCallback_ind_t, 733
CCETlv, 734
event_Index, 734
ParamPresenceMask, 734
unpack_ctl_GetClientId
ctl.h, 1295
unpack_ctl_GetClientId_t, 735
clientId, 736
ParamPresenceMask, 736
serviceType, 736
Tlvresult, 736
unpack_ctl_ReleaseClientId
ctl.h, 1295
unpack_ctl_ReleaseClientId_t, 736
clientId, 736
ParamPresenceMask, 736
serviceType, 736
Tlvresult, 737
unpack_dms_ActivateAutomatic
dms.h, 1333
unpack_dms_GetActivationState
dms.h, 1334
unpack_dms_GetActivationState_t, 737
ParamPresenceMask, 737
state, 737
unpack_dms_GetBandCapability
dms.h, 1334
unpack_dms_GetBandCapability_t, 737
BandCapability, 739
ParamPresenceMask, 739
Tlvresult, 739
unpack_dms_GetCrashAction
dms.h, 1335
unpack_dms_GetCrashAction_t, 739
DevCrashState, 739
ParamPresenceMask, 739
Tlvresult, 740
unpack_dms_GetCustFeature
dms.h, 1335
unpack_dms_GetCustFeature_t, 740
DHCPRelayEnabled, 742
DisableIMSI, 742
GPSLPM, 742
GPSSel, 742
GpsEnable, 742
IPFamSupport, 742
IsVoiceEnabled, 742
ParamPresenceMask, 742
RMAutoConnect, 742
SMSSupport, 742
Tlvresult, 742
unpack_dms_GetCustFeaturesV2
dms.h, 1335
unpack_dms_GetCustFeaturesV2_t, 742
GetCustomFeatureV2, 743
ParamPresenceMask, 743
Tlvresult, 743
unpack_dms_GetDeviceCap
dms.h, 1336
unpack_dms_GetDeviceCap_t, 743

DataServiceCapability, 744
MaxRXChannelRate, 744
MaxTXChannelRate, 744
ParamPresenceMask, 744
RadioIfaces, 744
RadioIfacesSize, 744
SimCapability, 744
Tlvresult, 744
unpack_dms_GetDeviceCapabilities
dms.h, 1336
unpack_dms_GetDeviceCapabilities_t, 744
dataServiceCaCapability, 745
maxRxChannelRate, 746
maxTxChannelRate, 746
ParamPresenceMask, 746
RadioIfaces, 746
radioIfacesSize, 746
simCapability, 746
unpack_dms_GetDeviceCapabilitiesV2
dms.h, 1336
unpack_dms_GetDeviceCapabilitiesV2_t, 746
DevCaps, 748
pDevCurSubsCaps, 748
pDevExplicitCfgIndex, 748
pDevMaxActDataSubsCaps, 748
pDevMaxCfgListCaps, 748
pDevMaxSubsCaps, 748
pDevMultiSimCaps, 748
pDevMultiSimVoiceDataCaps, 748
pDevSrvCaps, 748
pDevSubsFeatureModeCaps, 748
pDevSubsVoiceDataCaps, 748
pDevVoiceCaps, 748
pDevVoiceDataCaps, 748
ParamPresenceMask, 748
Tlvresult, 748
unpack_dms_GetDeviceHardwareRev
dms.h, 1337
unpack_dms_GetDeviceHardwareRev_t, 748
ParamPresenceMask, 749
String, 749
stringSize, 749
Tlvresult, 749
unpack_dms_GetDeviceMfr
dms.h, 1337
unpack_dms_GetDeviceMfr_t, 749
ParamPresenceMask, 750
String, 750
stringSize, 750
Tlvresult, 750
unpack_dms_GetDeviceSerialNumbers
dms.h, 1337
unpack_dms_GetDeviceSerialNumbers_t, 750
ESNString, 752
esnSize, 751
IMEIString, 752
imeiSize, 752
imeiSvnSize, 752
ImeiSvnString, 752
MEIDString, 752
meidSize, 752
ParamPresenceMask, 752
Tlvresult, 752
unpack_dms_GetFSN
dms.h, 1339
unpack_dms_GetFSN_t, 756
ParamPresenceMask, 756
String, 756
Tlvresult, 756
unpack_dms_GetFirmwareInfo
dms.h, 1338
unpack_dms_GetFirmwareInfo_t, 752
appversion_str, 753
bootversion_str, 753
carrier_str, 753
cur_carr_name, 753
cur_carr_rev, 753
modelid_str, 753
packageid_str, 753
ParamPresenceMask, 753
priversion_str, 753
sku_str, 753
Tlvresult, 753
unpack_dms_GetFirmwareRevision
dms.h, 1338
unpack_dms_GetFirmwareRevision_t, 754
AMSSString, 754
amssSize, 754
PRIString, 754
ParamPresenceMask, 754
Tlvresult, 754
unpack_dms_GetFirmwareRevisions
dms.h, 1338
unpack_dms_GetFirmwareRevisions_t, 755
AMSSString, 756
amssSize, 756
bootSize, 756
BootString, 756
PRIString, 756
ParamPresenceMask, 756
priSize, 756
Tlvresult, 756
unpack_dms_GetHardwareRevision
dms.h, 1339
unpack_dms_GetHardwareRevision_t, 757
hwVer, 757
ParamPresenceMask, 757
unpack_dms_GetIMSI
dms.h, 1339
unpack_dms_GetIMSI_t, 757
imsi, 757
ParamPresenceMask, 758
Tlvresult, 758
unpack_dms_GetManufacturer
dms.h, 1340
unpack_dms_GetManufacturer_t, 758

manufacturer, 758
ParamPresenceMask, 758
Tlvresult, 758
unpack_dms_GetModelID
dms.h, 1340
unpack_dms_GetModelID_t, 758
modelid, 759
ParamPresenceMask, 759
Tlvresult, 759
unpack_dms_GetNetworkTime
dms.h, 1341
unpack_dms_GetNetworkTime_t, 759
ParamPresenceMask, 760
source, 760
timestamp, 760
Tlvresult, 760
unpack_dms_GetNetworkTimeV2
dms.h, 1341
unpack_dms_GetNetworkTimeV2_t, 760
pSysTime, 761
pUsrTime, 761
ParamPresenceMask, 761
source, 761
timestamp, 761
Tlvresult, 761
unpack_dms_GetOfflineReason
dms.h, 1341
unpack_dms_GetOfflineReason_t, 761
pReasonMask, 762
ParamPresenceMask, 762
pbPlatform, 762
Tlvresult, 762
unpack_dms_GetPRLVersion
dms.h, 1342
unpack_dms_GetPRLVersion_t, 763
ParamPresenceMask, 764
Tlvresult, 764
u16PRLVersion, 764
u8PRLPreference, 764
unpack_dms_GetPower
dms.h, 1342
unpack_dms_GetPower_t, 762
HardwareControlledMode, 763
OfflineReason, 763
OperationMode, 763
ParamPresenceMask, 763
Tlvresult, 763
unpack_dms_GetSerialNumbers
dms.h, 1342
unpack_dms_GetSerialNumbers_t, 764
esn, 765
imei_no, 765
imeisv_svn, 765
meid, 765
ParamPresenceMask, 765
unpack_dms_GetUSBComp
dms.h, 1343
unpack_dms_GetUSBComp_t, 765
NumSupUSBComps, 767
ParamPresenceMask, 767
SupUSBComps, 767
Tlvresult, 767
USBComp, 767
unpack_dms_GetVoiceNumber
dms.h, 1343
unpack_dms_GetVoiceNumber_t, 767
MIN, 768
minSize, 768
ParamPresenceMask, 768
Tlvresult, 768
VoiceNumber, 768
voiceNumberSize, 768
unpack_dms_PSMCfgChange_ind
dms.h, 1343
unpack_dms_PSMCfgChange_ind_t, 769
ActiveTimer, 769
EnableState, 769
ParamPresenceMask, 769
PeriodicUpdateTimer, 769
Tlvresult, 769
unpack_dms_ResetToFactoryDefaults
dms.h, 1344
unpack_dms_ResetToFactoryDefaults_t, 769
ParamPresenceMask, 770
Tlvresult, 770
unpack_dms_SLQSDmsSwiGetPCInfo
dms.h, 1348
unpack_dms_SLQSDmsSwiGetPCInfo_t, 775
has_LpmFlag, 777
has_PersistentLpm, 777
has_Wdisable, 777
LpmFlag, 778
opMode, 778
ParamPresenceMask, 778
PersistentLpm, 778
PowerOffMode, 778
Wdisable, 778
unpack_dms_SLQSDmsSwiGetResetInfo
dms.h, 1348
unpack_dms_SLQSDmsSwiGetResetInfo_Ind
dms.h, 1348
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 778
ParamPresenceMask, 779
source, 779
Tlvresult, 779
type, 779
unpack_dms_SLQSDmsSwiGetResetInfo_t, 779
ParamPresenceMask, 780
source, 780
Tlvresult, 780
type, 780
unpack_dms_SLQSDmsSwiGetUimSelection
dms.h, 1349
unpack_dms_SLQSDmsSwiGetUimSelection_t, 780
pUimAutoSwitchActSlot, 781
ParamPresenceMask, 781

uimSelect, 781
unpack_dms_SLQSDmsSwiIndicationRegister
dms.h, 1349
unpack_dms_SLQSDmsSwiIndicationRegister_t, 781
ParamPresenceMask, 782
Tlvresult, 782
unpack_dms_SLQSGetBandCapability
dms.h, 1349
unpack_dms_SLQSGetBandCapability_t, 782
bandCapability, 786
LteBandCapability, 786
ParamPresenceMask, 786
TdsBandCapability, 786
unpack_dms_SLQSGetBandCapabilityExt
dms.h, 1350
unpack_dms_SLQSGetBandCapabilityExt_t, 786
bandCapability, 790
LteBandCapability, 790
LteBandsSupport, 790
ParamPresenceMask, 790
TdsBandCapability, 790
unpack_dms_SLQSGetERIFile
dms.h, 1350
unpack_dms_SLQSGetERIFile_t, 790
eriFile, 791
ParamPresenceMask, 791
Tlvresult, 791
unpack_dms_SLQSGetPowerSaveModeConfig
dms.h, 1351
unpack_dms_SLQSGetPowerSaveModeConfig_t, 791
pActiveTimer, 792
pDurationThreshold, 792
pEarlyWakeupTime, 792
pPeriodicUpdateTimer, 792
pPsmEnableState, 792
pRandomizationWindow, 792
ParamPresenceMask, 792
unpack_dms_SLQSSetPowerSaveModeConfig
dms.h, 1351
unpack_dms_SLQSSetPowerSaveModeConfig_t, 792
ParamPresenceMask, 793
Tlvresult, 793
unpack_dms_SLQSSwiClearDyingGaspStatistics
dms.h, 1351
unpack_dms_SLQSSwiClearDyingGaspStatistics_t, 793
ParamPresenceMask, 793
Tlvresult, 793
unpack_dms_SLQSSwiGetCrashInfo
dms.h, 1352
unpack_dms_SLQSSwiGetCrashInfo_t, 793
crashInfoParam, 794
ParamPresenceMask, 794
Tlvresult, 794
unpack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 1352
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 794
pGetDyingGaspCfg, 795
ParamPresenceMask, 795
Tlvresult, 795
unpack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 1352
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 795
pGetDyingGaspStatistics, 795
ParamPresenceMask, 795
Tlvresult, 795
unpack_dms_SLQSSwiGetFirmwareCurr
dms.h, 1353
unpack_dms_SLQSSwiGetFirmwareCurr_t, 795
carrier, 796
fwvers, 796
numEntries, 796
pCurrImgInfo, 796
ParamPresenceMask, 796
pkgver, 796
priver, 796
unpack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 1353
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 796
imgType, 798
logString, 798
ParamPresenceMask, 798
refData, 798
refString, 798
ResCode, 798
Tlvresult, 798
unpack_dms_SLQSSwiGetHostDevInfo
dms.h, 1353
unpack_dms_SLQSSwiGetHostDevInfo_t, 798
hostID, 799
manString, 799
modelString, 799
ParamPresenceMask, 799
plasmaIDString, 799
swVerString, 799
Tlvresult, 799
unpack_dms_SLQSSwiGetOSInfo
dms.h, 1354
unpack_dms_SLQSSwiGetOSInfo_t, 799
nameString, 800
ParamPresenceMask, 800
Tlvresult, 800
versionString, 800
unpack_dms_SLQSSwiGetSerialNoExt
dms.h, 1354
unpack_dms_SLQSSwiGetSerialNoExt_t, 800
meidString, 801
ParamPresenceMask, 801
Tlvresult, 801
unpack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 1354
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 801
ParamPresenceMask, 801
Tlvresult, 801
unpack_dms_SLQSSwiSetHostDevInfo
dms.h, 1355

unpack_dms_SLQSSwiSetHostDevInfo_t, 801
ParamPresenceMask, 802
Tlvresult, 802
unpack_dms_SLQSSwiSetOSInfo
dms.h, 1355
unpack_dms_SLQSSwiSetOSInfo_t, 802
ParamPresenceMask, 802
Tlvresult, 802
unpack_dms_SLQSUIMGetState
dms.h, 1355
unpack_dms_SLQSUIMGetState_t, 802
ParamPresenceMask, 803
state, 803
Tlvresult, 803
unpack_dms_SetActivationStatusCallback
dms.h, 1344
unpack_dms_SetActivationStatusCallback_t, 770
ParamPresenceMask, 770
Tlvresult, 770
unpack_dms_SetCrashAction
dms.h, 1344
unpack_dms_SetCrashAction_t, 770
notused, 771
ParamPresenceMask, 771
unpack_dms_SetCustFeature
dms.h, 1345
unpack_dms_SetCustFeature_t, 771
ParamPresenceMask, 771
Tlvresult, 771
unpack_dms_SetCustFeaturesV2
dms.h, 1345
unpack_dms_SetCustFeaturesV2_t, 771
ParamPresenceMask, 772
Tlvresult, 772
unpack_dms_SetEventReport
dms.h, 1345
unpack_dms_SetEventReport_ind
dms.h, 1346
unpack_dms_SetEventReport_ind_t, 772
ActivationStatusTlv, 772
OperatingModeTlv, 772
ParamPresenceMask, 773
Tlvresult, 773
unpack_dms_SetEventReport_t, 773
ParamPresenceMask, 773
Tlvresult, 773
unpack_dms_SetFirmwarePreference
dms.h, 1346
unpack_dms_SetFirmwarePreference_t, 773
ParamPresenceMask, 774
Tlvresult, 774
unpack_dms_SetIndicationRegister
dms.h, 1346
unpack_dms_SetIndicationRegister_t, 774
ParamPresenceMask, 774
Tlvresult, 774
unpack_dms_SetPower
dms.h, 1347
unpack_dms_SetPower_t, 774
ParamPresenceMask, 775
Tlvresult, 775
unpack_dms_SetUSBComp
dms.h, 1347
unpack_dms_SetUSBComp_t, 775
ParamPresenceMask, 775
Tlvresult, 775
unpack_dms_SwiEventReportCallBack_ind
dms.h, 1356
unpack_dms_SwiEventReportCallBack_ind_t, 803
ParamPresenceMask, 804
TempTlv, 804
UimStatusTlv, 804
VoltTlv, 804
unpack_dms_SwiSetEventReport
dms.h, 1356
unpack_dms_SwiSetEventReport_t, 804
ParamPresenceMask, 804
Tlvresult, 804
unpack_dms_SwiUimSelect
dms.h, 1356
unpack_dms_SwiUimSelect_t, 804
ParamPresenceMask, 805
Tlvresult, 805
unpack_dms_UIMChangePIN
dms.h, 1357
unpack_dms_UIMGetControlKeyStatus
dms.h, 1357
unpack_dms_UIMGetControlKeyStatus_t, 805
facilityState, 806
ParamPresenceMask, 806
Tlvresult, 806
unblockRetriesLeft, 806
verifyRetriesLeft, 806
unpack_dms_UIMGetICCID
dms.h, 1357
unpack_dms_UIMGetICCID_t, 806
ParamPresenceMask, 806
String, 806
stringSize, 806
Tlvresult, 806
unpack_dms_UIMGetPINStatus
dms.h, 1358
unpack_dms_UIMGetPINStatus_t, 807
p1Status, 808
p1UnblockRetriesLeft, 808
p1VerifyRetriesLeft, 808
p2Status, 808
p2UnblockRetriesLeft, 808
p2VerifyRetriesLeft, 808
ParamPresenceMask, 808
Tlvresult, 808
unpack_dms_UIMSetControlKeyProtection
dms.h, 1358
unpack_dms_UIMSetControlKeyProtection_t, 808
ParamPresenceMask, 809
Tlvresult, 809

verifyRetriesLeft, 809
unpack_dms_UIMSetPINProtection
dms.h, 1358
unpack_dms_UIMSetPINProtection_t, 809
ParamPresenceMask, 810
Tlvresult, 810
unblockRetriesLeft, 810
verifyRetriesLeft, 810
unpack_dms_UIMUnblockControlKey
dms.h, 1359
unpack_dms_UIMUnblockControlKey_t, 810
ParamPresenceMask, 810
Tlvresult, 810
unblockRetriesLeft, 810
unpack_dms_UIMUnblockPIN
dms.h, 1359
unpack_dms_UIMVerifyPIN
dms.h, 1360
unpack_dms_ValidateSPC
dms.h, 1360
unpack_dsd_RegisterUIInfoChange
dsd.h, 1364
unpack_dsd_RegisterUIInfoChange_t
dsd.h, 1363
unpack_dsd_SLQSGetDsdUIInfo_t, 810
Apn_info_len, 811
pDsdApnAvailableSystemInfo, 811
pDsdGlobalUIInformation, 811
pDsdNullBearerReason, 811
ParamPresenceMask, 811
unpack_dsd_SLQSGetSystemStatus
dsd.h, 1365
unpack_dsd_SLQSGetSystemStatus_t, 811
Apn_pref_sys_len, 812
Avail_sys_len, 812
pDsdApnAvailableSystemInfo, 812
pDsdAvailableSystem, 812
pDsdCurRecGlobalPrefSystem, 812
pDsdNullBearerReason, 812
ParamPresenceMask, 812
unpack_dsd_SLQSGetUIInfo
dsd.h, 1365
unpack_dsd_SLQSSystemStatusChange_Ind_t, 813
Avail_sys_len, 814
pDsdAvailableSystem, 814
pDsdNullBearerReason, 814
ParamPresenceMask, 814
Tlvresult, 814
unpack_dsd_SLQSSystemStatusChangeInd
dsd.h, 1365
unpack_dsd_SLQSUIInfo_Ind_t, 814
Apn_info_len, 815
NullBearerReasonTlvPresent, 815
pDsdNullBearerReason, 815
ParamPresenceMask, 815
Tlvresult, 815
unpack_dsd_SLQSUIInfoChangeInd
dsd.h, 1366
unpack_dsd_SystemStatusChange
dsd.h, 1366
unpack_dsd_SystemStatusChnage_t
dsd.h, 1363
unpack_fms_GetImagesPreference
fms.h, 1369
unpack_fms_GetImagesPreference_t, 815
ImageListSize, 816
pImageList, 816
ParamPresenceMask, 816
Tlvresult, 816
unpack_fms_GetStoredImages
fms.h, 1370
unpack_fms_GetStoredImages_t, 816
imageList, 817
imagelistSize, 817
ParamPresenceMask, 817
Tlvresult, 817
unpack_fms_SetImagesPreference
fms.h, 1370
unpack_fms_SetImagesPreference_t, 817
ImageTypes, 817
ImageTypesSize, 817
ParamPresenceMask, 817
Tlvresult, 817
unpack_ims_SLQSGetIMSSMSConfig
ims.h, 1377
unpack_ims_SLQSGetIMSSMSConfig_t, 817
pSettingResp, 818
ParamPresenceMask, 818
unpack_ims_SLQSGetIMSUserConfig
ims.h, 1377
unpack_ims_SLQSGetIMSUserConfig_t, 819
pIMSDomain, 819
pSettingResp, 819
ParamPresenceMask, 819
unpack_ims_SLQSGetIMSVoIPConfig
ims.h, 1378
unpack_ims_SLQSGetIMSVoIPConfig_t, 819
pAmrMode, 822
pRingingTimer, 822
pSettingResp, 822
ParamPresenceMask, 822
unpack_ims_SLQSGetRegMgrConfig
ims.h, 1378
unpack_ims_SLQSGetRegMgrConfig_t, 822
pIMSTestMode, 823
pPCSCFPort, 823
pSettingResp, 823
ParamPresenceMask, 823
unpack_ims_SLQSGetSIPConfig
ims.h, 1378
unpack_ims_SLQSGetSIPConfig_t, 823
pSIPLocalPort, 824
pSettingResp, 824
pSigCompEnabled, 824
pSubscribeTimer, 824
pTimerSIPReg, 825

pTimerT1, 825
pTimerT2, 825
pTimerTf, 825
ParamPresenceMask, 824
unpack_ims_SLQSImsConfigIndicationRegister
ims.h, 1379
unpack_ims_SLQSImsConfigIndicationRegister_t
ims.h, 1372
unpack_ims_SLQSRegMgrCfgCallBack_ind
ims.h, 1379
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 825
PCTlv, 825
PNTlv, 825
ParamPresenceMask, 825
TMTlv, 825
unpack_ims_SLQSSIPCfgCallBack_ind
ims.h, 1381
unpack_ims_SLQSSIPCfgCallBack_ind_t, 828
ParamPresenceMask, 829
SCTlv, 829
SPTlv, 829
SRTlv, 829
STTlv, 829
TT1Tlv, 829
TT2Tlv, 829
TTfTlv, 829
unpack_ims_SLQSSMSCfgCallBack_ind
ims.h, 1382
unpack_ims_SLQSSMSCfgCallBack_ind_t, 829
ParamPresenceMask, 830
SFTlv, 830
unpack_ims_SLQSSetIMSSMSConfig
ims.h, 1380
unpack_ims_SLQSSetIMSSMSConfig_t, 826
pSettingResp, 826
ParamPresenceMask, 826
unpack_ims_SLQSSetIMSUserConfig
ims.h, 1380
unpack_ims_SLQSSetIMSUserConfig_t, 826
pSettingResp, 826
ParamPresenceMask, 826
unpack_ims_SLQSSetIMSVoIPConfig
ims.h, 1380
unpack_ims_SLQSSetIMSVoIPConfig_t, 827
pSettingResp, 827
ParamPresenceMask, 827
unpack_ims_SLQSSetRegMgrConfig
ims.h, 1381
unpack_ims_SLQSSetRegMgrConfig_t, 827
pSettingResp, 827
ParamPresenceMask, 827
unpack_ims_SLQSSetSIPConfig
ims.h, 1381
unpack_ims_SLQSSetSIPConfig_t, 827
pSettingResp, 828
ParamPresenceMask, 828
unpack_ims_SLQSUserCfgCallBack_ind
ims.h, 1382
unpack_ims_SLQSUserCfgCallBack_ind_t, 830
IDTlv, 831
ParamPresenceMask, 831
unpack_ims_SLQSVoIPCfgCallBack_ind
ims.h, 1382
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 831
AMTlv, 833
ParamPresenceMask, 833
RTTlv, 833
SDTlv, 833
unpack_imsa_SLQSGetIMSARegStatus
imsa.h, 1385
unpack_imsa_SLQSGetIMSARegStatus_t, 833
ImsRegErrCode, 834
ImsRegStatus, 834
NewImsRegStatus, 834
ParamPresenceMask, 834
unpack_imsa_SLQSGetIMSAServiceStatus
imsa.h, 1386
unpack_imsa_SLQSGetIMSAServiceStatus_t, 834
ParamPresenceMask, 836
SmsServiceRat, 836
SmsServiceStatus, 836
UtServiceRat, 836
UtServiceStatus, 836
VoipServiceRat, 836
VoipServiceStatus, 836
VsServiceRat, 836
VsServiceStatus, 836
VtServiceRat, 836
VtServiceStatus, 837
unpack_imsa_SLQSImsaPdpStatusCallBack_ind
imsa.h, 1386
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 837
FailErrCode, 837
ParamPresenceMask, 837
PdpConnState, 837
unpack_imsa_SLQSImsaRatStatusCallBack_ind
imsa.h, 1387
unpack_imsa_SLQSImsaRatStatusCallBack_ind_t, 837
ParamPresenceMask, 838
RatHandover, 838
unpack_imsa_SLQSImsaRegStatusCallBack_ind
imsa.h, 1387
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 838
IMSRegistration, 839
ParamPresenceMask, 839
unpack_imsa_SLQSImsaSvcStatusCallBack_ind
imsa.h, 1387
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 839
ParamPresenceMask, 840
SmsRat, 840
SmsService, 840
UtRat, 840
UtService, 840
VoipRat, 840
VoipService, 840

VtRat, 840
VtService, 840
unpack_imsa_SLQSRegisterIMSAIndication
imsa.h, 1388
unpack_imsa_SLQSRegisterIMSAIndication_t
imsa.h, 1384
unpack_loc_BestAvailPos_Ind
loc.h, 1415
unpack_loc_BestAvailPos_Ind_t, 840
pAltitudeWrtEllipsoid, 846
pAltitudeWrtMeanSeaLevel, 846
pGpsTime, 846
pHeading, 846
pHeadingUnc, 846
pHorCirConf, 846
pHorEllpConf, 846
pHorReliability, 846
pHorUncCircular, 846
pHorUncEllipseOrientAzimuth, 846
pHorUncEllipseSemiMajor, 846
pHorUncEllipseSemiMinor, 846
pLatitude, 846
pLongitude, 846
pMagneticDeviation, 846
pPrecisionDilution, 846
pSensorDataUsage, 846
pSpeedHorizontal, 846
pSpeedUnc, 846
pSpeedVertical, 846
pSpeedVerticalUnc, 846
pSvUsedforFix, 846
pTechnologyMask, 846
pTimeSrc, 846
pTimeUnc, 847
pTimestampUtc, 847
pVertConfidence, 847
pVertReliability, 847
pVertUnc, 847
pXid, 847
ParamPresenceMask, 846
status, 847
Tlvresult, 847
unpack_loc_CradleMountCallback_Ind
loc.h, 1415
unpack_loc_CradleMountCallback_Ind_t, 847
cradleMountConfigStatus, 847
ParamPresenceMask, 848
unpack_loc_Delete_Assist_Data_t, 848
ParamPresenceMask, 848
Tlvresult, 848
unpack_loc_DeleteAssistData
loc.h, 1416
unpack_loc_DeleteAssistData_Ind
loc.h, 1416
unpack_loc_DeleteAssistData_Ind_t, 848
ParamPresenceMask, 849
status, 849
Tlvresult, 849
unpack_loc_DeleteAssistData_t
loc.h, 1409
unpack_loc_EngineState_Ind
loc.h, 1416
unpack_loc_EngineState_Ind_t, 849
engineState, 850
ParamPresenceMask, 850
Tlvresult, 850
unpack_loc_EventNMEA_Ind
loc.h, 1417
unpack_loc_EventNMEA_Ind_t, 850
NMEAData, 850
ParamPresenceMask, 851
Tlvresult, 851
unpack_loc_EventRegister
loc.h, 1417
unpack_loc_EventRegister_t, 851
ParamPresenceMask, 851
Tlvresult, 851
unpack_loc_EventTimeSyncCallback_Ind
loc.h, 1417
unpack_loc_EventTimeSyncCallback_Ind_t, 851
ParamPresenceMask, 852
timeSyncRefCounter, 852
unpack_loc_FixCriteria_Ind
loc.h, 1418
unpack_loc_FixCriteria_Ind_t, 852
loc_appProviderInfo, 852
loc_fixCriteriaStatus, 853
loc_horAccuracy, 853
loc_intermediateRptState, 853
loc_minInterval, 853
ParamPresenceMask, 853
unpack_loc_GetFixCriteria
loc.h, 1418
unpack_loc_GetFixCriteria_t
loc.h, 1409
unpack_loc_GetOpMode_Ind
loc.h, 1418
unpack_loc_GetOpMode_Ind_t, 853
pMode, 854
ParamPresenceMask, 854
Status, 854
unpack_loc_GetServer_Ind
loc.h, 1419
unpack_loc_GetServer_Ind_t, 854
pIPv4AddrInfo, 855
pIPv6AddrInfo, 855
pURL, 856
ParamPresenceMask, 855
serverStatus, 856
serverType, 856
Tlvresult, 856
unpack_loc_GnssSvInfo_Ind
loc.h, 1419
unpack_loc_GnssSvInfo_Ind_t, 856
altitudeAssumed, 856
pSatelliteInfo, 856

ParamPresenceMask, 856
TlvResult, 856
unpack_loc_InjectPositionCallback_Ind
loc.h, 1419
unpack_loc_InjectPositionCallback_Ind_t, 857
ParamPresenceMask, 857
status, 857
unpack_loc_InjectSensorDataCallback_Ind
loc.h, 1420
unpack_loc_InjectSensorDataCallback_Ind_t, 857
injectSensorDataStatus, 859
pAccelSamplesAccepted, 859
pAccelTempSamplesAccepted, 859
pGyroSamplesAccepted, 859
pGyroTempSamplesAccepted, 859
pOpaqueIdentifier, 859
ParamPresenceMask, 859
unpack_loc_InjectTimeSyncDataCallback_Ind
loc.h, 1420
unpack_loc_InjectTimeSyncDataCallback_Ind_t, 859
injectTimeSyncStatus, 860
ParamPresenceMask, 860
unpack_loc_InjectUTCTimeCallback_Ind
loc.h, 1420
unpack_loc_InjectUTCTimeCallback_Ind_t, 860
ParamPresenceMask, 860
status, 860
unpack_loc_PositionRpt_Ind
loc.h, 1421
unpack_loc_PositionRpt_Ind_t, 861
pAltitudeAssumed, 866
pAltitudeWrtEllipsoid, 866
pAltitudeWrtMeanSeaLevel, 866
pFixId, 866
pGpsTime, 866
pHeading, 866
pHeadingUnc, 866
pHorConfidence, 866
pHorReliability, 866
pHorUncCircular, 866
pHorUncEllipseOrientAzimuth, 866
pHorUncEllipseSemiMajor, 866
pHorUncEllipseSemiMinor, 866
pLatitude, 867
pLeapSeconds, 867
pLongitude, 867
pMagneticDeviation, 867
pPrecisionDilution, 867
pSensorDataUsage, 867
pSpeedHorizontal, 867
pSpeedUnc, 867
pSpeedVertical, 867
pSvUsedforFix, 867
pTechnologyMask, 867
pTimeSrc, 867
pTimeUnc, 867
pTimestampUtc, 867
pVertConfidence, 867
pVertReliability, 867
pVertUnc, 867
ParamPresenceMask, 866
sessionId, 867
sessionStatus, 867
TlvResult, 867
unpack_loc_SLQSLOCGetBestAvailPos
loc.h, 1423
unpack_loc_SLQSLOCGetBestAvailPos_t, 872
ParamPresenceMask, 873
TlvResult, 873
unpack_loc_SLQSLOCGetOpMode
loc.h, 1423
unpack_loc_SLQSLOCGetOpMode_t, 873
ParamPresenceMask, 873
TlvResult, 873
unpack_loc_SLQSLOCGetServer
loc.h, 1424
unpack_loc_SLQSLOCGetServer_t
loc.h, 1409
unpack_loc_SLQSLOCInjectPosition
loc.h, 1424
unpack_loc_SLQSLOCInjectPosition_t
loc.h, 1409
unpack_loc_SLQSLOCInjectSensorData
loc.h, 1424
unpack_loc_SLQSLOCInjectSensorData_t
loc.h, 1409
unpack_loc_SLQSLOCInjectUTCTime
loc.h, 1425
unpack_loc_SLQSLOCInjectUTCTime_t
loc.h, 1409
unpack_loc_SLQSLOCSetCradleMountConfig
loc.h, 1425
unpack_loc_SLQSLOCSetCradleMountConfig_t
loc.h, 1409
unpack_loc_SLQSLOCSetServer
loc.h, 1425
unpack_loc_SensorStreamingCallback_Ind
loc.h, 1421
unpack_loc_SensorStreamingCallback_Ind_t, 867
pAccelAcceptReady, 868
pAccelTempAcceptReady, 868
pGyroAcceptReady, 868
pGyroTempAcceptReady, 868
ParamPresenceMask, 868
unpack_loc_SetExtPowerConfig_Ind
loc.h, 1421
unpack_loc_SetExtPowerConfig_Ind_t, 868
ParamPresenceMask, 869
status, 869
TlvResult, 869
unpack_loc_SetExtPowerState
loc.h, 1422
unpack_loc_SetExtPowerState_t, 869
ParamPresenceMask, 870
TlvResult, 870
unpack_loc_SetOperationMode

loc.h, 1422
unpack_loc_SetOperationMode_Ind
loc.h, 1422
unpack_loc_SetOperationMode_Ind_t, 870
ParamPresenceMask, 870
status, 870
Tlvresult, 870
unpack_loc_SetOperationMode_t, 871
ParamPresenceMask, 871
Tlvresult, 871
unpack_loc_SetServer_Ind
loc.h, 1423
unpack_loc_SetServer_Ind_t, 871
ParamPresenceMask, 872
serverStatus, 872
Tlvresult, 872
unpack_loc_Start
loc.h, 1426
unpack_loc_Start_t, 873
ParamPresenceMask, 874
Tlvresult, 874
unpack_loc_Stop
loc.h, 1426
unpack_loc_Stop_t, 874
ParamPresenceMask, 874
Tlvresult, 874
unpack_nas_GetACCOLC
nas.h, 1479
unpack_nas_GetACCOLC_t, 874
pAccolc, 875
ParamPresenceMask, 875
unpack_nas_GetANAAAAuthenticationStatus
nas.h, 1479
unpack_nas_GetANAAAAuthenticationStatus_t, 875
pAuthStatus, 875
ParamPresenceMask, 875
unpack_nas_GetCDMANetworkParameters
nas.h, 1479
unpack_nas_GetCDMANetworkParameters_t, 875
Application, 877
Broadcast, 877
CustomSCP, 877
ForceRev0, 877
ParamPresenceMask, 877
Protocol, 878
RegForeignNID, 878
RegForeignSID, 878
RegHomeSID, 878
Roaming, 878
SCI, 878
SCM, 878
unpack_nas_GetHomeNetwork
nas.h, 1480
unpack_nas_GetHomeNetwork3GPP2
nas.h, 1480
unpack_nas_GetHomeNetwork3GPP2_t, 878
nameSize, 880
pMCC, 880
pMNC, 880
pNID, 880
pName, 880
pNw2DescDisp, 880
pNw2DescEnc, 880
pNw2DescLen, 880
pNw2MCC, 880
pNw2MNC, 880
pNw2Name, 880
pSID, 880
ParamPresenceMask, 880
unpack_nas_GetHomeNetwork_t, 880
mcc, 881
mnc, 881
name, 881
nid, 881
ParamPresenceMask, 881
sid, 881
unpack_nas_GetNetworkPreference
nas.h, 1481
unpack_nas_GetNetworkPreference_t, 881
ActiveTechPref, 882
Duration, 882
ParamPresenceMask, 882
PersistentTechPref, 882
Tlvresult, 882
unpack_nas_GetRFInfo
nas.h, 1481
unpack_nas_GetRFInfo_t, 883
instancesSize, 883
ParamPresenceMask, 883
RFBandInfoElements, 883
unpack_nas_GetServingNetwork
nas.h, 1481
unpack_nas_GetServingNetwork_t, 883
CSDomain, 885
DataCaps, 885
DataCapsLen, 885
MCC, 885
MNC, 886
Name, 886
nameSize, 886
PSDomain, 886
ParamPresenceMask, 886
RAN, 886
RadioIfaces, 886
RadioIfacesSize, 886
RegistrationState, 886
Roaming, 886
unpack_nas_GetServingNetworkCapabilities
nas.h, 1482
unpack_nas_GetServingNetworkCapabilities_t, 886
DataCaps, 887
DataCapsLen, 887
ParamPresenceMask, 887
unpack_nas_GetSignalStrengths
nas.h, 1482
unpack_nas_GetSignalStrengths_t, 887

len, 888
ParamPresenceMask, 888
radio, 888
rssi, 888
unpack_nas_InitiateDomainAttach
nas.h, 1482
unpack_nas_InitiateDomainAttach_t
nas.h, 1452
unpack_nas_PerformNetworkScan
nas.h, 1483
unpack_nas_PerformNetworkScan_t, 888
p3GppNetworkInfoInstances, 889
p3GppNetworkInstanceSize, 889
pLteOpModeTlv, 890
pPCIInfo, 890
pPCSInstance, 890
pPCSInstanceSize, 890
pRATINstance, 890
pRATInstanceSize, 890
pScanResult, 890
ParamPresenceMask, 889
unpack_nas_SLQSConfigSigInfo
nas.h, 1487
unpack_nas_SLQSConfigSigInfo_t
nas.h, 1453
unpack_nas_SLQSDDTMPrefIndicationCallback_Ind
nas.h, 1487
unpack_nas_SLQSDDTMPrefIndicationCallback_Ind_t, 895
unpack_nas_SLQSGetErrorRate
nas.h, 1487
unpack_nas_SLQSGetErrorRate_t, 895
pCDMAFrameErrRate, 896
pGSMBER, 896
pHDRPackErrRate, 896
pWCDMABER, 896
ParamPresenceMask, 896
unpack_nas_SLQSGetHomeNetwork
nas.h, 1488
unpack_nas_SLQSGetHomeNetwork_t, 896
mcc, 897
mnc, 897
name, 897
nid, 897
pHomeNwMNC3Gpp, 898
pNwNameSrc3Gpp, 898
ParamPresenceMask, 897
sid, 898
unpack_nas_SLQSGetNetworkTime
nas.h, 1488
unpack_nas_SLQSGetNetworkTime_t, 898
p3GPP2TimeInfo, 899
p3GPPTimeInfo, 899
ParamPresenceMask, 899
unpack_nas_SLQSGetOperatorNameData
nas.h, 1489
unpack_nas_SLQSGetOperatorNameData_t, 899
pNITZInformation, 900
pOperatorNameString, 900
pOperatorPLMNList, 900
pPLMNNetworkName, 900
pSrvcProviderName, 900
ParamPresenceMask, 900
unpack_nas_SLQSGetPLMNName
nas.h, 1489
unpack_nas_SLQSGetPLMNName_t, 900
longName, 903
longNameCI, 903
longNameEn, 903
longNameLen, 903
longNameSB, 903
ParamPresenceMask, 903
shortName, 903
shortNameCI, 903
shortNameEn, 903
shortNameLen, 903
shortNameSB, 903
spn, 903
spnEncoding, 903
spnLength, 903
unpack_nas_SLQSGetServingSystem
nas.h, 1489
unpack_nas_SLQSGetServingSystem_t, 903
BasestationID, 907
BasestationLatitude, 907
BasestationLongitude, 907
CDMASystemInfoExt, 907
CallBarStatus, 907
CellID, 907
ConcSvcInfo, 907
CurrentPLMN, 907
DTMInd, 907
DataSrvCapabilities, 907
DefaultRoamInd, 907
DetailedSvcInfo, 907
Gpp2TimeZone, 907
GppNetworkDSTAdjustment, 907
GppTimeZone, 907
HdrPersonality, 907
Lac, 907
NetworkID, 907
PRLInd, 907
ParamPresenceMask, 907
RoamIndicatorVal, 907
RoamingIndicatorList, 907
ServingSystem, 907
SystemID, 907
TrackAreaCode, 907
unpack_nas_SLQSGetServingSystemV2
nas.h, 1490
unpack_nas_SLQSGetServingSystemV2_t, 907
BasestationID, 911
BasestationLatitude, 911
BasestationLongitude, 911
CDMASystemInfoExt, 911
CallBarStatus, 911

CellID, 911
ConcSvcInfo, 911
CurrentPLMN, 911
DTMInd, 912
DataSrvCapabilities, 911
DefaultRoamInd, 911
DetailedSvcInfo, 912
Gpp2TimeZone, 912
GppNetworkDSTAdjustment, 912
GppTimeZone, 912
HdrPersonality, 912
Lac, 912
NetworkID, 912
PRLInd, 912
ParamPresenceMask, 912
RoamIndicatorVal, 912
RoamingIndicatorList, 912
ServingSystem, 912
SystemID, 912
TrackAreaCode, 912
unpack_nas_SLQSGetSignalStrength
nas.h, 1490
unpack_nas_SLQSGetSignalStrength_t, 912
ecioList, 914
ecioListLen, 914
errorRateList, 914
errorRateListLen, 914
Io, 914
ltersrp, 914
ltesnr, 914
ParamPresenceMask, 915
rsrqInfo, 915
rxSignalStrengthList, 915
rxSignalStrengthListLen, 915
signalStrengthReqMask, 915
sinr, 915
unpack_nas_SLQSGetSysInfo
nas.h, 1490
unpack_nas_SLQSGetSysInfo_t, 915
pAddCDMASysInfo, 917
pAddGSMSysInfo, 918
pAddHDRSysInfo, 918
pAddLTESysInfo, 918
pCDMASrvStatusInfo, 918
pCDMASysInfo, 918
pCampedCiotLteOpMode, 918
pGSMCallBarringSysInfo, 918
pGSMCipherDomainSysInfo, 918
pGSMSrvStatusInfo, 918
pGSMSysInfo, 918
pHDRSrvStatusInfo, 918
pHDRSysInfo, 918
pLTESrvStatusInfo, 918
pLTESysInfo, 918
pLTEVoiceSupportSysInfo, 918
pWCDMASysInfo, 918
ParamPresenceMask, 918
unpack_nas_SLQSGetSysInfoV2
nas.h, 1491
unpack_nas_SLQSGetSysInfoV2_t, 918
pAddGSMSysInfo, 922
pAddHDRSysInfo, 922
pAddLTESysInfo, 922
pCDMASysInfo, 922
pCampedCiotLteOpMode, 922
pGSMSrvStatusInfo, 922
pGSMSysInfo, 922
pHDRSrvStatusInfo, 922
pHDRSysInfo, 922
pImsVoiceSupportLte, 923
pLTESrvStatusInfo, 923
pLTESysInfo, 923
pLteEmbmsCoverage, 923
pLteEmbmsTraceId, 923
pLteRegDomain, 923
pLteVoiceDomain, 923
pNR5GCellStatusInfo, 923
pNR5GSrvStatusinfo, 923
pNR5GSysInfo, 923
pSimRejInfo, 923
pSrvRegRestriction, 923
ParamPresenceMask, 922
unpack_nas_SLQSGetSysSelectionPref
nas.h, 1491
unpack_nas_SLQSGetSysSelectionPref_t, 923
pBandPref, 928
pEmerMode, 928
pGWAcqOrderPref, 928
pLTEBandPref, 928
pModePref, 928
pNetSelPref, 928
pPRLPref, 928
pRoamPref, 928
pSrvDomainPref, 928
ParamPresenceMask, 928
unpack_nas_SLQSGetSysSelectionPrefExt
nas.h, 1492
unpack_nas_SLQSGetSysSelectionPrefExt_t, 928
pAcqOrderPref, 930
pBandPref, 930
pCiotAcqOrderPref, 930
pCiotLteOpModePref, 930
pEmerMode, 930
pGWAcqOrderPref, 930
pLTEBandPref, 930
pLteM1BandPref, 931
pLteNb1BandPref, 931
pModePref, 931
pNetSelPref, 931
pNr5gBandPref, 931
pPRLPref, 931
pRatDisabledMask, 931
pRoamPref, 931
pSrvDomainPref, 931
unpack_nas_SLQSGetSysSelectionPrefExtV2
nas.h, 1492

unpack_nas_SLQSGetSysSelectionPrefExtV2_t, 931
pAcqOrderPref, 934
pBandPref, 934
pCiotAcqOrderPref, 934
pEmerMode, 934
pLteM1BandPref, 934
pLteNb1BandPref, 934
pModePref, 934
pNetSelPref, 934
pNr5gBandPref, 934
pPRLPref, 934
pRatDisabledMask, 934
pRoamPref, 934
pSrvDomainPref, 934
pSrvRegRestric, 934
pUsageSetting, 934
pVoiceDomainPref, 934
ParamPresenceMask, 934
unpack_nas_SLQSInitiateNetworkRegistration
nas.h, 1492
unpack_nas_SLQSInitiateNetworkRegistration_t
nas.h, 1453
unpack_nas_SLQSNASGetForbiddenNetworks
nas.h, 1495
unpack_nas_SLQSNASGetForbiddenNetworks_t, 943
ParamPresenceMask, 943
unpack_nas_SLQSNASGeteDRXParams
nas.h, 1495
unpack_nas_SLQSNASGeteDRXParams_t, 940
pCycleLen, 941
pEdrxEnable, 941
ParamPresenceMask, 941
unpack_nas_SLQSNASGeteDRXParamsExt
nas.h, 1495
unpack_nas_SLQSNASGeteDRXParamsExt_t, 941
unpack_nas_SLQSNASSeteDRXParams
nas.h, 1499
unpack_nas_SLQSNASSeteDRXParams_t
nas.h, 1453
unpack_nas_SLQSNASSwiGetChannelLock
nas.h, 1500
unpack_nas_SLQSNASSwiGetChannelLock_t, 954
ParamPresenceMask, 955
unpack_nas_SLQSNASSwiSetChannelLock
nas.h, 1501
unpack_nas_SLQSNASSwiSetChannelLock_t
nas.h, 1453
unpack_nas_SLQSNasConfigSigInfo2
nas.h, 1493
unpack_nas_SLQSNasConfigSigInfo2_t
nas.h, 1453
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind
nas.h, 1493
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t, 935
unpack_nas_SLQSNasGet3GPP2Subscription
nas.h, 1494
unpack_nas_SLQSNasGet3GPP2Subscription_t, 936
pDirNum, 937
pMdn, 937
ParamPresenceMask, 937
unpack_nas_SLQSNasGetCellLocationInfo
nas.h, 1494
unpack_nas_SLQSNasGetCellLocationInfo_t, 937
pCDMAInfo, 938
pLTEInfoInterfreq, 938
pLTEInfoIntrafreq, 938
pUMTSInfo, 938
ParamPresenceMask, 938
unpack_nas_SLQSNasGetCellLocationInfoV2
nas.h, 1494
unpack_nas_SLQSNasGetCellLocationInfoV2_t, 938
pLteEarfcnInfo, 940
ParamPresenceMask, 940
unpack_nas_SLQSNasGetHDRColorCode
nas.h, 1496
unpack_nas_SLQSNasGetHDRColorCode_t, 943
pColorCode, 944
ParamPresenceMask, 944
unpack_nas_SLQSNasGetRFInfo
nas.h, 1496
unpack_nas_SLQSNasGetRFInfo_t, 944
pLTEOperationMode, 945
pRfBandInfoExtFormat, 945
pRfBandwidthInfo, 945
pRfDedicatedBandInfo, 945
rfbandInfoList, 945
Tlvresult, 945
unpack_nas_SLQSNasGetSigInfo
nas.h, 1496
unpack_nas_SLQSNasGetSigInfo_t, 945
CDMASSInfo, 946
GSMSSInfo, 946
HDRSSInfo, 946
LTESSInfo, 946
ParamPresenceMask, 946
unpack_nas_SLQSNasGetTxRxInfo
nas.h, 1497
unpack_nas_SLQSNasGetTxRxInfo_t, 946
pRXChain0Info, 946
pRXChain1Info, 946
pTXInfo, 946
ParamPresenceMask, 946
unpack_nas_SLQSNasIndicationRegisterExt
nas.h, 1497
unpack_nas_SLQSNasIndicationRegisterExt_t
nas.h, 1453
unpack_nas_SLQSNasIndicationRegisterV2
nas.h, 1497
unpack_nas_SLQSNasIndicationRegisterV2_t
nas.h, 1453
unpack_nas_SLQSNasNetworkRejectCallback_Ind
nas.h, 1498
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t, 947
pCsgId, 949

pLteOpMode, 949
pPlmnId, 949
ParamPresenceMask, 949
radioIf, 949
rejectCause, 949
serviceDomain, 949
unpack_nas_SLQSNasNetworkTimeCallBack_ind
nas.h, 1498
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 949
pDayltSavAdj, 950
pRadioInterface, 950
pTimeZone, 950
ParamPresenceMask, 950
universalTime, 950
unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind
nas.h, 1499
unpack_nas_SLQSNasPerformIncNetworkScanCallback_Ind_t, 950
unpack_nas_SLQSNasRFBandInfoCallback_Ind
nas.h, 1499
unpack_nas_SLQSNasRFBandInfoCallback_Ind_t, 951
unpack_nas_SLQSNasSigInfoCallback_ind
nas.h, 1500
unpack_nas_SLQSNasSigInfoCallback_ind_t, 953
pRscp, 954
ParamPresenceMask, 954
unpack_nas_SLQSNasSwiIndicationRegister
nas.h, 1500
unpack_nas_SLQSNasSwiIndicationRegister_t
nas.h, 1453
unpack_nas_SLQSNasSwiModemStatus
nas.h, 1501
unpack_nas_SLQSNasSwiModemStatus_t, 955
commonInfo, 955
pLTEInfo, 955
ParamPresenceMask, 955
unpack_nas_SLQSNasSwiOTAMessageCallback_ind
nas.h, 1501
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 955
Info, 956
Tlvresult, 956
unpack_nas_SLQSNasSysInfoCallback_ind
nas.h, 1502
unpack_nas_SLQSNasTimerCallback_ind
nas.h, 1502
unpack_nas_SLQSNasTimerCallback_ind_t, 956
ParamPresenceMask, 957
t3396_apn, 957
t3396_plmn_id, 957
t3396_val, 957
unpack_nas_SLQSPerformIncNetworkScan
nas.h, 1502
unpack_nas_SLQSPerformIncNetworkScan_t
nas.h, 1453
unpack_nas_SLQSPerformNetworkScanV2
nas.h, 1503
unpack_nas_SLQSPerformNetworkScanV2_t, 957
p3GppNetworkInfoInstances, 958
p3GppNetworkInstanceSize, 958
pLteOpModeTlv, 958
pNetworkNameSrcTlv, 958
pPCIInfo, 959
pPCSInstance, 959
pPCSInstanceSize, 959
pRATINstance, 959
pRATInstanceSize, 959
pScanResult, 959
ParamPresenceMask, 958
unpack_nas_SLQSSet3GPP2SubscriptionInfo
nas.h, 1503
unpack_nas_SLQSSet3GPP2SubscriptionInfo_t
nas.h, 1453
unpack_nas_SLQSSetBandPreference
nas.h, 1503
unpack_nas_SLQSSetBandPreference_t
nas.h, 1453
unpack_nas_SLQSSetDDTMPref
nas.h, 1504
unpack_nas_SLQSSetDDTMPref_t
nas.h, 1453
unpack_nas_SLQSSetSignalStrengthsCallback
nas.h, 1504
unpack_nas_SLQSSetSignalStrengthsCallback_t
nas.h, 1453
unpack_nas_SLQSSetSysSelectionPref
nas.h, 1504
unpack_nas_SLQSSetSysSelectionPref_t
nas.h, 1453
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind
nas.h, 1505
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 959
Info, 959
Tlvresult, 959
unpack_nas_SLQSSetSysSelectionPrefExt
nas.h, 1505
unpack_nas_SLQSSetSysSelectionPrefExt_t
nas.h, 1453
unpack_nas_SLQSSwiGetHDRPersonality
nas.h, 1505
unpack_nas_SLQSSwiGetHDRPersonality_t, 959
pCurrentPersonality, 960
pPersonalityListLength, 960
pProtocolSubtypeElement, 960
ParamPresenceMask, 960
unpack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 1506
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 960
pAppSubType, 961
pCurrentPrsnlty, 961
pPersonalityListLength, 961
ParamPresenceMask, 961
unpack_nas_SLQSSwiGetHRPDStats

nas.h, 1506
unpack_nas_SLQSSwiGetHRPDStats_t, 961
pPilotSetData, 962
pUATI, 962
ParamPresenceMask, 962
unpack_nas_SLQSSwiGetLteCQI
nas.h, 1506
unpack_nas_SLQSSwiGetLteCQI_t, 962
ParamPresenceMask, 963
ValidityCW0, 963
ValidityCW1, 963
unpack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 1507
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 963
pSccRxInfo, 963
ParamPresenceMask, 963
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind
nas.h, 1507
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t, 964
unpack_nas_SLQSSwiNetworkDebug
nas.h, 1507
unpack_nas_SLQSSwiNetworkDebug_t, 964
pDataStatusDetail, 965
pDeviceConfigDetail, 965
pNetworkStat1x, 965
pNetworkStatEVDO, 965
pObjectVer, 965
ParamPresenceMask, 965
unpack_nas_SLQSSwiPSDetach
nas.h, 1508
unpack_nas_SLQSSwiPSDetach_t
nas.h, 1453
unpack_nas_SLQSSwiRandIndicatorCallback_Ind
nas.h, 1508
unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t, 965
ParamPresenceMask, 966
rankIndicatorTlv, 966
unpack_nas_SLQSSysInfoCallback_ind_t, 966
pGSMSysInfo, 969
pHDRSysInfo, 969
pLTESysInfo, 969
pLteCiotOpModeTlv, 969
pNR5GCellStatus, 969
pSysInfoNoChange, 970
ParamPresenceMask, 969
unpack_nas_SetACCOLC
nas.h, 1483
unpack_nas_SetACCOLC_t
nas.h, 1453
unpack_nas_SetCDMANetworkParameters
nas.h, 1483
unpack_nas_SetCDMANetworkParameters_t
nas.h, 1453
unpack_nas_SetDataCapabilitiesCallback_ind
nas.h, 1484
unpack_nas_SetDataCapabilitiesCallback_ind_t, 890
dataCaps, 890
dataCapsSize, 890
ParamPresenceMask, 890
unpack_nas_SetEventReportInd
nas.h, 1484
unpack_nas_SetEventReportInd_t, 890
ParamPresenceMask, 891
RFTlv, 891
RRTlv, 891
SLQSSSTlv, 891
SSTlv, 891
unpack_nas_SetLURejectCallback
nas.h, 1484
unpack_nas_SetLURejectCallback_t
nas.h, 1453
unpack_nas_SetNasLTECphyCaIndCallback_ind
nas.h, 1485
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 892
ParamPresenceMask, 892
sNumScellsConfig, 893
unpack_nas_SetNetworkPreference
nas.h, 1485
unpack_nas_SetNetworkPreference_t, 893
ParamPresenceMask, 893
Tlvresult, 893
unpack_nas_SetRFInfoCallback
nas.h, 1486
unpack_nas_SetRFInfoCallback_t
nas.h, 1453
unpack_nas_SetRoamingIndicatorCallback_ind
nas.h, 1486
unpack_nas_SetRoamingIndicatorCallback_ind_t, 893
ParamPresenceMask, 894
roaming, 894
unpack_nas_SetServingSystemCallback_ind
nas.h, 1486
unpack_nas_SetServingSystemCallback_ind_t, 894
ParamPresenceMask, 894
SSInfo, 894
Tlvresult, 894
unpack_nas_SlqsGetLTECphyCAInfo
nas.h, 1488
unpack_nas_SlqsGetLTECphyCAInfo_t, 898
LTECphyCaInfo, 898
ParamPresenceMask, 898
Tlvresult, 898
unpack_omaDmConfigTlv_t, 970
alertmsg, 970
alertmsglength, 971
state, 971
userInputReq, 971
userInputTimeout, 971
unpack_omaDmFotaTlv_t, 971
description, 972
descriptionlength, 972
fwdloadsize, 972
fwloadComplete, 972
namelength, 972

package_name, 972
sessionType, 972
severity, 972
state, 973
updateCompleteStatus, 973
userInputReq, 973
userInputTimeout, 973
version, 973
versionlength, 973
unpack_omaDmNotificationsTlv_t, 973
notification, 974
sessionStatus, 974
unpack_pds_ForceXTRADownload
pds.h, 1523
unpack_pds_ForceXTRADownload_t
pds.h, 1513
unpack_pds_GetPDSDefaults
pds.h, 1523
unpack_pds_GetPDSDefaults_t, 974
pAccuracy, 974
pInterval, 974
pOperation, 974
pTimeout, 975
ParamPresenceMask, 974
unpack_pds_GetPDSState
pds.h, 1524
unpack_pds_GetPDSState_t, 975
pEnabledStatus, 975
pTrackingStatus, 975
ParamPresenceMask, 975
unpack_pds_GetPortAutomaticTracking
pds.h, 1524
unpack_pds_GetPortAutomaticTracking_t, 975
ParamPresenceMask, 976
pbAuto, 976
unpack_pds_GetServiceAutomaticTracking
pds.h, 1524
unpack_pds_GetServiceAutomaticTracking_t, 976
ParamPresenceMask, 976
pbAuto, 976
unpack_pds_GetXTRAAutomaticDownload
pds.h, 1525
unpack_pds_GetXTRAAutomaticDownload_t, 977
pInterval, 977
ParamPresenceMask, 977
pbEnabled, 977
unpack_pds_GetXTRANetwork
pds.h, 1525
unpack_pds_GetXTRANetwork_t, 977
pPreference, 978
ParamPresenceMask, 978
unpack_pds_GetXTRAValidity
pds.h, 1526
unpack_pds_GetXTRAValidity_t, 978
pDuration, 979
pGPSWeek, 979
pGPSWeekOffset, 979
ParamPresenceMask, 978
unpack_pds_PDSInjectTimeReference
pds.h, 1526
unpack_pds_PDSInjectTimeReference_t
pds.h, 1513
unpack_pds_ResetPDSData
pds.h, 1526
unpack_pds_ResetPDSData_t
pds.h, 1513
unpack_pds_SLQSGetAGPSConfig
pds.h, 1530
unpack_pds_SLQSGetAGPSConfig_t, 981
pServerAddress, 982
pServerPort, 982
pServerURL, 982
ParamPresenceMask, 982
unpack_pds_SLQSGetGPSStateInfo
pds.h, 1530
unpack_pds_SLQSGetGPSStateInfo_t, 982
Altitude, 986
EngineState, 986
HorizontalUncertainty, 987
Iono_valid, 987
Latitude, 987
Longitude, 987
ParamPresenceMask, 987
Time_uncert_ms, 987
ValidMask, 987
VerticalUncertainty, 987
unpack_pds_SLQSPDSDeterminePosition
pds.h, 1530
unpack_pds_SLQSPDSDeterminePosition_t
pds.h, 1514
unpack_pds_SLQSPDSInjectAbsoluteTimeReference
pds.h, 1531
unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t
pds.h, 1514
unpack_pds_SLQSPDSInjectPositionData
pds.h, 1531
unpack_pds_SLQSPDSInjectPositionData_t
pds.h, 1514
unpack_pds_SLQSSetAGPSConfig
pds.h, 1531
unpack_pds_SLQSSetAGPSConfig_t
pds.h, 1514
unpack_pds_SLQSSetPositionMethodState
pds.h, 1532
unpack_pds_SLQSSetPositionMethodState_t
pds.h, 1514
unpack_pds_SetEventReport_Ind
pds.h, 1527
unpack_pds_SetEventReport_Ind_t, 979
dLatitude, 980
dLongitude, 980
has_PositionDataNMEA, 980
has_SessionStatus, 980
has_dLatitude, 980
has_dLongitude, 980
has_posSrc, 980

ParamPresenceMask, 980
posSrc, 981
PositionDataNMEA, 980
SessionStatus, 981
unpack_pds_SetEventReportCallback
pds.h, 1527
unpack_pds_SetEventReportCallback_t
pds.h, 1513
unpack_pds_SetPDSDefaults
pds.h, 1527
unpack_pds_SetPDSDefaults_t
pds.h, 1513
unpack_pds_SetPDSState
pds.h, 1528
unpack_pds_SetPDSState_t
pds.h, 1513
unpack_pds_SetPdsState_Ind
pds.h, 1528
unpack_pds_SetPdsState_Ind_t, 981
EnabledStatus, 981
ParamPresenceMask, 981
TlvPresent, 981
TrackingStatus, 981
unpack_pds_SetPortAutomaticTracking
pds.h, 1528
unpack_pds_SetPortAutomaticTracking_t
pds.h, 1513
unpack_pds_SetServiceAutomaticTracking
pds.h, 1529
unpack_pds_SetServiceAutomaticTracking_t
pds.h, 1514
unpack_pds_SetXTRAAutomaticDownload
pds.h, 1529
unpack_pds_SetXTRAAutomaticDownload_t
pds.h, 1514
unpack_pds_SetXTRANetwork
pds.h, 1529
unpack_pds_SetXTRANetwork_t
pds.h, 1514
unpack_pds_StartPDSTrackingSessionExt
pds.h, 1532
unpack_pds_StartPDSTrackingSessionExt_t
pds.h, 1514
unpack_pds_StopPDSTrackingSession
pds.h, 1532
unpack_pds_StopPDSTrackingSession_t
pds.h, 1514
unpack_qmi_t, 987
msgid, 988
type, 988
xid, 988
unpack_qos_BindDataPort
qos.h, 1601
unpack_qos_BindDataPort_t, 988
ParamPresenceMask, 988
unpack_qos_IPv4Addr_t, 989
addr, 989
subnetMask, 989
unpack_qos_IPv6Addr_t, 989
addr, 990
prefixLen, 990
unpack_qos_IPv6TrafCls_t, 990
mask, 990
val, 990
unpack_qos_Port_t, 991
port, 991
range, 991
unpack_qos_QosFlowInfo_t, 991
BearerID, 992
is_RxQFlowGranted_Available, 992
is_TxQFlowGranted_Available, 992
NumRxFilters, 992
NumTxFilters, 992
QFlowState, 992
RxQFilter, 992
RxQFlowGranted, 993
TxQFilter, 993
TxQFlowGranted, 993
unpack_qos_QosFlowInfoState_t, 993
id, 993
isNewFlow, 993
state, 993
unpack_qos_SLQSQosGetNetworkStatus
qos.h, 1601
unpack_qos_SLQSQosGetNetworkStatus_t, 993
NWQoSStatus, 994
ParamPresenceMask, 994
unpack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 1601
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 994
ambr_dl, 995
ambr_dl_ext, 995
ambr_dl_ext2, 995
ambr_ul, 995
ambr_ul_ext, 995
ambr_ul_ext2, 995
apnId, 995
ParamPresenceMask, 995
unpack_qos_SLQSQosSwiReadDataStats
qos.h, 1602
unpack_qos_SLQSQosSwiReadDataStats_t, 995
apnId, 997
numQosFlow, 997
ParamPresenceMask, 997
qosFlow, 997
total_rx_bytes, 997
total_rx_pkt, 997
total_tx_bytes, 997
total_tx_pkt, 997
unpack_qos_SLQSSetQosEventCallback
qos.h, 1602
unpack_qos_SLQSSetQosEventCallback_ind
qos.h, 1603
unpack_qos_SLQSSetQosEventCallback_ind_t, 997
NumFlows, 997

ParamPresenceMask, 997
QosFlowInfo, 998
unpack_qos_SLQSSetQosEventCallback_t, 998
ParamPresenceMask, 998
unpack_qos_SLQSSetQosNWStatusCallback_ind
qos.h, 1603
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 998
status, 998
unpack_qos_SLQSSetQosPriEventCallback_ind
qos.h, 1604
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 999
event, 999
ParamPresenceMask, 999
unpack_qos_SLQSSetQosStatusCallback_ind
qos.h, 1605
unpack_qos_SLQSSetQosStatusCallback_ind_t, 999
event, 1000
id, 1000
ParamPresenceMask, 1000
reason, 1000
status, 1000
unpack_qos_Tos_t, 1009
mask, 1009
val, 1009
unpack_qos_dataRate_t, 988
dataRateMax, 989
guaranteedRate, 989
unpack_qos_pktErrRate_t, 990
exponent, 991
multiplier, 991
unpack_qos_swiQosFilter_t, 1001
EspSpi, 1003
IPv4DstAddr, 1003
IPv4SrcAddr, 1003
IPv4Tos, 1003
IPv6DstAddr, 1003
IPv6Label, 1003
IPv6SrcAddr, 1003
IPv6TrafCls, 1003
Id, 1003
index, 1003
is_EspSpi_Available, 1003
is_IPv4DstAddr_Available, 1003
is_IPv4SrcAddr_Available, 1003
is_IPv4Tos_Available, 1003
is_IPv6DstAddr_Available, 1003
is_IPv6Label_Available, 1003
is_IPv6SrcAddr_Available, 1003
is_IPv6TrafCls_Available, 1003
is_Id_Available, 1003
is_NxtHdrProto_Available, 1003
is_Precedence_Available, 1003
is_TCPDstPort_Available, 1003
is_TCPSrcPort_Available, 1003
is_TranDstPort_Available, 1004
is_TranSrcPort_Available, 1004
is_UDPDstPort_Available, 1004
is_UDPSrcPort_Available, 1004
NxtHdrProto, 1004
Precedence, 1004
TCPDstPort, 1004
TCPSrcPort, 1004
TranDstPort, 1004
TranSrcPort, 1004
UDPDstPort, 1004
UDPSrcPort, 1004
version, 1004
unpack_qos_swiQosFlow_t, 1004
DataRate, 1007
index, 1007
is_DataRate_Available, 1007
is_Jitter_Available, 1007
is_Latency_Available, 1007
is_LteQci_Available, 1007
is_MaxAllowedPktSz_Available, 1007
is_MinPolicedPktSz_Available, 1007
is_PktErrRate_Available, 1007
is_ProfileId3GPP2_Available, 1007
is_TokenBucket_Available, 1007
is_TrafficClass_Available, 1007
is_val_3GPP2Pri_Available, 1007
is_val_3GPPImCn_Available, 1007
is_val_3GPPSigInd_Available, 1007
Jitter, 1007
Latency, 1007
LteQci, 1008
MaxAllowedPktSz, 1008
MinPolicedPktSz, 1008
PktErrRate, 1008
ProfileId3GPP2, 1008
TokenBucket, 1008
TrafficClass, 1008
val_3GPP2Pri, 1008
val_3GPPImCn, 1008
val_3GPPResResidualBER, 1008
val_3GPPSigInd, 1008
val_3GPPTraHdlPri, 1008
unpack_qos_tokenBucket_t, 1008
bucketSz, 1008
peakRate, 1008
tokenRate, 1008
unpack_result_code_only
common.h, 1294
unpack_result_t, 1010
ParamPresenceMask, 1010
Tlvresult, 1010
unpack_rms_GetSMSWake
rms.h, 1606
unpack_rms_GetSMSWake_t, 1010
enabled, 1011
ParamPresenceMask, 1011
wake_mask, 1011
unpack_rms_SetSMSWake
rms.h, 1607
unpack_rms_SetSMSWake_t, 1011

ParamPresenceMask, 1011
unpack_sar_SLQSGetRfSarState
sar.h, 1608
unpack_sar_SLQSGetRfSarState_t, 1013
pSarRFState, 1013
ParamPresenceMask, 1013
unpack_sar_SLQSSetRfSarState
sar.h, 1609
unpack_sar_SLQSSetRfSarState_t
sar.h, 1608
unpack_sms_GetSMSCAddress
sms.h, 1622
unpack_sms_GetSMSCAddress_t, 1013
addressSize, 1014
pSMSCAddress, 1014
pSMSCType, 1014
ParamPresenceMask, 1014
typeSize, 1014
unpack_sms_SLQSDeleteSMS
sms.h, 1624
unpack_sms_SLQSDeleteSMS_t, 1018
ParamPresenceMask, 1018
unpack_sms_SLQSGetIndicationRegister
sms.h, 1624
unpack_sms_SLQSGetIndicationRegister_t, 1018
pGetIndicationRegInfo, 1019
ParamPresenceMask, 1019
unpack_sms_SLQSGetMessageWaiting
sms.h, 1624
unpack_sms_SLQSGetMessageWaiting_t, 1019
pGetMsgWaitingInfoResp, 1019
ParamPresenceMask, 1019
unpack_sms_SLQSGetSMS
sms.h, 1625
unpack_sms_SLQSGetSMS_t, 1019
message, 1020
messageFormat, 1020
messageSize, 1020
messageTag, 1020
ParamPresenceMask, 1020
unpack_sms_SLQSGetSMSList
sms.h, 1625
unpack_sms_SLQSGetSMSList_t, 1021
messageList, 1022
messageListSize, 1022
ParamPresenceMask, 1022
unpack_sms_SLQSGetSmsBroadcastConfig
sms.h, 1625
unpack_sms_SLQSGetSmsBroadcastConfig_t, 1020
pBroadcastConfig, 1021
pCDMABroadcastConfig, 1021
ParamPresenceMask, 1021
unpack_sms_SLQSGetTransLayerInfo
sms.h, 1626
unpack_sms_SLQSGetTransLayerInfo_t, 1022
pGetTransLayerInfo, 1022
ParamPresenceMask, 1022
unpack_sms_SLQSGetTransNWRegInfo
sms.h, 1626
unpack_sms_SLQSGetTransNWRegInfo_t, 1022
ParamPresenceMask, 1023
unpack_sms_SLQSModifySMSStatus
sms.h, 1626
unpack_sms_SLQSModifySMSStatus_t, 1023
ParamPresenceMask, 1023
unpack_sms_SLQSNWRegInfoCallback_ind
sms.h, 1627
unpack_sms_SLQSNWRegInfoCallback_ind_t, 1023
NWRegStat, 1024
ParamPresenceMask, 1024
unpack_sms_SLQSSendAsyncSMS
sms.h, 1627
unpack_sms_SLQSSendAsyncSMS_t, 1024
ParamPresenceMask, 1024
unpack_sms_SLQSSetIndicationRegister
sms.h, 1627
unpack_sms_SLQSSetIndicationRegister_t, 1024
ParamPresenceMask, 1025
unpack_sms_SLQSSetSmsBroadcastActivation
sms.h, 1628
unpack_sms_SLQSSetSmsBroadcastActivation_t, 1025
ParamPresenceMask, 1025
unpack_sms_SLQSSetSmsBroadcastConfig
sms.h, 1628
unpack_sms_SLQSSetSmsBroadcastConfig_t, 1025
ParamPresenceMask, 1026
unpack_sms_SLQSSetSmsStorage
sms.h, 1628
unpack_sms_SLQSSetSmsStorage_t, 1026
ParamPresenceMask, 1026
unpack_sms_SLQSSmsGetMaxStorageSize
sms.h, 1629
unpack_sms_SLQSSmsGetMaxStorageSize_t, 1026
pMaxStorageSizeResp, 1027
ParamPresenceMask, 1026
unpack_sms_SLQSSmsGetMessageProtocol
sms.h, 1629
unpack_sms_SLQSSmsGetMessageProtocol_t, 1027
pMessageProtocol, 1027
ParamPresenceMask, 1027
unpack_sms_SLQSSmsSetRoutes
sms.h, 1629
unpack_sms_SLQSSmsSetRoutes_t, 1027
ParamPresenceMask, 1027
unpack_sms_SLQSSwiGetSMSStorage
sms.h, 1630
unpack_sms_SLQSSwiGetSMSStorage_t, 1028
pSmsStorage, 1028
ParamPresenceMask, 1028
unpack_sms_SLQSTransLayerInfoCallback_ind
sms.h, 1630
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 1028
pTransLayerInfo, 1029
ParamPresenceMask, 1029
regInd, 1029
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind

sms.h, 1630
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t, 1029
causeCode, 1031
errorClass, 1031
messageID, 1031
RPCause, 1031
sendStatus, 1031
TPCause, 1031
userData, 1031
unpack_sms_SLQSWmsMemoryFullCallBack_ind
sms.h, 1631
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 1031
messageMode, 1032
ParamPresenceMask, 1032
storageType, 1032
unpack_sms_SLQSWmsMessageWaitingCallBack_ind
sms.h, 1631
unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t, 1032
msgWaitInfo, 1032
numInstances, 1032
ParamPresenceMask, 1032
unpack_sms_SaveSMS
sms.h, 1622
unpack_sms_SaveSMS_t, 1014
pMessageIndex, 1015
ParamPresenceMask, 1015
unpack_sms_SendSMS
sms.h, 1622
unpack_sms_SendSMS_t, 1015
messageFailureCode, 1015
messageID, 1015
ParamPresenceMask, 1015
unpack_sms_SetNewSMSCallback
sms.h, 1623
unpack_sms_SetNewSMSCallback_ind
sms.h, 1623
unpack_sms_SetNewSMSCallback_ind_t, 1015
ETWSTlv, 1017
IMSTlv, 1017
MMTlv, 1017
NewMMTlv, 1017
ParamPresenceMask, 1017
SMSCTlv, 1017
TRMessageTlv, 1017
unpack_sms_SetNewSMSCallback_t, 1017
ParamPresenceMask, 1017
unpack_sms_SetSMSCAddress
sms.h, 1623
unpack_sms_SetSMSCAddress_t, 1017
ParamPresenceMask, 1018
unpack_swiaudio_SLQSGetM2MAVMute
swiaudio.h, 1638
unpack_swiaudio_SLQSGetM2MAVMute_t, 1034
CwtMute, 1035
EarMute, 1035
MicMute, 1035
ParamPresenceMask, 1035
unpack_swiaudio_SLQSGetM2MAudioProfile
swiaudio.h, 1638
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1032
CwtMute, 1033
EarMute, 1033
Generator, 1033
MicMute, 1034
ParamPresenceMask, 1034
Profile, 1034
Volume, 1034
unpack_swiaudio_SLQSGetM2MAudioVolume
swiaudio.h, 1638
unpack_swiaudio_SLQSGetM2MAudioVolume_t, 1034
Level, 1034
ParamPresenceMask, 1034
unpack_swiaudio_SLQSGetM2MSpkrGain
swiaudio.h, 1639
unpack_swiaudio_SLQSGetM2MSpkrGain_t, 1035
ParamPresenceMask, 1036
Value, 1036
unpack_swiaudio_SLQSSetM2MAVMute
swiaudio.h, 1641
unpack_swiaudio_SLQSSetM2MAVMute_t
swiaudio.h, 1633
unpack_swiaudio_SLQSSetM2MAudioAVCFG
swiaudio.h, 1639
unpack_swiaudio_SLQSSetM2MAudioAVCFG_t
swiaudio.h, 1633
unpack_swiaudio_SLQSSetM2MAudioLPBK
swiaudio.h, 1639
unpack_swiaudio_SLQSSetM2MAudioLPBK_t
swiaudio.h, 1633
unpack_swiaudio_SLQSSetM2MAudioNVDef
swiaudio.h, 1640
unpack_swiaudio_SLQSSetM2MAudioNVDef_t
swiaudio.h, 1633
unpack_swiaudio_SLQSSetM2MAudioProfile
swiaudio.h, 1640
unpack_swiaudio_SLQSSetM2MAudioProfile_t
swiaudio.h, 1633
unpack_swiaudio_SLQSSetM2MAudioVolume
swiaudio.h, 1640
unpack_swiaudio_SLQSSetM2MAudioVolume_t
swiaudio.h, 1633
unpack_swiaudio_SLQSSetM2MSpkrGain
swiaudio.h, 1641
unpack_swiaudio_SLQSSetM2MSpkrGain_t
swiaudio.h, 1633
unpack_swiavms_SLQSAVMSEventReportInd
swiavms.h, 1650
unpack_swiavms_SLQSAVMSEventReportInd_t, 1036
pConfig, 1037
pConnectionRequest, 1037
pDataSessionStatus, 1037
pNotification, 1037
pPackageID, 1037

pRegStatus, 1037
pSessionType, 1037
ParamPresenceMask, 1037
resultcode, 1037
unpack_swiavms_SLQSAVMSGetSettings
swiavms.h, 1650
unpack_swiavms_SLQSAVMSGetSettings_t, 1037
AutoConnect, 1039
AutoReboot, 1040
FwAutoSDM, 1040
FwPromptUpdate, 1040
FwPromptdownload, 1040
pAPNInfo, 1040
pConnectionRetryTimers, 1040
pNotificationStore, 1040
pPeroidsInfo, 1040
pPollingTimer, 1040
ParamPresenceMask, 1040
resultcode, 1040
unpack_swiavms_SLQSAVMSGetSettings_v2
swiavms.h, 1651
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 1040
AutoConnect, 1042
FwPromptUpdate, 1042
FwPromptdownload, 1042
pAutoReboot, 1042
pNotificationStore, 1042
pPeroidsInfo, 1042
pPollingTimer, 1042
ParamPresenceMask, 1042
resultcode, 1042
unpack_swiavms_SLQSAVMSSendSelection
swiavms.h, 1651
unpack_swiavms_SLQSAVMSSendSelection_t, 1042
ParamPresenceMask, 1043
resultcode, 1043
unpack_swiavms_SLQSAVMSSessionGetInfo
swiavms.h, 1652
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 1043
pConfig, 1043
pNotification, 1044
pPackageID, 1044
ParamPresenceMask, 1043
resultcode, 1044
unpack_swiavms_SLQSAVMSSetSettings
swiavms.h, 1653
unpack_swiavms_SLQSAVMSSetSettings_t, 1044
ParamPresenceMask, 1044
resultcode, 1044
unpack_swiavms_SLQSAVMSSetSettings_v2
swiavms.h, 1653
unpack_swiavms_SLQSAVMSSetSettings_v2_t, 1045
ParamPresenceMask, 1045
resultcode, 1045
unpack_swiavms_SLQSAVMSStartSession
swiavms.h, 1654
unpack_swiavms_SLQSAVMSStartSession_t, 1045
ParamPresenceMask, 1045
resultcode, 1045
sessionResponse, 1046
unpack_swiavms_SLQSAVMSStopSession
swiavms.h, 1654
unpack_swiavms_SLQSAVMSStopSession_avc2
swiavms.h, 1655
unpack_swiavms_SLQSAVMSStopSession_avc2_t
swiavms.h, 1644
unpack_swiavms_SLQSAVMSStopSession_t, 1046
ParamPresenceMask, 1046
resultcode, 1046
unpack_swiavms_SLQSAvmsSetEventReport
swiavms.h, 1652
unpack_swiavms_SLQSAvmsSetEventReport_t, 1044
ParamPresenceMask, 1044
resultcode, 1044
unpack_swidms_SLQSSwiDmsGetHWWatchdog
swidms.h, 1662
unpack_swidms_SLQSSwiDmsGetHWWatchdog_t, 1046
pHWWatchdog, 1046
ParamPresenceMask, 1046
unpack_swidms_SLQSSwiDmsGetMTU
swidms.h, 1662
unpack_swidms_SLQSSwiDmsGetMTU_t, 1047
pMTUSize3gpp, 1047
ParamPresenceMask, 1047
unpack_swidms_SLQSSwiDmsGetSecureInfo
swidms.h, 1663
unpack_swidms_SLQSSwiDmsGetSecureInfo_t, 1047
jtagAccessAllowed, 1048
memoryDumpAllowed, 1048
ParamPresenceMask, 1048
secureBootEnabled, 1048
TlvResult, 1048
unpack_swidms_SLQSSwiDmsGetUimSlot
swidms.h, 1663
unpack_swidms_SLQSSwiDmsGetUimSlot_t, 1048
autoUimSwitchMode, 1049
multiplexedSimMode, 1049
ParamPresenceMask, 1049
uimSlotSelected, 1049
unpack_swidms_SLQSSwiDmsGetUsbComp
swidms.h, 1664
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 1049
pInterfaceCfg, 1050
pSupportedBitmasks, 1050
ParamPresenceMask, 1050
unpack_swidms_SLQSSwiDmsGetUsbNetNum
swidms.h, 1664
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t, 1050
ParamPresenceMask, 1050
usbNetNum, 1050
unpack_swidms_SLQSSwiDmsSetHWWatchdog
swidms.h, 1664
unpack_swidms_SLQSSwiDmsSetHWWatchdog_t, 1050
ParamPresenceMask, 1051

Tlvresult, 1051
unpack_swidms_SLQSSwiDmsSetMTU
swidms.h, 1665
unpack_swidms_SLQSSwiDmsSetMTU_t, 1051
ParamPresenceMask, 1051
Tlvresult, 1051
unpack_swidms_SLQSSwiDmsSetUimSlot
swidms.h, 1665
unpack_swidms_SLQSSwiDmsSetUimSlot_t, 1051
ParamPresenceMask, 1052
Tlvresult, 1052
unpack_swidms_SLQSSwiDmsSetUsbComp
swidms.h, 1665
unpack_swidms_SLQSSwiDmsSetUsbComp_t, 1052
ParamPresenceMask, 1052
Tlvresult, 1052
unpack_swidms_SLQSSwiDmsSetUsbNetNum
swidms.h, 1666
unpack_swidms_SLQSSwiDmsSetUsbNetNum_t
swidms.h, 1658
unpack_swiloc_SwiLocGetAutoStart
swiloc.h, 1667
unpack_swiloc_SwiLocGetAutoStart_t, 1052
fix_rate, 1054
fix_rate_reported, 1054
fix_type, 1054
fix_type_reported, 1054
function, 1054
function_reported, 1054
max_dist, 1054
max_dist_reported, 1054
max_time, 1054
max_time_reported, 1054
ParamPresenceMask, 1054
unpack_swiloc_SwiLocSetAutoStart
swiloc.h, 1668
unpack_swiloc_SwiLocSetAutoStart_t
swiloc.h, 1666
unpack_swioma_SLQSOMADMAlertCallback
swioma.h, 1674
unpack_swioma_SLQSOMADMAlertCallback_ind
swioma.h, 1674
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 1054
eventType, 1055
unpack_swioma_SLQSOMADMAlertCallback_t
swioma.h, 1669
unpack_swioma_SLQSOMADMCancelSession
swioma.h, 1675
unpack_swioma_SLQSOMADMCancelSession_t
swioma.h, 1669
unpack_swioma_SLQSOMADMCancelSessionExt
swiomaext.h, 1682
unpack_swioma_SLQSOMADMCancelSessionExt_t
swiomaext.h, 1678
unpack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 1675
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1055
Date, 1058
DateLength, 1058
ParamPresenceMask, 1058
PkgDescLength, 1058
PkgDescription, 1058
PkgName, 1058
PkgNameLength, 1058
RetryCount, 1058
SessionState, 1058
SessionType, 1058
Severity, 1058
Source, 1058
SourceLength, 1058
Status, 1058
Time, 1058
TimeLength, 1058
UpdateCompleteStatus, 1058
unpack_swioma_SLQSOMADMGetSessionInfoExt
swiomaext.h, 1682
unpack_swioma_SLQSOMADMGetSessionInfoExt_t, 1058
fumoState, 1061
hfaStatus, 1061
pkgDate, 1061
pkgDesc, 1061
pkgName, 1061
pkgSize, 1061
sessionState, 1061
status, 1061
unpack_swioma_SLQSOMADMGetSettings
swioma.h, 1676
unpack_swioma_SLQSOMADMGetSettings_t, 1062
Autosdm, 1063
FOTAdownload, 1063
FwAutoCheck, 1063
ParamPresenceMask, 1063
unpack_swioma_SLQSOMADMSendSelection
swioma.h, 1676
unpack_swioma_SLQSOMADMSendSelection_t
swioma.h, 1669
unpack_swioma_SLQSOMADMSendSelectionExt
swiomaext.h, 1683
unpack_swioma_SLQSOMADMSendSelectionExt_t
swiomaext.h, 1679
unpack_swioma_SLQSOMADMSetSettings
swioma.h, 1677
unpack_swioma_SLQSOMADMSetSettings_t
swioma.h, 1669
unpack_swioma_SLQSOMADMSetSettingsExt
swiomaext.h, 1683
unpack_swioma_SLQSOMADMSetSettingsExt_t
swiomaext.h, 1679
unpack_swioma_SLQSOMADMStartSession
swioma.h, 1677
unpack_swioma_SLQSOMADMStartSession_t, 1063
FwAvailability, 1064
ParamPresenceMask, 1064
unpack_swioma_SLQSOMADMStartSessionExt

swiomaext.h, 1684
unpack_swioma_SLQSOMADMStartSessionExt_t
swiomaext.h, 1679
unpack_tmd_SLQSTmdDeRegNotMitigationLvl
tmd.h, 1688
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 1064
ParamPresenceMask, 1064
Tlvresult, 1064
unpack_tmd_SLQSTmdGetMitigationDevList
tmd.h, 1688
unpack_tmd_SLQSTmdGetMitigationDevList_t, 1064
MitigationDevList, 1065
MitigationDevListLen, 1065
ParamPresenceMask, 1065
Tlvresult, 1065
unpack_tmd_SLQSTmdGetMitigationLvl
tmd.h, 1689
unpack_tmd_SLQSTmdGetMitigationLvl_t, 1065
CurrentmitigationLvl, 1066
ParamPresenceMask, 1066
ReqMitigationLvl, 1066
Tlvresult, 1066
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind
tmd.h, 1689
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t, 1066
deviceID, 1066
deviceIdLen, 1066
lvl, 1066
ParamPresenceMask, 1066
unpack_tmd_SLQSTmdRegNotMitigationLvl
tmd.h, 1689
unpack_tmd_SLQSTmdRegNotMitigationLvl_t, 1066
ParamPresenceMask, 1067
Tlvresult, 1067
unpack_uim_ChangePin
uim.h, 1708
unpack_uim_ChangePin_t, 1067
pEncryptedPIN1, 1068
pIndicationToken, 1068
pRemainingRetries, 1068
ParamPresenceMask, 1068
Tlvresult, 1068
unpack_uim_GetCardStatus
uim.h, 1709
unpack_uim_GetCardStatus_t, 1068
pCardStatus, 1068
pHotSwapStatus, 1068
ParamPresenceMask, 1068
Tlvresult, 1068
unpack_uim_GetCardStatusV2
uim.h, 1709
unpack_uim_GetCardStatusV2_t, 1069
pCardStatus, 1069
pHotSwapStatus, 1069
pSimBusyStatus, 1069
pValidCardStatus, 1069
ParamPresenceMask, 1069
Tlvresult, 1069
unpack_uim_ReadTransparent
uim.h, 1709
unpack_uim_ReadTransparent_t, 1070
pCardResult, 1070
pEncryptedData, 1070
pIndicationToken, 1070
pReadResult, 1070
ParamPresenceMask, 1070
Tlvresult, 1070
unpack_uim_SLQSUIMAddProfile
uim.h, 1711
unpack_uim_SLQSUIMAddProfile_t
uim.h, 1696
unpack_uim_SLQSUIMAuthenticate
uim.h, 1711
unpack_uim_SLQSUIMAuthenticate_t, 1072
pAuthenticateResult, 1073
pCardResult, 1073
pIndicationToken, 1073
ParamPresenceMask, 1073
unpack_uim_SLQSUIMDeleteProfile
uim.h, 1711
unpack_uim_SLQSUIMDeleteProfile_t
uim.h, 1696
unpack_uim_SLQSUIMDepersonalization
uim.h, 1712
unpack_uim_SLQSUIMDepersonalization_t, 1073
pRemainingRetries, 1073
ParamPresenceMask, 1073
unpack_uim_SLQSUIMEventRegister
uim.h, 1712
unpack_uim_SLQSUIMEventRegister_t, 1073
eventMask, 1074
ParamPresenceMask, 1074
unpack_uim_SLQSUIMGetConfiguration
uim.h, 1713
unpack_uim_SLQSUIMGetConfiguration_t, 1074
pAutoSelection, 1075
pHaltSubscription, 1075
pPersonalizationStatus, 1075
ParamPresenceMask, 1075
unpack_uim_SLQSUIMGetEID
uim.h, 1713
unpack_uim_SLQSUIMGetEID_t, 1075
pEidInfo, 1075
ParamPresenceMask, 1075
unpack_uim_SLQSUIMGetFileAttributes
uim.h, 1713
unpack_uim_SLQSUIMGetFileAttributes_t, 1075
pCardResult, 1076
pFileAttributes, 1076
pIndicationToken, 1076
ParamPresenceMask, 1076
unpack_uim_SLQSUIMGetProfileInfo
uim.h, 1714
unpack_uim_SLQSUIMGetProfileInfo_t, 1076
pProfileClass, 1077

pProfileIcon, 1078
pProfileIconType, 1078
pProfileName, 1078
pProfileNickname, 1078
pProfilePolicyRules, 1078
pProfileSPN, 1078
pProfileState, 1078
ParamPresenceMask, 1077
unpack_uim_SLQSUIMGetSIMProfile
uim.h, 1714
unpack_uim_SLQSUIMGetSIMProfile_t, 1079
pProfileInfo, 1079
ParamPresenceMask, 1079
unpack_uim_SLQSUIMGetServiceStatus
uim.h, 1714
unpack_uim_SLQSUIMGetServiceStatus_t, 1078
pFDNStatus, 1079
pHiddenKeyStatus, 1079
pIndex, 1079
ParamPresenceMask, 1078
unpack_uim_SLQSUIMGetSlotsStatus
uim.h, 1715
unpack_uim_SLQSUIMGetSlotsStatus_t, 1079
pNumberOfPhySlot, 1080
pUimSlotsStatus, 1080
ParamPresenceMask, 1080
unpack_uim_SLQSUIMGetSlotsStatusV2
uim.h, 1715
unpack_uim_SLQSUIMGetSlotsStatusV2_t, 1080
ParamPresenceMask, 1081
unpack_uim_SLQSUIMPowerDown
uim.h, 1716
unpack_uim_SLQSUIMPowerDown_t
uim.h, 1696
unpack_uim_SLQSUIMPowerUp
uim.h, 1716
unpack_uim_SLQSUIMPowerUp_t
uim.h, 1696
unpack_uim_SLQSUIMReadRecord
uim.h, 1716
unpack_uim_SLQSUIMReadRecord_t, 1081
pAdditionalReadResult, 1082
pCardResult, 1082
pIndicationToken, 1082
pReadResult, 1082
ParamPresenceMask, 1082
unpack_uim_SLQSUIMRefreshCallback_Ind
uim.h, 1717
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 1082
ParamPresenceMask, 1082
refreshEvent, 1082
TlvPresent, 1082
unpack_uim_SLQSUIMRefreshComplete
uim.h, 1717
unpack_uim_SLQSUIMRefreshComplete_t
uim.h, 1696
unpack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 1717
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 1082
pRefreshEvent, 1083
ParamPresenceMask, 1083
unpack_uim_SLQSUIMRefreshOK
uim.h, 1718
unpack_uim_SLQSUIMRefreshOK_t
uim.h, 1696
unpack_uim_SLQSUIMRefreshRegister
uim.h, 1718
unpack_uim_SLQSUIMRefreshRegister_t
uim.h, 1696
unpack_uim_SLQSUIMReset
uim.h, 1718
unpack_uim_SLQSUIMReset_t
uim.h, 1696
unpack_uim_SLQSUIMSetSIMProfile
uim.h, 1719
unpack_uim_SLQSUIMSetSIMProfile_t
uim.h, 1696
unpack_uim_SLQSUIMSetServiceStatus
uim.h, 1719
unpack_uim_SLQSUIMSetServiceStatus_t
uim.h, 1696
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind
uim.h, 1719
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t, 1083
unpack_uim_SLQSUIMSwitchSlot
uim.h, 1720
unpack_uim_SLQSUIMSwitchSlot_t
uim.h, 1696
unpack_uim_SLQSUIMUpdateProfile
uim.h, 1720
unpack_uim_SLQSUIMUpdateProfile_t
uim.h, 1696
unpack_uim_SLQSUIMWriteRecord
uim.h, 1720
unpack_uim_SLQSUIMWriteRecord_t, 1083
pCardResult, 1084
pIndicationToken, 1084
ParamPresenceMask, 1084
unpack_uim_SLQSUIMWriteTransparent
uim.h, 1721
unpack_uim_SLQSUIMWriteTransparent_t, 1084
pCardResult, 1085
pIndicationToken, 1085
ParamPresenceMask, 1085
unpack_uim_SLQSUIMeUICCMemReset
uim.h, 1712
unpack_uim_SLQSUIMeUICCMemReset_t
uim.h, 1696
unpack_uim_SetPinProtection
uim.h, 1710
unpack_uim_SetPinProtection_t, 1071
pEncryptedPIN1, 1071
pIndicationToken, 1071
pRemainingRetries, 1071
ParamPresenceMask, 1071

Tlvresult, 1071
unpack_uim_SetUimSlotStatusChangeCallback_ind
uim.h, 1710
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 1071
bNumberOfPhySlots, 1072
ParamPresenceMask, 1072
slotsstatusChange, 1072
unpack_uim_UnblockPin
uim.h, 1721
unpack_uim_UnblockPin_t, 1085
pEncryptedPIN1, 1086
pIndicationToken, 1086
pRemainingRetries, 1086
ParamPresenceMask, 1085
Tlvresult, 1086
unpack_uim_UnblockPinV2
uim.h, 1721
unpack_uim_UnblockPinV2_t, 1086
pCardResult, 1086
pEncryptedPIN1, 1087
pIndicationToken, 1087
pRemainingRetries, 1087
ParamPresenceMask, 1086
Tlvresult, 1087
unpack_uim_VerifyPin
uim.h, 1722
unpack_uim_VerifyPin_t, 1087
pEncryptedPIN1, 1087
pIndicationToken, 1087
pRemainingRetries, 1087
ParamPresenceMask, 1087
Tlvresult, 1087
unpack_valid_nas_GetCDMANetworkParameters
nas.h, 1508
unpack_valid_nas_SLQSGetServingSystem
nas.h, 1509
unpack_valid_nas_SLQSGetSignalStrength
nas.h, 1510
unpack_valid_nas_SLQSNasGetSigInfo
nas.h, 1510
unpack_voice_AnswerUSSD
voice.h, 1741
unpack_voice_AnswerUSSD_t
voice.h, 1728
unpack_voice_CancelUSSD
voice.h, 1741
unpack_voice_CancelUSSD_t
voice.h, 1728
unpack_voice_DTMFEventCallback_ind
voice.h, 1741
unpack_voice_DTMFEventCallback_ind_t, 1090
DTMFInformation, 1090
pOffLength, 1090
pOnLength, 1090
ParamPresenceMask, 1090
unpack_voice_OTASPStatusCallback_ind
voice.h, 1742
unpack_voice_OTASPStatusCallback_ind_t, 1091
callID, 1092
OTASPStatus, 1092
ParamPresenceMask, 1092
unpack_voice_OriginateUSSD
voice.h, 1742
unpack_voice_OriginateUSSD_t
voice.h, 1728
unpack_voice_SLQSOriginateUSSD
voice.h, 1742
unpack_voice_SLQSOriginateUSSD_t, 1092
pAlphaIDInfo, 1094
pCCSuppsType, 1094
pCallId, 1094
pCcResultType, 1094
pUSSDInfo, 1094
ParamPresenceMask, 1094
pfailureCause, 1094
unpack_voice_SLQSVoiceALSSelectLine
voice.h, 1743
unpack_voice_SLQSVoiceALSSelectLine_t
voice.h, 1728
unpack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 1743
unpack_voice_SLQSVoiceALSSetLineSwitching_t
voice.h, 1728
unpack_voice_SLQSVoiceAnswerCall
voice.h, 1743
unpack_voice_SLQSVoiceAnswerCall_t, 1094
pCallId, 1095
ParamPresenceMask, 1095
unpack_voice_SLQSVoiceBindSubscription
voice.h, 1744
unpack_voice_SLQSVoiceBindSubscription_t
voice.h, 1728
unpack_voice_SLQSVoiceBurstDTMF
voice.h, 1744
unpack_voice_SLQSVoiceBurstDTMF_t, 1095
pCallID, 1095
ParamPresenceMask, 1095
unpack_voice_SLQSVoiceDialCall
voice.h, 1744
unpack_voice_SLQSVoiceDialCall_t, 1095
pAlphaIDInfo, 1096
pCCResultType, 1096
pCCSUPSType, 1096
pCallID, 1096
ParamPresenceMask, 1096
unpack_voice_SLQSVoiceEndCall
voice.h, 1745
unpack_voice_SLQSVoiceEndCall_t, 1096
pCallId, 1097
ParamPresenceMask, 1097
unpack_voice_SLQSVoiceGetAllCallInfo
voice.h, 1745
unpack_voice_SLQSVoiceGetAllCallInfo_t, 1097
pArrAlertingPattern, 1100
pArrAlertingType, 1100

pArrAlphaID, 1100
pArrCallEndReason, 1100
pArrCallInfo, 1100
pArrCalledPartyNum, 1100
pArrConnectPartyNum, 1100
pArrDiagInfo, 1100
pArrRedirPartyNum, 1100
pArrRemotePartyName, 1100
pArrRemotePartyNum, 1100
pArrSvcOption, 1100
pArrUUSInfo, 1100
pVoicePrivacy, 1100
ParamPresenceMask, 1100
unpack_voice_SLQSVoiceGetCLIP
voice.h, 1747
unpack_voice_SLQSVoiceGetCLIP_t, 1108
pAlphaIDInfo, 1109
pCCResType, 1109
pCallID, 1109
pFailCause, 1110
ParamPresenceMask, 1109
unpack_voice_SLQSVoiceGetCLIR
voice.h, 1747
unpack_voice_SLQSVoiceGetCLIR_t, 1110
pAlphaIDInfo, 1111
pCCResType, 1111
pCallID, 1111
pFailCause, 1111
ParamPresenceMask, 1111
unpack_voice_SLQSVoiceGetCNAP
voice.h, 1747
unpack_voice_SLQSVoiceGetCNAP_t, 1111
pAlphaIDInfo, 1112
pCCResType, 1112
pCallID, 1112
pFailCause, 1112
ParamPresenceMask, 1112
unpack_voice_SLQSVoiceGetCOLP
voice.h, 1748
unpack_voice_SLQSVoiceGetCOLP_t, 1113
pAlphaIDInfo, 1114
pCCResType, 1114
pCallID, 1114
pFailCause, 1114
ParamPresenceMask, 1114
unpack_voice_SLQSVoiceGetCOLR
voice.h, 1748
unpack_voice_SLQSVoiceGetCOLR_t, 1114
pAlphaIDInfo, 1115
pCCResType, 1115
pCallID, 1115
pFailCause, 1115
ParamPresenceMask, 1115
unpack_voice_SLQSVoiceGetCallBarring
voice.h, 1745
unpack_voice_SLQSVoiceGetCallBarring_t, 1101
pAlphaIDInfo, 1102
pCCResType, 1102
pCallID, 1102
pFailCause, 1102
pSvcClass, 1102
ParamPresenceMask, 1102
unpack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 1746
unpack_voice_SLQSVoiceGetCallForwardingStatus_t, 1102
pAlphaIDInfo, 1103
pCCResType, 1103
pCallID, 1103
pFailCause, 1103
ParamPresenceMask, 1103
unpack_voice_SLQSVoiceGetCallInfo
voice.h, 1746
unpack_voice_SLQSVoiceGetCallInfo_t, 1104
pAlertType, 1106
pAlertingPattern, 1106
pAlphaIDInfo, 1106
pCallInfo, 1106
pConnectNumInfo, 1106
pDiagInfo, 1106
pOTASPStatus, 1106
pRemotePartyName, 1106
pRemotePartyNum, 1106
pSrvOpt, 1106
pUUSInfo, 1107
pVoicePrivacy, 1107
ParamPresenceMask, 1106
unpack_voice_SLQSVoiceGetCallWaiting
voice.h, 1746
unpack_voice_SLQSVoiceGetCallWaiting_t, 1107
pAlphaIDInfo, 1108
pCCResType, 1108
pCallID, 1108
pFailCause, 1108
pSvcClass, 1108
ParamPresenceMask, 1108
unpack_voice_SLQSVoiceGetConfig
voice.h, 1748
unpack_voice_SLQSVoiceGetConfig_t, 1115
pAirTimerCnt, 1117
pAutoAnswerStat, 1117
pCurAMRConfig, 1117
pCurPrefVoiceSO, 1117
pCurVoiceDomainPref, 1117
pCurVoicePrivacyPref, 1117
pCurrTTYMode, 1117
pRoamTimerCnt, 1117
ParamPresenceMask, 1117
unpack_voice_SLQSVoiceIndicationRegister
voice.h, 1749
unpack_voice_SLQSVoiceIndicationRegister_t
voice.h, 1728
unpack_voice_SLQSVoiceManageCalls
voice.h, 1749
unpack_voice_SLQSVoiceManageCalls_t, 1117
pFailCause, 1118

ParamPresenceMask, 1118
unpack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 1749
unpack_voice_SLQSVoiceOrigUSSDNoWait_t
voice.h, 1728
unpack_voice_SLQSVoiceSUPSCallback_ind
voice.h, 1752
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 1124
pDataSrc, 1127
pFailCause, 1127
pReason, 1127
pSvcClass, 1127
ParamPresenceMask, 1126
unpack_voice_SLQSVoiceSendFlash
voice.h, 1750
unpack_voice_SLQSVoiceSendFlash_t, 1118
pCallID, 1118
ParamPresenceMask, 1118
unpack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 1750
unpack_voice_SLQSVoiceSetCallBarringPassword_t, 1119
pAlphaIDInfo, 1120
pCCResType, 1120
pCallID, 1120
pFailCause, 1120
ParamPresenceMask, 1120
unpack_voice_SLQSVoiceSetConfig
voice.h, 1750
unpack_voice_SLQSVoiceSetConfig_t, 1120
pAirTimerStatus, 1121
pAutoAnsStatus, 1121
pPrefVoiceSOStatus, 1121
pRoamTimerStatus, 1121
pTTYConfigStatus, 1121
pVoiceDomainPrefStatus, 1121
ParamPresenceMask, 1121
unpack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 1751
unpack_voice_SLQSVoiceSetPreferredPrivacy_t
voice.h, 1728
unpack_voice_SLQSVoiceSetSUPSService
voice.h, 1751
unpack_voice_SLQSVoiceSetSUPSService_t, 1122
pCallID, 1123
pFailCause, 1123
ParamPresenceMask, 1123
unpack_voice_SLQSVoiceStartContDTMF
voice.h, 1751
unpack_voice_SLQSVoiceStartContDTMF_t, 1123
pCallID, 1123
ParamPresenceMask, 1123
unpack_voice_SLQSVoiceStopContDTMF
voice.h, 1752
unpack_voice_SLQSVoiceStopContDTMF_t, 1123
callID, 1124
ParamPresenceMask, 1124
unpack_voice_SUPSNotificationCallback_ind
voice.h, 1752
unpack_voice_SUPSNotificationCallback_ind_t, 1127
callID, 1129
notifType, 1129
pCUGIndex, 1129
pECTNum, 1129
ParamPresenceMask, 1129
unpack_voice_USSDNoWaitIndicationCallback_ind
voice.h, 1753
unpack_voice_USSDNoWaitIndicationCallback_ind_t, 1129
AIDTlv, 1130
FCTlv, 1130
NwETlv, 1130
ParamPresenceMask, 1130
UInfoTLV, 1130
unpack_voice_USSDNotificationCallback_ind
voice.h, 1753
unpack_voice_USSDNotificationCallback_ind_t, 1129
notification_Type, 1129
ParamPresenceMask, 1129
unpack_voice_VoiceInfoRecCallback_ind
voice.h, 1753
unpack_voice_VoiceInfoRecCallback_ind_t, 1130
callID, 1132
pCLIRCause, 1133
pCallWaitInd, 1133
pCalledPartyInfo, 1133
pCallerIDInfo, 1133
pCallerNameInfo, 1133
pCallingPartyInfo, 1133
pConnectNumInfo, 1133
pDispInfo, 1133
pExtDispInfo, 1133
pExtDispRecInfo, 1133
pLineCtrlInfo, 1133
pNSSAudioCtrl, 1133
pNSSRelease, 1133
pRedirNumInfo, 1133
pSignalInfo, 1133
ParamPresenceMask, 1132
unpack_voice_allCallStatusCallback_ind
voice.h, 1740
unpack_voice_allCallStatusCallback_ind_t, 1088
arrCallInfomation, 1089
pArrAlertingPattern, 1089
pArrAlertingType, 1089
pArrAlphaID, 1089
pArrCallEndReason, 1089
pArrCalledPartyNum, 1089
pArrConnectPartyNum, 1089
pArrDiagInfo, 1089
pArrRedirPartyNum, 1089
pArrRemotePartyName, 1089
pArrRemotePartyNum, 1089
pArrSvcOption, 1089
ParamPresenceMask, 1089
unpack_voice_voicePrivacyChangeCallback_ind

voice.h, 1754
unpack_voice_voicePrivacyChangeCallback_ind_t, 1133
callID, 1134
ParamPresenceMask, 1134
voicePrivacy, 1134
unpack_wda_GetDataFormat
wda.h, 1756
unpack_wda_GetDataFormat_t, 1134
Dl_data_aggregation_max_datagrams, 1136
Dl_data_aggregation_max_size, 1136
Dl_data_aggregation_protocol, 1136
Dl_minimum_padding, 1136
Flow_control, 1136
Ndp_signature, 1136
ParamPresenceMask, 1136
Qlink_prot, 1136
Qos_format, 1136
Qos_header_format, 1136
Tcp_coalescing, 1136
Tlvresult, 1136
Udp_coalescing, 1136
Ul_data_aggregation_max_datagrams, 1136
Ul_data_aggregation_max_size, 1136
Ul_data_aggregation_protocol, 1136
unpack_wda_GetQmapSettings
wda.h, 1757
unpack_wda_GetQmapSettings_t, 1136
pQmap_dfc, 1137
ParamPresenceMask, 1137
Tlvresult, 1137
unpack_wda_SetDataFormat
wda.h, 1757
unpack_wda_SetDataFormat_t, 1137
pDl_data_aggregation_max_datagrams, 1140
pDl_data_aggregation_max_size, 1140
pDl_data_aggregation_protocol, 1140
pDl_minimum_padding, 1140
pFlow_control, 1140
pNdp_signature, 1140
pQlink_prot, 1141
pQos_format, 1141
pQos_header_format, 1141
pTcp_coalescing, 1141
pUdp_coalescing, 1141
pUl_data_aggregation_max_datagrams, 1141
pUl_data_aggregation_max_size, 1141
pUl_data_aggregation_protocol, 1141
ParamPresenceMask, 1140
Tlvresult, 1141
unpack_wda_SetQmapSettings
wda.h, 1757
unpack_wda_SetQmapSettings_t, 1141
pQmap_dfc, 1141
ParamPresenceMask, 1141
Tlvresult, 1141
unpack_wds_DHCPv4ClientLease_ind
wds.h, 1789
unpack_wds_DHCPv4ClientLease_ind_t, 1142
DHCPv4LeaseOptTlv, 1142
DHCPv4LeaseStateTlv, 1142
IPv4AddrTlv, 1142
ParamPresenceMask, 1142
ProfileIdTlv, 1142
unpack_wds_DHCPv4ClientLeaseChange
wds.h, 1790
unpack_wds_DHCPv4ClientLeaseChange_t
wds.h, 1767
unpack_wds_GetAutoconnect
wds.h, 1790
unpack_wds_GetAutoconnect_t, 1142
ParamPresenceMask, 1143
psetting, 1143
unpack_wds_GetByteTotals
wds.h, 1791
unpack_wds_GetByteTotals_t, 1143
pRXTotalBytes, 1143
pTXTotalBytes, 1144
ParamPresenceMask, 1143
unpack_wds_GetConnectionRate
wds.h, 1791
unpack_wds_GetConnectionRate_t, 1144
currentChannelRXRate, 1144
currentChannelTXRate, 1144
maxChannelRXRate, 1144
maxChannelTXRate, 1144
ParamPresenceMask, 1144
unpack_wds_GetDataBearerTechnology
wds.h, 1791
unpack_wds_GetDataBearerTechnology_t, 1145
pDataBearer, 1145
ParamPresenceMask, 1145
unpack_wds_GetDefaultProfile
wds.h, 1792
unpack_wds_GetDefaultProfile_t, 1145
apnname, 1147
apnsize, 1147
auth, 1148
ipaddr, 1148
ipaddrv6, 1148
name, 1148
namesize, 1148
ParamPresenceMask, 1148
pdptype, 1148
pridns, 1148
pridnsv6, 1148
secdns, 1148
secdnsv6, 1148
username, 1148
usersize, 1148
unpack_wds_GetDefaultProfileNum
wds.h, 1792
unpack_wds_GetDefaultProfileNum_t, 1148
index, 1148
ParamPresenceMask, 1148
unpack_wds_GetDefaultProfileV2

wds.h, 1792
unpack_wds_GetDefaultProfileV2_t, 1149
apnname, 1151
apnsize, 1151
auth, 1151
ipaddr, 1151
ipaddrv6, 1151
name, 1151
namesize, 1151
ParamPresenceMask, 1151
pdptype, 1151
pridns, 1151
pridnsv6, 1151
pwd, 1151
pwdsize, 1151
secdns, 1151
secdnsv6, 1151
username, 1151
usersize, 1151
unpack_wds_GetDormancyState
wds.h, 1793
unpack_wds_GetDormancyState_t, 1151
dormancyState, 1152
ParamPresenceMask, 1152
unpack_wds_GetLastMobileIPError
wds.h, 1793
unpack_wds_GetLastMobileIPError_t, 1152
error, 1152
ParamPresenceMask, 1152
unpack_wds_GetMobileIP
wds.h, 1793
unpack_wds_GetMobileIP_t, 1153
mipMode, 1153
ParamPresenceMask, 1153
unpack_wds_GetMobileIPProfile
wds.h, 1794
unpack_wds_GetMobileIPProfile_t, 1153
AAASPI, 1155
AAAState, 1155
address, 1155
enabled, 1155
HASPI, 1155
HAState, 1155
NAI, 1155
naiSize, 1155
ParamPresenceMask, 1155
primaryHA, 1155
revTunneling, 1155
secondaryHA, 1155
unpack_wds_GetPacketStatistics
wds.h, 1794
unpack_wds_GetPacketStatistics_t, 1155
pRXDroppedCount, 1157
pRXOKBytesLastCall, 1157
pRXOkBytesCount, 1157
pRXPacketErrors, 1157
pRXPacketOverflows, 1157
pRXPacketSuccesses, 1157
pTXDroppedCount, 1157
pTXOKBytesLastCall, 1157
pTXOkBytesCount, 1157
pTXPacketErrors, 1157
pTXPacketOverflows, 1157
pTXPacketSuccesses, 1157
ParamPresenceMask, 1157
unpack_wds_GetPacketStatus
wds.h, 1794
unpack_wds_GetPacketStatus_t, 1157
ParamPresenceMask, 1159
rXDroppedCount, 1159
rXOKBytesLastCall, 1159
rXOkBytesCount, 1159
rXPacketErrors, 1159
rXPacketOverflows, 1159
rXPacketSuccesses, 1159
tXDroppedCount, 1159
tXOKBytesLastCall, 1159
tXOkBytesCount, 1159
tXPacketErrors, 1159
tXPacketOverflows, 1159
tXPacketSuccesses, 1159
unpack_wds_GetSessionDuration
wds.h, 1795
unpack_wds_GetSessionDuration_t, 1159
callDuration, 1160
ParamPresenceMask, 1160
unpack_wds_GetSessionDurationV2
wds.h, 1795
unpack_wds_GetSessionDurationV2_t, 1160
callDuration, 1161
pCallActiveDuration, 1161
pLastCallActiveDuration, 1161
pLastCallDuration, 1161
ParamPresenceMask, 1161
unpack_wds_GetSessionState
wds.h, 1795
unpack_wds_GetSessionState_t, 1161
connectionStatus, 1161
ParamPresenceMask, 1161
unpack_wds_RMSetTransferStatistics
wds.h, 1796
unpack_wds_RMSetTransferStatistics_t, 1161
ParamPresenceMask, 1162
unpack_wds_RMTransferStatistics_ind
wds.h, 1796
unpack_wds_RMTransferStatistics_ind_t
wds.h, 1767
unpack_wds_SLQSCreateProfile
wds.h, 1799
unpack_wds_SLQSCreateProfile_t, 1162
pCreateProfileOut, 1163
pProfileID, 1163
ParamPresenceMask, 1163
Tlvresult, 1163
unpack_wds_SLQSDUNCallInfoCallBack_ind
wds.h, 1799

unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 1163
unpack_wds_SLQSDeleteProfile
wds.h, 1799
unpack_wds_SLQSDeleteProfile_t, 1163
extendedErrorCode, 1163
ParamPresenceMask, 1163
unpack_wds_SLQSGet3GPPConfigItem
wds.h, 1800
unpack_wds_SLQSGet3GPPConfigItem_t, 1165
_3gppRelease, 1166
defaultPDNEnabled, 1166
LTEAttachProfile, 1167
ParamPresenceMask, 1167
profileList, 1167
unpack_wds_SLQSGetCurrDataSystemStat
wds.h, 1800
unpack_wds_SLQSGetCurrDataSystemStat_t, 1167
currNetworkInfo, 1167
networkInfoLen, 1167
ParamPresenceMask, 1168
prefNetwork, 1168
unpack_wds_SLQSGetCurrentChannelRate
wds.h, 1800
unpack_wds_SLQSGetCurrentChannelRate_t, 1168
max_channel_rx_rate, 1169
max_channel_tx_rate, 1169
ParamPresenceMask, 1169
unpack_wds_SLQSGetDUNCallInfo
wds.h, 1801
unpack_wds_SLQSGetDUNCallInfo_t, 1170
callEndReason, 1172
channelRate, 1172
connectionStatus, 1172
dataBearerTech, 1172
dormancyStatus, 1172
lastCallDataBearerTech, 1172
mdmCallDurationActive, 1173
ParamPresenceMask, 1173
rxOKBytesCount, 1173
txOKBytesCount, 1173
unpack_wds_SLQSGetDataBearerTechnology
wds.h, 1801
unpack_wds_SLQSGetDataBearerTechnology_t, 1169
curDataBearerTechnology, 1169
dataBearerMask, 1169
lastCallDataBearerTechnology, 1170
ParamPresenceMask, 1170
unpack_wds_SLQSGetProfileSettings
wds.h, 1801
unpack_wds_SLQSGetProfileSettings_t, 1173
pProfileSettings, 1173
ParamPresenceMask, 1173
ProfileType, 1173
Tlvresult, 1173
unpack_wds_SLQSGetProfileSettingsV2
wds.h, 1802
unpack_wds_SLQSGetProfileSettingsV2_t, 1173
pProfileSettings, 1174
ParamPresenceMask, 1174
ProfileType, 1174
Tlvresult, 1174
unpack_wds_SLQSGetRuntimeSettings
wds.h, 1802
unpack_wds_SLQSGetRuntimeSettings_t, 1174
APNName, 1176
Authentication, 1176
DomainList, 1176
GPRSGrantedQoS, 1176
GWAddressV4, 1176
IMCNflag, 1177
IPFamilyPreference, 1177
IPV6AddrInfo, 1177
IPV6GWAddrInfo, 1177
IPv4, 1177
Mtu, 1177
PDPType, 1177
ParamPresenceMask, 1177
PrimaryDNSV4, 1177
PrimaryDNSV6, 1177
ProfileID, 1177
ProfileName, 1177
SecondaryDNSV4, 1177
SecondaryDNSV6, 1177
ServerAddrList, 1177
SubnetMaskV4, 1177
Technology, 1177
UMTSGrantedQoS, 1177
Username, 1177
unpack_wds_SLQSModifyProfile
wds.h, 1802
unpack_wds_SLQSModifyProfile_t, 1177
pExtErrorCode, 1178
ParamPresenceMask, 1178
unpack_wds_SLQSResetPacketStatics
wds.h, 1803
unpack_wds_SLQSResetPacketStatics_t
wds.h, 1767
unpack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 1805
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 1186
pHwConfig, 1187
ParamPresenceMask, 1187
unpack_wds_SLQSSGetLoopback
wds.h, 1805
unpack_wds_SLQSSGetLoopback_t, 1187
ByteLoopbackMode, 1188
ByteLoopbackMultiplier, 1188
ParamPresenceMask, 1188
unpack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 1805
unpack_wds_SLQSSSetDHCPv4ClientConfig_t
wds.h, 1767
unpack_wds_SLQSSSetLoopback
wds.h, 1806
unpack_wds_SLQSSSetLoopback_t
wds.h, 1767

unpack_wds_SLQSSet3GPPConfigItem
wds.h, 1803
unpack_wds_SLQSSet3GPPConfigItem_t
wds.h, 1767
unpack_wds_SLQSSetIPFamilyPreference
wds.h, 1803
unpack_wds_SLQSSetIPFamilyPreference_t, 1178
ParamPresenceMask, 1178
Tlvresult, 1178
unpack_wds_SLQSSetPacketSrvStatusCallback
wds.h, 1804
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 1178
bearerID, 1180
conn_status, 1180
ipFamily, 1180
ParamPresenceMask, 1180
reconfigReqd, 1180
sessionEndReason, 1180
techName, 1180
verboseSessnEndReason, 1180
verboseSessnEndReasonType, 1180
unpack_wds_SLQSSetWdsEventCallback
wds.h, 1804
unpack_wds_SLQSSetWdsEventCallback_ind
wds.h, 1804
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1180
currDBTechAvail, 1185
currNWInfo, 1185
dBTechAvail, 1185
dBTechnology, 1186
dBtechExtAvail, 1185
dBtechnologyExt, 1186
dataSysStatAvail, 1185
dormancyStatAvail, 1186
dormancyStatus, 1186
mipStatus, 1186
mipstatAvail, 1186
netInfoLen, 1186
ParamPresenceMask, 1186
prefNetwork, 1186
ratMask, 1186
rx_bytes, 1186
rx_pkts, 1186
soMask, 1186
tx_bytes, 1186
tx_pkts, 1186
xferStatAvail, 1186
unpack_wds_SLQSSetWdsEventCallback_t
wds.h, 1767
unpack_wds_SLQSStartDataSession
wds.h, 1806
unpack_wds_SLQSStartDataSession_t, 1188
pFailureReason, 1188
pVerboseFailReasonType, 1189
pVerboseFailureReason, 1189
ParamPresenceMask, 1188
psid, 1188
unpack_wds_SLQSStopDataSession
wds.h, 1806
unpack_wds_SLQSStopDataSession_t
wds.h, 1767
unpack_wds_SLQSSwiProfileChangeCallback
wds.h, 1807
unpack_wds_SLQSSwiProfileChangeCallback_Ind
wds.h, 1807
unpack_wds_SLQSSwiProfileChangeCallback_Ind_t, 1189
ParamPresenceMask, 1189
ProfileTlv, 1189
srcTlv, 1189
unpack_wds_SLQSSwiProfileChangeCallback_t
wds.h, 1767
unpack_wds_SLQSWdsGoActive
wds.h, 1807
unpack_wds_SLQSWdsGoActive_t
wds.h, 1767
unpack_wds_SLQSWdsGoDormant
wds.h, 1808
unpack_wds_SLQSWdsGoDormant_t
wds.h, 1767
unpack_wds_SLQSWdsSetEventReport
wds.h, 1808
unpack_wds_SLQSWdsSetEventReport_t
wds.h, 1767
unpack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 1808
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 1189
apnName, 1191
bearerId, 1191
contextId, 1191
ipv4Address, 1191
ipv4GWAddress, 1191
ipv6Address, 1191
ipv6GWAddress, 1191
ParamPresenceMask, 1191
unpack_wds_SetAutoconnect
wds.h, 1796
unpack_wds_SetAutoconnect_t
wds.h, 1767
unpack_wds_SetDefaultProfile
wds.h, 1797
unpack_wds_SetDefaultProfile_t
wds.h, 1767
unpack_wds_SetDefaultProfileNum
wds.h, 1797
unpack_wds_SetDefaultProfileNum_t
wds.h, 1767
unpack_wds_SetMobileIP
wds.h, 1797
unpack_wds_SetMobileIP_t
wds.h, 1767
unpack_wds_SetMobileIPParameters
wds.h, 1798
unpack_wds_SetMobileIPParameters_t

wds.h, 1767
unpack_wds_SetMobileIPProfile
wds.h, 1798
unpack_wds_SetMobileIPProfile_t, 1162
ParamPresenceMask, 1162
unpack_wds_SetMuxID
wds.h, 1798
unpack_wds_SetMuxID_t
wds.h, 1767
UnpackQmiProfileInfo
wds.h, 1767
UnpackQmiProfileInfoV2
wds.h, 1768
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 1193
bBinaryType, 1194
bSerity, 1194
bState, 1194
bUserInputRequest, 1194
szDescription, 1194
szName, 1194
szVersion, 1195
TlvPresent, 1195
ulPkgDownloadComplete, 1195
ulPkgDownloadSize, 1195
wDescriptionLength, 1195
wNameLength, 1195
wUpdateCompeteStatus, 1195
wUserInputTimeout, 1195
wVersionLength, 1195
UnpackSwiAvmsEventReportConfig, 1195
bState, 1196
bUserInputRequest, 1196
szAlertMsg, 1196
TlvPresent, 1196
wAlertMsgLength, 1196
wUserInputTimeout, 1196
UnpackSwiAvmsEventReportConnectionRequest, 1196
bUserInputRequest, 1196
TlvPresent, 1196
wUserInputTimeout, 1196
UnpackSwiAvmsEventReportDataSessionStatus, 1196
bType, 1197
TlvPresent, 1197
wErrorCode, 1197
UnpackSwiAvmsEventReportHTTPStatus, 1197
TlvPresent, 1198
wHTTPStatus, 1198
UnpackSwiAvmsEventReportNotification, 1199
bNotification, 1199
TlvPresent, 1200
wSessionStatus, 1200
UnpackSwiAvmsEventReportPackageID, 1200
bPackageID, 1200
TlvPresent, 1200
UnpackSwiAvmsEventReportRegStatus, 1200
bRegStatus, 1200
TlvPresent, 1200
UnpackSwiAvmsEventReportSessionType, 1201
bType, 1201
TlvPresent, 1201
UnpackSwiAvmsEventReportWAMSParamChange, 1201
TlvPresent, 1202
wWamsChangeMask, 1202
unpackWdsProfileParam, 1202
SlqsProfile3GPP, 1202
SlqsProfile3GPP2, 1202
unpackWdsProfileParamV2, 1202
SlqsProfile3GPP, 1203
SlqsProfile3GPP2, 1203
upLink
voice_NSSAudioCtrl, 1239
UpdateCompleteStatus
unpack_swioma_SLQSOMADMGetSessionInfo_t, 1058
updateCompleteStatus
unpack_omaDmFotaTlv_t, 973
upinRetries
slotInf, 657
uim_slotInfo, 722
upinState
slotInf, 657
uim_slotInfo, 722
upukRetries
slotInf, 657
uim_slotInfo, 722
urlAddr
loc_URLAddrInfo, 183
usageMask
loc_sensorDataUsage, 180
UsageSetting
nas_UsageSettingTlv, 377
UsbMTUSize
swidms_usbMTUSizeTlv, 684
usbNetNum
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t, 1050
useMbim
QmuxTransport, 649
useQcqmi
QcqmiService, 642
userData
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 1031
userInputReq
unpack_omaDmConfigTlv_t, 971
unpack_omaDmFotaTlv_t, 973
userInputTimeout
unpack_omaDmConfigTlv_t, 971
unpack_omaDmFotaTlv_t, 973
Username
unpack_wds_SLQSGetRuntimeSettings_t, 1177
username
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151

usersize
unpack_wds_GetDefaultProfile_t, 1148
unpack_wds_GetDefaultProfileV2_t, 1151
ussDCS
pack_voice_SLQSOriginateUSSD_t, 587
voice_USSInfo, 1249
ussData
pack_voice_SLQSOriginateUSSD_t, 587
voice_USSInfo, 1249
ussLen
pack_voice_SLQSOriginateUSSD_t, 587
voice_USSInfo, 1249
UtRat
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
UtRatVal
imsa_UtRatInfo, 115
UtService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
UtServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
UtServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
UtSvcStatus
imsa_UtSvcStatusInfo, 115
uusInfo
voice_allCallsUUSInfo, 1205

VDOP
loc_precisionDilution, 177
val
unpack_qos_IPv6TrafCls_t, 990
unpack_qos_Tos_t, 1009
val_3GPP2Pri
unpack_qos_swiQosFlow_t, 1008
val_3GPPImCn
unpack_qos_swiQosFlow_t, 1008
val_3GPPResResidualBER
unpack_qos_swiQosFlow_t, 1008
val_3GPPSigInd
unpack_qos_swiQosFlow_t, 1008
val_3GPPTraHdlPri
unpack_qos_swiQosFlow_t, 1008
ValidBitmasks
swidms_supportedIntBitmaskTlv, 682
validCard
uim_validCardStatus, 727
validCardLength
uim_validCardStatus, 727
ValidMask
unpack_pds_SLQSGetGPSStateInfo_t, 987
validMask
loc_satelliteInfo, 180
ValidityCW0
unpack_nas_SLQSSwiGetLteCQI_t, 963
ValidityCW1
unpack_nas_SLQSSwiGetLteCQI_t, 963
Value
pack_swiaudio_SLQSSetM2MSpkrGain_t, 546
unpack_swiaudio_SLQSGetM2MSpkrGain_t, 1036
value
pack_dms_UIMSetPINProtection_t, 432
pack_dms_UIMVerifyPIN_t, 433
value_length
DMScustSettingInfo, 79
pack_dms_SetCustFeaturesV2_t, 421
verValid
loc_AppProviderInfoTlv, 164
verboseSessnEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 1180
verboseSessnEndReasonType
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 1180
verifyLeft
uim_personalizationStatus, 706
uim_remainingRetries, 718
verifyPIN
pack_uim_VerifyPin_t, 586
verifyRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 806
unpack_dms_UIMSetControlKeyProtection_t, 809
unpack_dms_UIMSetPINProtection_t, 810
version
loc_AppProviderInfoTlv, 164
unpack_omaDmFotaTlv_t, 973
unpack_qos_swiQosFilter_t, 1004
versionLen
loc_AppProviderInfoTlv, 164
versionStr
_litefw_FirmwareFileInfo, 40
versionString
pack_dms_SLQSSwiSetOSInfo_t, 427
unpack_dms_SLQSSwiGetOSInfo_t, 800
versionlength
unpack_omaDmFotaTlv_t, 973
vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 455
vertReliability
pack_loc_SLQSLOCInjectPosition_t, 455
vertUnc
pack_loc_SLQSLOCInjectPosition_t, 455
VerticalUncertainty
unpack_pds_SLQSGetGPSStateInfo_t, 987
VirtStream
nas_protocolSubtypeElement, 322
voice.h
LITE_VOICE_SUPS_SRV_CLASS_DATA, 1728
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC, 1728
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC, 1728
LITE_VOICE_SUPS_SRV_CLASS_FAX, 1728
LITE_VOICE_SUPS_SRV_CLASS_NONE, 1728
LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS, 1728

LITE_VOICE_SUPS_SRV_CLASS_PADACCESS, 1729
LITE_VOICE_SUPS_SRV_CLASS_SMS, 1728
LITE_VOICE_SUPS_SRV_CLASS_VOICE, 1728
voice.h, 1722
liteServiceClassInformation, 1728
MAXVOICEUSSDLENGTH, 1728
pack_voice_AnswerUSSD, 1729
pack_voice_CancelUSSD, 1729
pack_voice_OriginateUSSD, 1729
pack_voice_SLQSOriginateUSSD, 1730
pack_voice_SLQSVoiceALSSelectLine, 1730
pack_voice_SLQSVoiceALSSetLineSwitching, 1730
pack_voice_SLQSVoiceAnswerCall, 1731
pack_voice_SLQSVoiceBindSubscription, 1731
pack_voice_SLQSVoiceBurstDTMF, 1731
pack_voice_SLQSVoiceDialCall, 1732
pack_voice_SLQSVoiceEndCall, 1732
pack_voice_SLQSVoiceGetAllCallInfo, 1733
pack_voice_SLQSVoiceGetCLIP, 1734
pack_voice_SLQSVoiceGetCLIR, 1735
pack_voice_SLQSVoiceGetCNAP, 1735
pack_voice_SLQSVoiceGetCOLP, 1735
pack_voice_SLQSVoiceGetCOLR, 1736
pack_voice_SLQSVoiceGetCallBarring, 1733
pack_voice_SLQSVoiceGetCallForwardingStatus, 1733
pack_voice_SLQSVoiceGetCallInfo, 1734
pack_voice_SLQSVoiceGetCallWaiting, 1734
pack_voice_SLQSVoiceGetConfig, 1736
pack_voice_SLQSVoiceIndicationRegister, 1736
pack_voice_SLQSVoiceManageCalls, 1737
pack_voice_SLQSVoiceOrigUSSDNoWait, 1737
pack_voice_SLQSVoiceSendFlash, 1738
pack_voice_SLQSVoiceSetCallBarringPassword, 1738
pack_voice_SLQSVoiceSetConfig, 1738
pack_voice_SLQSVoiceSetPreferredPrivacy, 1739
pack_voice_SLQSVoiceSetSUPSService, 1739
pack_voice_SLQSVoiceStartContDTMF, 1739
pack_voice_SLQSVoiceStopContDTMF, 1740
unpack_voice_AnswerUSSD, 1741
unpack_voice_AnswerUSSD_t, 1728
unpack_voice_CancelUSSD, 1741
unpack_voice_CancelUSSD_t, 1728
unpack_voice_DTMFEventCallback_ind, 1741
unpack_voice_OTASPStatusCallback_ind, 1742
unpack_voice_OriginateUSSD, 1742
unpack_voice_OriginateUSSD_t, 1728
unpack_voice_SLQSOriginateUSSD, 1742
unpack_voice_SLQSVoiceALSSelectLine, 1743
unpack_voice_SLQSVoiceALSSelectLine_t, 1728
unpack_voice_SLQSVoiceALSSetLineSwitching, 1743
unpack_voice_SLQSVoiceALSSetLineSwitching_t, 1728
unpack_voice_SLQSVoiceAnswerCall, 1743
unpack_voice_SLQSVoiceBindSubscription, 1744
unpack_voice_SLQSVoiceBindSubscription_t, 1728
unpack_voice_SLQSVoiceBurstDTMF, 1744
unpack_voice_SLQSVoiceDialCall, 1744
unpack_voice_SLQSVoiceEndCall, 1745
unpack_voice_SLQSVoiceGetAllCallInfo, 1745
unpack_voice_SLQSVoiceGetCLIP, 1747
unpack_voice_SLQSVoiceGetCLIR, 1747
unpack_voice_SLQSVoiceGetCNAP, 1747
unpack_voice_SLQSVoiceGetCOLP, 1748
unpack_voice_SLQSVoiceGetCOLR, 1748
unpack_voice_SLQSVoiceGetCallBarring, 1745
unpack_voice_SLQSVoiceGetCallForwarding-Status, 1746
unpack_voice_SLQSVoiceGetCallInfo, 1746
unpack_voice_SLQSVoiceGetCallWaiting, 1746
unpack_voice_SLQSVoiceGetConfig, 1748
unpack_voice_SLQSVoiceIndicationRegister, 1749
unpack_voice_SLQSVoiceIndicationRegister_t, 1728
unpack_voice_SLQSVoiceManageCalls, 1749
unpack_voice_SLQSVoiceOrigUSSDNoWait, 1749
unpack_voice_SLQSVoiceOrigUSSDNoWait_t, 1728
unpack_voice_SLQSVoiceSUPSCallback_ind, 1752
unpack_voice_SLQSVoiceSendFlash, 1750
unpack_voice_SLQSVoiceSetCallBarringPassword, 1750
unpack_voice_SLQSVoiceSetConfig, 1750
unpack_voice_SLQSVoiceSetPreferredPrivacy, 1751
unpack_voice_SLQSVoiceSetPreferredPrivacy_t, 1728
unpack_voice_SLQSVoiceSetSUPSService, 1751
unpack_voice_SLQSVoiceStartContDTMF, 1751
unpack_voice_SLQSVoiceStopContDTMF, 1752
unpack_voice_SUPSNotificationCallback_ind, 1752
unpack_voice_USSDNoWaitIndicationCallback_-ind, 1753
unpack_voice_USSDNotificationCallback_ind, 1753
unpack_voice_VoiceInfoRecCallback_ind, 1753
unpack_voice_allCallStatusCallback_ind, 1740
unpack_voice_voicePrivacyChangeCallback_ind, 1754
voice_CLIPResp, 1225
ActiveStatus, 1225
ProvisionStatus, 1225
voice_CLIRResp, 1225
ActiveStatus, 1226
ProvisionStatus, 1226
voice_CNAPResp, 1226
ActiveStatus, 1227
ProvisionStatus, 1227
voice_COLPResp, 1227

ActiveStatus, 1227
ProvisionStatus, 1227
voice_COLRResp, 1227
ActiveStatus, 1228
ProvisionStatus, 1228
voice_CUGInfo, 1230
CUGIndex, 1230
SuppOA, 1230
SuppPrefCUG, 1230
voice_DTMFInfo, 1232
callID, 1232
DTMFEvent, 1232
digitBuff, 1232
digitCnt, 1232
voice_DTMFLengths, 1232
DTMFInterdigitInterval, 1233
DTMFPulseWidth, 1233
voice_ECTNum, 1233
ECTCallState, 1234
number, 1234
presentationInd, 1234
voice_NSSAudioCtrl, 1239
downLink, 1239
upLink, 1239
voice_SUPSInfo, 1248
isModByCC, 1248
svcType, 1248
voice_USSDNotificationNetworkInfo, 1248
networkInfo, 1249
tlvPresent, 1249
voice_USSInfo, 1249
ussDCS, 1249
ussData, 1249
ussLen, 1249
voice_UUSInfo, 1249
UUSData, 1250
UUSDatalen, 1250
UUSDcs, 1250
UUSType, 1250
voice_VoiceUssdNoWaitAlphaIDTlv, 1251
Alpha_Identifier, 1251
TlvPresent, 1251
voice_VoiceUssdNoWaitErrorTlv, 1251
Error, 1251
TlvPresent, 1252
voice_VoiceUssdNoWaitFailCauseTlv, 1252
Failure_Cause, 1252
TlvPresent, 1252
voice_VoiceUssdNoWaitUSSINFOTlv, 1252
NetworkInfo, 1253
TlvPresent, 1253
voice_airTimer, 1203
airTimerValue, 1203
namID, 1203
voice_allCallsAlphaIDInfo, 1203
AlphaIDInfo, 1204
callID, 1204
voice_allCallsDiagInfo, 1204
callID, 1204
DiagInfo, 1204
voice_allCallsUUSInfo, 1204
callID, 1205
uusInfo, 1205
voice_alphaIDInfo, 1205
alphaDcs, 1205
alphaLen, 1205
alphaText, 1205
voice_arrAlertingPattern, 1205
alertingPattern, 1206
callID, 1206
numInstances, 1206
voice_arrAlertingType, 1206
AlertingType, 1207
callID, 1207
numInstances, 1207
voice_arrAlphaID, 1207
allCallsAlphaIDInfoArr, 1207
numInstances, 1207
voice_arrCallEndReason, 1208
callEndReason, 1209
callID, 1209
numInstances, 1209
voice_arrCallInfo, 1209
getAllCallInfo, 1209
numInstances, 1209
voice_arrCalledPartyNum, 1207
CalledPartyNum, 1208
numInstances, 1208
voice_arrConnectPartyNum, 1209
ConnectedPartyNum, 1210
numInstances, 1210
voice_arrDiagInfo, 1210
DiagInfo, 1210
numInstances, 1210
voice_arrRedirPartyNum, 1210
numInstances, 1211
RedirPartyNum, 1211
voice_arrRemotePartyName, 1211
GetAllCallRmtPtyName, 1211
numInstances, 1211
voice_arrRemotePartyNum, 1211
numInstances, 1212
RmtPtyNum, 1212
voice_arrSvcOption, 1212
callID, 1212
numInstances, 1212
srvOption, 1212
voice_arrUUSInfo, 1213
AllCallsUUSInfo, 1213
numInstances, 1213
voice_burstDTMFInfo, 1213
digitCnt, 1214
pCallID, 1214
pDigitBuff, 1214
voice_callFWExtInfo, 1218
noReplyTimer, 1219

numLen, 1219
numPlan, 1220
numType, 1220
number, 1219
PI, 1220
SI, 1220
SvcClass, 1220
SvcStatus, 1220
voice_callFWInfo, 1220
noReplyTimer, 1220
numLen, 1221
number, 1220
SvcClass, 1221
SvcStatus, 1221
voice_callFwdTypeAndPlan, 1217
numberPlan, 1218
numberType, 1218
voice_callInfo, 1221
callID, 1222
callState, 1222
callType, 1222
direction, 1222
mode, 1222
voice_calledPartyInfo, 1214
numLen, 1215
numPlan, 1215
numType, 1215
number, 1215
PI, 1215
SI, 1215
voice_calledPartySubAdd, 1215
extBit, 1216
oddEvenInd, 1216
subAddr, 1216
subAddrLen, 1216
subAddrType, 1216
voice_callerIDInfo, 1216
callerID, 1217
callerIDLen, 1217
PI, 1217
voice_callingPartyInfo, 1222
numLen, 1224
numPlan, 1224
numType, 1224
number, 1224
PI, 1224
SI, 1224
voice_ccSUPSType, 1224
reason, 1224
svcType, 1225
voice_connectNumInfo, 1228
callerID, 1229
callerIDLen, 1229
numPlan, 1229
numPresInd, 1229
numType, 1230
screeningInd, 1230
voice_curAMRConfig, 1230
gsmAmrStat, 1231
wcdmaAmrStat, 1231
voice_diagInfo, 1231
diagInfoLen, 1231
diagnosticInfo, 1231
voice_extDispRecInfo, 1234
dispType, 1234
extDispInfo, 1234
extDispInfoLen, 1234
voice_getAllCallInformation, 1235
ALS, 1235
Callinfo, 1235
isEmpty, 1235
voice_getAllCallRmtPtyName, 1235
callID, 1236
RemotePartyName, 1236
voice_getAllCallRmtPtyNum, 1236
callID, 1236
RemotePartyNum, 1236
voice_getCallFWExtInfo, 1236
CallFWExtInfo, 1237
numInstances, 1237
voice_getCallFWInfo, 1237
CallFWInfo, 1237
numInstances, 1237
voice_lineCtrlInfo, 1237
polarityIncluded, 1238
pwrDenialTime, 1238
revPolarity, 1238
toggleMode, 1238
voice_newPwdData, 1238
newPwd, 1239
newPwdAgain, 1239
voice_peerNumberInfo, 1239
callID, 1240
numLen, 1241
numPI, 1241
numPlan, 1241
numSI, 1241
numType, 1241
number, 1241
voice_prefVoiceSO, 1241
evrcCapability, 1243
homeOrigVoiceSO, 1243
homePageVoiceSO, 1243
namID, 1243
roamOrigVoiceSO, 1243
voice_redirNumInfo, 1243
numLen, 1244
numPlan, 1244
numType, 1244
number, 1244
PI, 1244
reason, 1244
SI, 1244
voice_remotePartyName, 1245
callerName, 1245
codingScheme, 1245

nameLen, 1245
namePI, 1245
voice_remotePartyNum, 1245
numLen, 1246
presentationInd, 1246
remPartyNumber, 1246
voice_roamTimer, 1246
namID, 1247
roamTimerValue, 1247
voice_signalInfo, 1247
alertPitch, 1247
signal, 1247
signalType, 1247
VoiceDomainPref
nas_VoiceDomainPrefTlv, 378
VoiceNumber
unpack_dms_GetVoiceNumber_t, 768
voiceNumberSize
unpack_dms_GetVoiceNumber_t, 768
voicePrivacy
unpack_voice_voicePrivacyChangeCallback_ind_t, 1134
voiceSvc
pack_voice_SLQSVoiceSetSUPSService_t, 604
VoipRat
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
VoipRatVal
imsa_VoipRatInfo, 116
VoipService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
VoipServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
VoipServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
VoipSvcStatus
imsa_VoipSvcStatusInfo, 116
VolValue
pack_audio_SLQSSetAudioVolTLBConfig_t, 412
VoltStat
dms_VoltageTlv, 78
VoltTlv
unpack_dms_SwiEventReportCallBack_ind_t, 804
Voltage
dms_VoltageTlv, 78
Volume
pack_audio_SLQSGetAudioVolTLBConfig_t, 408
pack_audio_SLQSSetAudioProfile_t, 411
pack_audio_SLQSSetAudioVolTLBConfig_t, 412
unpack_audio_SLQSGetAudioProfile_t, 732
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 1034
voteForInit
uim_registerRefresh, 718
VsServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
VsServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
VtRat
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
VtRatVal
imsa_VtRatInfo, 117
VtService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 840
VtServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 836
VtServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 837
VtSvcStatus
imsa_VtSvcStatusInfo, 117

wAgc
nas_WCDMACellInfoExt, 380
wAlertMsgLength
UnpackSwiAvmsEventReportConfig, 1196
WCDMACellInfo
nas_lteWcdmaCellInfo, 280
WCDMAECIOThreshListLen
nas_WCDMAECIOThresh, 381
WCDMARSSIThreshListLen
nas_WCDMARSSIThresh, 382
WCDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 946
WDS_PROFILE_3GPP
wds.h, 1767
WDS_PROFILE_3GPP2
wds.h, 1767
wDescriptionLength
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 1195
wDlBler
nas_WCDMACellInfoExt, 380
wErrorCode
UnpackSwiAvmsEventReportDataSessionStatus, 1197
wHTTPStatus
UnpackSwiAvmsEventReportHTTPStatus, 1198
wNameLength
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 1195
WORD
SwiDataTypes.h, 1657
wSessionStatus
UnpackSwiAvmsEventReportNotification, 1200
wTxAgc
nas_WCDMACellInfoExt, 380
wUpdateCompeteStatus
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 1195
wUserInputTimeout
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 1195
UnpackSwiAvmsEventReportConfig, 1196

UnpackSwiAvmsEventReportConnectionRequest, 1196
wVersionLength
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 1195
wWamsChangeMask
UnpackSwiAvmsEventReportWAMSParam-Change, 1202
wake_mask
pack_rms_SetSMSWake_t, 530
unpack_rms_GetSMSWake_t, 1011
wcdmaAmrStat
voice_curAMRConfig, 1231
wcdmaArfcn
nas_WCDMAARFCNTlv, 378
wcdmaRRCState
nas_WCDMAInfoLTENeighborCell, 381
wda.h, 1754
helper_get_wda_resp_ctx, 1755
pack_wda_GetDataFormat, 1755
pack_wda_GetQmapSettings, 1755
pack_wda_SetDataFormat, 1756
pack_wda_SetQmapSettings, 1756
unpack_wda_GetDataFormat, 1756
unpack_wda_GetQmapSettings, 1757
unpack_wda_SetDataFormat, 1757
unpack_wda_SetQmapSettings, 1757
Wdisable
unpack_dms_SLQSDmsSwiGetPCInfo_t, 778
wds.h
QMI_LITE_WDS_CURRENT_CALL_DB_MASK, 1768
QMI_LITE_WDS_LAST_CALL_DB_MASK, 1768
wds.h, 1758
BYT_STAT_STAT_MASK, 1766
liteQmiDataBearerMasks, 1768
PACK_WDS_IPV4, 1767
PACK_WDS_IPV6, 1767
pack_wds_DHCPv4ClientLeaseChange, 1768
pack_wds_GetAutoconnect, 1769
pack_wds_GetByteTotals, 1769
pack_wds_GetConnectionRate, 1769
pack_wds_GetDataBearerTechnology, 1770
pack_wds_GetDefaultProfile, 1770
pack_wds_GetDefaultProfileNum, 1771
pack_wds_GetDefaultProfileV2, 1771
pack_wds_GetDormancyState, 1771
pack_wds_GetLastMobileIPError, 1772
pack_wds_GetMobileIP, 1772
pack_wds_GetMobileIPProfile, 1773
pack_wds_GetPacketStatistics, 1773
pack_wds_GetPacketStatus, 1774
pack_wds_GetSessionDuration, 1774
pack_wds_GetSessionDurationV2, 1774
pack_wds_GetSessionState, 1775
pack_wds_RMSetTransferStatistics, 1775
pack_wds_SLQSCreateProfile, 1779
pack_wds_SLQSDeleteProfile, 1779
pack_wds_SLQSGet3GPPConfigItem, 1779
pack_wds_SLQSGetCurrDataSystemStat, 1780
pack_wds_SLQSGetCurrentChannelRate, 1780
pack_wds_SLQSGetDUNCallInfo, 1781
pack_wds_SLQSGetDataBearerTechnology, 1781
pack_wds_SLQSGetProfileSettings, 1782
pack_wds_SLQSGetProfileSettingsV2, 1782
pack_wds_SLQSGetRuntimeSettings, 1782
pack_wds_SLQSModifyProfile, 1783
pack_wds_SLQSResetPacketStatics, 1783
pack_wds_SLQSSGetDHCPv4ClientConfig, 1785
pack_wds_SLQSSGetLoopback, 1785
pack_wds_SLQSSSetDHCPv4ClientConfig, 1786
pack_wds_SLQSSSetLoopback, 1786
pack_wds_SLQSSet3GPPConfigItem, 1784
pack_wds_SLQSSetIPFamilyPreference, 1784
pack_wds_SLQSSetWdsEventCallback, 1784
pack_wds_SLQSStartDataSession, 1786
pack_wds_SLQSStopDataSession, 1787
pack_wds_SLQSSwiProfileChangeCallback, 1787
pack_wds_SLQSWdsGoActive, 1788
pack_wds_SLQSWdsGoDormant, 1788
pack_wds_SLQSWdsSetEventReport, 1789
pack_wds_SLQSWdsSwiPDPRuntimeSettings, 1789
pack_wds_SetAutoconnect, 1776
pack_wds_SetDefaultProfile, 1776
pack_wds_SetDefaultProfileNum, 1776
pack_wds_SetMobileIP, 1777
pack_wds_SetMobileIPParameters, 1777
pack_wds_SetMobileIPProfile, 1778
pack_wds_SetMuxID, 1778
unpack_wds_DHCPv4ClientLease_ind, 1789
unpack_wds_DHCPv4ClientLeaseChange, 1790
unpack_wds_DHCPv4ClientLeaseChange_t, 1767
unpack_wds_GetAutoconnect, 1790
unpack_wds_GetByteTotals, 1791
unpack_wds_GetConnectionRate, 1791
unpack_wds_GetDataBearerTechnology, 1791
unpack_wds_GetDefaultProfile, 1792
unpack_wds_GetDefaultProfileNum, 1792
unpack_wds_GetDefaultProfileV2, 1792
unpack_wds_GetDormancyState, 1793
unpack_wds_GetLastMobileIPError, 1793
unpack_wds_GetMobileIP, 1793
unpack_wds_GetMobileIPProfile, 1794
unpack_wds_GetPacketStatistics, 1794
unpack_wds_GetPacketStatus, 1794
unpack_wds_GetSessionDuration, 1795
unpack_wds_GetSessionDurationV2, 1795
unpack_wds_GetSessionState, 1795
unpack_wds_RMSetTransferStatistics, 1796
unpack_wds_RMTransferStatistics_ind, 1796
unpack_wds_RMTransferStatistics_ind_t, 1767
unpack_wds_SLQSCreateProfile, 1799
unpack_wds_SLQSDUNCallInfoCallBack_ind, 1799
unpack_wds_SLQSDeleteProfile, 1799

unpack_wds_SLQSGet3GPPConfigItem, 1800
unpack_wds_SLQSGetCurrDataSystemStat, 1800
unpack_wds_SLQSGetCurrentChannelRate, 1800
unpack_wds_SLQSGetDUNCallInfo, 1801
unpack_wds_SLQSGetDataBearerTechnology, 1801
unpack_wds_SLQSGetProfileSettings, 1801
unpack_wds_SLQSGetProfileSettingsV2, 1802
unpack_wds_SLQSGetRuntimeSettings, 1802
unpack_wds_SLQSModifyProfile, 1802
unpack_wds_SLQSResetPacketStatics, 1803
unpack_wds_SLQSResetPacketStatics_t, 1767
unpack_wds_SLQSSGetDHCPv4ClientConfig, 1805
unpack_wds_SLQSSGetLoopback, 1805
unpack_wds_SLQSSSetDHCPv4ClientConfig, 1805
unpack_wds_SLQSSSetDHCPv4ClientConfig_t, 1767
unpack_wds_SLQSSSetLoopback, 1806
unpack_wds_SLQSSSetLoopback_t, 1767
unpack_wds_SLQSSet3GPPConfigItem, 1803
unpack_wds_SLQSSet3GPPConfigItem_t, 1767
unpack_wds_SLQSSetIPFamilyPreference, 1803
unpack_wds_SLQSSetPacketSrvStatusCallback, 1804
unpack_wds_SLQSSetWdsEventCallback, 1804
unpack_wds_SLQSSetWdsEventCallback_ind, 1804
unpack_wds_SLQSSetWdsEventCallback_t, 1767
unpack_wds_SLQSStartDataSession, 1806
unpack_wds_SLQSStopDataSession, 1806
unpack_wds_SLQSStopDataSession_t, 1767
unpack_wds_SLQSSwiProfileChangeCallback, 1807
unpack_wds_SLQSSwiProfileChangeCallback_Ind, 1807
unpack_wds_SLQSSwiProfileChangeCallback_t, 1767
unpack_wds_SLQSWdsGoActive, 1807
unpack_wds_SLQSWdsGoActive_t, 1767
unpack_wds_SLQSWdsGoDormant, 1808
unpack_wds_SLQSWdsGoDormant_t, 1767
unpack_wds_SLQSWdsSetEventReport, 1808
unpack_wds_SLQSWdsSetEventReport_t, 1767
unpack_wds_SLQSWdsSwiPDPRuntimeSettings, 1808
unpack_wds_SetAutoconnect, 1796
unpack_wds_SetAutoconnect_t, 1767
unpack_wds_SetDefaultProfile, 1797
unpack_wds_SetDefaultProfile_t, 1767
unpack_wds_SetDefaultProfileNum, 1797
unpack_wds_SetDefaultProfileNum_t, 1767
unpack_wds_SetMobileIP, 1797
unpack_wds_SetMobileIP_t, 1767
unpack_wds_SetMobileIPParameters, 1798
unpack_wds_SetMobileIPParameters_t, 1767
unpack_wds_SetMobileIPProfile, 1798
unpack_wds_SetMuxID, 1798
unpack_wds_SetMuxID_t, 1767
UnpackQmiProfileInfo, 1767
UnpackQmiProfileInfoV2, 1768
WDS_PROFILE_3GPP, 1767
WDS_PROFILE_3GPP2, 1767
wds_ChannelRateTlv, 1253
ChannelRate, 1254
TlvPresent, 1254
wds_ConnStatusTlv, 1254
MDMConnStatus, 1254
TlvPresent, 1254
wds_DHCPLeaseOptTlv, 1257
numOpt, 1257
optList, 1257
optListData, 1257
TlvPresent, 1257
wds_DHCPLeaseStateTlv, 1258
leaseState, 1258
TlvPresent, 1258
wds_DHCPOpt, 1258
optCode, 1259
optValLen, 1259
pOptVal, 1259
wds_DHCPProfileIdTlv, 1259
profileId, 1259
profileType, 1259
TlvPresent, 1259
wds_DHCPv4HWConfig, 1259
chaddr, 1260
chaddrLen, 1260
hwType, 1260
wds_DHCPv4Option, 1260
optCode, 1260
optVal, 1260
optValLen, 1261
wds_DHCPv4OptionList, 1261
numOpt, 1261
pOptList, 1261
wds_DHCPv4ProfileId, 1261
profileId, 1261
profileType, 1262
wds_DataBearTechTlv, 1255
DataBearerTech, 1256
TlvPresent, 1256
wds_DataULongLongTlv, 1256
TlvPresent, 1256
ullData, 1256
wds_DataULongTlv, 1257
TlvPresent, 1257
ulData, 1257
wds_Domain, 1262
domainLen, 1262
domainName, 1262
wds_DomainNameList, 1262
domain, 1263
numInstances, 1263
wds_DormStatTlv, 1263

DormancyStat, 1263
TlvPresent, 1263
wds_GPRSQoS, 1263
delayClass, 1264
meanThroughputClass, 1264
peakThroughputClass, 1264
precedenceClass, 1264
reliabilityClass, 1264
wds_IPV6AddressInfo, 1265
IPAddressV6, 1265
IPV6PrefixLen, 1265
wds_IPV6GWAddressInfo, 1265
gwAddressV6, 1266
gwV6PrefixLen, 1266
wds_IPv4AdTlv, 1264
IPv4Addr, 1264
TlvPresent, 1264
wds_LastMdmCallEndRsnTlv, 1266
CallEndReason, 1266
TlvPresent, 1266
wds_PCSCFFQDNAddress, 1266
fqdnAddr, 1267
fqdnLen, 1267
wds_PCSCFFQDNAddressList, 1267
numInstances, 1267
pcsfFQDNAddress, 1267
wds_PCSCFIPv4ServerAddressList, 1267
numInstances, 1268
pscsfIPv4Addr, 1268
wds_ProfileIdentifier, 1268
profileIndex, 1269
profileType, 1269
wds_RXBytesOKTlv, 1269
RxByteOKCnt, 1270
TlvPresent, 1270
wds_TXBytesOKTlv, 1271
TlvPresent, 1272
TxByteOKCnt, 1272
wds_TrStatInd, 1271
statsMask, 1271
statsPeriod, 1271
wds_UMTSMinQoS, 1272
deliveryErrSDU, 1274
grntDownlinkBitrate, 1274
grntUplinkBitrate, 1274
maxDownlinkBitrate, 1274
maxSDUSize, 1274
maxUplinkBitrate, 1274
qosDeliveryOrder, 1274
resBerRatio, 1274
sduErrorRatio, 1274
trafficClass, 1274
trafficPriority, 1274
transferDelay, 1274
wds_channelRate, 1253
CurrChanRxRate, 1253
CurrChanTxRate, 1253
wds_currNetworkInfo, 1254
NetworkType, 1255
RATMask, 1255
SOMask, 1255
wds_profileChange, 1268
profileInx, 1268
profileType, 1268
wds_profileInfo, 1269
SlqsProfile3GPP, 1269
SlqsProfile3GPP2, 1269
wds_sourceOfChange, 1270
source, 1270
wds_transferStatInd, 1270
StatsMask, 1271
StatsPeriod, 1271
wdsDhcpv4HwConfig, 1274
chaddr, 1275
chaddrLen, 1275
hwType, 1275
wdsDhcpv4Option, 1275
optCode, 1275
optVal, 1275
optValLen, 1275
wdsDhcpv4OptionList, 1275
numOpt, 1276
pOptList, 1276
wdsDhcpv4ProfileId, 1276
profileId, 1276
profileType, 1276
word
swi_uint256_t, 676
writeLock
MbimTransport, 192
QmuxTransport, 649
writeRecord
pack_uim_SLQSUIMWriteRecord_t, 583
writeTransparent
pack_uim_SLQSUIMWriteTransparent_t, 584

xAxis
sensorData_t, 654
xferStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 1186
xid
pack_loc_SLQSLOCGetBestAvailPos_t, 449
pack_qmi_t, 528
unpack_qmi_t, 988
xtra_start_gps_minutes
unpack_pds_SLQSGetGPSStateInfo_t, 987
xtra_start_gps_week
unpack_pds_SLQSGetGPSStateInfo_t, 987
xtra_valid_duration_hours
unpack_pds_SLQSGetGPSStateInfo_t, 987

yAxis
sensorData_t, 654
year
nas_timeInfo, 368
nas_UniversalTime, 377

zAxis
sensorData_t, 655