LinuxQMISDK-Lite

SLQS04.00.21

Generated by Doxygen 1.8.6

Thu Jul 4 2019 02:28:49

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 **27**

5.1 File List . . . 27

6 Module Documentation **29**

6.1 QMI pack/unpack (lite-qmi) . . . 29

6.1.1 Detailed Description . . . 29

6.2 Streaming Download Protocol (lite-fw) . . . 30

6.2.1 Detailed Description . . . 30

7 Namespace Documentation **31**

7.1 Tables Namespace Reference . . . 31

7.1.1 Detailed Description . . . 31

8 Data Structure Documentation **33**

8.1 _litefw_FirmwareInfo_ Struct Reference . . . 33

8.1.1 Detailed Description . . . 33

8.1.2 Field Documentation . . . 33

8.1.2.1 szCarrier_str . . . 34
8.1.2.2 szCarrierPriversion_str . . . 34
8.1.2.3 szFwversion_str . . . 34
8.1.2.4 szModelid_str . . . 34
8.1.2.5 szPackageid_str . . . 34
8.1.2.6 szSku_str . . . 34
8.2 _litefw_FirmwarePartNo_ Struct Reference . . . 34
8.2.1 Detailed Description . . . 34
8.2.2 Field Documentation . . . 34
8.2.2.1 szPartno_str . . . 34
8.3 altSrcInfo_t Struct Reference . . . 34
8.3.1 Detailed Description . . . 34
8.3.2 Field Documentation . . . 35
8.3.2.1 coverage . . . 35
8.3.2.2 linkage . . . 35
8.3.2.3 source . . . 35
8.4 appStats Struct Reference . . . 35
8.4.1 Detailed Description . . . 36
8.4.2 Field Documentation . . . 38
8.4.2.1 aidLength . . . 38
8.4.2.2 aidVal . . . 38
8.4.2.3 appState . . . 38
8.4.2.4 appType . . . 38
8.4.2.5 persoFeature . . . 38
8.4.2.6 persoRetries . . . 38
8.4.2.7 persoState . . . 38
8.4.2.8 persoUnblockRetries . . . 38
8.4.2.9 pin1Retries . . . 38
8.4.2.10 pin1State . . . 38
8.4.2.11 pin2Retries . . . 38
8.4.2.12 pin2State . . . 38
8.4.2.13 puk1Retries . . . 38
8.4.2.14 puk2Retries . . . 38
8.4.2.15 univPin . . . 38
8.5 audio_RXAGCList Struct Reference . . . 38
8.5.1 Detailed Description . . . 39
8.5.2 Field Documentation . . . 39
8.5.2.1 pRXAIG . . . 39
8.5.2.2 pRXComprSlope . . . 39
8.5.2.3 pRXComprThres . . . 39

8.5.2.4 pRXExpSlope . . . 39
8.5.2.5 pRXExpThres . . . 39
8.5.2.6 pRXStaticGain . . . 39
8.6 audio_RXAVCList Struct Reference . . . 39
8.6.1 Detailed Description . . . 40
8.6.2 Field Documentation . . . 40
8.6.2.1 pAVRXAVCHeadroom . . . 40
8.6.2.2 pAVRXAVCSens . . . 40
8.7 audio_RXPCMIIRFltr Struct Reference . . . 40
8.7.1 Detailed Description . . . 40
8.7.2 Field Documentation . . . 41
8.7.2.1 pFlag . . . 41
8.7.2.2 pStage0Val . . . 41
8.7.2.3 pStage1Val . . . 41
8.7.2.4 pStage2Val . . . 41
8.7.2.5 pStage3Val . . . 41
8.7.2.6 pStage4Val . . . 41
8.7.2.7 pStageCnt . . . 41
8.8 audio_TXAGCList Struct Reference . . . 41
8.8.1 Detailed Description . . . 42
8.8.2 Field Documentation . . . 42
8.8.2.1 pTXAIG . . . 42
8.8.2.2 pTXComprSlope . . . 42
8.8.2.3 pTXComprThres . . . 42
8.8.2.4 pTXExpSlope . . . 42
8.8.2.5 pTXExpThres . . . 42
8.8.2.6 pTXStaticGain . . . 42
8.9 audio_TXPCMIIRFltr Struct Reference . . . 42
8.9.1 Detailed Description . . . 43
8.9.2 Field Documentation . . . 44
8.9.2.1 pFlag . . . 44
8.9.2.2 pStage0Val . . . 44
8.9.2.3 pStage1Val . . . 44
8.9.2.4 pStage2Val . . . 44
8.9.2.5 pStage3Val . . . 44
8.9.2.6 pStage4Val . . . 44
8.9.2.7 pStageCnt . . . 44
8.10 CarrierImage_t Struct Reference . . . 44
8.10.1 Detailed Description . . . 44
8.10.2 Field Documentation . . . 45

8.10.2.1 m_FwBuildId . . . 45
8.10.2.2 m_FwImageId . . . 45
8.10.2.3 m_nCarrierId . . . 45
8.10.2.4 m_nFolderId . . . 45
8.10.2.5 m_nStorage . . . 45
8.10.2.6 m_PriBuildId . . . 45
8.10.2.7 m_PriImageId . . . 45
8.11 cat_AlPhaIdentifierTlv Struct Reference . . . 45
8.11.1 Detailed Description . . . 45
8.11.2 Field Documentation . . . 46
8.11.2.1 AlphaID . . . 46
8.11.2.2 AlphaIDLength . . . 46
8.11.2.3 ReferenceID . . . 46
8.12 cat_commonEventTlv Struct Reference . . . 46
8.12.1 Detailed Description . . . 46
8.12.2 Field Documentation . . . 46
8.12.2.1 CatEvent . . . 46
8.12.2.2 EventID . . . 46
8.12.2.3 EventLength . . . 46
8.12.2.4 TlvPresent . . . 46
8.13 cat_currentCatEvent Union Reference . . . 46
8.13.1 Detailed Description . . . 47
8.13.2 Field Documentation . . . 47
8.13.2.1 CatAlphaIdtfr . . . 47
8.13.2.2 CatEndPS . . . 47
8.13.2.3 CatEventLst . . . 47
8.13.2.4 CatEvIDData . . . 47
8.13.2.5 CatRefresh . . . 47
8.14 cat_EndProactiveSessionTlv Struct Reference . . . 47
8.14.1 Detailed Description . . . 47
8.14.2 Field Documentation . . . 47
8.14.2.1 EndProactiveSession . . . 48
8.15 cat_EventIDDataTlv Struct Reference . . . 48
8.15.1 Detailed Description . . . 48
8.15.2 Field Documentation . . . 48
8.15.2.1 Data . . . 48
8.15.2.2 DataLength . . . 48
8.15.2.3 ReferenceID . . . 48
8.16 cat_EventListTlv Struct Reference . . . 48
8.16.1 Detailed Description . . . 48

8.16.2 Field Documentation . . . 48
8.16.2.1 SetupEventList . . . 48
8.17 cat_RefreshTlv Struct Reference . . . 49
8.17.1 Detailed Description . . . 49
8.17.2 Field Documentation . . . 49
8.17.2.1 RefreshMode . . . 49
8.17.2.2 RefreshStage . . . 49
8.18 cdmaSSInfo Struct Reference . . . 49
8.18.1 Detailed Description . . . 49
8.18.2 Field Documentation . . . 49
8.18.2.1 ecio . . . 49
8.18.2.2 rssi . . . 50
8.19 connectionStatus Struct Reference . . . 50
8.19.1 Detailed Description . . . 50
8.19.2 Field Documentation . . . 50
8.19.2.1 MDMCallDuration . . . 50
8.19.2.2 MDMConnStatus . . . 50
8.20 crashInfoParams Struct Reference . . . 50
8.20.1 Detailed Description . . . 50
8.20.2 Field Documentation . . . 51
8.20.2.1 crashInfo . . . 51
8.20.2.2 crashStatus . . . 51
8.21 crashInformation Struct Reference . . . 51
8.21.1 Detailed Description . . . 51
8.21.2 Field Documentation . . . 52
8.21.2.1 crashData . . . 52
8.21.2.2 crashId . . . 52
8.21.2.3 crashString . . . 52
8.21.2.4 crashStrlen . . . 52
8.21.2.5 gcdumpString . . . 52
8.21.2.6 gcdumpStrlen . . . 52
8.21.2.7 numCrashes . . . 52
8.22 currNetworkInfo Struct Reference . . . 52
8.22.1 Detailed Description . . . 52
8.22.2 Field Documentation . . . 52
8.22.2.1 NetworkType . . . 52
8.22.2.2 RATMask . . . 52
8.22.2.3 SOMask . . . 52
8.23 dms_ActivationStatusTlv Struct Reference . . . 52
8.23.1 Detailed Description . . . 53

8.23.2 Field Documentation . . . 53
8.23.2.1 activationStatus . . . 53
8.23.2.2 TlvPresent . . . 53
8.24 dms_OperatingModeTlv Struct Reference . . . 53
8.24.1 Detailed Description . . . 53
8.24.2 Field Documentation . . . 54
8.24.2.1 operatingMode . . . 54
8.24.2.2 TlvPresent . . . 54
8.25 dms_PSMActiveTimerIndTlv Struct Reference . . . 54
8.25.1 Detailed Description . . . 54
8.25.2 Field Documentation . . . 54
8.25.2.1 ActiveTimerInd . . . 54
8.25.2.2 TlvPresent . . . 54
8.26 dms_PSMActiveTimerTlv Struct Reference . . . 54
8.26.1 Detailed Description . . . 55
8.26.2 Field Documentation . . . 55
8.26.2.1 activeTimer . . . 55
8.26.2.2 TlvPresent . . . 55
8.27 dms_PSMDurationDueToOOSTlv Struct Reference . . . 55
8.27.1 Detailed Description . . . 55
8.27.2 Field Documentation . . . 55
8.27.2.1 durationDueToOOS . . . 55
8.27.2.2 TlvPresent . . . 55
8.28 dms_PSMDurationThresholdTlv Struct Reference . . . 55
8.28.1 Detailed Description . . . 56
8.28.2 Field Documentation . . . 56
8.28.2.1 durationThreshold . . . 56
8.28.2.2 TlvPresent . . . 56
8.29 dms_PSMEarlyWakeupTimeTlv Struct Reference . . . 56
8.29.1 Detailed Description . . . 56
8.29.2 Field Documentation . . . 56
8.29.2.1 earlyWakeupTime . . . 56
8.29.2.2 TlvPresent . . . 56
8.30 dms_PSMEnableStateIndTlv Struct Reference . . . 56
8.30.1 Detailed Description . . . 57
8.30.2 Field Documentation . . . 57
8.30.2.1 EnableStateInd . . . 57
8.30.2.2 TlvPresent . . . 57
8.31 dms_PSMEnableStateTlv Struct Reference . . . 57
8.31.1 Detailed Description . . . 57

8.31.2 Field Documentation . . . 57
8.31.2.1 enableState . . . 57
8.31.2.2 TlvPresent . . . 57
8.32 dms_PSMPeriodicUpdateTimerIndTlv Struct Reference . . . 57
8.32.1 Detailed Description . . . 58
8.32.2 Field Documentation . . . 58
8.32.2.1 PeriodicUpdateTimerInd . . . 58
8.32.2.2 TlvPresent . . . 58
8.33 dms_PSMPeriodicUpdateTimerTlv Struct Reference . . . 58
8.33.1 Detailed Description . . . 58
8.33.2 Field Documentation . . . 58
8.33.2.1 periodicUpdateTimer . . . 58
8.33.2.2 TlvPresent . . . 58
8.34 dms_PSMRandomizationWindowTlv Struct Reference . . . 58
8.34.1 Detailed Description . . . 59
8.34.2 Field Documentation . . . 59
8.34.2.1 randomizationWindow . . . 59
8.34.2.2 TlvPresent . . . 59
8.35 dms_TemperatureTlv Struct Reference . . . 59
8.35.1 Detailed Description . . . 59
8.35.2 Field Documentation . . . 60
8.35.2.1 Temperature . . . 60
8.35.2.2 TempStat . . . 60
8.35.2.3 TlvPresent . . . 60
8.36 dms_UimAutoSwitchActSlotTlv Struct Reference . . . 60
8.36.1 Detailed Description . . . 60
8.36.2 Field Documentation . . . 60
8.36.2.1 TlvPresent . . . 60
8.36.2.2 uimAutoSwitchActSlot . . . 60
8.37 dms_UimStatusTlv Struct Reference . . . 60
8.37.1 Detailed Description . . . 60
8.37.2 Field Documentation . . . 61
8.37.2.1 event . . . 61
8.37.2.2 intf . . . 61
8.37.2.3 TlvPresent . . . 61
8.38 dms_VoltageTlv Struct Reference . . . 61
8.38.1 Detailed Description . . . 61
8.38.2 Field Documentation . . . 62
8.38.2.1 TlvPresent . . . 62
8.38.2.2 Voltage . . . 62

8.38.2.3 VoltStat . . . 62
8.39 DMScustSettingInfo Struct Reference . . . 62
8.39.1 Detailed Description . . . 62
8.39.2 Field Documentation . . . 62
8.39.2.1 cust_attr . . . 62
8.39.2.2 cust_id . . . 62
8.39.2.3 cust_value . . . 62
8.39.2.4 id_length . . . 63
8.39.2.5 value_length . . . 63
8.40 DMScustSettingList Struct Reference . . . 63
8.40.1 Detailed Description . . . 63
8.40.2 Field Documentation . . . 63
8.40.2.1 custSetting . . . 63
8.40.2.2 list_type . . . 63
8.40.2.3 num_instances . . . 63
8.41 DMSgetCustomFeatureV2 Struct Reference . . . 63
8.41.1 Detailed Description . . . 63
8.41.2 Field Documentation . . . 64
8.41.2.1 pCustSettingInfo . . . 64
8.41.2.2 pCustSettingList . . . 64
8.41.2.3 pGetCustomInput . . . 64
8.42 DMSgetCustomInput Struct Reference . . . 64
8.42.1 Detailed Description . . . 64
8.42.2 Field Documentation . . . 64
8.42.2.1 cust_id . . . 64
8.42.2.2 list_type . . . 64
8.43 dunchannelRate Struct Reference . . . 64
8.43.1 Detailed Description . . . 65
8.43.2 Field Documentation . . . 65
8.43.2.1 CurrChanRxRate . . . 65
8.43.2.2 CurrChanTxRate . . . 65
8.43.2.3 MaxChanRxRate . . . 65
8.43.2.4 MaxChanTxRate . . . 65
8.44 eriDataparams Struct Reference . . . 65
8.44.1 Detailed Description . . . 65
8.44.2 Field Documentation . . . 66
8.44.2.1 eriData . . . 66
8.44.2.2 eriDataLen . . . 66
8.45 eTWSPLMNInfoTlv Struct Reference . . . 66
8.45.1 Detailed Description . . . 66

8.45.2 Field Documentation . . . 66
8.45.2.1 ETWSPLMNInfo . . . 66
8.45.2.2 TlvPresent . . . 66
8.46 FMSImageElement Struct Reference . . . 66
8.46.1 Detailed Description . . . 66
8.46.2 Field Documentation . . . 67
8.46.2.1 buildId . . . 67
8.46.2.2 buildIdLength . . . 67
8.46.2.3 imageId . . . 67
8.46.2.4 imageType . . . 67
8.47 FMSImageIdElement Struct Reference . . . 67
8.47.1 Detailed Description . . . 67
8.47.2 Field Documentation . . . 68
8.47.2.1 buildID . . . 68
8.47.2.2 buildIDLength . . . 68
8.47.2.3 failureCount . . . 68
8.47.2.4 imageID . . . 68
8.47.2.5 storageIndex . . . 68
8.48 FMSImageIDEntries Struct Reference . . . 68
8.48.1 Detailed Description . . . 68
8.48.2 Field Documentation . . . 68
8.48.2.1 executingImage . . . 68
8.48.2.2 imageIDElement . . . 68
8.48.2.3 imageIDSize . . . 69
8.48.2.4 imageType . . . 69
8.48.2.5 maxImages . . . 69
8.49 FMSImageList Struct Reference . . . 69
8.49.1 Detailed Description . . . 69
8.49.2 Field Documentation . . . 69
8.49.2.1 imageIDEntries . . . 69
8.49.2.2 listSize . . . 69
8.50 FMSPrefImageList Struct Reference . . . 69
8.50.1 Detailed Description . . . 69
8.50.2 Field Documentation . . . 70
8.50.2.1 listEntries . . . 70
8.50.2.2 listSize . . . 70
8.51 hdrSSInfo Struct Reference . . . 70
8.51.1 Detailed Description . . . 70
8.51.2 Field Documentation . . . 70
8.51.2.1 ecio . . . 70

8.51.2.2 io . . . 71
8.51.2.3 rssi . . . 71
8.51.2.4 sinr . . . 71
8.52 image_info_t Struct Reference . . . 71
8.52.1 Detailed Description . . . 71
8.52.2 Field Documentation . . . 71
8.52.2.1 buildID . . . 71
8.52.2.2 buildIDLen . . . 71
8.52.2.3 imageType . . . 71
8.52.2.4 uniqueID . . . 71
8.53 ims_AMRModeInfo Struct Reference . . . 71
8.53.1 Detailed Description . . . 72
8.53.2 Field Documentation . . . 72
8.53.2.1 amrMode . . . 72
8.53.2.2 TlvPresent . . . 72
8.54 ims_AMROctAlgnInfo Struct Reference . . . 72
8.54.1 Detailed Description . . . 72
8.54.2 Field Documentation . . . 72
8.54.2.1 amrOctAlgn . . . 72
8.54.2.2 TlvPresent . . . 73
8.55 ims_AMRWBModeInfo Struct Reference . . . 73
8.55.1 Detailed Description . . . 73
8.55.2 Field Documentation . . . 73
8.55.2.1 amrWBMode . . . 73
8.55.2.2 TlvPresent . . . 73
8.56 ims_AMRWBOctAlgnInfo Struct Reference . . . 73
8.56.1 Detailed Description . . . 73
8.56.2 Field Documentation . . . 74
8.56.2.1 amrWBOctAlgn . . . 74
8.56.2.2 TlvPresent . . . 74
8.57 ims_CSCFPortNameInfo Struct Reference . . . 74
8.57.1 Detailed Description . . . 74
8.57.2 Field Documentation . . . 74
8.57.2.1 cscfPortName . . . 74
8.57.2.2 TlvPresent . . . 74
8.58 ims_EnabAMRWBInfo Struct Reference . . . 74
8.58.1 Detailed Description . . . 74
8.58.2 Field Documentation . . . 74
8.58.2.1 amrWBEnable . . . 74
8.58.2.2 TlvPresent . . . 75

8.59 ims_EnabSCRAMRInfo Struct Reference . . . 75
8.59.1 Detailed Description . . . 75
8.59.2 Field Documentation . . . 75
8.59.2.1 scrAmrEnable . . . 75
8.59.2.2 TlvPresent . . . 75
8.60 ims_EnabSCRAMRWBInfo Struct Reference . . . 75
8.60.1 Detailed Description . . . 75
8.60.2 Field Documentation . . . 75
8.60.2.1 scrAmrWBEnable . . . 75
8.60.2.2 TlvPresent . . . 75
8.61 ims_IMSDomainInfo Struct Reference . . . 76
8.61.1 Detailed Description . . . 76
8.61.2 Field Documentation . . . 76
8.61.2.1 imsDomainName . . . 76
8.61.2.2 TlvPresent . . . 76
8.62 ims_IMSTestModeInfo Struct Reference . . . 76
8.62.1 Detailed Description . . . 76
8.62.2 Field Documentation . . . 76
8.62.2.1 imsTestMode . . . 76
8.62.2.2 TlvPresent . . . 76
8.63 ims_MinSessExpInfo Struct Reference . . . 76
8.63.1 Detailed Description . . . 77
8.63.2 Field Documentation . . . 77
8.63.2.1 minSessExp . . . 77
8.63.2.2 TlvPresent . . . 77
8.64 ims_PCSCFPortInfo Struct Reference . . . 77
8.64.1 Detailed Description . . . 77
8.64.2 Field Documentation . . . 77
8.64.2.1 priCSCFPort . . . 77
8.64.2.2 TlvPresent . . . 77
8.65 ims_PhCtxtURIInfo Struct Reference . . . 77
8.65.1 Detailed Description . . . 78
8.65.2 Field Documentation . . . 78
8.65.2.1 PhCtxtURI . . . 78
8.65.2.2 TlvPresent . . . 78
8.66 ims_RngBkTmrInfo Struct Reference . . . 78
8.66.1 Detailed Description . . . 78
8.66.2 Field Documentation . . . 78
8.66.2.1 RingBkTmr . . . 78
8.66.2.2 TlvPresent . . . 78

8.67 ims_RngTmrInfo Struct Reference . . . 78
8.67.1 Detailed Description . . . 79
8.67.2 Field Documentation . . . 79
8.67.2.1 RingTmr . . . 79
8.67.2.2 TlvPresent . . . 79
8.68 ims_RTPRTCPInactTmrDurInfo Struct Reference . . . 79
8.68.1 Detailed Description . . . 79
8.68.2 Field Documentation . . . 79
8.68.2.1 InactTmr . . . 79
8.68.2.2 TlvPresent . . . 79
8.69 ims_SessDurInfo Struct Reference . . . 79
8.69.1 Detailed Description . . . 80
8.69.2 Field Documentation . . . 80
8.69.2.1 sessExp . . . 80
8.69.2.2 TlvPresent . . . 80
8.70 ims_SigCompEnInfo Struct Reference . . . 80
8.70.1 Detailed Description . . . 80
8.70.2 Field Documentation . . . 80
8.70.2.1 SigCompEn . . . 80
8.70.2.2 TlvPresent . . . 80
8.71 ims_SIPPortInfo Struct Reference . . . 80
8.71.1 Detailed Description . . . 81
8.71.2 Field Documentation . . . 81
8.71.2.1 SIPLocalPort . . . 81
8.71.2.2 TlvPresent . . . 81
8.72 ims_SIPRegnTmrInfo Struct Reference . . . 81
8.72.1 Detailed Description . . . 81
8.72.2 Field Documentation . . . 81
8.72.2.1 TlvPresent . . . 81
8.72.2.2 tmrSIPRegn . . . 81
8.73 ims_SMSFmtInfo Struct Reference . . . 81
8.73.1 Detailed Description . . . 82
8.73.2 Field Documentation . . . 82
8.73.2.1 smsFormat . . . 82
8.73.2.2 TlvPresent . . . 82
8.74 ims_SMSoIPNwInfo Struct Reference . . . 82
8.74.1 Detailed Description . . . 82
8.74.2 Field Documentation . . . 82
8.74.2.1 smsoIPNW . . . 82
8.74.2.2 TlvPresent . . . 82

8.75 ims_SubscrTmrInfo Struct Reference . . . 82
8.75.1 Detailed Description . . . 83
8.75.2 Field Documentation . . . 83
8.75.2.1 subscrTmr . . . 83
8.75.2.2 TlvPresent . . . 83
8.76 ims_TmrT1Info Struct Reference . . . 83
8.76.1 Detailed Description . . . 83
8.76.2 Field Documentation . . . 83
8.76.2.1 TlvPresent . . . 83
8.76.2.2 tmrT1 . . . 83
8.77 ims_TmrT2Info Struct Reference . . . 83
8.77.1 Detailed Description . . . 83
8.77.2 Field Documentation . . . 84
8.77.2.1 TlvPresent . . . 84
8.77.2.2 tmrT2 . . . 84
8.78 ims_TmrTfInfo Struct Reference . . . 84
8.78.1 Detailed Description . . . 84
8.78.2 Field Documentation . . . 84
8.78.2.1 TlvPresent . . . 84
8.78.2.2 tmrTf . . . 84
8.79 imsa_IMSFailErrCodeTlv Struct Reference . . . 84
8.79.1 Detailed Description . . . 84
8.79.2 Field Documentation . . . 85
8.79.2.1 ImsFailErrCode . . . 85
8.79.2.2 TlvPresent . . . 85
8.80 imsa_IMSRegStatusErrorCodeInfo Struct Reference . . . 85
8.80.1 Detailed Description . . . 85
8.80.2 Field Documentation . . . 85
8.80.2.1 ErrorCode . . . 85
8.80.2.2 TlvPresent . . . 85
8.81 imsa_IMSRegStatusInfo Struct Reference . . . 85
8.81.1 Detailed Description . . . 86
8.81.2 Field Documentation . . . 86
8.81.2.1 ImsRegistered . . . 86
8.81.2.2 TlvPresent . . . 86
8.82 imsa_NewIMSRegStatusInfo Struct Reference . . . 86
8.82.1 Detailed Description . . . 86
8.82.2 Field Documentation . . . 86
8.82.2.1 ImsRegStatus . . . 86
8.82.2.2 TlvPresent . . . 86

8.83 imsa_RatHandoverStatusInfo Struct Reference . . . 86
8.83.1 Detailed Description . . . 87
8.83.2 Field Documentation . . . 87
8.83.2.1 ErrorCodeData . . . 87
8.83.2.2 ErrorCodeLen . . . 87
8.83.2.3 RatHandoverStatus . . . 87
8.83.2.4 SourceRAT . . . 87
8.83.2.5 TargetRAT . . . 87
8.83.2.6 TlvPresent . . . 87
8.84 imsa_SmsRatInfo Struct Reference . . . 87
8.84.1 Detailed Description . . . 88
8.84.2 Field Documentation . . . 88
8.84.2.1 SmsRatVal . . . 88
8.84.2.2 TlvPresent . . . 88
8.85 imsa_SmsSvcStatusInfo Struct Reference . . . 88
8.85.1 Detailed Description . . . 88
8.85.2 Field Documentation . . . 88
8.85.2.1 SmsSvcStatus . . . 88
8.85.2.2 TlvPresent . . . 88
8.86 imsa_UtRatInfo Struct Reference . . . 88
8.86.1 Detailed Description . . . 89
8.86.2 Field Documentation . . . 89
8.86.2.1 TlvPresent . . . 89
8.86.2.2 UtRatVal . . . 89
8.87 imsa_UtSvcStatusInfo Struct Reference . . . 89
8.87.1 Detailed Description . . . 89
8.87.2 Field Documentation . . . 89
8.87.2.1 TlvPresent . . . 89
8.87.2.2 UtSvcStatus . . . 89
8.88 imsa_VoipRatInfo Struct Reference . . . 90
8.88.1 Detailed Description . . . 90
8.88.2 Field Documentation . . . 90
8.88.2.1 TlvPresent . . . 90
8.88.2.2 VoipRatVal . . . 90
8.89 imsa_VoipSvcStatusInfo Struct Reference . . . 90
8.89.1 Detailed Description . . . 90
8.89.2 Field Documentation . . . 90
8.89.2.1 TlvPresent . . . 90
8.89.2.2 VoipSvcStatus . . . 91
8.90 imsa_VtRatInfo Struct Reference . . . 91

8.90.1 Detailed Description . . . 91
8.90.2 Field Documentation . . . 91
8.90.2.1 TlvPresent . . . 91
8.90.2.2 VtRatVal . . . 91
8.91 imsa_VtSvcStatusInfo Struct Reference . . . 91
8.91.1 Detailed Description . . . 91
8.91.2 Field Documentation . . . 92
8.91.2.1 TlvPresent . . . 92
8.91.2.2 VtSvcStatus . . . 92
8.92 ipv6AddressInfo Struct Reference . . . 92
8.92.1 Detailed Description . . . 92
8.92.2 Field Documentation . . . 92
8.92.2.1 IPAddressV6 . . . 92
8.92.2.2 IPV6PrefixLen . . . 92
8.93 LibPackGPRSRequestedQoS Struct Reference . . . 92
8.93.1 Detailed Description . . . 92
8.93.2 Field Documentation . . . 93
8.93.2.1 delayClass . . . 93
8.93.2.2 meanThroughputClass . . . 93
8.93.2.3 peakThroughputClass . . . 93
8.93.2.4 precedenceClass . . . 93
8.93.2.5 reliabilityClass . . . 93
8.94 LibpackProfile3GPP Struct Reference . . . 93
8.94.1 Detailed Description . . . 94
8.94.2 Field Documentation . . . 99
8.94.2.1 pAddrAllocPref . . . 99
8.94.2.2 pAPNClass . . . 99
8.94.2.3 pAPNDisabledFlag . . . 99
8.94.2.4 pAPNName . . . 99
8.94.2.5 pAPNnameSize . . . 99
8.94.2.6 pAuthenticationPref . . . 99
8.94.2.7 pGPRSMinimumQoS . . . 99
8.94.2.8 pGPRSRequestedQos . . . 99
8.94.2.9 pImCnFlag . . . 99
8.94.2.10 pIPv4AddrPref . . . 99
8.94.2.11 pIPv6AddPref . . . 99
8.94.2.12 pPassword . . . 99
8.94.2.13 pPasswordSize . . . 99
8.94.2.14 pPcscfAddrUsingDhcp . . . 99
8.94.2.15 pPcscfAddrUsingPCO . . . 99

8.94.2.16 pPDNInactivTimeout . . . 99
8.94.2.17 pPdpAccessConFlag . . . 99
8.94.2.18 pPdpContext . . . 99
8.94.2.19 pPdpDataCompType . . . 99
8.94.2.20 pPdpHdrCompType . . . 99
8.94.2.21 pPDPtype . . . 99
8.94.2.22 pPriDNSIPv4AddPref . . . 99
8.94.2.23 pPriDNSIPv6addpref . . . 99
8.94.2.24 pPrimaryID . . . 99
8.94.2.25 pProfilename . . . 99
8.94.2.26 pProfilenameSize . . . 99
8.94.2.27 pQosClassID . . . 99
8.94.2.28 pSecDNSIPv4AddPref . . . 100
8.94.2.29 pSecDNSIPv6addpref . . . 100
8.94.2.30 pSecondaryFlag . . . 100
8.94.2.31 pSupportEmergencyCalls . . . 100
8.94.2.32 pTFTID1Params . . . 100
8.94.2.33 pTFTID2Params . . . 100
8.94.2.34 pUMTSMinQoS . . . 100
8.94.2.35 pUMTSMinQosSigInd . . . 100
8.94.2.36 pUMTSReqQoS . . . 100
8.94.2.37 pUMTSReqQoSSigInd . . . 100
8.94.2.38 pUsername . . . 100
8.94.2.39 pUsernameSize . . . 100
8.95 LibpackProfile3GPP2 Struct Reference . . . 100
8.95.1 Detailed Description . . . 101
8.95.2 Field Documentation . . . 105
8.95.2.1 pAllowLinger . . . 105
8.95.2.2 pAPNClass3GPP2 . . . 105
8.95.2.3 pAPNEnabled3GPP2 . . . 105
8.95.2.4 pApnString . . . 105
8.95.2.5 pApnStringSize . . . 105
8.95.2.6 pAppPriority . . . 105
8.95.2.7 pAppType . . . 105
8.95.2.8 pAuthPassword . . . 105
8.95.2.9 pAuthPasswordSize . . . 105
8.95.2.10 pAuthProtocol . . . 105
8.95.2.11 pAuthRetryCount . . . 105
8.95.2.12 pAuthTimeout . . . 105
8.95.2.13 pDataMode . . . 105

8.95.2.14 pDataRate . . . 105
8.95.2.15 pIpcpAckTimeout . . . 105
8.95.2.16 pIpcpCreqRetryCount . . . 105
8.95.2.17 pIsPcscfAddressNedded . . . 105
8.95.2.18 pLcpAckTimeout . . . 105
8.95.2.19 pLcpCreqRetryCount . . . 105
8.95.2.20 pNegoDnsSrvrPref . . . 105
8.95.2.21 pPDNInactivTimeout3GPP2 . . . 105
8.95.2.22 pPdnType . . . 105
8.95.2.23 pPppSessCloseTimer1x . . . 106
8.95.2.24 pPppSessCloseTimerDO . . . 106
8.95.2.25 pPrimaryV4DnsAddress . . . 106
8.95.2.26 pPriV6DnsAddress . . . 106
8.95.2.27 pRATType . . . 106
8.95.2.28 pSecondaryV4DnsAddress . . . 106
8.95.2.29 pSecV6DnsAddress . . . 106
8.95.2.30 pUserId . . . 106
8.95.2.31 pUserIdSize . . . 106
8.96 LibPackprofile_3GPP Struct Reference . . . 106
8.96.1 Detailed Description . . . 107
8.96.2 Field Documentation . . . 110
8.96.2.1 pAddrAllocPref . . . 110
8.96.2.2 pAPNClass . . . 110
8.96.2.3 pAPNDisabledFlag . . . 110
8.96.2.4 pAPNName . . . 111
8.96.2.5 pAPNnameSize . . . 111
8.96.2.6 pAuthenticationPref . . . 111
8.96.2.7 pGPRSMinimumQoS . . . 111
8.96.2.8 pGPRSRequestedQos . . . 111
8.96.2.9 pImCnFlag . . . 111
8.96.2.10 pIPv4AddrPref . . . 111
8.96.2.11 pIPv6AddPref . . . 111
8.96.2.12 pPassword . . . 111
8.96.2.13 pPasswordSize . . . 111
8.96.2.14 pPcscfAddrUsingDhcp . . . 111
8.96.2.15 pPcscfAddrUsingPCO . . . 111
8.96.2.16 pPDNInactivTimeout . . . 111
8.96.2.17 pPdpAccessConFlag . . . 111
8.96.2.18 pPdpContext . . . 111
8.96.2.19 pPdpDataCompType . . . 111

8.96.2.20 pPdpHdrCompType . . . 111
8.96.2.21 pPDPtype . . . 111
8.96.2.22 pPriDNSIPv4AddPref . . . 111
8.96.2.23 pPriDNSIPv6addpref . . . 111
8.96.2.24 pPrimaryID . . . 111
8.96.2.25 pProfilename . . . 111
8.96.2.26 pProfilenameSize . . . 111
8.96.2.27 pQosClassID . . . 111
8.96.2.28 pSecDNSIPv4AddPref . . . 111
8.96.2.29 pSecDNSIPv6addpref . . . 111
8.96.2.30 pSecondaryFlag . . . 111
8.96.2.31 pSupportEmergencyCalls . . . 111
8.96.2.32 pTFTID1Params . . . 112
8.96.2.33 pTFTID2Params . . . 112
8.96.2.34 pUMTSMinQoS . . . 112
8.96.2.35 pUMTSMinQosSigInd . . . 112
8.96.2.36 pUMTSReqQoS . . . 112
8.96.2.37 pUMTSReqQoSSigInd . . . 112
8.96.2.38 pUsername . . . 112
8.96.2.39 pUsernameSize . . . 112
8.97 LibPackprofile_3GPP2 Struct Reference . . . 112
8.97.1 Detailed Description . . . 113
8.97.2 Field Documentation . . . 116
8.97.2.1 pAllowLinger . . . 116
8.97.2.2 pAPNClass3GPP2 . . . 116
8.97.2.3 pAPNEnabled3GPP2 . . . 116
8.97.2.4 pApnString . . . 116
8.97.2.5 pApnStringSize . . . 116
8.97.2.6 pAppPriority . . . 116
8.97.2.7 pAppType . . . 116
8.97.2.8 pAuthPassword . . . 116
8.97.2.9 pAuthPassword_tSize . . . 116
8.97.2.10 pAuthProtocol . . . 116
8.97.2.11 pAuthRetryCount . . . 116
8.97.2.12 pAuthTimeout . . . 116
8.97.2.13 pDataMode . . . 116
8.97.2.14 pDataRate . . . 116
8.97.2.15 pIpcpAckTimeout . . . 117
8.97.2.16 pIpcpCreqRetryCount . . . 117
8.97.2.17 pIsPcscfAddressNedded . . . 117

8.97.2.18 pLcpAckTimeout . . . 117
8.97.2.19 pLcpCreqRetryCount . . . 117
8.97.2.20 pNegoDnsSrvrPref . . . 117
8.97.2.21 pPDNInactivTimeout3GPP2 . . . 117
8.97.2.22 pPdnType . . . 117
8.97.2.23 pPppSessCloseTimer1x . . . 117
8.97.2.24 pPppSessCloseTimerDO . . . 117
8.97.2.25 pPrimaryV4DnsAddress . . . 117
8.97.2.26 pPriV6DnsAddress . . . 117
8.97.2.27 pRATType . . . 117
8.97.2.28 pSecondaryV4DnsAddress . . . 117
8.97.2.29 pSecV6DnsAddress . . . 117
8.97.2.30 pUserId . . . 117
8.97.2.31 pUserIdSize . . . 117
8.98 LibPackQosClassID Struct Reference . . . 117
8.98.1 Detailed Description . . . 117
8.98.2 Field Documentation . . . 118
8.98.2.1 gDlBitRate . . . 118
8.98.2.2 gUlBitRate . . . 118
8.98.2.3 maxDlBitRate . . . 118
8.98.2.4 maxUlBitRate . . . 118
8.98.2.5 QCI . . . 118
8.99 LibPackTFTIDParams Struct Reference . . . 118
8.99.1 Detailed Description . . . 118
8.99.2 Field Documentation . . . 119
8.99.2.1 destPortRangeEnd . . . 119
8.99.2.2 destPortRangeStart . . . 119
8.99.2.3 eValid . . . 120
8.99.2.4 filterId . . . 120
8.99.2.5 flowLabel . . . 120
8.99.2.6 IPSECSPI . . . 120
8.99.2.7 ipVersion . . . 120
8.99.2.8 nextHeader . . . 120
8.99.2.9 pSourceIP . . . 120
8.99.2.10 sourceIPMask . . . 120
8.99.2.11 srcPortRangeEnd . . . 120
8.99.2.12 srcPortRangeStart . . . 120
8.99.2.13 tosMask . . . 120
8.100 LibPackUMTSQoS Struct Reference . . . 120
8.100.1 Detailed Description . . . 120

8.100.2 Field Documentation . . . 122
8.100.2.1 deliveryErrSDU . . . 122
8.100.2.2 grntDownlinkBitrate . . . 122
8.100.2.3 grntUplinkBitrate . . . 122
8.100.2.4 maxDownlinkBitrate . . . 122
8.100.2.5 maxSDUSize . . . 122
8.100.2.6 maxUplinkBitrate . . . 122
8.100.2.7 qosDeliveryOrder . . . 122
8.100.2.8 resBerRatio . . . 122
8.100.2.9 sduErrorRatio . . . 122
8.100.2.10 trafficClass . . . 122
8.100.2.11 trafficPriority . . . 122
8.100.2.12 transferDelay . . . 122
8.101 LibPackUMTSReqQoSSigInd Struct Reference . . . 123
8.101.1 Detailed Description . . . 123
8.101.2 Field Documentation . . . 123
8.101.2.1 SigInd . . . 123
8.101.2.2 UMTSReqQoS . . . 123
8.102 loc_accelAcceptReady Struct Reference . . . 123
8.102.1 Detailed Description . . . 123
8.102.2 Field Documentation . . . 124
8.102.2.1 batchPerSec . . . 124
8.102.2.2 injectEnable . . . 124
8.102.2.3 samplesPerBatch . . . 124
8.103 loc_accelTempAcceptReady Struct Reference . . . 124
8.103.1 Detailed Description . . . 124
8.103.2 Field Documentation . . . 125
8.103.2.1 batchPerSec . . . 125
8.103.2.2 injectEnable . . . 125
8.103.2.3 samplesPerBatch . . . 125
8.104 loc_BdsSV Struct Reference . . . 125
8.104.1 Detailed Description . . . 125
8.104.2 Field Documentation . . . 125
8.104.2.1 id . . . 125
8.104.2.2 mask . . . 125
8.105 loc_BdsSVInfo Struct Reference . . . 125
8.105.1 Detailed Description . . . 125
8.105.2 Field Documentation . . . 126
8.105.2.1 len . . . 126
8.105.2.2 pSV . . . 126

8.106loc_CellDb Struct Reference . . . 126
8.106.1 Detailed Description . . . 126
8.106.2 Field Documentation . . . 126
8.106.2.1 mask . . . 126
8.107loc_ClkInfo Struct Reference . . . 126
8.107.1 Detailed Description . . . 127
8.107.2 Field Documentation . . . 127
8.107.2.1 mask . . . 128
8.108loc_GnssData Struct Reference . . . 128
8.108.1 Detailed Description . . . 128
8.108.2 Field Documentation . . . 129
8.108.2.1 mask . . . 129
8.109loc_gpsTime Struct Reference . . . 129
8.109.1 Detailed Description . . . 130
8.109.2 Field Documentation . . . 130
8.109.2.1 gpsTimeOfWeekMs . . . 130
8.109.2.2 gpsWeek . . . 130
8.110loc_gyroAcceptReady Struct Reference . . . 130
8.110.1 Detailed Description . . . 130
8.110.2 Field Documentation . . . 131
8.110.2.1 batchPerSec . . . 131
8.110.2.2 injectEnable . . . 131
8.110.2.3 samplesPerBatch . . . 131
8.111loc_gyroTempAcceptReady Struct Reference . . . 131
8.111.1 Detailed Description . . . 131
8.111.2 Field Documentation . . . 131
8.111.2.1 batchPerSec . . . 131
8.111.2.2 injectEnable . . . 131
8.111.2.3 samplesPerBatch . . . 132
8.112loc_IPv4Config Struct Reference . . . 132
8.112.1 Detailed Description . . . 132
8.112.2 Field Documentation . . . 132
8.112.2.1 IPv4Addr . . . 132
8.112.2.2 IPv4Port . . . 132
8.113loc_IPv4Info Struct Reference . . . 132
8.113.1 Detailed Description . . . 132
8.113.2 Field Documentation . . . 133
8.113.2.1 address . . . 133
8.113.2.2 port . . . 133
8.113.2.3 TlvPresent . . . 133

8.114loc_IPv6Config Struct Reference . . . 133
8.114.1 Detailed Description . . . 133
8.114.2 Field Documentation . . . 133
8.114.2.1 IPv6Addr . . . 133
8.114.2.2 IPv6Port . . . 133
8.115loc_IPv6Info Struct Reference . . . 133
8.115.1 Detailed Description . . . 133
8.115.2 Field Documentation . . . 134
8.115.2.1 address . . . 134
8.115.2.2 port . . . 134
8.115.2.3 TlvPresent . . . 134
8.116loc_LocApplicationInfo Struct Reference . . . 134
8.116.1 Detailed Description . . . 134
8.116.2 Field Documentation . . . 135
8.116.2.1 appNameLength . . . 135
8.116.2.2 appProviderLength . . . 135
8.116.2.3 appVersionLength . . . 135
8.116.2.4 appVersionValid . . . 135
8.116.2.5 pAppName . . . 135
8.116.2.6 pAppProvider . . . 135
8.116.2.7 pAppVersion . . . 135
8.117loc_precisionDilution Struct Reference . . . 135
8.117.1 Detailed Description . . . 135
8.117.2 Field Documentation . . . 136
8.117.2.1 HDOP . . . 136
8.117.2.2 PDOP . . . 136
8.117.2.3 VDOP . . . 136
8.118loc_satelliteInfo Struct Reference . . . 136
8.118.1 Detailed Description . . . 136
8.118.2 Field Documentation . . . 138
8.118.2.1 azimuth . . . 138
8.118.2.2 elevation . . . 138
8.118.2.3 gnssSvId . . . 138
8.118.2.4 healthStatus . . . 138
8.118.2.5 snr . . . 138
8.118.2.6 svInfoMask . . . 138
8.118.2.7 svListLen . . . 138
8.118.2.8 svStatus . . . 138
8.118.2.9 system . . . 138
8.118.2.10validMask . . . 138

8.119loc_sensorDataUsage Struct Reference . . . 138
8.119.1 Detailed Description . . . 138
8.119.2 Field Documentation . . . 139
8.119.2.1 aidingIndicatorMask . . . 139
8.119.2.2 usageMask . . . 139
8.120loc_SV Struct Reference . . . 139
8.120.1 Detailed Description . . . 139
8.120.2 Field Documentation . . . 140
8.120.2.1 id . . . 140
8.120.2.2 mask . . . 140
8.120.2.3 system . . . 140
8.121loc_SVInfo Struct Reference . . . 140
8.121.1 Detailed Description . . . 140
8.121.2 Field Documentation . . . 140
8.121.2.1 len . . . 140
8.121.2.2 pSV . . . 140
8.122loc_svUsedforFix Struct Reference . . . 140
8.122.1 Detailed Description . . . 140
8.122.2 Field Documentation . . . 141
8.122.2.1 gnssSvUsedList . . . 141
8.122.2.2 gnssSvUsedList_len . . . 141
8.123loc_urlAddr Struct Reference . . . 141
8.123.1 Detailed Description . . . 141
8.123.2 Field Documentation . . . 141
8.123.2.1 address . . . 141
8.123.2.2 TlvPresent . . . 141
8.124loc_URLAddrInfo Struct Reference . . . 141
8.124.1 Detailed Description . . . 142
8.124.2 Field Documentation . . . 142
8.124.2.1 urlAddr . . . 142
8.125lteSSInfo Struct Reference . . . 142
8.125.1 Detailed Description . . . 142
8.125.2 Field Documentation . . . 142
8.125.2.1 rsrp . . . 143
8.125.2.2 rsrq . . . 143
8.125.2.3 rssi . . . 143
8.125.2.4 snr . . . 143
8.126messageModeTlv Struct Reference . . . 143
8.126.1 Detailed Description . . . 143
8.126.2 Field Documentation . . . 143

8.126.2.1 MessageModeInfo . . . 143
8.126.2.2 TlvPresent . . . 143
8.127 nas_acqOrderPref Struct Reference . . . 143
8.127.1 Detailed Description . . . 143
8.127.2 Field Documentation . . . 144
8.127.2.1 acqOrdeLen . . . 144
8.127.2.2 pAcqOrder . . . 144
8.128 nas_AcqOrderPrefTlv Struct Reference . . . 144
8.128.1 Detailed Description . . . 144
8.128.2 Field Documentation . . . 144
8.128.2.1 acqOrdeLen . . . 144
8.128.2.2 pAcqOrder . . . 145
8.128.2.3 TlvPresent . . . 145
8.129 nas_ActPilotPNElement Struct Reference . . . 145
8.129.1 Detailed Description . . . 145
8.129.2 Field Documentation . . . 145
8.129.2.1 ActSetPilotPN . . . 145
8.129.2.2 ActSetPilotPNStrength . . . 145
8.130 nas_AddCDMASysInfo Struct Reference . . . 145
8.130.1 Detailed Description . . . 145
8.130.2 Field Documentation . . . 146
8.130.2.1 geoSysIdx . . . 146
8.130.2.2 regPrd . . . 146
8.131 nas_AddSysInfo Struct Reference . . . 146
8.131.1 Detailed Description . . . 146
8.131.2 Field Documentation . . . 146
8.131.2.1 cellBroadcastCap . . . 146
8.131.2.2 geoSysIdx . . . 146
8.132 nas_BandPrefInfoTlv Struct Reference . . . 146
8.132.1 Detailed Description . . . 146
8.132.2 Field Documentation . . . 147
8.132.2.1 bits_129_192 . . . 147
8.132.2.2 bits_193_256 . . . 147
8.132.2.3 bits_1_64 . . . 147
8.132.2.4 bits_65_128 . . . 147
8.132.2.5 TlvPresent . . . 147
8.133 nas_BandPrefTlv Struct Reference . . . 147
8.133.1 Detailed Description . . . 147
8.133.2 Field Documentation . . . 149
8.133.2.1 BandPref . . . 149

8.133.2.2 TlvPresent . . . 149
8.134 nas_CallBarringSysInfo Struct Reference . . . 149
8.134.1 Detailed Description . . . 149
8.134.2 Field Documentation . . . 149
8.134.2.1 csBarStatus . . . 149
8.134.2.2 psBarStatus . . . 149
8.135 nas_callBarStatus Struct Reference . . . 149
8.135.1 Detailed Description . . . 150
8.135.2 Field Documentation . . . 150
8.135.2.1 csBarStatus . . . 150
8.135.2.2 psBarStatus . . . 150
8.136 nas_CDMAChannel Struct Reference . . . 150
8.136.1 Detailed Description . . . 150
8.136.2 Field Documentation . . . 151
8.136.2.1 priChA . . . 151
8.136.2.2 priChB . . . 151
8.136.2.3 secChA . . . 151
8.136.2.4 secChB . . . 151
8.137 nas_CDMAECIOThresh Struct Reference . . . 151
8.137.1 Detailed Description . . . 151
8.137.2 Field Documentation . . . 151
8.137.2.1 CDMAECIOThreshListLen . . . 151
8.137.2.2 pCDMAECIOThreshList . . . 152
8.138 nas_CDMAInfo Struct Reference . . . 152
8.138.1 Detailed Description . . . 152
8.138.2 Field Documentation . . . 152
8.138.2.1 baseId . . . 152
8.138.2.2 baseLat . . . 152
8.138.2.3 baseLong . . . 152
8.138.2.4 nid . . . 153
8.138.2.5 refpn . . . 153
8.138.2.6 sid . . . 153
8.139 nas_CDMARSSIThresh Struct Reference . . . 153
8.139.1 Detailed Description . . . 153
8.139.2 Field Documentation . . . 153
8.139.2.1 CDMARSSIThreshListLen . . . 153
8.139.2.2 pCDMARSSIThreshList . . . 153
8.140 nas_CDMASysInfo Struct Reference . . . 153
8.140.1 Detailed Description . . . 154
8.140.2 Field Documentation . . . 156

8.140.2.1 baseId . . . 156
8.140.2.2 baseLat . . . 156
8.140.2.3 baseLong . . . 156
8.140.2.4 bsInfoValid . . . 156
8.140.2.5 bsPRev . . . 156
8.140.2.6 bsPRevValid . . . 156
8.140.2.7 ccsSupported . . . 156
8.140.2.8 ccsSupportedValid . . . 156
8.140.2.9 cdmaSysIdValid . . . 156
8.140.2.10 isSysPrlMatch . . . 156
8.140.2.11 isSysPrlMatchValid . . . 156
8.140.2.12 MCC . . . 156
8.140.2.13 MNC . . . 156
8.140.2.14 networkID . . . 156
8.140.2.15 networkIdValid . . . 157
8.140.2.16 packetZone . . . 157
8.140.2.17 packetZoneValid . . . 157
8.140.2.18 pRevInUse . . . 157
8.140.2.19 pRevInUseValid . . . 157
8.140.2.20 sysInfoCDMA . . . 157
8.140.2.21 systemID . . . 157
8.141 nas_CDMASysInfoExt Struct Reference . . . 157
8.141.1 Detailed Description . . . 157
8.141.2 Field Documentation . . . 157
8.141.2.1 imsi_11_12 . . . 157
8.141.2.2 MCC . . . 157
8.142 nas_cellParams Struct Reference . . . 157
8.142.1 Detailed Description . . . 158
8.142.2 Field Documentation . . . 158
8.142.2.1 pci . . . 158
8.142.2.2 rsrp . . . 158
8.142.2.3 rsrq . . . 158
8.142.2.4 rssi . . . 158
8.142.2.5 srxlev . . . 158
8.143 nas_ciotAcqOrderPref Struct Reference . . . 158
8.143.1 Detailed Description . . . 158
8.143.2 Field Documentation . . . 159
8.143.2.1 ciotAcqOrderLen . . . 159
8.143.2.2 pCiotAcqOrder . . . 159
8.144 nas_CiotAcqOrderPrefTlv Struct Reference . . . 159

8.144.1 Detailed Description . . . 159
8.144.2 Field Documentation . . . 160
8.144.2.1 ciotAcqOrderLen . . . 160
8.144.2.2 pCiotAcqOrder . . . 160
8.144.2.3 TlvPresent . . . 160
8.145nas_CiotLteOpModePrefTlv Struct Reference . . . 160
8.145.1 Detailed Description . . . 160
8.145.2 Field Documentation . . . 160
8.145.2.1 ciotLteOpModePref . . . 160
8.145.2.2 TlvPresent . . . 160
8.146nas_CommInfo Struct Reference . . . 160
8.146.1 Detailed Description . . . 161
8.146.2 Field Documentation . . . 162
8.146.2.1 imsRegState . . . 162
8.146.2.2 modemMode . . . 162
8.146.2.3 psState . . . 162
8.146.2.4 systemMode . . . 162
8.146.2.5 temperature . . . 162
8.147nas_CSGID Struct Reference . . . 162
8.147.1 Detailed Description . . . 162
8.147.2 Field Documentation . . . 163
8.147.2.1 id . . . 163
8.147.2.2 mcc . . . 163
8.147.2.3 mnc . . . 163
8.147.2.4 mncPcsDigits . . . 163
8.147.2.5 rat . . . 163
8.148nas_CsgId Struct Reference . . . 163
8.148.1 Detailed Description . . . 163
8.148.2 Field Documentation . . . 163
8.148.2.1 csgId . . . 163
8.148.2.2 TlvPresent . . . 163
8.149nas_currentPLMN Struct Reference . . . 163
8.149.1 Detailed Description . . . 164
8.149.2 Field Documentation . . . 164
8.149.2.1 MCC . . . 164
8.149.2.2 MNC . . . 164
8.149.2.3 netDescr . . . 164
8.149.2.4 netDescrLength . . . 164
8.150nas_dataSrvCapabilities Struct Reference . . . 164
8.150.1 Detailed Description . . . 164

8.150.2 Field Documentation . . . 165
8.150.2.1 dataCapabilities . . . 165
8.150.2.2 dataCapabilitiesLen . . . 165
8.151 nas_DataStatusDetail Struct Reference . . . 165
8.151.1 Detailed Description . . . 165
8.151.2 Field Documentation . . . 166
8.151.2.1 IPAddress . . . 166
8.151.2.2 LastErrCode . . . 166
8.152 nas_detailSvcInfo Struct Reference . . . 167
8.152.1 Detailed Description . . . 167
8.152.2 Field Documentation . . . 168
8.152.2.1 hdrHybrid . . . 168
8.152.2.2 hdrSrvStatus . . . 168
8.152.2.3 isSysForbidden . . . 168
8.152.2.4 srvCapability . . . 168
8.152.2.5 srvStatus . . . 168
8.153 nas_DeviceConfigDetail Struct Reference . . . 168
8.153.1 Detailed Description . . . 168
8.153.2 Field Documentation . . . 169
8.153.2.1 Chipset . . . 169
8.153.2.2 HWVersion . . . 169
8.153.2.3 QLIC . . . 169
8.153.2.4 Technology . . . 169
8.154 nas_dirNum Struct Reference . . . 169
8.154.1 Detailed Description . . . 169
8.154.2 Field Documentation . . . 169
8.154.2.1 dirNum . . . 169
8.154.2.2 dirNumLen . . . 169
8.155 nas_DRCParams Struct Reference . . . 169
8.155.1 Detailed Description . . . 170
8.155.2 Field Documentation . . . 170
8.155.2.1 DRCCover . . . 170
8.155.2.2 DRCValue . . . 170
8.156 nas_ecioListElement Struct Reference . . . 170
8.156.1 Detailed Description . . . 170
8.156.2 Field Documentation . . . 170
8.156.2.1 ecio . . . 171
8.156.2.2 radioIf . . . 171
8.157 nas_ECIOThresh Struct Reference . . . 171
8.157.1 Detailed Description . . . 171

8.157.2 Field Documentation . . . 171
8.157.2.1 ECIOThresListLen . . . 171
8.157.2.2 pECIOThresList . . . 171
8.158 nas_EdrxCiotLteMode Struct Reference . . . 171
8.158.1 Detailed Description . . . 171
8.158.2 Field Documentation . . . 172
8.158.2.1 lteOpMode . . . 172
8.158.2.2 TlvPresent . . . 172
8.159 nas_EdrxCycleLength Struct Reference . . . 172
8.159.1 Detailed Description . . . 172
8.159.2 Field Documentation . . . 172
8.159.2.1 cycleLength . . . 172
8.159.2.2 TlvPresent . . . 172
8.160 nas_EdrxEnableType Struct Reference . . . 172
8.160.1 Detailed Description . . . 173
8.160.2 Field Documentation . . . 173
8.160.2.1 edrxEnabled . . . 173
8.160.2.2 TlvPresent . . . 173
8.161 nas_EdrxPagingTimeWindow Struct Reference . . . 173
8.161.1 Detailed Description . . . 173
8.161.2 Field Documentation . . . 173
8.161.2.1 edrxPtw . . . 173
8.161.2.2 TlvPresent . . . 173
8.162 nas_EdrxRatType Struct Reference . . . 173
8.162.1 Detailed Description . . . 174
8.162.2 Field Documentation . . . 174
8.162.2.1 edrxRatType . . . 174
8.162.2.2 TlvPresent . . . 174
8.163 nas_EmerModeTlv Struct Reference . . . 174
8.163.1 Detailed Description . . . 174
8.163.2 Field Documentation . . . 175
8.163.2.1 EmerMode . . . 175
8.163.2.2 TlvPresent . . . 175
8.164 nas_errorRateListElement Struct Reference . . . 175
8.164.1 Detailed Description . . . 175
8.164.2 Field Documentation . . . 176
8.164.2.1 errorRate . . . 176
8.164.2.2 radioIf . . . 176
8.165 nas_ForbiddenNetworks3GPP Struct Reference . . . 176
8.165.1 Detailed Description . . . 176

8.165.2 Field Documentation . . . 177
8.165.2.1 forbiddenNwInstLen . . . 177
8.165.2.2 MCC . . . 177
8.165.2.3 MNC . . . 177
8.165.2.4 TlvPresent . . . 177
8.166 nas_GERANInfo Struct Reference . . . 177
8.166.1 Detailed Description . . . 177
8.166.2 Field Documentation . . . 178
8.166.2.1 arfcn . . . 178
8.166.2.2 bsic . . . 178
8.166.2.3 cellID . . . 178
8.166.2.4 insNmrCellInfo . . . 178
8.166.2.5 lac . . . 178
8.166.2.6 nmrInst . . . 178
8.166.2.7 plmn . . . 178
8.166.2.8 rxLev . . . 178
8.166.2.9 timingAdvance . . . 178
8.167 nas_geranInstInfo Struct Reference . . . 178
8.167.1 Detailed Description . . . 179
8.167.2 Field Documentation . . . 179
8.167.2.1 geranArfcn . . . 179
8.167.2.2 geranBsicBcc . . . 179
8.167.2.3 geranBsicNcc . . . 179
8.167.2.4 geranRssi . . . 179
8.168 nas_gsmCellInfo Struct Reference . . . 179
8.168.1 Detailed Description . . . 179
8.168.2 Field Documentation . . . 180
8.168.2.1 arfcn . . . 180
8.168.2.2 band1900 . . . 180
8.168.2.3 bsicId . . . 180
8.168.2.4 cellIdValid . . . 180
8.168.2.5 rssi . . . 180
8.168.2.6 srxlev . . . 180
8.169 nas_GSMRSSIThresh Struct Reference . . . 180
8.169.1 Detailed Description . . . 180
8.169.2 Field Documentation . . . 181
8.169.2.1 GSMRSSIThreshListLen . . . 181
8.169.2.2 pGSMRSSIThreshList . . . 181
8.170 nas_GSMSrvStatusInfo Struct Reference . . . 181
8.170.1 Detailed Description . . . 181

8.170.2 Field Documentation . . . 182
8.170.2.1 isPrefDataPath . . . 182
8.170.2.2 srvStatus . . . 182
8.170.2.3 trueSrvStatus . . . 182
8.171 nas_GSMSysInfo Struct Reference . . . 182
8.171.1 Detailed Description . . . 182
8.171.2 Field Documentation . . . 184
8.171.2.1 cellId . . . 184
8.171.2.2 cellIdValid . . . 184
8.171.2.3 dtmSupp . . . 184
8.171.2.4 dtmSuppValid . . . 184
8.171.2.5 egprsSupp . . . 184
8.171.2.6 egprsSuppValid . . . 184
8.171.2.7 lac . . . 184
8.171.2.8 lacValid . . . 184
8.171.2.9 MCC . . . 184
8.171.2.10 MNC . . . 184
8.171.2.11 networkIdValid . . . 184
8.171.2.12 regRejectInfoValid . . . 184
8.171.2.13 rejCause . . . 185
8.171.2.14 rejectSrvDomain . . . 185
8.171.2.15 sysInfoGSM . . . 185
8.172 nas_GWAcqOrderPrefTlv Struct Reference . . . 185
8.172.1 Detailed Description . . . 185
8.172.2 Field Documentation . . . 185
8.172.2.1 GWAcqOrderPref . . . 185
8.172.2.2 TlvPresent . . . 185
8.173 nas_HDRECIOThresh Struct Reference . . . 185
8.173.1 Detailed Description . . . 185
8.173.2 Field Documentation . . . 186
8.173.2.1 HDRECIOThreshListLen . . . 186
8.173.2.2 pHDRECIOThreshList . . . 186
8.174 nas_HDRIOThresh Struct Reference . . . 186
8.174.1 Detailed Description . . . 186
8.174.2 Field Documentation . . . 186
8.174.2.1 HDRIOThreshListLen . . . 186
8.174.2.2 pHDRIOThreshList . . . 186
8.175 nas_HDRPersonality_Ind_Data Struct Reference . . . 186
8.175.1 Detailed Description . . . 187
8.175.2 Field Documentation . . . 187

8.175.2.1 pCurrentPersonality . . . 187
8.175.2.2 pPersonalityListLength . . . 187
8.175.2.3 pProtocolSubtypeElement . . . 187
8.176 nas_HDRRSSIThresh Struct Reference . . . 187
8.176.1 Detailed Description . . . 187
8.176.2 Field Documentation . . . 187
8.176.2.1 HDRRSSIThreshListLen . . . 187
8.176.2.2 pHDRRSSIThreshList . . . 188
8.177 nas_HDRSINRThresh Struct Reference . . . 188
8.177.1 Detailed Description . . . 188
8.177.2 Field Documentation . . . 188
8.177.2.1 HDRSINRThresListLen . . . 188
8.177.2.2 pHDRSINRThresList . . . 188
8.178 nas_HDRSINRThreshold Struct Reference . . . 188
8.178.1 Detailed Description . . . 188
8.178.2 Field Documentation . . . 189
8.178.2.1 HDRSINRThreshListLen . . . 189
8.178.2.2 pHDRSINRThreshList . . . 189
8.179 nas_HDRSysInfo Struct Reference . . . 189
8.179.1 Detailed Description . . . 189
8.179.2 Field Documentation . . . 191
8.179.2.1 hdrActiveProt . . . 191
8.179.2.2 hdrActiveProtValid . . . 191
8.179.2.3 hdrPersonality . . . 191
8.179.2.4 hdrPersonalityValid . . . 191
8.179.2.5 is856SysId . . . 191
8.179.2.6 is856SysIdValid . . . 191
8.179.2.7 isSysPrlMatch . . . 191
8.179.2.8 isSysPrlMatchValid . . . 191
8.179.2.9 sysInfoHDR . . . 191
8.180 nas_homeSIDNID Struct Reference . . . 191
8.180.1 Detailed Description . . . 191
8.180.2 Field Documentation . . . 191
8.180.2.1 numInstances . . . 191
8.180.2.2 SidNid . . . 191
8.181 nas_infoInterFreq Struct Reference . . . 191
8.181.1 Detailed Description . . . 192
8.181.2 Field Documentation . . . 192
8.181.2.1 cell_resel_priority . . . 192
8.181.2.2 cellInterFreqParams . . . 192

8.181.2.3 cells_len . . . 192
8.181.2.4 earfcn . . . 192
8.181.2.5 threshXHigh . . . 193
8.181.2.6 threshXLow . . . 193
8.182 nas_IOThresh Struct Reference . . . 193
8.182.1 Detailed Description . . . 193
8.182.2 Field Documentation . . . 193
8.182.2.1 IOThresListLen . . . 193
8.182.2.2 pIOThresList . . . 193
8.183 nas_lteBandPrefExt Struct Reference . . . 193
8.183.1 Detailed Description . . . 193
8.183.2 Field Documentation . . . 194
8.183.2.1 bits_129_192 . . . 194
8.183.2.2 bits_193_256 . . . 194
8.183.2.3 bits_1_64 . . . 194
8.183.2.4 bits_65_128 . . . 194
8.184 nas_LTEBandPrefTlv Struct Reference . . . 194
8.184.1 Detailed Description . . . 194
8.184.2 Field Documentation . . . 195
8.184.2.1 LTEBandPref . . . 196
8.184.2.2 TlvPresent . . . 196
8.185 nas_LteCiotOpModeTlv Struct Reference . . . 196
8.185.1 Detailed Description . . . 196
8.185.2 Field Documentation . . . 196
8.185.2.1 campedCiotLteOpMode . . . 196
8.185.2.2 TlvPresent . . . 196
8.186 nas_lteEARFCN Struct Reference . . . 196
8.186.1 Detailed Description . . . 196
8.186.2 Field Documentation . . . 197
8.186.2.1 earfcn0 . . . 197
8.186.2.2 earfcn1 . . . 197
8.186.2.3 status . . . 197
8.187 nas_lteGsmCellInfo Struct Reference . . . 197
8.187.1 Detailed Description . . . 197
8.187.2 Field Documentation . . . 198
8.187.2.1 cellReselPriority . . . 198
8.187.2.2 cells_len . . . 198
8.187.2.3 GsmCellInfo . . . 198
8.187.2.4 nccPermitted . . . 198
8.187.2.5 threshGsmHigh . . . 198

8.187.2.6 threshGsmLow . . . 198
8.188 nas_LTEInfo Struct Reference . . . 198
8.188.1 Detailed Description . . . 198
8.188.2 Field Documentation . . . 200
8.188.2.1 band . . . 200
8.188.2.2 bandwidth . . . 200
8.188.2.3 emmConnState . . . 200
8.188.2.4 emmState . . . 200
8.188.2.5 emmSubState . . . 200
8.188.2.6 RXChan . . . 200
8.188.2.7 TXChan . . . 200
8.189 nas_LTEInfoInterfreq Struct Reference . . . 201
8.189.1 Detailed Description . . . 201
8.189.2 Field Documentation . . . 201
8.189.2.1 freqsLen . . . 201
8.189.2.2 InfoInterfreq . . . 201
8.189.2.3 ueInIdle . . . 201
8.190 nas_LTEInfoIntrafreq Struct Reference . . . 201
8.190.1 Detailed Description . . . 202
8.190.2 Field Documentation . . . 203
8.190.2.1 CellParams . . . 203
8.190.2.2 cellReselPriority . . . 203
8.190.2.3 cellsLen . . . 203
8.190.2.4 earfcn . . . 203
8.190.2.5 globalCellId . . . 203
8.190.2.6 plmn . . . 203
8.190.2.7 servingCellId . . . 203
8.190.2.8 sIntraSearch . . . 203
8.190.2.9 sNonIntraSearch . . . 203
8.190.2.10 tac . . . 203
8.190.2.11 threshServingLow . . . 203
8.190.2.12 ueInIdle . . . 203
8.191 nas_LTEInfoNeighboringGSM Struct Reference . . . 203
8.191.1 Detailed Description . . . 204
8.191.2 Field Documentation . . . 204
8.191.2.1 freqsLen . . . 204
8.191.2.2 LteGsmCellInfo . . . 204
8.191.2.3 ueInIdle . . . 204
8.192 nas_LTEInfoNeighboringWCDMA Struct Reference . . . 204
8.192.1 Detailed Description . . . 204

8.192.2 Field Documentation . . . 205
8.192.2.1 freqsLen . . . 205
8.192.2.2 LTEWCDMACellInfo . . . 205
8.192.2.3 ueInIdle . . . 205
8.193 nas_LteM1BandPrefTlv Struct Reference . . . 205
8.193.1 Detailed Description . . . 205
8.193.2 Field Documentation . . . 206
8.193.2.1 lteM1BandPref . . . 207
8.193.2.2 TlvPresent . . . 207
8.194 nas_LteNb1BandPrefTlv Struct Reference . . . 207
8.194.1 Detailed Description . . . 207
8.194.2 Field Documentation . . . 208
8.194.2.1 lteNb1BandPref . . . 209
8.194.2.2 TlvPresent . . . 209
8.195 nas_LTEOperationMode Struct Reference . . . 209
8.195.1 Detailed Description . . . 209
8.195.2 Field Documentation . . . 209
8.195.2.1 pLTEOperationMode . . . 209
8.195.2.2 TlvPresent . . . 209
8.196 nas_LteOpMode Struct Reference . . . 209
8.196.1 Detailed Description . . . 209
8.196.2 Field Documentation . . . 210
8.196.2.1 lteOpMode . . . 210
8.196.2.2 TlvPresent . . . 210
8.197 nas_lteOpModeTlv Struct Reference . . . 210
8.197.1 Detailed Description . . . 210
8.197.2 Field Documentation . . . 211
8.197.2.1 lteOpMode . . . 211
8.197.2.2 lteOpModeLen . . . 211
8.197.2.3 MCC . . . 211
8.197.2.4 MNC . . . 211
8.197.2.5 TlvPresent . . . 211
8.198 nas_ltePCI Struct Reference . . . 211
8.198.1 Detailed Description . . . 211
8.198.2 Field Documentation . . . 211
8.198.2.1 earfcn . . . 211
8.198.2.2 pci . . . 211
8.198.2.3 status . . . 211
8.199 nas_lteRsrpinformation Struct Reference . . . 211
8.199.1 Detailed Description . . . 212

8.199.2 Field Documentation . . . 212
8.199.2.1 rsrplevel . . . 212
8.200 nas_LTERSRPThresh Struct Reference . . . 212
8.200.1 Detailed Description . . . 212
8.200.2 Field Documentation . . . 212
8.200.2.1 LTERSRPThreshListLen . . . 212
8.200.2.2 pLTERSRPThreshList . . . 212
8.201 nas_LTERSRQThresh Struct Reference . . . 212
8.201.1 Detailed Description . . . 213
8.201.2 Field Documentation . . . 213
8.201.2.1 LTERSRQThreshListLen . . . 213
8.201.2.2 pLTERSRQThreshList . . . 213
8.202 nas_LTERSSIThresh Struct Reference . . . 213
8.202.1 Detailed Description . . . 213
8.202.2 Field Documentation . . . 213
8.202.2.1 LTERSSIThreshListLen . . . 213
8.202.2.2 pLTERSSIThreshList . . . 213
8.203 nas_LTESigRptCfg Struct Reference . . . 214
8.203.1 Detailed Description . . . 214
8.203.2 Field Documentation . . . 214
8.203.2.1 avgPeriod . . . 214
8.203.2.2 rptRate . . . 214
8.204 nas_LTESigRptConfig Struct Reference . . . 214
8.204.1 Detailed Description . . . 215
8.204.2 Field Documentation . . . 215
8.204.2.1 avgPeriod . . . 215
8.204.2.2 rptRate . . . 215
8.205 nas_lteSnrinformation Struct Reference . . . 215
8.205.1 Detailed Description . . . 215
8.205.2 Field Documentation . . . 216
8.205.2.1 snrlevel . . . 216
8.206 nas_LTESNRThresh Struct Reference . . . 216
8.206.1 Detailed Description . . . 216
8.206.2 Field Documentation . . . 216
8.206.2.1 LTESNRThresListLen . . . 216
8.206.2.2 pLTESNRThresList . . . 216
8.207 nas_LTESNRThreshold Struct Reference . . . 216
8.207.1 Detailed Description . . . 216
8.207.2 Field Documentation . . . 217
8.207.2.1 LTESNRThreshListLen . . . 217

8.207.2.2 pLTESNRThreshList . . . 217
8.208 nas_LTESysInfo Struct Reference . . . 217
8.208.1 Detailed Description . . . 217
8.208.2 Field Documentation . . . 219
8.208.2.1 cellId . . . 219
8.208.2.2 cellIdValid . . . 219
8.208.2.3 lac . . . 219
8.208.2.4 lacValid . . . 219
8.208.2.5 MCC . . . 219
8.208.2.6 MNC . . . 219
8.208.2.7 networkIdValid . . . 219
8.208.2.8 regRejectInfoValid . . . 219
8.208.2.9 rejCause . . . 219
8.208.2.10 rejectSrvDomain . . . 219
8.208.2.11 sysInfoLTE . . . 219
8.208.2.12 tac . . . 219
8.208.2.13 tacValid . . . 219
8.209 nas_lteWcdmaCellInfo Struct Reference . . . 219
8.209.1 Detailed Description . . . 220
8.209.2 Field Documentation . . . 220
8.209.2.1 cellReselPriority . . . 220
8.209.2.2 cellsLen . . . 220
8.209.2.3 threshXhigh . . . 220
8.209.2.4 threshXlow . . . 220
8.209.2.5 uarfcn . . . 220
8.209.2.6 WCDMACellInfo . . . 220
8.210 nas_minBasedIMSI Struct Reference . . . 220
8.210.1 Detailed Description . . . 221
8.210.2 Field Documentation . . . 221
8.210.2.1 imsiM1112 . . . 221
8.210.2.2 imsiMS1 . . . 221
8.210.2.3 imsiMS2 . . . 221
8.210.2.4 mccM . . . 221
8.211 nas_MNRInfo Struct Reference . . . 221
8.211.1 Detailed Description . . . 221
8.211.2 Field Documentation . . . 222
8.211.2.1 mcc . . . 222
8.211.2.2 mnc . . . 222
8.211.2.3 rat . . . 222
8.212 nas_ModePrefTlv Struct Reference . . . 222

8.212.1 Detailed Description . . . 222
8.212.2 Field Documentation . . . 222
8.212.2.1 ModePref . . . 222
8.212.2.2 TlvPresent . . . 222
8.213nas_namName Struct Reference . . . 222
8.213.1 Detailed Description . . . 223
8.213.2 Field Documentation . . . 223
8.213.2.1 namName . . . 223
8.213.2.2 namNameLen . . . 223
8.214nas_netSelectionPref Struct Reference . . . 223
8.214.1 Detailed Description . . . 223
8.214.2 Field Documentation . . . 224
8.214.2.1 mcc . . . 224
8.214.2.2 mnc . . . 224
8.214.2.3 netReg . . . 224
8.215nas_NetSelPrefTlv Struct Reference . . . 224
8.215.1 Detailed Description . . . 224
8.215.2 Field Documentation . . . 224
8.215.2.1 NetSelPref . . . 224
8.215.2.2 TlvPresent . . . 224
8.216nas_NetworkStat1x Struct Reference . . . 224
8.216.1 Detailed Description . . . 225
8.216.2 Field Documentation . . . 226
8.216.2.1 ActSetCnt . . . 226
8.216.2.2 NeighborSetCnt . . . 226
8.216.2.3 pActPilotPNElements . . . 226
8.216.2.4 pNeighborSetPilotPN . . . 226
8.216.2.5 RX_EC_IO . . . 226
8.216.2.6 RX_PWR . . . 226
8.216.2.7 SO . . . 226
8.216.2.8 State . . . 226
8.216.2.9 TX_PWR . . . 226
8.217nas_NetworkStatEVDO Struct Reference . . . 226
8.217.1 Detailed Description . . . 227
8.217.2 Field Documentation . . . 228
8.217.2.1 MACIndex . . . 228
8.217.2.2 PER . . . 228
8.217.2.3 PilotEnergy . . . 228
8.217.2.4 pSectorID . . . 228
8.217.2.5 RX_PWR . . . 228

8.217.2.6 SectorIDLen . . . 228
8.217.2.7 SNR . . . 228
8.217.2.8 State . . . 228
8.218 nas_nmrCellInfo Struct Reference . . . 228
8.218.1 Detailed Description . . . 228
8.218.2 Field Documentation . . . 229
8.218.2.1 nmrArfcn . . . 229
8.218.2.2 nmrBsic . . . 229
8.218.2.3 nmrCellID . . . 229
8.218.2.4 nmrLac . . . 229
8.218.2.5 nmrPlmn . . . 229
8.218.2.6 nmrRxLev . . . 229
8.219 nas_nr5gBandPref Struct Reference . . . 229
8.219.1 Detailed Description . . . 229
8.219.2 Field Documentation . . . 230
8.219.2.1 bits_129_192 . . . 230
8.219.2.2 bits_193_256 . . . 230
8.219.2.3 bits_1_64 . . . 230
8.219.2.4 bits_65_128 . . . 230
8.220 nas_NR5GCellStatusTlv Struct Reference . . . 230
8.220.1 Detailed Description . . . 230
8.220.2 Field Documentation . . . 230
8.220.2.1 nr5gCellStatus . . . 230
8.220.2.2 TlvPresent . . . 230
8.221 nas_NR5GSerStatTlv Struct Reference . . . 231
8.221.1 Detailed Description . . . 231
8.221.2 Field Documentation . . . 231
8.221.2.1 isPrefDataPath . . . 231
8.221.2.2 srvStatus . . . 231
8.221.2.3 TlvPresent . . . 231
8.221.2.4 trueSrvStatus . . . 231
8.222 nas_NR5GSystemInfoTlv Struct Reference . . . 232
8.222.1 Detailed Description . . . 232
8.222.2 Field Documentation . . . 234
8.222.2.1 cellId . . . 234
8.222.2.2 cellIdValid . . . 234
8.222.2.3 lac . . . 234
8.222.2.4 lacValid . . . 234
8.222.2.5 MCC . . . 234
8.222.2.6 MNC . . . 234

8.222.2.7 nwIdValid . . . 235
8.222.2.8 regRejectInfoValid . . . 235
8.222.2.9 rejCause . . . 235
8.222.2.10 rejectSrvDomain . . . 235
8.222.2.11 roamStatus . . . 235
8.222.2.12 roamStatusValid . . . 235
8.222.2.13 srvcapability . . . 235
8.222.2.14 srvCapValid . . . 235
8.222.2.15 srvDomain . . . 235
8.222.2.16 srvDomainValid . . . 235
8.222.2.17 sysForbidden . . . 235
8.222.2.18 sysForbiddenValid . . . 235
8.222.2.19 tac . . . 235
8.222.2.20 tacValid . . . 235
8.222.2.21 TlvPresent . . . 235
8.223 nas_operatorNameString Struct Reference . . . 235
8.223.1 Detailed Description . . . 235
8.223.2 Field Documentation . . . 235
8.223.2.1 PLMNName . . . 235
8.224 nas_OperatorPLMNData Struct Reference . . . 235
8.224.1 Detailed Description . . . 236
8.224.2 Field Documentation . . . 236
8.224.2.1 lac1 . . . 236
8.224.2.2 lac2 . . . 236
8.224.2.3 mcc . . . 236
8.224.2.4 mnc . . . 236
8.224.2.5 PLMNRecID . . . 236
8.225 nas_operatorPLMNList Struct Reference . . . 236
8.225.1 Detailed Description . . . 237
8.225.2 Field Documentation . . . 237
8.225.2.1 numInstance . . . 237
8.225.2.2 PLMNData . . . 237
8.226 nas_PhyCaAggPcellInfo Struct Reference . . . 237
8.226.1 Detailed Description . . . 237
8.226.2 Field Documentation . . . 238
8.226.2.1 dl_bw_value . . . 238
8.226.2.2 freq . . . 238
8.226.2.3 iLTEbandValue . . . 238
8.226.2.4 pci . . . 238
8.226.2.5 TlvPresent . . . 238

8.227 nas_PhyCaAggScellDlBw Struct Reference . . . 238
8.227.1 Detailed Description . . . 238
8.227.2 Field Documentation . . . 238
8.227.2.1 dl_bw_value . . . 238
8.227.2.2 TlvPresent . . . 238
8.228 nas_PhyCaAggScellIndex Struct Reference . . . 238
8.228.1 Detailed Description . . . 238
8.228.2 Field Documentation . . . 239
8.228.2.1 scell_idx . . . 239
8.228.2.2 TlvPresent . . . 239
8.229 nas_PhyCaAggScellIndType Struct Reference . . . 239
8.229.1 Detailed Description . . . 239
8.229.2 Field Documentation . . . 239
8.229.2.1 freq . . . 239
8.229.2.2 pci . . . 239
8.229.2.3 scell_state . . . 239
8.229.2.4 TlvPresent . . . 239
8.230 nas_PhyCaAggScellInfo Struct Reference . . . 240
8.230.1 Detailed Description . . . 240
8.230.2 Field Documentation . . . 242
8.230.2.1 dl_bw_value . . . 242
8.230.2.2 freq . . . 242
8.230.2.3 iLTEbandValue . . . 242
8.230.2.4 pci . . . 242
8.230.2.5 scell_state . . . 242
8.230.2.6 TlvPresent . . . 242
8.231 nas_PilotSetData Struct Reference . . . 242
8.231.1 Detailed Description . . . 242
8.231.2 Field Documentation . . . 242
8.231.2.1 NumPilots . . . 242
8.231.2.2 pPilotSetInfo . . . 242
8.232 nas_PilotSetParams Struct Reference . . . 243
8.232.1 Detailed Description . . . 243
8.232.2 Field Documentation . . . 243
8.232.2.1 PilotPN . . . 243
8.232.2.2 PilotStrength . . . 243
8.232.2.3 PilotType . . . 243
8.233 nas_PlmnID Struct Reference . . . 243
8.233.1 Detailed Description . . . 243
8.233.2 Field Documentation . . . 244

8.233.2.1 mcc . . . 244
8.233.2.2 mnc . . . 244
8.233.2.3 pcsDigit . . . 244
8.233.2.4 TlvPresent . . . 244
8.234 nas_PLMNNetworkName Struct Reference . . . 244
8.234.1 Detailed Description . . . 244
8.234.2 Field Documentation . . . 244
8.234.2.1 numInstance . . . 244
8.234.2.2 PLMNNetName . . . 244
8.235 nas_PLMNNetworkNameData Struct Reference . . . 244
8.235.1 Detailed Description . . . 245
8.235.2 Field Documentation . . . 246
8.235.2.1 codingScheme . . . 246
8.235.2.2 countryInitials . . . 246
8.235.2.3 longName . . . 246
8.235.2.4 longNameLen . . . 246
8.235.2.5 longNameSpareBits . . . 246
8.235.2.6 shortName . . . 246
8.235.2.7 shortNameLen . . . 246
8.235.2.8 shortNameSpareBits . . . 246
8.236 nas_PRLPrefTlv Struct Reference . . . 246
8.236.1 Detailed Description . . . 246
8.236.2 Field Documentation . . . 247
8.236.2.1 PRLPref . . . 247
8.236.2.2 TlvPresent . . . 247
8.237 nas_protocolSubtypeElement Struct Reference . . . 247
8.237.1 Detailed Description . . . 247
8.237.2 Field Documentation . . . 248
8.237.2.1 AccessMac . . . 248
8.237.2.2 AuthProt . . . 248
8.237.2.3 ControlMac . . . 248
8.237.2.4 EncryptProt . . . 248
8.237.2.5 ForwardMac . . . 248
8.237.2.6 IdleState . . . 248
8.237.2.7 KeyExchange . . . 248
8.237.2.8 MultDisc . . . 248
8.237.2.9 PhysicalLayer . . . 248
8.237.2.10 ReverseMac . . . 248
8.237.2.11 SecProt . . . 248
8.237.2.12 VirtStream . . . 248

8.238 nas_qaQmi3Gpp2TimeZone Struct Reference . . . 248
8.238.1 Detailed Description . . . 249
8.238.2 Field Documentation . . . 249
8.238.2.1 daylightSavings . . . 249
8.238.2.2 leapSeconds . . . 249
8.238.2.3 localTimeOffset . . . 249
8.239 nas_QmiNas3GppNetworkInfo Struct Reference . . . 249
8.239.1 Detailed Description . . . 249
8.239.2 Field Documentation . . . 250
8.239.2.1 Desription . . . 250
8.239.2.2 Forbidden . . . 250
8.239.2.3 InUse . . . 250
8.239.2.4 MCC . . . 250
8.239.2.5 MNC . . . 250
8.239.2.6 Preferred . . . 250
8.239.2.7 Roaming . . . 250
8.240 nas_QmiNas3GppNetworkRAT Struct Reference . . . 250
8.240.1 Detailed Description . . . 251
8.240.2 Field Documentation . . . 251
8.240.2.1 MCC . . . 251
8.240.2.2 MNC . . . 251
8.240.2.3 RAT . . . 251
8.241 nas_QmisNasPcsDigit Struct Reference . . . 251
8.241.1 Detailed Description . . . 251
8.241.2 Field Documentation . . . 252
8.241.2.1 includes_pcs_digit . . . 252
8.241.2.2 MCC . . . 252
8.241.2.3 MNC . . . 252
8.242 nas_QmisNasSlqsNasPCICellInfo Struct Reference . . . 252
8.242.1 Detailed Description . . . 252
8.242.2 Field Documentation . . . 252
8.242.2.1 cellID . . . 252
8.242.2.2 freq . . . 252
8.242.2.3 GlobalCellID . . . 252
8.242.2.4 nasQmisNasPcsDigit . . . 252
8.242.2.5 PlmnLen . . . 252
8.243 nas_QmisNasSlqsNasPCIInfo Struct Reference . . . 253
8.243.1 Detailed Description . . . 253
8.243.2 Field Documentation . . . 253
8.243.2.1 nasQmisNasSlqsNasPCICellInfo . . . 253

8.243.2.2 PCICellInfoLen . . . 253
8.243.2.3 rsrp . . . 253
8.243.2.4 rsrpRx0 . . . 253
8.243.2.5 rsrpRx1 . . . 254
8.243.2.6 rsrq . . . 254
8.243.2.7 rsrqRx0 . . . 254
8.243.2.8 rsrqRx1 . . . 254
8.244 nas_RankIndicatorTlv Struct Reference . . . 254
8.244.1 Detailed Description . . . 254
8.244.2 Field Documentation . . . 254
8.244.2.1 count0 . . . 254
8.244.2.2 count1 . . . 254
8.244.2.3 TlvPresent . . . 254
8.245 nas_RatDisabledMaskTlv Struct Reference . . . 254
8.245.1 Detailed Description . . . 254
8.245.2 Field Documentation . . . 255
8.245.2.1 ratDisabledMask . . . 255
8.245.2.2 TlvPresent . . . 255
8.246 nas_RejectReasonTlv Struct Reference . . . 255
8.246.1 Detailed Description . . . 255
8.246.2 Field Documentation . . . 255
8.246.2.1 rejectCause . . . 255
8.246.2.2 serviceDomain . . . 255
8.246.2.3 TlvPresent . . . 255
8.247 nas_RFBandInfoElements Struct Reference . . . 255
8.247.1 Detailed Description . . . 256
8.247.2 Field Documentation . . . 256
8.247.2.1 activeBandClass . . . 256
8.247.2.2 activeChannel . . . 256
8.247.2.3 radioInterface . . . 256
8.248 nas_RfBandInfoExtFormat Struct Reference . . . 256
8.248.1 Detailed Description . . . 256
8.248.2 Field Documentation . . . 257
8.248.2.1 pInstancesSize . . . 257
8.248.2.2 pRfBandInfoExtFormatParam . . . 257
8.248.2.3 TlvPresent . . . 257
8.249 nas_RfBandInfoExtFormatElements Struct Reference . . . 257
8.249.1 Detailed Description . . . 257
8.249.2 Field Documentation . . . 257
8.249.2.1 activeBand . . . 257

8.249.2.2 activeChannel . . . 257
8.249.2.3 radioInterface . . . 258
8.250 nas_RfBandInfoList Struct Reference . . . 258
8.250.1 Detailed Description . . . 258
8.250.2 Field Documentation . . . 258
8.250.2.1 pInstanceSize . . . 258
8.250.2.2 pRFBandInfoParam . . . 258
8.251 nas_RfBandwidthInfo Struct Reference . . . 258
8.251.1 Detailed Description . . . 258
8.251.2 Field Documentation . . . 259
8.251.2.1 pInstancesSize . . . 259
8.251.2.2 pRfBandwidthInfoParam . . . 259
8.251.2.3 TlvPresent . . . 259
8.252 nas_RfBandwidthInfoElements Struct Reference . . . 259
8.252.1 Detailed Description . . . 259
8.252.2 Field Documentation . . . 259
8.252.2.1 bandwidth . . . 259
8.252.2.2 radioInterface . . . 260
8.253 nas_RfDedicatedBandInfo Struct Reference . . . 260
8.253.1 Detailed Description . . . 260
8.253.2 Field Documentation . . . 260
8.253.2.1 pInstancesSize . . . 260
8.253.2.2 pRfDedicatedBandInfoParam . . . 260
8.253.2.3 TlvPresent . . . 260
8.254 nas_RfDedicatedBandInfoElements Struct Reference . . . 260
8.254.1 Detailed Description . . . 260
8.254.2 Field Documentation . . . 261
8.254.2.1 dedicatedBand . . . 261
8.254.2.2 radioInterface . . . 261
8.255 nas_RFInfoTlv Struct Reference . . . 261
8.255.1 Detailed Description . . . 261
8.255.2 Field Documentation . . . 262
8.255.2.1 activeBandClass . . . 262
8.255.2.2 activeChannel . . . 262
8.255.2.3 radioInterface . . . 262
8.255.2.4 radioInterfaceSize . . . 262
8.255.2.5 TlvPresent . . . 262
8.256 nas_roamIndList Struct Reference . . . 262
8.256.1 Detailed Description . . . 262
8.256.2 Field Documentation . . . 262

8.256.2.1 numInstances . . . 263
8.256.2.2 radioInterface . . . 263
8.256.2.3 roamIndicator . . . 263
8.257 nas_RoamPrefTlv Struct Reference . . . 263
8.257.1 Detailed Description . . . 263
8.257.2 Field Documentation . . . 263
8.257.2.1 RoamPref . . . 263
8.257.2.2 TlvPresent . . . 263
8.258 nas_RSRPThresh Struct Reference . . . 263
8.258.1 Detailed Description . . . 263
8.258.2 Field Documentation . . . 264
8.258.2.1 pRSRPThresList . . . 264
8.258.2.2 RSRPThresListLen . . . 264
8.259 nas_rsrqInformation Struct Reference . . . 264
8.259.1 Detailed Description . . . 264
8.259.2 Field Documentation . . . 264
8.259.2.1 radioIf . . . 264
8.259.2.2 rsrq . . . 264
8.260 nas_RSRQThresh Struct Reference . . . 264
8.260.1 Detailed Description . . . 265
8.260.2 Field Documentation . . . 265
8.260.2.1 pRSRQThresList . . . 265
8.260.2.2 RSRQThresListLen . . . 265
8.261 nas_RSSIThresh Struct Reference . . . 265
8.261.1 Detailed Description . . . 265
8.261.2 Field Documentation . . . 266
8.261.2.1 pRSSIThresList . . . 266
8.261.2.2 RSSIThresListLen . . . 266
8.262 nas_rxInfo Struct Reference . . . 266
8.262.1 Detailed Description . . . 266
8.262.2 Field Documentation . . . 267
8.262.2.1 ecio . . . 267
8.262.2.2 isRadioTuned . . . 267
8.262.2.3 phase . . . 267
8.262.2.4 rscp . . . 267
8.262.2.5 rsrp . . . 267
8.262.2.6 rxPower . . . 267
8.263 nas_RxSigInfo Struct Reference . . . 267
8.263.1 Detailed Description . . . 267
8.263.2 Field Documentation . . . 268

8.263.2.1 isRadioTuned . . . 268
8.263.2.2 rsrp . . . 268
8.263.2.3 rxChainIndex . . . 268
8.263.2.4 rxPower . . . 268
8.264 nas_rxSignalStrengthListElement Struct Reference . . . 268
8.264.1 Detailed Description . . . 268
8.264.2 Field Documentation . . . 269
8.264.2.1 radioIf . . . 269
8.264.2.2 rxSignalStrength . . . 269
8.265 nas_SccRxInfo Struct Reference . . . 269
8.265.1 Detailed Description . . . 269
8.265.2 Field Documentation . . . 270
8.265.2.1 numInstances . . . 270
8.265.2.2 rsrq . . . 270
8.265.2.3 sigInfo . . . 270
8.265.2.4 snr . . . 270
8.265.2.5 TlvPresent . . . 270
8.266 nas_serviceProviderName Struct Reference . . . 270
8.266.1 Detailed Description . . . 270
8.266.2 Field Documentation . . . 270
8.266.2.1 displayCondition . . . 270
8.266.2.2 spn . . . 270
8.266.2.3 spnLength . . . 270
8.267 nas_servSystem Struct Reference . . . 271
8.267.1 Detailed Description . . . 271
8.267.2 Field Documentation . . . 272
8.267.2.1 csAttachState . . . 272
8.267.2.2 numRadioInterfaces . . . 272
8.267.2.3 psAttachState . . . 272
8.267.2.4 radioInterface . . . 272
8.267.2.5 regState . . . 272
8.267.2.6 selNetwork . . . 272
8.268 nas_sidNid Struct Reference . . . 272
8.268.1 Detailed Description . . . 272
8.268.2 Field Documentation . . . 272
8.268.2.1 nid . . . 272
8.268.2.2 sid . . . 272
8.269 nas_SignalStrengthTlv Struct Reference . . . 273
8.269.1 Detailed Description . . . 273
8.269.2 Field Documentation . . . 273

8.269.2.1 radioInterface . . . 273
8.269.2.2 signalStrength . . . 273
8.269.2.3 TlvPresent . . . 273
8.270 nas_SLQSSignalStrengthsIndReq Struct Reference . . . 273
8.270.1 Detailed Description . . . 273
8.270.2 Field Documentation . . . 274
8.270.2.1 ecioDelta . . . 274
8.270.2.2 ecioThresholdList . . . 274
8.270.2.3 ecioThresholdListLen . . . 275
8.270.2.4 ioDelta . . . 275
8.270.2.5 lteRsrpDelta . . . 275
8.270.2.6 lteSnrDelta . . . 275
8.270.2.7 rsrqDelta . . . 275
8.270.2.8 rxSignalStrengthDelta . . . 275
8.270.2.9 sinrDelta . . . 275
8.270.2.10 sinrThresholdList . . . 275
8.270.2.11 sinrThresholdListLen . . . 275
8.271 nas_SLQSSignalStrengthsInformation Struct Reference . . . 275
8.271.1 Detailed Description . . . 275
8.271.2 Field Documentation . . . 276
8.271.2.1 ecioInfo . . . 276
8.271.2.2 errorRateInfo . . . 276
8.271.2.3 io . . . 276
8.271.2.4 lteRsrpinfo . . . 276
8.271.2.5 lteSnrinfo . . . 276
8.271.2.6 rsrqInfo . . . 276
8.271.2.7 rxSignalStrengthInfo . . . 276
8.271.2.8 sinr . . . 276
8.272 nas_SLQSSignalStrengthsTlv Struct Reference . . . 276
8.272.1 Detailed Description . . . 276
8.272.2 Field Documentation . . . 277
8.272.2.1 sSLQSSignalStrengthsInfo . . . 277
8.272.2.2 TlvPresent . . . 277
8.273 nas_SrvDomainPrefTlv Struct Reference . . . 277
8.273.1 Detailed Description . . . 277
8.273.2 Field Documentation . . . 277
8.273.2.1 SrvDomainPref . . . 277
8.273.2.2 TlvPresent . . . 277
8.274 nas_SrvStatusInfo Struct Reference . . . 277
8.274.1 Detailed Description . . . 277

8.274.2 Field Documentation . . . 278
8.274.2.1 isPrefDataPath . . . 278
8.274.2.2 srvStatus . . . 278
8.275 nas_sysInfoCommon Struct Reference . . . 278
8.275.1 Detailed Description . . . 278
8.275.2 Field Documentation . . . 280
8.275.2.1 isSysForbidden . . . 280
8.275.2.2 isSysForbiddenValid . . . 280
8.275.2.3 roamStatus . . . 280
8.275.2.4 roamStatusValid . . . 280
8.275.2.5 srvCapability . . . 280
8.275.2.6 srvCapabilityValid . . . 280
8.275.2.7 srvDomain . . . 280
8.275.2.8 srvDomainValid . . . 280
8.276 nas_TDSCDMAECIOThresh Struct Reference . . . 281
8.276.1 Detailed Description . . . 281
8.276.2 Field Documentation . . . 281
8.276.2.1 pTDSCDMAECIOThreshList . . . 281
8.276.2.2 TDSCDMAECIOThreshListLen . . . 281
8.277 nas_TDSCDMARSCPThresh Struct Reference . . . 281
8.277.1 Detailed Description . . . 281
8.277.2 Field Documentation . . . 281
8.277.2.1 pTDSCDMARSCPThreshList . . . 281
8.277.2.2 TDSCDMARSCPThreshListLen . . . 282
8.278 nas_TDSCDMARSSIThresh Struct Reference . . . 282
8.278.1 Detailed Description . . . 282
8.278.2 Field Documentation . . . 282
8.278.2.1 pTDSCDMARSSIThreshList . . . 282
8.278.2.2 TDSCDMARSSIThreshListLen . . . 282
8.279 nas_TDSCDMASINRCONFThresh Struct Reference . . . 282
8.279.1 Detailed Description . . . 282
8.279.2 Field Documentation . . . 282
8.279.2.1 pTDSCDMASINRCONFThreshList . . . 283
8.279.2.2 TDSCDMASINRCONFThreshListLen . . . 283
8.280 nas_TDSCDMASINRThresh Struct Reference . . . 283
8.280.1 Detailed Description . . . 283
8.280.2 Field Documentation . . . 283
8.280.2.1 pTDSCDMASINRThreshList . . . 283
8.280.2.2 TDSCDMASINRThreshListLen . . . 283
8.281 nas_timeInfo Struct Reference . . . 283

8.281.1 Detailed Description . . . 283
8.281.2 Field Documentation . . . 285
8.281.2.1 day . . . 285
8.281.2.2 dayLtSavingAdj . . . 285
8.281.2.3 dayOfWeek . . . 285
8.281.2.4 hour . . . 285
8.281.2.5 minute . . . 285
8.281.2.6 month . . . 285
8.281.2.7 radioInterface . . . 285
8.281.2.8 second . . . 285
8.281.2.9 timeZone . . . 285
8.281.2.10 TlvPresent . . . 285
8.281.2.11 year . . . 285
8.282 nas_trueIMSI Struct Reference . . . 285
8.282.1 Detailed Description . . . 285
8.282.2 Field Documentation . . . 286
8.282.2.1 imsiT1112 . . . 286
8.282.2.2 imsiTaddrNum . . . 286
8.282.2.3 imsiTS1 . . . 286
8.282.2.4 imsiTS2 . . . 286
8.282.2.5 mccT . . . 286
8.283 nas_txInfo Struct Reference . . . 286
8.283.1 Detailed Description . . . 286
8.283.2 Field Documentation . . . 286
8.283.2.1 isInTraffic . . . 286
8.283.2.2 txPower . . . 286
8.284 nas_UMTSInfo Struct Reference . . . 286
8.284.1 Detailed Description . . . 287
8.284.2 Field Documentation . . . 288
8.284.2.1 cellID . . . 288
8.284.2.2 ecio . . . 288
8.284.2.3 geranInst . . . 288
8.284.2.4 GeranInstInfo . . . 288
8.284.2.5 lac . . . 288
8.284.2.6 plmn . . . 288
8.284.2.7 psc . . . 288
8.284.2.8 rscp . . . 288
8.284.2.9 uarfcn . . . 288
8.284.2.10 umtsInst . . . 288
8.284.2.11 UMTSInstInfo . . . 288

8.285 nas_UMTSinstInfo Struct Reference . . . 288
8.285.1 Detailed Description . . . 288
8.285.2 Field Documentation . . . 289
8.285.2.1 umtsEcio . . . 289
8.285.2.2 umtsPsc . . . 289
8.285.2.3 umtsRscp . . . 289
8.285.2.4 umtsUarfcn . . . 289
8.286 nas_umtsLTENbrCell Struct Reference . . . 289
8.286.1 Detailed Description . . . 289
8.286.2 Field Documentation . . . 290
8.286.2.1 cellIsTDD . . . 290
8.286.2.2 earfcn . . . 290
8.286.2.3 pci . . . 290
8.286.2.4 rsrp . . . 290
8.286.2.5 rsrq . . . 290
8.286.2.6 srxlev . . . 290
8.287 nas_UniversalTime Struct Reference . . . 290
8.287.1 Detailed Description . . . 290
8.287.2 Field Documentation . . . 291
8.287.2.1 day . . . 291
8.287.2.2 dayOfWeek . . . 291
8.287.2.3 hour . . . 291
8.287.2.4 minute . . . 291
8.287.2.5 month . . . 291
8.287.2.6 second . . . 291
8.287.2.7 year . . . 291
8.288 nas_wcdmaCellInfo Struct Reference . . . 291
8.288.1 Detailed Description . . . 291
8.288.2 Field Documentation . . . 292
8.288.2.1 cpich_ecno . . . 292
8.288.2.2 cpich_rscp . . . 292
8.288.2.3 psc . . . 292
8.288.2.4 srxlev . . . 292
8.289 nas_WCDMAECIOThresh Struct Reference . . . 292
8.289.1 Detailed Description . . . 292
8.289.2 Field Documentation . . . 292
8.289.2.1 pWCDMAECIOThreshList . . . 292
8.289.2.2 WCDMAECIOThreshListLen . . . 292
8.290 nas_WCDMAInfoLTENeighborCell Struct Reference . . . 292
8.290.1 Detailed Description . . . 293

8.290.2 Field Documentation . . . 293
8.290.2.1 UMTSLTENbrCell . . . 293
8.290.2.2 umtsLTENbrCellLen . . . 293
8.290.2.3 wcdmaRRCState . . . 293
8.291 nas_WCDMARSSIThresh Struct Reference . . . 293
8.291.1 Detailed Description . . . 293
8.291.2 Field Documentation . . . 294
8.291.2.1 pWCDMARSSIThreshList . . . 294
8.291.2.2 WCDMARSSIThreshListLen . . . 294
8.292 nas_WCDMASysInfo Struct Reference . . . 294
8.292.1 Detailed Description . . . 294
8.292.2 Field Documentation . . . 296
8.292.2.1 cellId . . . 297
8.292.2.2 cellIdValid . . . 297
8.292.2.3 hsCallStatus . . . 297
8.292.2.4 hsCallStatusValid . . . 297
8.292.2.5 hsInd . . . 297
8.292.2.6 hsIndValid . . . 297
8.292.2.7 lac . . . 297
8.292.2.8 lacValid . . . 297
8.292.2.9 MCC . . . 297
8.292.2.10 MNC . . . 297
8.292.2.11 networkIdValid . . . 297
8.292.2.12 psc . . . 297
8.292.2.13 pscValid . . . 297
8.292.2.14 regRejectInfoValid . . . 297
8.292.2.15 rejCause . . . 297
8.292.2.16 rejectSrvDomain . . . 297
8.292.2.17 sysInfoWCDMA . . . 297
8.293 nas_wcdmaUARFCN Struct Reference . . . 297
8.293.1 Detailed Description . . . 297
8.293.2 Field Documentation . . . 298
8.293.2.1 status . . . 298
8.293.2.2 uarfcn . . . 298
8.294 NASAcqOrderPrefTlv Struct Reference . . . 298
8.294.1 Detailed Description . . . 298
8.294.2 Field Documentation . . . 298
8.294.2.1 AcqOrderLen . . . 298
8.294.2.2 AcqOrderPref . . . 298
8.294.2.3 TlvPresent . . . 298

8.295 NASBandPreferenceTlv Struct Reference . . . 298
8.295.1 Detailed Description . . . 298
8.295.2 Field Documentation . . . 298
8.295.2.1 band_pref . . . 298
8.295.2.2 TlvPresent . . . 298
8.296 NASCiotAcqOrderPrefTlv Struct Reference . . . 299
8.296.1 Detailed Description . . . 299
8.296.2 Field Documentation . . . 299
8.296.2.1 CiotAcqOrderLen . . . 299
8.296.2.2 CiotAcqOrderPref . . . 299
8.296.2.3 TlvPresent . . . 299
8.297 NASCiotLteOpModePrefTlv Struct Reference . . . 299
8.297.1 Detailed Description . . . 299
8.297.2 Field Documentation . . . 299
8.297.2.1 CiotLteOpModePref . . . 299
8.297.2.2 TlvPresent . . . 299
8.298 NASEmergencyModeTlv Struct Reference . . . 299
8.298.1 Detailed Description . . . 300
8.298.2 Field Documentation . . . 300
8.298.2.1 EmerMode . . . 300
8.298.2.2 TlvPresent . . . 300
8.299 NasGetLTECphyCaInfo Struct Reference . . . 300
8.299.1 Detailed Description . . . 300
8.299.2 Field Documentation . . . 301
8.299.2.1 PhyCaAggPcellInfo . . . 301
8.299.2.2 PhyCaAggScellArray . . . 301
8.299.2.3 PhyCaAggScellDlBw . . . 301
8.299.2.4 PhyCaAggScellIndex . . . 301
8.299.2.5 PhyCaAggScellIndType . . . 301
8.299.2.6 PhyCaAggScellInfo . . . 301
8.300 NASGWAcqOrderPrefTlv Struct Reference . . . 301
8.300.1 Detailed Description . . . 301
8.300.2 Field Documentation . . . 301
8.300.2.1 GWAcqOrderPref . . . 301
8.300.2.2 TlvPresent . . . 301
8.301 NASLTEBandPreferenceTlv Struct Reference . . . 301
8.301.1 Detailed Description . . . 302
8.301.2 Field Documentation . . . 302
8.301.2.1 LTEBandPref . . . 302
8.301.2.2 TlvPresent . . . 302

8.302NASLteM1BandPrefTlv Struct Reference . . . 302
8.302.1 Detailed Description . . . 302
8.302.2 Field Documentation . . . 302
8.302.2.1 LteM1BandPref . . . 302
8.302.2.2 TlvPresent . . . 302
8.303NASLteNasReleaseInfoTlv Struct Reference . . . 302
8.303.1 Detailed Description . . . 302
8.303.2 Field Documentation . . . 303
8.303.2.1 nas_major . . . 303
8.303.2.2 nas_minor . . . 303
8.303.2.3 nas_release . . . 303
8.303.2.4 TlvPresent . . . 303
8.304NASLteNB1BandPrefTlv Struct Reference . . . 303
8.304.1 Detailed Description . . . 303
8.304.2 Field Documentation . . . 303
8.304.2.1 LteNB1BandPref . . . 303
8.304.2.2 TlvPresent . . . 303
8.305NASModePreferenceTlv Struct Reference . . . 303
8.305.1 Detailed Description . . . 304
8.305.2 Field Documentation . . . 304
8.305.2.1 ModePref . . . 304
8.305.2.2 TlvPresent . . . 304
8.306NASNetSelPreferenceTlv Struct Reference . . . 304
8.306.1 Detailed Description . . . 304
8.306.2 Field Documentation . . . 304
8.306.2.1 NetSelPref . . . 304
8.306.2.2 TlvPresent . . . 304
8.307NASNr5gBandPrefTlv Struct Reference . . . 304
8.307.1 Detailed Description . . . 304
8.307.2 Field Documentation . . . 305
8.307.2.1 Nr5gBandPrefbits1_64 . . . 305
8.307.2.2 Nr5gBandPrefbits_129_192 . . . 305
8.307.2.3 Nr5gBandPrefbits_193_256 . . . 305
8.307.2.4 Nr5gBandPrefbits_65_128 . . . 305
8.307.2.5 TlvPresent . . . 305
8.308NASOTAMessageTlv Struct Reference . . . 305
8.308.1 Detailed Description . . . 305
8.308.2 Field Documentation . . . 305
8.308.2.1 data_buf . . . 306
8.308.2.2 data_len . . . 306

8.308.2.3 message_type . . . 306
8.308.2.4 TlvPresent . . . 306
8.309 NASPhyCaAggPcellInfo Struct Reference . . . 306
8.309.1 Detailed Description . . . 306
8.309.2 Field Documentation . . . 306
8.309.2.1 dl_bw_value . . . 306
8.309.2.2 freq . . . 306
8.309.2.3 iLTEbandValue . . . 306
8.309.2.4 pci . . . 306
8.309.2.5 TlvPresent . . . 307
8.310 NASPhyCaAggScellArray Struct Reference . . . 307
8.310.1 Detailed Description . . . 307
8.310.2 Field Documentation . . . 308
8.310.2.1 band . . . 308
8.310.2.2 cphy_ca_dl_bandwidth . . . 308
8.310.2.3 cphy_scell_info_list_len . . . 308
8.310.2.4 freq . . . 308
8.310.2.5 pci . . . 308
8.310.2.6 scell_idx . . . 308
8.310.2.7 scell_state . . . 308
8.310.2.8 TlvPresent . . . 308
8.311 NASPhyCaAggScellDlBw Struct Reference . . . 308
8.311.1 Detailed Description . . . 308
8.311.2 Field Documentation . . . 308
8.311.2.1 dl_bw_value . . . 308
8.311.2.2 TlvPresent . . . 308
8.312 NASPhyCaAggScellIndex Struct Reference . . . 308
8.312.1 Detailed Description . . . 309
8.312.2 Field Documentation . . . 309
8.312.2.1 scell_idx . . . 309
8.312.2.2 TlvPresent . . . 309
8.313 NASPhyCaAggScellIndType Struct Reference . . . 309
8.313.1 Detailed Description . . . 309
8.313.2 Field Documentation . . . 309
8.313.2.1 freq . . . 310
8.313.2.2 pci . . . 310
8.313.2.3 scell_state . . . 310
8.313.2.4 TlvPresent . . . 310
8.314 NASPhyCaAggScellInfo Struct Reference . . . 310
8.314.1 Detailed Description . . . 310

8.314.2 Field Documentation . . . 310
8.314.2.1 dl_bw_value . . . 310
8.314.2.2 freq . . . 311
8.314.2.3 iLTEbandValue . . . 311
8.314.2.4 pci . . . 311
8.314.2.5 scell_state . . . 311
8.314.2.6 TlvPresent . . . 311
8.315 NASPRLPreferenceTlv Struct Reference . . . 311
8.315.1 Detailed Description . . . 311
8.315.2 Field Documentation . . . 311
8.315.2.1 PRLPref . . . 311
8.315.2.2 TlvPresent . . . 311
8.316 NASQmiCbkNasSwiOTAMessageInd Struct Reference . . . 311
8.316.1 Detailed Description . . . 311
8.316.2 Field Documentation . . . 312
8.316.2.1 nasRelInfoTlv . . . 312
8.316.2.2 otaMsgTlv . . . 312
8.316.2.3 timeTlv . . . 312
8.317 NASQmiCbkNasSystemSelPrefInd Struct Reference . . . 312
8.317.1 Detailed Description . . . 312
8.317.2 Field Documentation . . . 314
8.317.2.1 AOPTlv . . . 314
8.317.2.2 BPTlv . . . 314
8.317.2.3 CiotAOPTlv . . . 314
8.317.2.4 CiotOpMPTlv . . . 314
8.317.2.5 EMTlv . . . 314
8.317.2.6 GWAOPTlv . . . 314
8.317.2.7 LBPTlv . . . 314
8.317.2.8 M1BandPTlv . . . 314
8.317.2.9 MPTlv . . . 314
8.317.2.10 NB1BandPTlv . . . 314
8.317.2.11 NR5gBandPTlv . . . 314
8.317.2.12 NSPTlv . . . 314
8.317.2.13 PRLPTlv . . . 314
8.317.2.14 RatDMTlv . . . 314
8.317.2.15 RPTlv . . . 314
8.317.2.16 SDPTlv . . . 314
8.318 NASRatDisabledMaskTlv Struct Reference . . . 314
8.318.1 Detailed Description . . . 314
8.318.2 Field Documentation . . . 315

8.318.2.1 RatDisabledMask . . . 315
8.318.2.2 TlvPresent . . . 315
8.319 NASRoamPreferenceTlv Struct Reference . . . 315
8.319.1 Detailed Description . . . 315
8.319.2 Field Documentation . . . 315
8.319.2.1 RoamPref . . . 315
8.319.2.2 TlvPresent . . . 315
8.320 NASServDomainPrefTlv Struct Reference . . . 315
8.320.1 Detailed Description . . . 315
8.320.2 Field Documentation . . . 315
8.320.2.1 SrvDomainPref . . . 315
8.320.2.2 TlvPresent . . . 315
8.321 NASServingSystemInfo Struct Reference . . . 315
8.321.1 Detailed Description . . . 316
8.321.2 Field Documentation . . . 317
8.321.2.1 csAttachState . . . 317
8.321.2.2 hdrPersonality . . . 317
8.321.2.3 psAttachState . . . 317
8.321.2.4 radioInterfaceList . . . 317
8.321.2.5 radioInterfaceNo . . . 317
8.321.2.6 registrationState . . . 317
8.321.2.7 selectedNetwork . . . 317
8.322 NASTimeInfoTlv Struct Reference . . . 317
8.322.1 Detailed Description . . . 317
8.322.2 Field Documentation . . . 317
8.322.2.1 time . . . 317
8.322.2.2 TlvPresent . . . 317
8.323 newMTMessageTlv Struct Reference . . . 318
8.323.1 Detailed Description . . . 318
8.323.2 Field Documentation . . . 318
8.323.2.1 MTMessageInfo . . . 318
8.323.2.2 TlvPresent . . . 318
8.324 pack_audio_SLQSGetAudioPathConfig_t Struct Reference . . . 318
8.324.1 Detailed Description . . . 318
8.324.2 Field Documentation . . . 319
8.324.2.1 Item . . . 319
8.324.2.2 Profile . . . 319
8.325 pack_audio_SLQSGetAudioProfile_t Struct Reference . . . 319
8.325.1 Detailed Description . . . 319
8.325.2 Field Documentation . . . 319

8.325.2.1 Generator . . . 319
8.326 pack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference . . . 320
8.326.1 Detailed Description . . . 320
8.326.2 Field Documentation . . . 320
8.326.2.1 Generator . . . 320
8.326.2.2 Item . . . 320
8.326.2.3 Profile . . . 320
8.326.2.4 Volume . . . 320
8.327 pack_audio_SLQSSetAudioPathConfig_t Struct Reference . . . 320
8.327.1 Detailed Description . . . 321
8.327.2 Field Documentation . . . 322
8.327.2.1 pCodecSTGain . . . 322
8.327.2.2 pDTMFTXGain . . . 322
8.327.2.3 pECMode . . . 322
8.327.2.4 pNSEnable . . . 322
8.327.2.5 Profile . . . 322
8.327.2.6 pRXAGCList . . . 322
8.327.2.7 pRXAVCAGCSwitch . . . 322
8.327.2.8 pRXAVCList . . . 322
8.327.2.9 pRXPCMIIRFltr . . . 322
8.327.2.10 pTXAGCList . . . 322
8.327.2.11 pTXAVCSwitch . . . 322
8.327.2.12 pTXGain . . . 322
8.327.2.13 pTXPCMIIRFltr . . . 322
8.328 pack_audio_SLQSSetAudioProfile_t Struct Reference . . . 322
8.328.1 Detailed Description . . . 323
8.328.2 Field Documentation . . . 323
8.328.2.1 EarMute . . . 323
8.328.2.2 Generator . . . 323
8.328.2.3 MicMute . . . 323
8.328.2.4 Profile . . . 323
8.328.2.5 Volume . . . 323
8.329 pack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference . . . 324
8.329.1 Detailed Description . . . 324
8.329.2 Field Documentation . . . 324
8.329.2.1 Generator . . . 324
8.329.2.2 Item . . . 324
8.329.2.3 Profile . . . 324
8.329.2.4 Volume . . . 324
8.329.2.5 VolValue . . . 324

8.330 pack_cat_CATSendEnvelopeCommand_t Struct Reference . . . 324
8.330.1 Detailed Description . . . 325
8.330.2 Field Documentation . . . 325
8.330.2.1 cmdID . . . 325
8.330.2.2 dataLen . . . 325
8.330.2.3 pData . . . 325
8.331 pack_cat_CATSendTerminalResponse_t Struct Reference . . . 325
8.331.1 Detailed Description . . . 325
8.331.2 Field Documentation . . . 326
8.331.2.1 dataLen . . . 326
8.331.2.2 pData . . . 326
8.331.2.3 refID . . . 326
8.332 pack_cat_SetCATEventCallback_t Struct Reference . . . 326
8.332.1 Detailed Description . . . 326
8.332.2 Field Documentation . . . 326
8.332.2.1 eventMask . . . 326
8.333 pack_dms_ActivateAutomatic_t Struct Reference . . . 327
8.333.1 Detailed Description . . . 327
8.333.2 Field Documentation . . . 327
8.333.2.1 actCode . . . 327
8.334 pack_dms_GetCustFeaturesV2_t Struct Reference . . . 327
8.334.1 Detailed Description . . . 327
8.334.2 Field Documentation . . . 327
8.334.2.1 cust_id . . . 327
8.334.2.2 list_type . . . 327
8.334.2.3 Tlvresult . . . 328
8.335 pack_dms_ResetToFactoryDefaults_t Struct Reference . . . 328
8.335.1 Detailed Description . . . 328
8.335.2 Field Documentation . . . 328
8.335.2.1 spc . . . 328
8.336 pack_dms_SetActivationStatusCallback_t Struct Reference . . . 328
8.336.1 Detailed Description . . . 328
8.336.2 Field Documentation . . . 328
8.336.2.1 activationState . . . 328
8.337 pack_dms_SetCrashAction_t Struct Reference . . . 328
8.337.1 Detailed Description . . . 329
8.337.2 Field Documentation . . . 329
8.337.2.1 crashAction . . . 329
8.338 pack_dms_SetCustFeature_t Struct Reference . . . 329
8.338.1 Detailed Description . . . 329

8.338.2 Field Documentation . . . 331
8.338.2.1 DHCPRelayEnabled . . . 331
8.338.2.2 DisableIMSI . . . 331
8.338.2.3 GpsEnable . . . 331
8.338.2.4 GPSLPM . . . 331
8.338.2.5 GPSSel . . . 331
8.338.2.6 IPFamSupport . . . 331
8.338.2.7 IsVoiceEnabled . . . 331
8.338.2.8 RMAutoConnect . . . 331
8.338.2.9 SMSSupport . . . 331
8.339 pack_dms_SetCustFeaturesV2_t Struct Reference . . . 331
8.339.1 Detailed Description . . . 331
8.339.2 Field Documentation . . . 332
8.339.2.1 cust_id . . . 332
8.339.2.2 cust_value . . . 332
8.339.2.3 Tlvresult . . . 332
8.339.2.4 value_length . . . 332
8.340 pack_dms_SetEventReport_t Struct Reference . . . 332
8.340.1 Detailed Description . . . 332
8.340.2 Field Documentation . . . 332
8.340.2.1 mode . . . 332
8.341 pack_dms_SetIndicationRegister_t Struct Reference . . . 332
8.341.1 Detailed Description . . . 332
8.341.2 Field Documentation . . . 333
8.341.2.1 PSMCfgChangeInfo . . . 333
8.341.2.2 PSMStatus . . . 333
8.341.2.3 RptIMSCapability . . . 333
8.342 pack_dms_SetPower_t Struct Reference . . . 333
8.342.1 Detailed Description . . . 333
8.342.2 Field Documentation . . . 334
8.342.2.1 mode . . . 334
8.342.2.2 Tlvresult . . . 334
8.343 pack_dms_SetUSBComp_t Struct Reference . . . 334
8.343.1 Detailed Description . . . 334
8.343.2 Field Documentation . . . 334
8.343.2.1 Tlvresult . . . 334
8.343.2.2 USBComp . . . 335
8.344 pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 335
8.344.1 Detailed Description . . . 335
8.344.2 Field Documentation . . . 335

8.344.2.1 resetInfoInd . . . 335
8.345 pack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference . . . 335
8.345.1 Detailed Description . . . 335
8.345.2 Field Documentation . . . 336
8.345.2.1 pActiveTimer . . . 336
8.345.2.2 pPeriodicUpdateTimer . . . 336
8.345.2.3 pPsmEnableState . . . 336
8.346 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 336
8.346.1 Detailed Description . . . 336
8.346.2 Field Documentation . . . 336
8.346.2.1 clear . . . 336
8.347 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 336
8.347.1 Detailed Description . . . 336
8.347.2 Field Documentation . . . 337
8.347.2.1 pDestSMSContent . . . 337
8.347.2.2 pDestSMSNum . . . 337
8.348 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 337
8.348.1 Detailed Description . . . 337
8.348.2 Field Documentation . . . 338
8.348.2.1 hostID . . . 338
8.348.2.2 manString . . . 338
8.348.2.3 modelString . . . 338
8.348.2.4 plasmaIDString . . . 338
8.348.2.5 swVerString . . . 338
8.349 pack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 338
8.349.1 Detailed Description . . . 338
8.349.2 Field Documentation . . . 338
8.349.2.1 nameString . . . 338
8.349.2.2 versionString . . . 338
8.350 pack_dms_SwiSetEventReport_t Struct Reference . . . 338
8.350.1 Detailed Description . . . 338
8.350.2 Field Documentation . . . 339
8.350.2.1 pTempReport . . . 339
8.350.2.2 pUIMStatusReport . . . 339
8.350.2.3 pVoltReport . . . 339
8.351 pack_dms_SwiUimSelect_t Struct Reference . . . 339
8.351.1 Detailed Description . . . 339
8.351.2 Field Documentation . . . 339
8.351.2.1 uim_select . . . 340
8.352 pack_dms_UIMChangePIN_t Struct Reference . . . 340

8.352.1 Detailed Description . . . 340
8.352.2 Field Documentation . . . 340
8.352.2.1 id . . . 340
8.352.2.2 newValue . . . 340
8.352.2.3 oldValue . . . 340
8.353 pack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 340
8.353.1 Detailed Description . . . 340
8.353.2 Field Documentation . . . 341
8.353.2.1 facility . . . 341
8.354 pack_dms_UIMGetICCID_t Struct Reference . . . 341
8.354.1 Detailed Description . . . 341
8.354.2 Field Documentation . . . 341
8.354.2.1 ParamPresenceMask . . . 341
8.354.2.2 Tlvresult . . . 341
8.355 pack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 341
8.355.1 Detailed Description . . . 341
8.355.2 Field Documentation . . . 342
8.355.2.1 facility . . . 342
8.355.2.2 facilityCk . . . 342
8.355.2.3 facilityState . . . 342
8.356 pack_dms_UIMSetPINProtection_t Struct Reference . . . 342
8.356.1 Detailed Description . . . 342
8.356.2 Field Documentation . . . 343
8.356.2.1 bEnable . . . 343
8.356.2.2 id . . . 343
8.356.2.3 value . . . 343
8.357 pack_dms_UIMUnblockControlKey_t Struct Reference . . . 343
8.357.1 Detailed Description . . . 343
8.357.2 Field Documentation . . . 343
8.357.2.1 facility . . . 343
8.357.2.2 facilityCk . . . 343
8.358 pack_dms_UIMUnblockPIN_t Struct Reference . . . 343
8.358.1 Detailed Description . . . 343
8.358.2 Field Documentation . . . 344
8.358.2.1 id . . . 344
8.358.2.2 newPin . . . 344
8.358.2.3 pukValue . . . 344
8.359 pack_dms_UIMVerifyPIN_t Struct Reference . . . 344
8.359.1 Detailed Description . . . 344
8.359.2 Field Documentation . . . 344

8.359.2.1 id . . . 344
8.359.2.2 value . . . 344
8.360 pack_fms_GetImagesPreference_t Struct Reference . . . 344
8.360.1 Detailed Description . . . 345
8.360.2 Field Documentation . . . 345
8.360.2.1 TlvResult . . . 345
8.361 pack_fms_GetStoredImages_t Struct Reference . . . 345
8.361.1 Detailed Description . . . 345
8.361.2 Field Documentation . . . 345
8.361.2.1 TlvResult . . . 345
8.362 pack_fms_SetImagesPreference_t Struct Reference . . . 345
8.362.1 Detailed Description . . . 345
8.362.2 Field Documentation . . . 346
8.362.2.1 bForceDownload . . . 346
8.362.2.2 imageListSize . . . 346
8.362.2.3 modemIndex . . . 346
8.362.2.4 pImageList . . . 346
8.362.2.5 TlvResult . . . 346
8.363 pack_ims_SLQSImsConfigIndicationRegister_t Struct Reference . . . 346
8.363.1 Detailed Description . . . 346
8.363.2 Field Documentation . . . 347
8.363.2.1 pRegMgrConfigEvents . . . 347
8.363.2.2 pSIPConfigEvents . . . 347
8.363.2.3 pSMSConfigEvents . . . 347
8.363.2.4 pUserConfigEvents . . . 347
8.363.2.5 pVoIPConfigEvents . . . 347
8.364 pack_ims_SLQSSetIMSSMSConfig_t Struct Reference . . . 347
8.364.1 Detailed Description . . . 348
8.364.2 Field Documentation . . . 348
8.364.2.1 pPhoneCtxtURI . . . 348
8.364.2.2 pPhoneCtxtURILen . . . 348
8.364.2.3 pSMSFormat . . . 348
8.364.2.4 pSMSOverIPNwInd . . . 348
8.365 pack_ims_SLQSSetIMSUserConfig_t Struct Reference . . . 348
8.365.1 Detailed Description . . . 348
8.365.2 Field Documentation . . . 349
8.365.2.1 pIMSDomain . . . 349
8.365.2.2 pIMSDomainLen . . . 349
8.366 pack_ims_SLQSSetIMSVoIPConfig_t Struct Reference . . . 349
8.366.1 Detailed Description . . . 349

8.366.2 Field Documentation . . . 351
8.366.2.1 pAmrMode . . . 351
8.366.2.2 pAmrOctetAligned . . . 351
8.366.2.3 pAmrWbEnable . . . 351
8.366.2.4 pAmrWBMode . . . 351
8.366.2.5 pAmrWBOctetAligned . . . 351
8.366.2.6 pMinSessionExpiryTimer . . . 351
8.366.2.7 pRingBackTimer . . . 351
8.366.2.8 pRingingTimer . . . 351
8.366.2.9 pRTPRTCPInactTimer . . . 351
8.366.2.10 pScrAmrEnable . . . 351
8.366.2.11 pScrAmrWbEnable . . . 351
8.366.2.12 pSessionExpiryTimer . . . 351
8.367 pack_ims_SLQSSetRegMgrConfig_t Struct Reference . . . 351
8.367.1 Detailed Description . . . 351
8.367.2 Field Documentation . . . 352
8.367.2.1 pCSCFPortName . . . 352
8.367.2.2 pCSCFPortNameLen . . . 352
8.367.2.3 pIMSTestMode . . . 352
8.367.2.4 pPriCSCFPort . . . 352
8.368 pack_ims_SLQSSetSIPConfig_t Struct Reference . . . 352
8.368.1 Detailed Description . . . 352
8.368.2 Field Documentation . . . 352
8.368.2.1 pSigCompEnabled . . . 353
8.368.2.2 pSIPLocalPort . . . 353
8.368.2.3 pSubscribeTimer . . . 353
8.368.2.4 pTimerSIPReg . . . 353
8.368.2.5 pTimerT1 . . . 353
8.368.2.6 pTimerT2 . . . 353
8.368.2.7 pTimerTf . . . 353
8.369 pack_imsa_SLQSRegisterIMSAIndication_t Struct Reference . . . 353
8.369.1 Detailed Description . . . 353
8.369.2 Field Documentation . . . 354
8.369.2.1 has_PdpStatusConfig . . . 354
8.369.2.2 has_RatHandoverStatusConfig . . . 354
8.369.2.3 has_RegStatusConfig . . . 354
8.369.2.4 has_ServiceStatusConfig . . . 354
8.369.2.5 PdpStatusConfig . . . 354
8.369.2.6 RatHandoverStatusConfig . . . 354
8.369.2.7 RegStatusConfig . . . 354

8.369.2.8 ServiceStatusConfig . . . 354
8.370 pack_loc_Delete_Assist_Data_t Struct Reference . . . 354
8.370.1 Detailed Description . . . 354
8.370.2 Field Documentation . . . 355
8.370.2.1 pBdsSVInfo . . . 355
8.370.2.2 pCellDb . . . 355
8.370.2.3 pClkInfo . . . 355
8.370.2.4 pGnssData . . . 355
8.370.2.5 pSVInfo . . . 355
8.370.2.6 TlvResult . . . 355
8.371 pack_loc_EventRegister_t Struct Reference . . . 355
8.371.1 Detailed Description . . . 355
8.371.2 Field Documentation . . . 357
8.371.2.1 eventRegister . . . 357
8.371.2.2 TlvResult . . . 357
8.372 pack_loc_SetExtPowerState_t Struct Reference . . . 357
8.372.1 Detailed Description . . . 357
8.372.2 Field Documentation . . . 357
8.372.2.1 extPowerState . . . 357
8.372.2.2 TlvResult . . . 357
8.373 pack_loc_SetOperationMode_t Struct Reference . . . 357
8.373.1 Detailed Description . . . 358
8.373.2 Field Documentation . . . 358
8.373.2.1 mode . . . 358
8.373.2.2 TlvResult . . . 358
8.374 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 358
8.374.1 Detailed Description . . . 358
8.374.2 Field Documentation . . . 358
8.374.2.1 TlvResult . . . 358
8.374.2.2 xid . . . 359
8.375 pack_loc_SLQSLOCGetServer_t Struct Reference . . . 359
8.375.1 Detailed Description . . . 359
8.375.2 Field Documentation . . . 359
8.375.2.1 has_serverAddrTypeMask . . . 359
8.375.2.2 serverAddrTypeMask . . . 359
8.375.2.3 serverType . . . 359
8.376 pack_loc_SLQSLOCInjectPosition_t Struct Reference . . . 359
8.376.1 Detailed Description . . . 360
8.376.2 Field Documentation . . . 363
8.376.2.1 altitudeSrcInfo . . . 363

8.376.2.2 altitudeWrtEllipsoid . . . 363
8.376.2.3 altitudeWrtMeanSeaLevel . . . 363
8.376.2.4 has_altitudeSrcInfo . . . 363
8.376.2.5 has_altitudeWrtEllipsoid . . . 363
8.376.2.6 has_altitudeWrtMeanSeaLevel . . . 363
8.376.2.7 has_horConfidence . . . 363
8.376.2.8 has_horReliability . . . 363
8.376.2.9 has_horUncCircular . . . 363
8.376.2.10 has_latitude . . . 363
8.376.2.11 has_longitude . . . 363
8.376.2.12 has_positionSrc . . . 363
8.376.2.13 has_rawHorConfidence . . . 363
8.376.2.14 has_rawHorUncCircular . . . 363
8.376.2.15 has_timestampAge . . . 363
8.376.2.16 has_timestampUtc . . . 363
8.376.2.17 has_vertConfidence . . . 363
8.376.2.18 has_vertRelicability . . . 364
8.376.2.19 has_vertUnc . . . 364
8.376.2.20 horConfidence . . . 364
8.376.2.21 horReliability . . . 364
8.376.2.22 horUncCircular . . . 364
8.376.2.23 latitude . . . 364
8.376.2.24 longitude . . . 364
8.376.2.25 positionSrc . . . 364
8.376.2.26 rawHorConfidence . . . 364
8.376.2.27 rawHorUncCircular . . . 364
8.376.2.28 timestampAge . . . 364
8.376.2.29 timestampUtc . . . 364
8.376.2.30 vertConfidence . . . 364
8.376.2.31 vertReliability . . . 364
8.376.2.32 vertUnc . . . 364
8.377 pack_loc_SLQSLOCInjectSensorData_t Struct Reference . . . 364
8.377.1 Detailed Description . . . 364
8.377.2 Field Documentation . . . 365
8.377.2.1 acceleroData . . . 365
8.377.2.2 acceleroTimeSrc . . . 365
8.377.2.3 accelTemp . . . 365
8.377.2.4 gyroData . . . 365
8.377.2.5 gyroTemp . . . 366
8.377.2.6 gyroTimeSrc . . . 366

8.377.2.7 has_acceleroTimeSrc . . . 366
8.377.2.8 has_accelTemp . . . 366
8.377.2.9 has_accleroData . . . 366
8.377.2.10 has_gyroData . . . 366
8.377.2.11 has_gyroTemp . . . 366
8.377.2.12 has_gyroTimeSrc . . . 366
8.377.2.13 has_opaqueId . . . 366
8.377.2.14 opaqueId . . . 366
8.378 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference . . . 366
8.378.1 Detailed Description . . . 366
8.378.2 Field Documentation . . . 366
8.378.2.1 timeMsec . . . 366
8.378.2.2 timeUncMsec . . . 366
8.379 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference . . . 366
8.379.1 Detailed Description . . . 367
8.379.2 Field Documentation . . . 367
8.379.2.1 confidence . . . 367
8.379.2.2 has_confidence . . . 367
8.379.2.3 state . . . 367
8.380 pack_loc_SLQSLOCSetServer_t Struct Reference . . . 367
8.380.1 Detailed Description . . . 367
8.380.2 Field Documentation . . . 368
8.380.2.1 pIPv4Config . . . 368
8.380.2.2 pIPv6Config . . . 368
8.380.2.3 pURLAddr . . . 368
8.380.2.4 serverType . . . 368
8.381 pack_loc_Start_t Struct Reference . . . 368
8.381.1 Detailed Description . . . 368
8.381.2 Field Documentation . . . 369
8.381.2.1 pApplicationInfo . . . 369
8.381.2.2 pConfigAltitudeAssumed . . . 369
8.381.2.3 pHorizontalAccuracyLvl . . . 369
8.381.2.4 pIntermediateReportState . . . 369
8.381.2.5 pMinIntervalTime . . . 369
8.381.2.6 pRecurrenceType . . . 370
8.381.2.7 SessionId . . . 370
8.381.2.8 Tlvresult . . . 370
8.382 pack_loc_Stop_t Struct Reference . . . 370
8.382.1 Detailed Description . . . 370
8.382.2 Field Documentation . . . 370

8.382.2.1 SessionId . . . 370
8.382.2.2 Tlvresult . . . 370
8.383pack_nas_InitiateDomainAttach_t Struct Reference . . . 370
8.383.1 Detailed Description . . . 370
8.383.2 Field Documentation . . . 370
8.383.2.1 action . . . 371
8.384pack_nas_PerformNetworkScanPCI_t Struct Reference . . . 371
8.384.1 Field Documentation . . . 371
8.384.1.1 pCiotOpModePref . . . 371
8.384.1.2 pLteM1BandPref . . . 371
8.384.1.3 pLteNB1BandPref . . . 371
8.384.1.4 pNetworkType . . . 371
8.384.1.5 pScanType . . . 371
8.385pack_nas_SetACCOLC_t Struct Reference . . . 371
8.385.1 Detailed Description . . . 371
8.385.2 Field Documentation . . . 371
8.385.2.1 accolc . . . 371
8.385.2.2 spc . . . 371
8.386pack_nas_SetCDMANetworkParameters_t Struct Reference . . . 371
8.386.1 Detailed Description . . . 372
8.386.2 Field Documentation . . . 373
8.386.2.1 pApplication . . . 373
8.386.2.2 pBroadcast . . . 373
8.386.2.3 pCustomSCP . . . 373
8.386.2.4 pForceRev0 . . . 373
8.386.2.5 pProtocol . . . 373
8.386.2.6 pRoaming . . . 373
8.386.2.7 pSPC . . . 373
8.387pack_nas_SetNetworkPreference_t Struct Reference . . . 373
8.387.1 Detailed Description . . . 373
8.387.2 Field Documentation . . . 374
8.387.2.1 Duration . . . 374
8.387.2.2 TechnologyPref . . . 374
8.387.2.3 Tlvresult . . . 374
8.388pack_nas_SLQSConfigSigInfo_t Struct Reference . . . 374
8.388.1 Detailed Description . . . 374
8.388.2 Field Documentation . . . 375
8.388.2.1 pECIOThresh . . . 375
8.388.2.2 pHDRSINRThresh . . . 375
8.388.2.3 pIOThresh . . . 375

8.388.2.4 pLTESigRptCfg . . . 375
8.388.2.5 pLTESNRThresh . . . 375
8.388.2.6 pRSRPThresh . . . 375
8.388.2.7 pRSRQThresh . . . 375
8.388.2.8 pRSSIThresh . . . 375
8.388.2.9 pTDSCDMASINRCONFThresh . . . 375
8.389 pack_nas_SLQSGetPLMNName_t Struct Reference . . . 375
8.389.1 Detailed Description . . . 376
8.389.2 Field Documentation . . . 376
8.389.2.1 mcc . . . 376
8.389.2.2 mnc . . . 376
8.389.2.3 pMncPcsStatus . . . 376
8.390 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference . . . 376
8.390.1 Detailed Description . . . 376
8.390.2 Field Documentation . . . 377
8.390.2.1 pChangeDuration . . . 377
8.390.2.2 pMncPcsDigitStatus . . . 377
8.390.2.3 pMNRInfo . . . 377
8.390.2.4 regAction . . . 377
8.391 pack_nas_SLQSNasConfigSigInfo2_t Struct Reference . . . 377
8.391.1 Detailed Description . . . 378
8.391.2 Field Documentation . . . 380
8.391.2.1 pCDMAECIODelta . . . 380
8.391.2.2 pCDMAECIOThresh . . . 380
8.391.2.3 pCDMARSSIDelta . . . 381
8.391.2.4 pCDMARSSIThresh . . . 381
8.391.2.5 pGSMRSSIDelta . . . 381
8.391.2.6 pGSMRSSIThresh . . . 381
8.391.2.7 pHDRECIODelta . . . 381
8.391.2.8 pHDRECIOThresh . . . 381
8.391.2.9 pHDRIODelta . . . 381
8.391.2.10 pHDRIOThresh . . . 381
8.391.2.11 pHDRRSSIDelta . . . 381
8.391.2.12 pHDRRSSIThresh . . . 381
8.391.2.13 pHDRSINRDelta . . . 381
8.391.2.14 pHDRSINRThresh . . . 381
8.391.2.15 pLTERSRPDelta . . . 381
8.391.2.16 pLTERSRPThresh . . . 381
8.391.2.17 pLTERSRQDelta . . . 381
8.391.2.18 pLTERSRQThresh . . . 381

8.391.2.19 pLTERSSIDelta . . . 381
8.391.2.20 pLTERSSIThresh . . . 381
8.391.2.21 pLTESigRptConfig . . . 381
8.391.2.22 pLTESNRDelta . . . 381
8.391.2.23 pLTESNRThresh . . . 381
8.391.2.24 pTDSCDMAECIODelta . . . 381
8.391.2.25 pTDSCDMAECIOThresh . . . 381
8.391.2.26 pTDSCDMARSCPDelta . . . 381
8.391.2.27 pTDSCDMARSCPThresh . . . 381
8.391.2.28 pTDSCDMARSSIDelta . . . 381
8.391.2.29 pTDSCDMARSSIThresh . . . 381
8.391.2.30 pTDSCDMASINRDelta . . . 381
8.391.2.31 pTDSCDMASINRThresh . . . 382
8.391.2.32 pWCDMAECIODelta . . . 382
8.391.2.33 pWCDMAECIOThresh . . . 382
8.391.2.34 pWCDMARSSIDelta . . . 382
8.391.2.35 pWCDMARSSIThresh . . . 382
8.392 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 382
8.392.1 Detailed Description . . . 382
8.392.2 Field Documentation . . . 382
8.392.2.1 namID . . . 382
8.393 pack_nas_SLQSNASGeteDRXParamsExt_t Struct Reference . . . 382
8.393.1 Detailed Description . . . 382
8.393.2 Field Documentation . . . 383
8.393.2.1 pEdrxRAT . . . 383
8.393.2.2 pLteOpMode . . . 383
8.394 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 383
8.394.1 Detailed Description . . . 383
8.394.2 Field Documentation . . . 384
8.394.2.1 radio_if . . . 384
8.395 pack_nas_SLQSNasIndicationRegisterExt_t Struct Reference . . . 384
8.395.1 Detailed Description . . . 384
8.395.2 Field Documentation . . . 386
8.395.2.1 pDDTMInd . . . 386
8.395.2.2 pDualStandByPrefInd . . . 386
8.395.2.3 pEdrxChangeInfoInd . . . 386
8.395.2.4 pErrorRateInd . . . 386
8.395.2.5 pHDRNewUATIAssInd . . . 386
8.395.2.6 pHDRSessionCloseInd . . . 386
8.395.2.7 pLTECphyCa . . . 386

8.395.2.8 pManagedRoamingInd . . . 387
8.395.2.9 pNetworkRejectInd . . . 387
8.395.2.10 pNetworkTimeInd . . . 387
8.395.2.11 pServingSystemInd . . . 387
8.395.2.12 pSignalStrengthInd . . . 387
8.395.2.13 pSubscriptionInfoInd . . . 387
8.395.2.14 pSuppressSysInfoInd . . . 387
8.395.2.15 pSysInfoInd . . . 387
8.395.2.16 pSystemSelectionInd . . . 387
8.396 pack_nas_SLQSNASSeteDRXParams_t Struct Reference . . . 387
8.396.1 Detailed Description . . . 387
8.396.2 Field Documentation . . . 388
8.396.2.1 pCycleLen . . . 388
8.396.2.2 pEdrxCiotLteMode . . . 388
8.396.2.3 pEdrxEnable . . . 388
8.396.2.4 pEdrxRatType . . . 388
8.396.2.5 pPagingTimeWindow . . . 388
8.397 pack_nas_SLQSNasSwiIndicationRegister_t Struct Reference . . . 388
8.397.1 Detailed Description . . . 389
8.397.2 Field Documentation . . . 389
8.397.2.1 gsmUmtsDl . . . 389
8.397.2.2 gsmUmtsUl . . . 389
8.397.2.3 lteEmmDl . . . 389
8.397.2.4 lteEmmUl . . . 389
8.397.2.5 lteEsmDl . . . 389
8.397.2.6 lteEsmUl . . . 390
8.397.2.7 pRankIndicatorInd . . . 390
8.397.2.8 pTimer . . . 390
8.398 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference . . . 390
8.398.1 Detailed Description . . . 390
8.398.2 Field Documentation . . . 390
8.398.2.1 pLteEARFCN . . . 390
8.398.2.2 pLtePCI . . . 390
8.398.2.3 pWcdmaUARFCN . . . 390
8.399 pack_nas_SLQSSetBandPreference_t Struct Reference . . . 390
8.399.1 Detailed Description . . . 390
8.399.2 Field Documentation . . . 391
8.399.2.1 bandPref . . . 392
8.400 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference . . . 392
8.400.1 Detailed Description . . . 392

8.400.2 Field Documentation . . . 392
8.400.2.1 bEnable . . . 392
8.400.2.2 pSigIndReq . . . 392
8.401 pack_nas_SLQSSetSysSelectionPref_t Struct Reference . . . 392
8.401.1 Detailed Description . . . 392
8.401.2 Field Documentation . . . 397
8.401.2.1 pAcqOrderPref . . . 397
8.401.2.2 pBandPref . . . 397
8.401.2.3 pChgDuration . . . 397
8.401.2.4 pCSGID . . . 397
8.401.2.5 pEmerMode . . . 397
8.401.2.6 pGWAcqOrderPref . . . 397
8.401.2.7 pLTEBandPref . . . 397
8.401.2.8 pMNCIncPCSDigStat . . . 397
8.401.2.9 pModePref . . . 397
8.401.2.10 pNetSelPref . . . 397
8.401.2.11 pPRLPref . . . 397
8.401.2.12 pRAT . . . 397
8.401.2.13 pRoamPref . . . 397
8.401.2.14 pSrvDomainPref . . . 397
8.401.2.15 pSrvRegRestriction . . . 398
8.401.2.16 pTdscdmaBandPref . . . 398
8.402 pack_nas_SLQSSetSysSelectionPrefExt_t Struct Reference . . . 398
8.402.1 Detailed Description . . . 398
8.402.2 Field Documentation . . . 407
8.402.2.1 pAcqOrderPref . . . 407
8.402.2.2 pBandPref . . . 407
8.402.2.3 pChgDuration . . . 407
8.402.2.4 pCiotAcqOrderPref . . . 407
8.402.2.5 pCiotLteOpMode . . . 407
8.402.2.6 pCiotLteOpModePref . . . 407
8.402.2.7 pCSGID . . . 407
8.402.2.8 pEmerMode . . . 407
8.402.2.9 pGWAcqOrderPref . . . 407
8.402.2.10 pLTEBandPref . . . 407
8.402.2.11 pLTEBandPrefExt . . . 407
8.402.2.12 pLteM1BandPref . . . 407
8.402.2.13 pLteNb1BandPref . . . 407
8.402.2.14 pMNCIncPCSDigStat . . . 407
8.402.2.15 pModePref . . . 407

8.402.2.16 pNetSelPref . . . 407
8.402.2.17 pNr5gBandPref . . . 407
8.402.2.18 pPRLPref . . . 408
8.402.2.19 pRAT . . . 408
8.402.2.20 pRoamPref . . . 408
8.402.2.21 pSrvDomainPref . . . 408
8.402.2.22 pSrvRegRestriction . . . 408
8.402.2.23 pTdscdmaBandPref . . . 408
8.403 pack_nas_SLQSSwiPSDetach_t Struct Reference . . . 408
8.403.1 Detailed Description . . . 408
8.403.2 Field Documentation . . . 408
8.403.2.1 pDetachAction . . . 408
8.404 pack_pds_PDSInjectTimeReference_t Struct Reference . . . 408
8.404.1 Detailed Description . . . 408
8.404.2 Field Documentation . . . 409
8.404.2.1 systemDiscontinuities . . . 409
8.404.2.2 systemTime . . . 409
8.405 pack_pds_ResetPDSData_t Struct Reference . . . 409
8.405.1 Detailed Description . . . 409
8.405.2 Field Documentation . . . 409
8.405.2.1 pCellDataMask . . . 410
8.405.2.2 pGPSDataMask . . . 410
8.406 pack_pds_SetEventReportCallback_t Struct Reference . . . 410
8.406.1 Detailed Description . . . 410
8.406.2 Field Documentation . . . 410
8.406.2.1 posDataNmea . . . 410
8.406.2.2 rptPosData . . . 410
8.407 pack_pds_SetPDSDefaults_t Struct Reference . . . 410
8.407.1 Detailed Description . . . 410
8.407.2 Field Documentation . . . 411
8.407.2.1 accuracy . . . 411
8.407.2.2 interval . . . 411
8.407.2.3 operation . . . 411
8.407.2.4 timeout . . . 411
8.408 pack_pds_SetPDSState_t Struct Reference . . . 411
8.408.1 Detailed Description . . . 411
8.408.2 Field Documentation . . . 411
8.408.2.1 enable . . . 411
8.409 pack_pds_SetPortAutomaticTracking_t Struct Reference . . . 411
8.409.1 Detailed Description . . . 412

8.409.2 Field Documentation . . . 412
8.409.2.1 bAuto . . . 412
8.410 pack_pds_SetServiceAutomaticTracking_t Struct Reference . . . 412
8.410.1 Detailed Description . . . 412
8.410.2 Field Documentation . . . 412
8.410.2.1 bAuto . . . 412
8.411 pack_pds_SetXTRAAutomaticDownload_t Struct Reference . . . 412
8.411.1 Detailed Description . . . 413
8.411.2 Field Documentation . . . 413
8.411.2.1 bEnabled . . . 413
8.411.2.2 interval . . . 413
8.412 pack_pds_SetXTRANetwork_t Struct Reference . . . 413
8.412.1 Detailed Description . . . 413
8.412.2 Field Documentation . . . 413
8.412.2.1 preference . . . 413
8.413 pack_pds_SLQSGetAGPSConfig_t Struct Reference . . . 413
8.413.1 Detailed Description . . . 414
8.413.2 Field Documentation . . . 414
8.413.2.1 pNetworkMode . . . 414
8.414 pack_pds_SLQSPDSInjectAbsoluteTimeReference_t Struct Reference . . . 414
8.414.1 Detailed Description . . . 414
8.414.2 Field Documentation . . . 414
8.414.2.1 forceFlag . . . 414
8.414.2.2 timeBase . . . 414
8.414.2.3 timeMsec . . . 415
8.414.2.4 timeUncMsec . . . 415
8.415 pack_pds_SLQSPDSInjectPositionData_t Struct Reference . . . 415
8.415.1 Detailed Description . . . 415
8.415.2 Field Documentation . . . 416
8.415.2.1 pAltitudeWrtEllipsoid . . . 416
8.415.2.2 pAltitudeWrtSealevel . . . 416
8.415.2.3 pHorizontalConfidence . . . 416
8.415.2.4 pHorizontalUncCircular . . . 416
8.415.2.5 pLatitude . . . 416
8.415.2.6 pLongitude . . . 416
8.415.2.7 pPositionSource . . . 416
8.415.2.8 pTimeStamp . . . 416
8.415.2.9 pTimeType . . . 416
8.415.2.10 pVerticalConfidence . . . 416
8.415.2.11 pVerticalUnc . . . 417

8.416pack_pds_SLQSSetAGPSConfig_t Struct Reference . . . 417
8.416.1 Detailed Description . . . 417
8.416.2 Field Documentation . . . 417
8.416.2.1 pNetworkMode . . . 417
8.416.2.2 pServerAddress . . . 417
8.416.2.3 pServerPort . . . 417
8.416.2.4 pServerURL . . . 417
8.416.2.5 pServerURLLength . . . 417
8.417pack_pds_SLQSSetPositionMethodState_t Struct Reference . . . 417
8.417.1 Detailed Description . . . 418
8.417.2 Field Documentation . . . 418
8.417.2.1 pWifiState . . . 418
8.417.2.2 pXtraDataState . . . 418
8.417.2.3 pXtraTimeState . . . 418
8.418pack_pds_StartPDSTrackingSessionExt_t Struct Reference . . . 418
8.418.1 Detailed Description . . . 418
8.418.2 Field Documentation . . . 419
8.418.2.1 fixAccuracy . . . 419
8.418.2.2 fixCount . . . 419
8.418.2.3 fixInterval . . . 419
8.418.2.4 fixTimeout . . . 419
8.418.2.5 sessionControl . . . 419
8.418.2.6 sessionOperation . . . 419
8.418.2.7 sessionServerOption . . . 419
8.418.2.8 sessionType . . . 419
8.419pack_qmi_t Struct Reference . . . 419
8.419.1 Detailed Description . . . 420
8.419.2 Field Documentation . . . 420
8.419.2.1 msgid . . . 420
8.419.2.2 svc . . . 420
8.419.2.3 timeout . . . 420
8.419.2.4 xid . . . 420
8.420pack_qos_BindDataPort_t Struct Reference . . . 420
8.420.1 Detailed Description . . . 420
8.420.2 Field Documentation . . . 421
8.420.2.1 pMuxID . . . 421
8.420.2.2 pPeripheralEndPointID . . . 421
8.420.2.3 pSIODataPort . . . 421
8.421pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 421
8.421.1 Detailed Description . . . 421

8.421.2 Field Documentation . . . 421
8.421.2.1 apnId . . . 421
8.422pack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 421
8.422.1 Detailed Description . . . 421
8.422.2 Field Documentation . . . 421
8.422.2.1 apnId . . . 421
8.423pack_qos_SLQSSetQosEventCallback_t Struct Reference . . . 421
8.423.1 Detailed Description . . . 422
8.423.2 Field Documentation . . . 422
8.423.2.1 enable . . . 422
8.424pack_rms_SetSMSWake_t Struct Reference . . . 422
8.424.1 Detailed Description . . . 422
8.424.2 Field Documentation . . . 422
8.424.2.1 enabled . . . 422
8.424.2.2 wake_mask . . . 422
8.425pack_sar_SLQSSetRfSarState_t Struct Reference . . . 422
8.425.1 Detailed Description . . . 423
8.425.2 Field Documentation . . . 423
8.425.2.1 RfSarState . . . 423
8.426pack_sms_SaveSMS_t Struct Reference . . . 423
8.426.1 Detailed Description . . . 423
8.426.2 Field Documentation . . . 424
8.426.2.1 messageFormat . . . 424
8.426.2.2 messageSize . . . 424
8.426.2.3 pMessage . . . 424
8.426.2.4 storageType . . . 424
8.427pack_sms_SendSMS_t Struct Reference . . . 424
8.427.1 Detailed Description . . . 424
8.427.2 Field Documentation . . . 424
8.427.2.1 messageFormat . . . 424
8.427.2.2 messageSize . . . 424
8.427.2.3 pLinktimer . . . 425
8.427.2.4 pMessage . . . 425
8.428pack_sms_SetNewSMSCallback_t Struct Reference . . . 425
8.428.1 Detailed Description . . . 425
8.428.2 Field Documentation . . . 425
8.428.2.1 status . . . 425
8.429pack_sms_SetSMSCAddress_t Struct Reference . . . 425
8.429.1 Detailed Description . . . 425
8.429.2 Field Documentation . . . 425

8.429.2.1 pSMSCAddress . . . 425
8.429.2.2 pSMSCType . . . 425
8.430 pack_sms_SLQSDeleteSMS_t Struct Reference . . . 425
8.430.1 Detailed Description . . . 426
8.430.2 Field Documentation . . . 426
8.430.2.1 pMessageIndex . . . 426
8.430.2.2 pMessageMode . . . 426
8.430.2.3 pMessageTag . . . 426
8.430.2.4 storageType . . . 426
8.431 pack_sms_SLQSGetSMS_t Struct Reference . . . 426
8.431.1 Detailed Description . . . 427
8.431.2 Field Documentation . . . 427
8.431.2.1 messageIndex . . . 427
8.431.2.2 pMessageMode . . . 427
8.431.2.3 storageType . . . 427
8.432 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 427
8.432.1 Detailed Description . . . 427
8.432.2 Field Documentation . . . 427
8.432.2.1 mode . . . 427
8.433 pack_sms_SLQSGetSMSList_t Struct Reference . . . 427
8.433.1 Detailed Description . . . 428
8.433.2 Field Documentation . . . 428
8.433.2.1 pMessageMode . . . 428
8.433.2.2 pRequestedTag . . . 428
8.433.2.3 storageType . . . 428
8.434 pack_sms_SLQSModifySMSStatus_t Struct Reference . . . 428
8.434.1 Detailed Description . . . 428
8.434.2 Field Documentation . . . 429
8.434.2.1 messageIndex . . . 429
8.434.2.2 messageTag . . . 429
8.434.2.3 pMessageMode . . . 429
8.434.2.4 storageType . . . 429
8.435 pack_sms_SLQSSendAsyncSMS_t Struct Reference . . . 429
8.435.1 Detailed Description . . . 429
8.435.2 Field Documentation . . . 429
8.435.2.1 pSendSmsParams . . . 429
8.436 pack_sms_SLQSSetIndicationRegister_t Struct Reference . . . 429
8.436.1 Detailed Description . . . 430
8.436.2 Field Documentation . . . 430
8.436.2.1 pSetIndicationRegReq . . . 430

8.437 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference . . . 430
8.437.1 Detailed Description . . . 430
8.437.2 Field Documentation . . . 430
8.437.2.1 broadcastActivate . . . 430
8.437.2.2 mode . . . 430
8.438 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference . . . 430
8.438.1 Detailed Description . . . 431
8.438.2 Field Documentation . . . 431
8.438.2.1 mode . . . 431
8.438.2.2 pBroadcastConfig . . . 431
8.438.2.3 pCDMABroadcastConfig . . . 431
8.439 pack_sms_SLQSSetSmsStorage_t Struct Reference . . . 431
8.439.1 Detailed Description . . . 431
8.439.2 Field Documentation . . . 431
8.439.2.1 smsStorage . . . 432
8.440 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 432
8.440.1 Detailed Description . . . 432
8.440.2 Field Documentation . . . 432
8.440.2.1 pMaxStorageSizeReq . . . 432
8.441 pack_sms_SLQSSmsSetRoutes_t Struct Reference . . . 432
8.441.1 Detailed Description . . . 432
8.441.2 Field Documentation . . . 432
8.441.2.1 pSetRoutesReq . . . 432
8.442 pack_swiaudio_SLQSGetM2MAudioProfile_t Struct Reference . . . 432
8.442.1 Detailed Description . . . 433
8.442.2 Field Documentation . . . 433
8.442.2.1 pGenerator . . . 433
8.443 pack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference . . . 433
8.443.1 Detailed Description . . . 433
8.443.2 Field Documentation . . . 433
8.443.2.1 Generator . . . 433
8.443.2.2 Profile . . . 433
8.444 pack_swiaudio_SLQSGetM2MAVMute_t Struct Reference . . . 433
8.444.1 Detailed Description . . . 433
8.444.2 Field Documentation . . . 434
8.444.2.1 Profile . . . 434
8.445 pack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference . . . 434
8.445.1 Detailed Description . . . 434
8.445.2 Field Documentation . . . 434
8.445.2.1 Profile . . . 434

8.446 pack_swiaudio_SLQSSetM2MAudioAVCFG_t Struct Reference . . . 434
8.446.1 Detailed Description . . . 434
8.446.2 Field Documentation . . . 435
8.446.2.1 Device . . . 435
8.446.2.2 PIFACEId . . . 435
8.446.2.3 pPCMParams . . . 435
8.446.2.4 Profile . . . 435
8.447 pack_swiaudio_SLQSSetM2MAudioLPBK_t Struct Reference . . . 435
8.447.1 Detailed Description . . . 435
8.447.2 Field Documentation . . . 435
8.447.2.1 Enable . . . 435
8.448 pack_swiaudio_SLQSSetM2MAudioProfile_t Struct Reference . . . 435
8.448.1 Detailed Description . . . 436
8.448.2 Field Documentation . . . 436
8.448.2.1 pCwtMute . . . 436
8.448.2.2 pEarMute . . . 436
8.448.2.3 pGenerator . . . 436
8.448.2.4 pMicMute . . . 436
8.448.2.5 Profile . . . 436
8.448.2.6 pVolume . . . 436
8.449 pack_swiaudio_SLQSSetM2MAudioVolume_t Struct Reference . . . 437
8.449.1 Detailed Description . . . 437
8.449.2 Field Documentation . . . 437
8.449.2.1 Generator . . . 437
8.449.2.2 Level . . . 437
8.449.2.3 Profile . . . 437
8.450 pack_swiaudio_SLQSSetM2MAVMute_t Struct Reference . . . 437
8.450.1 Detailed Description . . . 437
8.450.2 Field Documentation . . . 438
8.450.2.1 EarMute . . . 438
8.450.2.2 MicMute . . . 438
8.450.2.3 pCwtMute . . . 438
8.450.2.4 Profile . . . 438
8.451 pack_swiaudio_SLQSSetM2MSpkrGain_t Struct Reference . . . 438
8.451.1 Detailed Description . . . 438
8.451.2 Field Documentation . . . 438
8.451.2.1 Profile . . . 438
8.451.2.2 Value . . . 438
8.452 pack_swiavms_SLQSAVMSSendSelection_t Struct Reference . . . 439
8.452.1 Detailed Description . . . 439

8.452.2 Field Documentation . . . 439
8.452.2.1 pClientPerformOperationFlag . . . 439
8.452.2.2 pDeferTime . . . 439
8.452.2.3 pPackageID . . . 439
8.452.2.4 pRejectReason . . . 439
8.452.2.5 selection . . . 439
8.453 pack_swiavms_SLQSAVMSSetSettings_t Struct Reference . . . 439
8.453.1 Detailed Description . . . 440
8.453.2 Field Documentation . . . 441
8.453.2.1 AutoConnect . . . 441
8.453.2.2 AutoReboot . . . 441
8.453.2.3 pAPNInfo . . . 441
8.453.2.4 pConnectionRetryTimers . . . 441
8.453.2.5 pFwAutoSDM . . . 441
8.453.2.6 pNotifStore . . . 441
8.453.2.7 pPeriodInfo . . . 441
8.453.2.8 pPollingTimer . . . 441
8.453.2.9 PromptFwDownload . . . 441
8.453.2.10 PromptFwUpdate . . . 441
8.454 pack_swiavms_SLQSAVMSSetSettings_v2_t Struct Reference . . . 441
8.454.1 Detailed Description . . . 441
8.454.2 Field Documentation . . . 442
8.454.2.1 AutoConnect . . . 443
8.454.2.2 pAPNInfo . . . 443
8.454.2.3 pAutoReboot . . . 443
8.454.2.4 pConnectionRetryTimers . . . 443
8.454.2.5 pFwAutoSDM . . . 443
8.454.2.6 pNotifStore . . . 443
8.454.2.7 pPeriodInfo . . . 443
8.454.2.8 pPollingTimer . . . 443
8.454.2.9 PromptFwDownload . . . 443
8.454.2.10 PromptFwUpdate . . . 443
8.455 pack_swiavms_SLQSAVMSStartSession_t Struct Reference . . . 443
8.455.1 Detailed Description . . . 443
8.455.2 Field Documentation . . . 443
8.455.2.1 sessionType . . . 443
8.456 pack_swiavms_SLQSAVMSStopSession_t Struct Reference . . . 443
8.456.1 Detailed Description . . . 443
8.456.2 Field Documentation . . . 444
8.456.2.1 sessionType . . . 444

8.457 pack_swidms_SLQSSwiDmsSetMTU_t Struct Reference . . . 444
8.457.1 Detailed Description . . . 444
8.457.2 Field Documentation . . . 444
8.457.2.1 MTUSize . . . 444
8.458 pack_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference . . . 444
8.458.1 Detailed Description . . . 444
8.458.2 Field Documentation . . . 445
8.458.2.1 CfgValue . . . 445
8.459 pack_swidms_SLQSSwiDmsSetUsbNetNum_t Struct Reference . . . 445
8.459.1 Detailed Description . . . 445
8.459.2 Field Documentation . . . 445
8.459.2.1 nUsbNetNum . . . 446
8.460 pack_swiloc_SwiLocSetAutoStart_t Struct Reference . . . 446
8.460.1 Detailed Description . . . 446
8.460.2 Field Documentation . . . 447
8.460.2.1 fix_rate . . . 447
8.460.2.2 fix_type . . . 447
8.460.2.3 function . . . 447
8.460.2.4 max_dist . . . 447
8.460.2.5 max_time . . . 447
8.460.2.6 set_fix_rate . . . 447
8.460.2.7 set_fix_type . . . 447
8.460.2.8 set_function . . . 447
8.460.2.9 set_max_dist . . . 447
8.460.2.10 set_max_time . . . 447
8.461 pack_swioma_SLQSOMADMCancelSession_t Struct Reference . . . 447
8.461.1 Detailed Description . . . 447
8.461.2 Field Documentation . . . 448
8.461.2.1 sessionType . . . 448
8.462 pack_swioma_SLQSOMADMCancelSessionExt_t Struct Reference . . . 448
8.462.1 Detailed Description . . . 448
8.462.2 Field Documentation . . . 448
8.462.2.1 sessionType . . . 448
8.463 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 448
8.463.1 Detailed Description . . . 448
8.463.2 Field Documentation . . . 449
8.463.2.1 SessionType . . . 449
8.464 pack_swioma_SLQSOMADMSendSelection_t Struct Reference . . . 449
8.464.1 Detailed Description . . . 449
8.464.2 Field Documentation . . . 449

8.464.2.1 pDeferTime . . . 449
8.464.2.2 pRejectReason . . . 449
8.464.2.3 selection . . . 449
8.465 pack_swioma_SLQSOMADMSendSelectionExt_t Struct Reference . . . 449
8.465.1 Detailed Description . . . 449
8.465.2 Field Documentation . . . 450
8.465.2.1 selection . . . 450
8.466 pack_swioma_SLQSOMADMSetSettings_t Struct Reference . . . 450
8.466.1 Detailed Description . . . 450
8.466.2 Field Documentation . . . 451
8.466.2.1 FOTAdownload . . . 451
8.466.2.2 FOTAUpdate . . . 451
8.466.2.3 pAutosdm . . . 451
8.466.2.4 pFwAutoCheck . . . 451
8.467 pack_swioma_SLQSOMADMSetSettingsExt_t Struct Reference . . . 451
8.467.1 Detailed Description . . . 451
8.467.2 Field Documentation . . . 452
8.467.2.1 FOTAdownload . . . 452
8.467.2.2 FOTAUpdate . . . 452
8.467.2.3 FUMOEnable . . . 452
8.467.2.4 OMADMEnable . . . 452
8.467.2.5 OMADMLogEnable . . . 452
8.467.2.6 PRLEnable . . . 452
8.468 pack_swioma_SLQSOMADMStartSession_t Struct Reference . . . 452
8.468.1 Detailed Description . . . 452
8.468.2 Field Documentation . . . 452
8.468.2.1 sessionType . . . 452
8.469 pack_swioma_SLQSOMADMStartSessionExt_t Struct Reference . . . 452
8.469.1 Detailed Description . . . 453
8.469.2 Field Documentation . . . 453
8.469.2.1 sessionType . . . 453
8.470 pack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference . . . 453
8.470.1 Detailed Description . . . 453
8.470.2 Field Documentation . . . 453
8.470.2.1 mitigationDevID . . . 453
8.470.2.2 mitigationDevIDLen . . . 453
8.471 pack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference . . . 453
8.471.1 Detailed Description . . . 454
8.471.2 Field Documentation . . . 454
8.471.2.1 mitigationDevID . . . 454

8.471.2.2 mitigationDevIDLen . . . 454
8.472 pack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference . . . 454
8.472.1 Detailed Description . . . 454
8.472.2 Field Documentation . . . 454
8.472.2.1 mitigationDevID . . . 454
8.472.2.2 mitigationDevIDLen . . . 454
8.473 pack_uim_ChangePin_t Struct Reference . . . 454
8.473.1 Detailed Description . . . 455
8.473.2 Field Documentation . . . 455
8.473.2.1 changePIN . . . 455
8.473.2.2 EncryptedPIN1 . . . 455
8.473.2.3 pIndicationToken . . . 455
8.473.2.4 pKeyReferenceID . . . 455
8.473.2.5 sessionInfo . . . 455
8.473.2.6 Tlvresult . . . 455
8.474 pack_uim_ReadTransparent_t Struct Reference . . . 455
8.474.1 Detailed Description . . . 456
8.474.2 Field Documentation . . . 456
8.474.2.1 fileIndex . . . 456
8.474.2.2 pEncryptData . . . 456
8.474.2.3 pIndicationToken . . . 456
8.474.2.4 readTransparent . . . 456
8.474.2.5 sessionInfo . . . 456
8.474.2.6 Tlvresult . . . 456
8.475 pack_uim_SetPinProtection_t Struct Reference . . . 457
8.475.1 Detailed Description . . . 457
8.475.2 Field Documentation . . . 457
8.475.2.1 EncryptedPIN1 . . . 457
8.475.2.2 pIndicationToken . . . 457
8.475.2.3 pinProtection . . . 457
8.475.2.4 pKeyReferenceID . . . 457
8.475.2.5 sessionInfo . . . 457
8.475.2.6 Tlvresult . . . 458
8.476 pack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 458
8.476.1 Detailed Description . . . 458
8.476.2 Field Documentation . . . 458
8.476.2.1 authData . . . 458
8.476.2.2 pIndicationToken . . . 458
8.476.2.3 sessionInfo . . . 458
8.477 pack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 458

8.477.1 Detailed Description . . . 458
8.477.2 Field Documentation . . . 459
8.477.2.1 depersonilisationInfo . . . 459
8.478 pack_uim_SLQSUIMEventRegister_t Struct Reference . . . 459
8.478.1 Detailed Description . . . 459
8.478.2 Field Documentation . . . 459
8.478.2.1 eventMask . . . 459
8.479 pack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 459
8.479.1 Detailed Description . . . 459
8.479.2 Field Documentation . . . 459
8.479.2.1 pConfigurationMask . . . 459
8.480 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 460
8.480.1 Detailed Description . . . 460
8.480.2 Field Documentation . . . 460
8.480.2.1 fileIndex . . . 460
8.480.2.2 pIndicationToken . . . 460
8.480.2.3 sessionInfo . . . 460
8.481 pack_uim_SLQSUIMPowerDown_t Struct Reference . . . 460
8.481.1 Detailed Description . . . 460
8.481.2 Field Documentation . . . 461
8.481.2.1 slot . . . 461
8.482 pack_uim_SLQSUIMPowerUp_t Struct Reference . . . 461
8.482.1 Detailed Description . . . 461
8.482.2 Field Documentation . . . 461
8.482.2.1 pIgnoreHotSwapSwitch . . . 461
8.482.2.2 slot . . . 461
8.483 pack_uim_SLQSUIMRefreshComplete_t Struct Reference . . . 461
8.483.1 Detailed Description . . . 461
8.483.2 Field Documentation . . . 462
8.483.2.1 refreshComplete . . . 462
8.483.2.2 sessionInfo . . . 462
8.484 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 462
8.484.1 Detailed Description . . . 462
8.484.2 Field Documentation . . . 462
8.484.2.1 sessionInfo . . . 462
8.485 pack_uim_SLQSUIMRefreshOK_t Struct Reference . . . 462
8.485.1 Detailed Description . . . 462
8.485.2 Field Documentation . . . 463
8.485.2.1 OKtoRefresh . . . 463
8.485.2.2 sessionInfo . . . 463

8.486 pack_uim_SLQSUIMRefreshRegister_t Struct Reference . . . 463
8.486.1 Detailed Description . . . 463
8.486.2 Field Documentation . . . 463
8.486.2.1 regRefresh . . . 463
8.486.2.2 sessionInfo . . . 463
8.487 pack_uim_SLQSUIMSwitchSlot_t Struct Reference . . . 463
8.487.1 Detailed Description . . . 463
8.487.2 Field Documentation . . . 464
8.487.2.1 bLogicalSlot . . . 464
8.487.2.2 ulPhysicalSlot . . . 464
8.488 pack_uim_UnblockPin_t Struct Reference . . . 464
8.488.1 Detailed Description . . . 464
8.488.2 Field Documentation . . . 465
8.488.2.1 EncryptedPIN1 . . . 465
8.488.2.2 pIndicationToken . . . 465
8.488.2.3 pinProtection . . . 465
8.488.2.4 pKeyReferenceID . . . 465
8.488.2.5 sessionInfo . . . 465
8.488.2.6 Tlvresult . . . 465
8.489 pack_uim_VerifyPin_t Struct Reference . . . 465
8.489.1 Detailed Description . . . 465
8.489.2 Field Documentation . . . 466
8.489.2.1 pEncryptedPIN1 . . . 466
8.489.2.2 pIndicationToken . . . 466
8.489.2.3 pKeyReferenceID . . . 466
8.489.2.4 sessionInfo . . . 466
8.489.2.5 Tlvresult . . . 466
8.489.2.6 verifyPIN . . . 466
8.490 pack_voice_AnswerUSSD_t Struct Reference . . . 466
8.490.1 Detailed Description . . . 466
8.490.2 Field Documentation . . . 467
8.490.2.1 pInfo . . . 467
8.491 pack_voice_OriginateUSSD_t Struct Reference . . . 467
8.491.1 Detailed Description . . . 467
8.491.2 Field Documentation . . . 467
8.491.2.1 pInfo . . . 467
8.492 pack_voice_SLQSOriginateUSSD_t Struct Reference . . . 467
8.492.1 Detailed Description . . . 467
8.492.2 Field Documentation . . . 468
8.492.2.1 ussData . . . 468

8.492.2.2 ussDCS . . . 468
8.492.2.3 ussLen . . . 468
8.493 pack_voice_SLQSVoiceALSSelectLine_t Struct Reference . . . 468
8.493.1 Detailed Description . . . 468
8.493.2 Field Documentation . . . 468
8.493.2.1 lineValue . . . 468
8.494 pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference . . . 468
8.494.1 Detailed Description . . . 468
8.494.2 Field Documentation . . . 468
8.494.2.1 switchOption . . . 468
8.495 pack_voice_SLQSVoiceAnswerCall_t Struct Reference . . . 469
8.495.1 Detailed Description . . . 469
8.495.2 Field Documentation . . . 469
8.495.2.1 pCallId . . . 469
8.496 pack_voice_SLQSVoiceBindSubscription_t Struct Reference . . . 469
8.496.1 Detailed Description . . . 469
8.496.2 Field Documentation . . . 469
8.496.2.1 subsType . . . 469
8.497 pack_voice_SLQSVoiceBurstDTMF_t Struct Reference . . . 469
8.497.1 Detailed Description . . . 470
8.497.2 Field Documentation . . . 470
8.497.2.1 BurstDTMFInfo . . . 470
8.497.2.2 pBurstDTMFLengths . . . 470
8.498 pack_voice_SLQSVoiceDialCall_t Struct Reference . . . 470
8.498.1 Detailed Description . . . 470
8.498.2 Field Documentation . . . 471
8.498.2.1 callNumber . . . 471
8.498.2.2 pCallPartySubAdd . . . 471
8.498.2.3 pCallType . . . 471
8.498.2.4 pCLIRType . . . 471
8.498.2.5 pCUGInfo . . . 471
8.498.2.6 pEmergencyCategory . . . 472
8.498.2.7 pSvcType . . . 472
8.498.2.8 pUUSInFo . . . 472
8.499 pack_voice_SLQSVoiceEndCall_t Struct Reference . . . 472
8.499.1 Detailed Description . . . 472
8.499.2 Field Documentation . . . 472
8.499.2.1 pCallId . . . 472
8.500 pack_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 472
8.500.1 Detailed Description . . . 472

8.500.2 Field Documentation . . . 473
8.500.2.1 pSvcClass . . . 473
8.500.2.2 reason . . . 473
8.501 pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 473
8.501.1 Detailed Description . . . 473
8.501.2 Field Documentation . . . 474
8.501.2.1 pSvcClass . . . 474
8.501.2.2 Reason . . . 474
8.502 pack_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 474
8.502.1 Detailed Description . . . 474
8.502.2 Field Documentation . . . 474
8.502.2.1 callID . . . 474
8.503 pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 474
8.503.1 Detailed Description . . . 475
8.503.2 Field Documentation . . . 475
8.503.2.1 pSvcClass . . . 475
8.504 pack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 475
8.504.1 Detailed Description . . . 475
8.504.2 Field Documentation . . . 476
8.504.2.1 pAirTimer . . . 476
8.504.2.2 pAMRStatus . . . 476
8.504.2.3 pAutoAnswer . . . 476
8.504.2.4 pNamID . . . 476
8.504.2.5 pPrefVoicePrivacy . . . 476
8.504.2.6 pPrefVoiceSO . . . 476
8.504.2.7 pRoamTimer . . . 476
8.504.2.8 pTTYMode . . . 476
8.504.2.9 pVoiceDomainPref . . . 476
8.505 pack_voice_SLQSVoiceIndicationRegister_t Struct Reference . . . 477
8.505.1 Detailed Description . . . 477
8.505.2 Field Documentation . . . 477
8.505.2.1 pRegDTMFEvents . . . 477
8.505.2.2 pRegVoicePrivacyEvents . . . 477
8.505.2.3 pSuppsNotifEvents . . . 477
8.506 pack_voice_SLQSVoiceManageCalls_t Struct Reference . . . 477
8.506.1 Detailed Description . . . 478
8.506.2 Field Documentation . . . 478
8.506.2.1 pCallID . . . 478
8.506.2.2 SUPSType . . . 478
8.507 pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference . . . 478

8.507.1 Detailed Description . . . 478
8.507.2 Field Documentation . . . 479
8.507.2.1 USSInformation . . . 479
8.508 pack_voice_SLQSVoiceSendFlash_t Struct Reference . . . 479
8.508.1 Detailed Description . . . 479
8.508.2 Field Documentation . . . 479
8.508.2.1 pCallID . . . 479
8.508.2.2 pFlashPayLd . . . 479
8.508.2.3 pFlashType . . . 479
8.509 pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 479
8.509.1 Detailed Description . . . 480
8.509.2 Field Documentation . . . 480
8.509.2.1 newPasswd . . . 480
8.509.2.2 newPasswdAgain . . . 480
8.509.2.3 oldPasswd . . . 480
8.509.2.4 Reason . . . 480
8.510 pack_voice_SLQSVoiceSetConfig_t Struct Reference . . . 480
8.510.1 Detailed Description . . . 481
8.510.2 Field Documentation . . . 482
8.510.2.1 pAirTimerConfig . . . 482
8.510.2.2 pAutoAnswer . . . 482
8.510.2.3 pPrefVoiceDomain . . . 482
8.510.2.4 pPrefVoiceSO . . . 482
8.510.2.5 pRoamTimerConfig . . . 482
8.510.2.6 pTTYMode . . . 482
8.511 pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference . . . 482
8.511.1 Detailed Description . . . 482
8.511.2 Field Documentation . . . 482
8.511.2.1 privacyPref . . . 482
8.512 pack_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 482
8.512.1 Detailed Description . . . 482
8.512.2 Field Documentation . . . 484
8.512.2.1 pCallBarringPasswd . . . 484
8.512.2.2 pCallForwardingNumber . . . 484
8.512.2.3 pCallFwdTypeAndPlan . . . 484
8.512.2.4 pServiceClass . . . 484
8.512.2.5 pTimerVal . . . 484
8.512.2.6 reason . . . 484
8.512.2.7 voiceSvc . . . 484
8.513 pack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 484

8.513.1 Detailed Description . . . 484
8.513.2 Field Documentation . . . 485
8.513.2.1 DTMFdigit . . . 485
8.513.2.2 pCallID . . . 485
8.514 pack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 485
8.514.1 Detailed Description . . . 485
8.514.2 Field Documentation . . . 485
8.514.2.1 callID . . . 485
8.515 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference . . . 485
8.515.1 Detailed Description . . . 485
8.515.2 Field Documentation . . . 486
8.515.2.1 pEnableNotification . . . 486
8.516 pack_wds_GetDefaultProfile_t Struct Reference . . . 486
8.516.1 Detailed Description . . . 486
8.516.2 Field Documentation . . . 486
8.516.2.1 profiletype . . . 486
8.517 pack_wds_GetDefaultProfileNum_t Struct Reference . . . 486
8.517.1 Detailed Description . . . 486
8.517.2 Field Documentation . . . 486
8.517.2.1 family . . . 486
8.517.2.2 type . . . 487
8.518 pack_wds_GetDormancyState_t Struct Reference . . . 487
8.518.1 Detailed Description . . . 487
8.519 pack_wds_GetLastMobileIPError_t Struct Reference . . . 487
8.519.1 Detailed Description . . . 487
8.520 pack_wds_GetMobileIP_t Struct Reference . . . 487
8.520.1 Detailed Description . . . 487
8.521 pack_wds_GetMobileIPProfile_t Struct Reference . . . 487
8.521.1 Detailed Description . . . 487
8.521.2 Field Documentation . . . 488
8.521.2.1 index . . . 488
8.522 pack_wds_GetPacketStatistics_t Struct Reference . . . 488
8.522.1 Detailed Description . . . 488
8.522.2 Field Documentation . . . 488
8.522.2.1 pStatMask . . . 488
8.523 pack_wds_GetPacketStatus_t Struct Reference . . . 488
8.523.1 Detailed Description . . . 488
8.523.2 Field Documentation . . . 489
8.523.2.1 statmask . . . 489
8.524 pack_wds_GetSessionDuration_t Struct Reference . . . 489

8.524.1 Detailed Description . . . 489
8.525 pack_wds_RMSetTransferStatistics_t Struct Reference . . . 489
8.525.1 Detailed Description . . . 489
8.525.2 Field Documentation . . . 489
8.525.2.1 RmTrasnferStaticsReq . . . 489
8.526 pack_wds_SetAutoconnect_t Struct Reference . . . 490
8.526.1 Detailed Description . . . 490
8.526.2 Field Documentation . . . 490
8.526.2.1 acroamsetting . . . 490
8.526.2.2 acsetting . . . 490
8.527 pack_wds_SetDefaultProfile_t Struct Reference . . . 490
8.527.1 Detailed Description . . . 490
8.527.2 Field Documentation . . . 491
8.527.2.1 authentication . . . 492
8.527.2.2 ipAddress . . . 492
8.527.2.3 pApnname . . . 492
8.527.2.4 pdpType . . . 492
8.527.2.5 pName . . . 492
8.527.2.6 pPassword . . . 492
8.527.2.7 primaryDNS . . . 492
8.527.2.8 profileType . . . 492
8.527.2.9 pUsername . . . 492
8.527.2.10 secondaryDNS . . . 492
8.528 pack_wds_SetDefaultProfileNum_t Struct Reference . . . 492
8.528.1 Detailed Description . . . 492
8.528.2 Field Documentation . . . 492
8.528.2.1 family . . . 492
8.528.2.2 index . . . 492
8.528.2.3 type . . . 492
8.529 pack_wds_SetMobileIP_t Struct Reference . . . 493
8.529.1 Detailed Description . . . 493
8.529.2 Field Documentation . . . 493
8.529.2.1 mode . . . 493
8.530 pack_wds_SetMobileIPParameters_t Struct Reference . . . 493
8.530.1 Detailed Description . . . 493
8.530.2 Field Documentation . . . 494
8.530.2.1 pHA2002bis . . . 494
8.530.2.2 pHAAuthenticator . . . 494
8.530.2.3 pMode . . . 494
8.530.2.4 pReRegPeriod . . . 494

8.530.2.5 pReRegTraffic . . . 494
8.530.2.6 pRetryInterval . . . 494
8.530.2.7 pRetryLimit . . . 494
8.530.2.8 pSPC . . . 494
8.531 pack_wds_SetMobileIPProfile_t Struct Reference . . . 494
8.531.1 Detailed Description . . . 495
8.531.2 Field Documentation . . . 495
8.531.2.1 index . . . 496
8.531.2.2 pAAASPI . . . 496
8.531.2.3 pAddress . . . 496
8.531.2.4 pEnabled . . . 496
8.531.2.5 pHASPI . . . 496
8.531.2.6 pMNAAA . . . 496
8.531.2.7 pMNHA . . . 496
8.531.2.8 pNAI . . . 496
8.531.2.9 pPrimaryHA . . . 496
8.531.2.10 pRevTunneling . . . 496
8.531.2.11 pSecondaryHA . . . 496
8.531.2.12 spc . . . 496
8.532 pack_wds_SLQSCreateProfile_t Struct Reference . . . 496
8.532.1 Detailed Description . . . 496
8.532.2 Field Documentation . . . 497
8.532.2.1 pCurProfile . . . 497
8.532.2.2 pProfileId . . . 497
8.532.2.3 pProfileType . . . 497
8.533 pack_wds_SLQSDeleteProfile_t Struct Reference . . . 497
8.533.1 Detailed Description . . . 497
8.533.2 Field Documentation . . . 497
8.533.2.1 profileIndex . . . 497
8.533.2.2 profileType . . . 497
8.534 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 497
8.534.1 Detailed Description . . . 497
8.535 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 498
8.535.1 Detailed Description . . . 498
8.536 pack_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 498
8.536.1 Detailed Description . . . 498
8.536.2 Field Documentation . . . 499
8.536.2.1 Mask . . . 499
8.536.2.2 pReportChannelRate . . . 499
8.536.2.3 pReportConnStatus . . . 499

8.536.2.4 pReportDataBearerTech . . . 499
8.536.2.5 pReportDormStatus . . . 499
8.536.2.6 pTransferStatInd . . . 499
8.537 pack_wds_SLQSGetProfileSettings_t Struct Reference . . . 499
8.537.1 Detailed Description . . . 499
8.537.2 Field Documentation . . . 500
8.537.2.1 ProfileId . . . 500
8.537.2.2 ProfileType . . . 500
8.538 pack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 500
8.538.1 Detailed Description . . . 500
8.538.2 Field Documentation . . . 501
8.538.2.1 pReqSettings . . . 501
8.539 pack_wds_SLQSModifyProfile_t Struct Reference . . . 501
8.539.1 Detailed Description . . . 501
8.539.2 Field Documentation . . . 502
8.539.2.1 curProfile . . . 502
8.539.2.2 pProfileId . . . 502
8.539.2.3 pProfileType . . . 502
8.540 pack_wds_SLQSSet3GPPConfigItem_t Struct Reference . . . 502
8.540.1 Detailed Description . . . 502
8.540.2 Field Documentation . . . 503
8.540.2.1 LTEAttachProfileListLen . . . 503
8.540.2.2 p3gppRelease . . . 503
8.540.2.3 pDefaultPDNEnabled . . . 503
8.540.2.4 pLTEAttachProfile . . . 503
8.540.2.5 pLTEAttachProfileList . . . 503
8.540.2.6 pProfileList . . . 504
8.541 pack_wds_SLQSSetIPFamilyPreference_t Struct Reference . . . 504
8.541.1 Detailed Description . . . 504
8.541.2 Field Documentation . . . 504
8.541.2.1 IPFamilyPreference . . . 504
8.542 pack_wds_SLQSSetWdsEventCallback_t Struct Reference . . . 504
8.542.1 Detailed Description . . . 504
8.542.2 Field Documentation . . . 505
8.542.2.1 currentDataBearer . . . 505
8.542.2.2 dataBearer . . . 505
8.542.2.3 dataBearerTechExt . . . 505
8.542.2.4 dataSystemStatus . . . 505
8.542.2.5 dormancyStatus . . . 505
8.542.2.6 interval . . . 505

8.542.2.7 mobileIP . . . 505
8.542.2.8 transferStats . . . 505
8.543 pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 505
8.543.1 Detailed Description . . . 505
8.543.2 Field Documentation . . . 505
8.543.2.1 pProfileId . . . 505
8.544 pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference . . . 505
8.544.1 Detailed Description . . . 505
8.544.2 Field Documentation . . . 506
8.544.2.1 pHwConfig . . . 506
8.544.2.2 pProfileId . . . 506
8.544.2.3 pRequestOptionList . . . 506
8.545 pack_wds_SLQSSSetLoopback_t Struct Reference . . . 506
8.545.1 Detailed Description . . . 506
8.545.2 Field Documentation . . . 506
8.545.2.1 loopbackMode . . . 506
8.545.2.2 loopbackMultiplier . . . 506
8.546 pack_wds_SLQSStartDataSession_t Struct Reference . . . 506
8.546.1 Detailed Description . . . 507
8.546.2 Field Documentation . . . 507
8.546.2.1 pAuth . . . 507
8.546.2.2 pPass . . . 507
8.546.2.3 pprofileid3gpp . . . 507
8.546.2.4 pprofileid3gpp2 . . . 507
8.546.2.5 pTech . . . 507
8.546.2.6 pUser . . . 508
8.547 pack_wds_SLQSStopDataSession_t Struct Reference . . . 508
8.547.1 Detailed Description . . . 508
8.547.2 Field Documentation . . . 508
8.547.2.1 psid . . . 508
8.548 pack_wds_SLQSWdsSetEventReport_t Struct Reference . . . 508
8.548.1 Detailed Description . . . 508
8.548.2 Field Documentation . . . 509
8.548.2.1 pCurrChannelRateInd . . . 509
8.548.2.2 pCurrDataBearerTechInd . . . 509
8.548.2.3 pCurrPrefDataSysInd . . . 509
8.548.2.4 pDataBearerTechInd . . . 509
8.548.2.5 pDataCallStatusChangeInd . . . 509
8.548.2.6 pDataSystemStatusChangeInd . . . 510
8.548.2.7 pDormancyStatusInd . . . 510

8.548.2.8 pEVDOPageMonPerChangeInd . . . 510
8.548.2.9 pMIPStatusInd . . . 510
8.548.2.10 pTransferStatInd . . . 510
8.549 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 510
8.549.1 Detailed Description . . . 510
8.549.2 Field Documentation . . . 510
8.549.2.1 contextId . . . 510
8.549.2.2 contextType . . . 510
8.550 PackCreateProfileOut Struct Reference . . . 510
8.550.1 Detailed Description . . . 510
8.550.2 Field Documentation . . . 511
8.550.2.1 ExtErrorCode . . . 511
8.550.2.2 ProfileIndex . . . 511
8.550.2.3 ProfileType . . . 511
8.551 packgetDyingGaspCfg Struct Reference . . . 511
8.551.1 Detailed Description . . . 511
8.551.2 Field Documentation . . . 511
8.551.2.1 pDestSMSContent . . . 511
8.551.2.2 pDestSMSNum . . . 511
8.552 packgetDyingGaspStatistics Struct Reference . . . 511
8.552.1 Detailed Description . . . 512
8.552.2 Field Documentation . . . 512
8.552.2.1 pSMSAttemptedFlag . . . 512
8.552.2.2 pTimeStamp . . . 512
8.553 PackSwiAvmsSetSettingsAPNInfo Struct Reference . . . 512
8.553.1 Detailed Description . . . 512
8.553.2 Field Documentation . . . 513
8.553.2.1 bAPNLength . . . 513
8.553.2.2 bPWDLength . . . 513
8.553.2.3 bUnameLength . . . 513
8.553.2.4 szAPN . . . 513
8.553.2.5 szPWD . . . 513
8.553.2.6 szUname . . . 513
8.554 PackSwiAvmsSetSettingsConnectionRetryTimers Struct Reference . . . 513
8.554.1 Detailed Description . . . 513
8.554.2 Field Documentation . . . 513
8.554.2.1 Timers . . . 514
8.555 PackSwiAvmsSetSettingsPeriodInfo Struct Reference . . . 514
8.555.1 Detailed Description . . . 514
8.555.2 Field Documentation . . . 514

8.555.2.1 ulMax . . . 514
8.555.2.2 ulMin . . . 514
8.556 PackSwiAVMSSettingsAPNInfo Struct Reference . . . 514
8.556.1 Detailed Description . . . 514
8.556.2 Field Documentation . . . 514
8.556.2.1 bAPNLength . . . 514
8.556.2.2 bPWDLength . . . 515
8.556.2.3 bUnameLength . . . 515
8.556.2.4 pAPN . . . 515
8.556.2.5 pPWD . . . 515
8.556.2.6 pUname . . . 515
8.557 PackSwiAVMSSettingsConnectionRetryTimers Struct Reference . . . 515
8.557.1 Detailed Description . . . 515
8.557.2 Field Documentation . . . 515
8.557.2.1 Timers . . . 515
8.558 PackSwiAVMSSettingsPeriodsInfo Struct Reference . . . 516
8.558.1 Detailed Description . . . 516
8.558.2 Field Documentation . . . 516
8.558.2.1 max . . . 516
8.558.2.2 min . . . 516
8.559 qmiSmsMessageList Struct Reference . . . 516
8.559.1 Detailed Description . . . 516
8.559.2 Field Documentation . . . 516
8.559.2.1 messageIndex . . . 516
8.559.2.2 messageTag . . . 516
8.560 qmiWDSDataBearerTechnology Struct Reference . . . 516
8.560.1 Detailed Description . . . 517
8.560.2 Field Documentation . . . 517
8.560.2.1 currentNetwork . . . 517
8.560.2.2 ratMask . . . 517
8.560.2.3 soMask . . . 517
8.561 qmTlvResult Struct Reference . . . 517
8.561.1 Field Documentation . . . 517
8.561.1.1 DeviceError . . . 517
8.561.1.2 DeviceResult . . . 517
8.561.1.3 TlvPresenceMask . . . 517
8.561.1.4 TlvResultCode . . . 517
8.562 qos_BindDataPortMuxID_t Struct Reference . . . 517
8.562.1 Detailed Description . . . 518
8.562.2 Field Documentation . . . 518

8.562.2.1 MuxID . . . 518
8.563 qos_BindDataPortPeripheralEndPointID_t Struct Reference . . . 518
8.563.1 Detailed Description . . . 518
8.563.2 Field Documentation . . . 518
8.563.2.1 EndPointType . . . 518
8.563.2.2 IfaceID . . . 518
8.564 qos_BindDataPortSIODataPort_t Struct Reference . . . 518
8.564.1 Detailed Description . . . 519
8.564.2 Field Documentation . . . 519
8.564.2.1 SIODataPort . . . 519
8.565 RFBandInfoElements Struct Reference . . . 519
8.565.1 Detailed Description . . . 519
8.565.2 Field Documentation . . . 519
8.565.2.1 activeBandClass . . . 519
8.565.2.2 activeChannel . . . 519
8.565.2.3 radioInterface . . . 519
8.566 rmTrasnferStaticsReq Struct Reference . . . 519
8.566.1 Detailed Description . . . 520
8.566.2 Field Documentation . . . 520
8.566.2.1 bResetStatistics . . . 520
8.566.2.2 ulMask . . . 520
8.567 sensorData_t Struct Reference . . . 520
8.567.1 Detailed Description . . . 520
8.567.2 Field Documentation . . . 521
8.567.2.1 flags . . . 521
8.567.2.2 sensorDataLen . . . 521
8.567.2.3 timeOfFirstSample . . . 521
8.567.2.4 timeOffset . . . 521
8.567.2.5 xAxis . . . 521
8.567.2.6 yAxis . . . 521
8.567.2.7 zAxis . . . 521
8.568 slot_t Struct Reference . . . 522
8.568.1 Detailed Description . . . 522
8.568.2 Field Documentation . . . 522
8.568.2.1 bICCID . . . 522
8.568.2.2 bICCIDLength . . . 522
8.568.2.3 bLogicalSlot . . . 522
8.568.2.4 uPhyCardStatus . . . 522
8.568.2.5 uPhySlotStatus . . . 523
8.569 slotInf Struct Reference . . . 523

8.569.1 Detailed Description . . . 523
8.569.2 Field Documentation . . . 524
8.569.2.1 AppStatus . . . 524
8.569.2.2 cardState . . . 524
8.569.2.3 errorState . . . 524
8.569.2.4 numApp . . . 524
8.569.2.5 upinRetries . . . 524
8.569.2.6 upinState . . . 524
8.569.2.7 upukRetries . . . 524
8.570 slots_t Struct Reference . . . 524
8.570.1 Detailed Description . . . 524
8.570.2 Field Documentation . . . 525
8.570.2.1 uimSlotStatus . . . 525
8.571 sms_BroadcastConfig Struct Reference . . . 525
8.571.1 Detailed Description . . . 525
8.571.2 Field Documentation . . . 525
8.571.2.1 fromServiceId . . . 525
8.571.2.2 selected . . . 525
8.571.2.3 toServiceId . . . 525
8.572 sms_CDMABroadcastConfig Struct Reference . . . 525
8.572.1 Detailed Description . . . 526
8.572.2 Field Documentation . . . 526
8.572.2.1 language . . . 526
8.572.2.2 selected . . . 526
8.572.2.3 serviceCategory . . . 526
8.573 sms_getIndicationReg Struct Reference . . . 526
8.573.1 Detailed Description . . . 526
8.573.2 Field Documentation . . . 527
8.573.2.1 pRegCallStatInfoEvt . . . 527
8.573.2.2 pRegTransLayerInfoEvt . . . 527
8.573.2.3 pRegTransNWRegInfoEvt . . . 527
8.574 sms_getMsgWaitingInfo Struct Reference . . . 527
8.574.1 Detailed Description . . . 527
8.574.2 Field Documentation . . . 527
8.574.2.1 msgWaitInfo . . . 527
8.574.2.2 numInstances . . . 527
8.575 sms_getTransLayerInfo Struct Reference . . . 527
8.575.1 Detailed Description . . . 528
8.575.2 Field Documentation . . . 528
8.575.2.1 pRegInd . . . 528

8.575.2.2 pTransLayerInfo . . . 528
8.576 sms_getTransNWRegInfo Struct Reference . . . 528
8.576.1 Detailed Description . . . 528
8.576.2 Field Documentation . . . 528
8.576.2.1 pRegStatus . . . 529
8.577 sms_maxStorageSizeReq Struct Reference . . . 529
8.577.1 Detailed Description . . . 529
8.577.2 Field Documentation . . . 529
8.577.2.1 pMessageMode . . . 529
8.577.2.2 storageType . . . 529
8.578 sms_maxStorageSizeResp Struct Reference . . . 529
8.578.1 Detailed Description . . . 529
8.578.2 Field Documentation . . . 530
8.578.2.1 freeSlots . . . 530
8.578.2.2 maxStorageSize . . . 530
8.579 sms_messageWaitingInfoContent Struct Reference . . . 530
8.579.1 Detailed Description . . . 530
8.579.2 Field Documentation . . . 530
8.579.2.1 activeInd . . . 530
8.579.2.2 msgCount . . . 530
8.579.2.3 msgType . . . 530
8.580 sms_msgProtocolResp Struct Reference . . . 531
8.580.1 Detailed Description . . . 531
8.580.2 Field Documentation . . . 531
8.580.2.1 msgProtocol . . . 531
8.581 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference . . . 531
8.581.1 Detailed Description . . . 531
8.581.2 Field Documentation . . . 532
8.581.2.1 activated_ind . . . 532
8.581.2.2 CDMABroadcastConfig . . . 532
8.581.2.3 num_instances . . . 532
8.582 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference . . . 532
8.582.1 Detailed Description . . . 532
8.582.2 Field Documentation . . . 532
8.582.2.1 activated_ind . . . 532
8.582.2.2 broadcastConfig . . . 532
8.582.2.3 num_instances . . . 532
8.583 sms_routeEntry Struct Reference . . . 532
8.583.1 Detailed Description . . . 533
8.583.2 Field Documentation . . . 533

8.583.2.1 messageClass . . . 533
8.583.2.2 messageType . . . 533
8.583.2.3 receiptAction . . . 533
8.583.2.4 routeStorage . . . 533
8.584 sms_sendAsyncsmsParams Struct Reference . . . 533
8.584.1 Detailed Description . . . 534
8.584.2 Field Documentation . . . 535
8.584.2.1 messageFormat . . . 535
8.584.2.2 messageSize . . . 535
8.584.2.3 pFollowOnDC . . . 535
8.584.2.4 pForceOnDC . . . 535
8.584.2.5 pLinktimer . . . 535
8.584.2.6 pMessage . . . 535
8.584.2.7 pRetryMessage . . . 535
8.584.2.8 pRetryMessageId . . . 535
8.584.2.9 pServiceOption . . . 535
8.584.2.10 pSmsOnIms . . . 535
8.584.2.11 pUserData . . . 535
8.585 sms_setIndicationReg Struct Reference . . . 536
8.585.1 Detailed Description . . . 536
8.585.2 Field Documentation . . . 536
8.585.2.1 pRegCallStatInfoEvt . . . 536
8.585.2.2 pRegTransLayerInfoEvt . . . 536
8.585.2.3 pRegTransNWRegInfoEvt . . . 536
8.586 sms_setRoutesReq Struct Reference . . . 536
8.586.1 Detailed Description . . . 537
8.586.2 Field Documentation . . . 537
8.586.2.1 numOfRoutes . . . 537
8.586.2.2 pTransferStatusReport . . . 537
8.586.2.3 routeList . . . 537
8.587 sms_transLayerInfo Struct Reference . . . 537
8.587.1 Detailed Description . . . 537
8.587.2 Field Documentation . . . 537
8.587.2.1 TransCap . . . 538
8.587.2.2 TransType . . . 538
8.588 sMSCAddressInfo Struct Reference . . . 538
8.588.1 Detailed Description . . . 538
8.588.2 Field Documentation . . . 538
8.588.2.1 data . . . 538
8.588.2.2 length . . . 538

8.589 sMSCAddressTlv Struct Reference . . . 538
8.589.1 Detailed Description . . . 538
8.589.2 Field Documentation . . . 538
8.589.2.1 SMSCInfo . . . 539
8.589.2.2 TlvPresent . . . 539
8.590 sMSEtwsMessageInfo Struct Reference . . . 539
8.590.1 Detailed Description . . . 539
8.590.2 Field Documentation . . . 539
8.590.2.1 data . . . 539
8.590.2.2 length . . . 539
8.590.2.3 notificationType . . . 539
8.591 sMSEtwsMessageTlv Struct Reference . . . 539
8.591.1 Detailed Description . . . 539
8.591.2 Field Documentation . . . 540
8.591.2.1 EtwsMessageInfo . . . 540
8.591.2.2 TlvPresent . . . 540
8.592 sMSEtwsPlmnInfo Struct Reference . . . 540
8.592.1 Detailed Description . . . 540
8.592.2 Field Documentation . . . 540
8.592.2.1 mobileCountryCode . . . 540
8.592.2.2 mobileNetworkCode . . . 540
8.593 sMSMessageModeInfo Struct Reference . . . 540
8.593.1 Detailed Description . . . 540
8.593.2 Field Documentation . . . 541
8.593.2.1 messageMode . . . 541
8.594 sMSMTMessageInfo Struct Reference . . . 541
8.594.1 Detailed Description . . . 541
8.594.2 Field Documentation . . . 541
8.594.2.1 messageIndex . . . 541
8.594.2.2 storageType . . . 541
8.595 sMSOnIMSInfo Struct Reference . . . 541
8.595.1 Detailed Description . . . 541
8.595.2 Field Documentation . . . 541
8.595.2.1 smsOnIMS . . . 541
8.596 sMSOnIMSTlv Struct Reference . . . 542
8.596.1 Detailed Description . . . 542
8.596.2 Field Documentation . . . 542
8.596.2.1 IMSInfo . . . 542
8.596.2.2 TlvPresent . . . 542
8.597 sMSTransferRouteMTMessageInfo Struct Reference . . . 542

8.597.1 Detailed Description . . . 542
8.597.2 Field Documentation . . . 543
8.597.2.1 ackIndicator . . . 543
8.597.2.2 data . . . 543
8.597.2.3 format . . . 543
8.597.2.4 length . . . 543
8.597.2.5 transactionID . . . 543
8.598 swi_uint256_t Struct Reference . . . 543
8.598.1 Detailed Description . . . 543
8.598.2 Field Documentation . . . 543
8.598.2.1 word . . . 543
8.599 swiaudio_PCMparams Struct Reference . . . 543
8.599.1 Detailed Description . . . 543
8.599.2 Field Documentation . . . 544
8.599.2.1 iFaceTab . . . 544
8.599.2.2 iFaceTabLen . . . 544
8.600 swidms_ehrpdMTUSizeTlv Struct Reference . . . 544
8.600.1 Detailed Description . . . 544
8.600.2 Field Documentation . . . 544
8.600.2.1 ehrpdMTUSize . . . 544
8.600.2.2 TlvPresent . . . 544
8.601 swidms_hrpdMTUSizeTlv Struct Reference . . . 544
8.601.1 Detailed Description . . . 544
8.601.2 Field Documentation . . . 545
8.601.2.1 hrpdMTUSize . . . 545
8.601.2.2 TlvPresent . . . 545
8.602 swidms_intfaceCfgTlv Struct Reference . . . 545
8.602.1 Detailed Description . . . 545
8.602.2 Field Documentation . . . 546
8.602.2.1 CfgValue . . . 546
8.602.2.2 CurrentCfgType . . . 546
8.602.2.3 TlvPresent . . . 546
8.603 swidms_mtuSize3gppTlv Struct Reference . . . 546
8.603.1 Detailed Description . . . 546
8.603.2 Field Documentation . . . 547
8.603.2.1 MTUSize3gpp . . . 547
8.603.2.2 TlvPresent . . . 547
8.604 swidms_supportedIntBitmaskTlv Struct Reference . . . 547
8.604.1 Detailed Description . . . 547
8.604.2 Field Documentation . . . 547

8.604.2.1 TlvPresent . . . 547
8.604.2.2 ValidBitmasks . . . 547
8.605 swidms_usbMTUSizeTlv Struct Reference . . . 547
8.605.1 Detailed Description . . . 547
8.605.2 Field Documentation . . . 548
8.605.2.1 TlvPresent . . . 548
8.605.2.2 UsbMTUSize . . . 548
8.606 tdscdmaSigInfoExt Struct Reference . . . 548
8.606.1 Detailed Description . . . 548
8.606.2 Field Documentation . . . 548
8.606.2.1 ecio . . . 548
8.606.2.2 rscp . . . 548
8.606.2.3 rssi . . . 548
8.606.2.4 sinr . . . 548
8.607 tempData_t Struct Reference . . . 548
8.607.1 Detailed Description . . . 549
8.607.2 Field Documentation . . . 549
8.607.2.1 temperature . . . 549
8.607.2.2 temperatureDataLen . . . 549
8.607.2.3 timeOfFirstSample . . . 549
8.607.2.4 timeOffset . . . 549
8.607.2.5 timeSource . . . 549
8.608 tmd_mitigationDevList Struct Reference . . . 550
8.608.1 Detailed Description . . . 550
8.608.2 Field Documentation . . . 550
8.608.2.1 maxMitigationLevel . . . 550
8.608.2.2 mitigationDevId . . . 550
8.608.2.3 mitigationDevIdLen . . . 550
8.609 transferRouteMessageTlv Struct Reference . . . 550
8.609.1 Detailed Description . . . 550
8.609.2 Field Documentation . . . 551
8.609.2.1 TlvPresent . . . 551
8.609.2.2 TransferRouteMTMessageInfo . . . 551
8.610 uim_appStatus Struct Reference . . . 551
8.610.1 Detailed Description . . . 551
8.610.2 Field Documentation . . . 553
8.610.2.1 aidLength . . . 553
8.610.2.2 aidVal . . . 553
8.610.2.3 appState . . . 553
8.610.2.4 appType . . . 553

8.610.2.5 persoFeature . . . 553
8.610.2.6 persoRetries . . . 553
8.610.2.7 persoState . . . 553
8.610.2.8 persoUnblockRetries . . . 553
8.610.2.9 pin1Retries . . . 553
8.610.2.10 pin1State . . . 553
8.610.2.11 pin2Retries . . . 554
8.610.2.12 pin2State . . . 554
8.610.2.13 puk1Retries . . . 554
8.610.2.14 puk2Retries . . . 554
8.610.2.15 univPin . . . 554
8.611 uim_authenticateResult Struct Reference . . . 554
8.611.1 Detailed Description . . . 554
8.611.2 Field Documentation . . . 554
8.611.2.1 content . . . 554
8.611.2.2 contentLen . . . 554
8.612 uim_authenticationData Struct Reference . . . 554
8.612.1 Detailed Description . . . 554
8.612.2 Field Documentation . . . 555
8.612.2.1 context . . . 555
8.612.2.2 data . . . 555
8.612.2.3 dataLen . . . 555
8.613 uim_cardResult Struct Reference . . . 555
8.613.1 Detailed Description . . . 555
8.613.2 Field Documentation . . . 556
8.613.2.1 sw1 . . . 556
8.613.2.2 sw2 . . . 556
8.614 uim_cardStatus Struct Reference . . . 556
8.614.1 Detailed Description . . . 556
8.614.2 Field Documentation . . . 557
8.614.2.1 index1xPri . . . 557
8.614.2.2 index1xSec . . . 557
8.614.2.3 indexGwPri . . . 557
8.614.2.4 indexGwSec . . . 557
8.614.2.5 numSlot . . . 557
8.614.2.6 SlotInfo . . . 557
8.615 uim_changeUIMPIN Struct Reference . . . 557
8.615.1 Detailed Description . . . 557
8.615.2 Field Documentation . . . 558
8.615.2.1 oldPINLen . . . 558

8.615.2.2 oldPINVal . . . 558
8.615.2.3 pinID . . . 558
8.615.2.4 pinLen . . . 558
8.615.2.5 pinVal . . . 558
8.616 uim_depersonalizationInformation Struct Reference . . . 558
8.616.1 Detailed Description . . . 558
8.616.2 Field Documentation . . . 559
8.616.2.1 ckLen . . . 559
8.616.2.2 ckVal . . . 559
8.616.2.3 feature . . . 559
8.616.2.4 operation . . . 559
8.617 uim_encryptedPIN1 Struct Reference . . . 559
8.617.1 Detailed Description . . . 559
8.617.2 Field Documentation . . . 560
8.617.2.1 pin1Len . . . 560
8.617.2.2 pin1Val . . . 560
8.618 uim_fileAttributes Struct Reference . . . 560
8.618.1 Detailed Description . . . 560
8.618.2 Field Documentation . . . 563
8.618.2.1 fileID . . . 563
8.618.2.2 fileSize . . . 563
8.618.2.3 fileType . . . 563
8.618.2.4 rawLen . . . 563
8.618.2.5 rawValue . . . 563
8.618.2.6 recordCount . . . 563
8.618.2.7 recordSize . . . 563
8.618.2.8 secActivate . . . 563
8.618.2.9 secActivateMask . . . 563
8.618.2.10 secDeactivate . . . 563
8.618.2.11 secDeactivateMask . . . 563
8.618.2.12 secIncrease . . . 563
8.618.2.13 secIncreaseMask . . . 563
8.618.2.14 secRead . . . 563
8.618.2.15 secReadMask . . . 563
8.618.2.16 secWrite . . . 563
8.618.2.17 secWriteMask . . . 563
8.619 uim_fileInfo Struct Reference . . . 563
8.619.1 Detailed Description . . . 563
8.619.2 Field Documentation . . . 564
8.619.2.1 fileID . . . 564

8.619.2.2 path . . . 564
8.619.2.3 pathLen . . . 564
8.620 uim_hotSwapStatus Struct Reference . . . 564
8.620.1 Detailed Description . . . 564
8.620.2 Field Documentation . . . 564
8.620.2.1 hotSwap . . . 564
8.620.2.2 hotSwapLength . . . 564
8.621 uim_personalizationStatus Struct Reference . . . 564
8.621.1 Detailed Description . . . 565
8.621.2 Field Documentation . . . 565
8.621.2.1 feature . . . 565
8.621.2.2 numFeatures . . . 565
8.621.2.3 unblockLeft . . . 565
8.621.2.4 verifyLeft . . . 565
8.622 uim_readResult Struct Reference . . . 565
8.622.1 Detailed Description . . . 566
8.622.2 Field Documentation . . . 566
8.622.2.1 content . . . 566
8.622.2.2 contentLen . . . 566
8.623 uim_readTransparentInfo Struct Reference . . . 566
8.623.1 Detailed Description . . . 566
8.623.2 Field Documentation . . . 566
8.623.2.1 length . . . 566
8.623.2.2 offset . . . 566
8.624 uim_refreshevent Struct Reference . . . 566
8.624.1 Detailed Description . . . 567
8.624.2 Field Documentation . . . 568
8.624.2.1 aid . . . 568
8.624.2.2 aidLength . . . 568
8.624.2.3 arrfileInfo . . . 568
8.624.2.4 mode . . . 568
8.624.2.5 numOfFiles . . . 568
8.624.2.6 sessionType . . . 568
8.624.2.7 stage . . . 568
8.625 uim_registerRefresh Struct Reference . . . 568
8.625.1 Detailed Description . . . 568
8.625.2 Field Documentation . . . 569
8.625.2.1 arrfileInfo . . . 569
8.625.2.2 numFiles . . . 569
8.625.2.3 registerFlag . . . 569

8.625.2.4 voteForInit . . . 569
8.626 uim_remainingRetries Struct Reference . . . 569
8.626.1 Detailed Description . . . 569
8.626.2 Field Documentation . . . 569
8.626.2.1 unblockLeft . . . 569
8.626.2.2 verifyLeft . . . 570
8.627 uim_sessionInformation Struct Reference . . . 570
8.627.1 Detailed Description . . . 570
8.627.2 Field Documentation . . . 570
8.627.2.1 aid . . . 570
8.627.2.2 aidLength . . . 570
8.627.2.3 sessionType . . . 570
8.628 uim_setPINProtection Struct Reference . . . 570
8.628.1 Detailed Description . . . 571
8.628.2 Field Documentation . . . 571
8.628.2.1 pinID . . . 571
8.628.2.2 pinLength . . . 571
8.628.2.3 pinOperation . . . 571
8.628.2.4 pinValue . . . 571
8.629 uim_slotInfo Struct Reference . . . 571
8.629.1 Detailed Description . . . 572
8.629.2 Field Documentation . . . 573
8.629.2.1 AppStatus . . . 573
8.629.2.2 cardState . . . 573
8.629.2.3 errorState . . . 573
8.629.2.4 numApp . . . 573
8.629.2.5 upinRetries . . . 573
8.629.2.6 upinState . . . 573
8.629.2.7 upukRetries . . . 573
8.630 uim_UIMSessionInformation Struct Reference . . . 573
8.630.1 Detailed Description . . . 573
8.630.2 Field Documentation . . . 574
8.630.2.1 aid . . . 574
8.630.2.2 aidLength . . . 574
8.630.2.3 sessionType . . . 574
8.631 uim_unblockUIMPIN Struct Reference . . . 574
8.631.1 Detailed Description . . . 574
8.631.2 Field Documentation . . . 574
8.631.2.1 newPINLen . . . 574
8.631.2.2 newPINVal . . . 574

8.631.2.3 pinID . . . 574
8.631.2.4 pukLen . . . 575
8.631.2.5 pukVal . . . 575
8.632 uim_verifyUIMPIN Struct Reference . . . 575
8.632.1 Detailed Description . . . 575
8.632.2 Field Documentation . . . 575
8.632.2.1 pinID . . . 575
8.632.2.2 pinLen . . . 575
8.632.2.3 pinVal . . . 575
8.633 unpack_audio_SLQSGetAudioPathConfig_t Struct Reference . . . 575
8.633.1 Detailed Description . . . 576
8.633.2 Field Documentation . . . 577
8.633.2.1 ParamPresenceMask . . . 577
8.633.2.2 pCodecSTGain . . . 577
8.633.2.3 pDTMFTXGain . . . 577
8.633.2.4 pECMode . . . 577
8.633.2.5 pMICGainSelect . . . 577
8.633.2.6 pNSEnable . . . 577
8.633.2.7 pRXAGCList . . . 577
8.633.2.8 pRXAVCAGCSwitch . . . 577
8.633.2.9 pRXAVCList . . . 577
8.633.2.10 pRXPCMIIRFltr . . . 577
8.633.2.11 pTXAGCList . . . 577
8.633.2.12 pTXAVCSwitch . . . 577
8.633.2.13 pTXGain . . . 577
8.633.2.14 pTXPCMIIRFltr . . . 578
8.634 unpack_audio_SLQSGetAudioProfile_t Struct Reference . . . 578
8.634.1 Detailed Description . . . 578
8.634.2 Field Documentation . . . 578
8.634.2.1 EarMute . . . 579
8.634.2.2 MicMute . . . 579
8.634.2.3 ParamPresenceMask . . . 579
8.634.2.4 Profile . . . 579
8.634.2.5 Volume . . . 579
8.635 unpack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference . . . 579
8.635.1 Detailed Description . . . 579
8.635.2 Field Documentation . . . 579
8.635.2.1 ParamPresenceMask . . . 579
8.635.2.2 ResCode . . . 579
8.636 unpack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference . . . 579

8.636.1 Detailed Description . . . 579
8.636.2 Field Documentation . . . 580
8.636.2.1 ParamPresenceMask . . . 580
8.636.2.2 ResCode . . . 580
8.637 unpack_cat_SetCatEventCallback_ind_t Struct Reference . . . 580
8.637.1 Detailed Description . . . 580
8.637.2 Field Documentation . . . 580
8.637.2.1 CCETlv . . . 580
8.637.2.2 event_Index . . . 580
8.637.2.3 ParamPresenceMask . . . 580
8.638 unpack_cat_SetCATEventCallback_t Struct Reference . . . 580
8.638.1 Detailed Description . . . 581
8.638.2 Field Documentation . . . 581
8.638.2.1 errorMask . . . 581
8.638.2.2 ParamPresenceMask . . . 581
8.638.2.3 Tlvresult . . . 581
8.639 unpack_dms_GetActivationState_t Struct Reference . . . 581
8.639.1 Detailed Description . . . 581
8.639.2 Field Documentation . . . 582
8.639.2.1 ParamPresenceMask . . . 582
8.639.2.2 state . . . 582
8.640 unpack_dms_GetBandCapability_t Struct Reference . . . 582
8.640.1 Detailed Description . . . 582
8.640.2 Field Documentation . . . 584
8.640.2.1 BandCapability . . . 584
8.640.2.2 ParamPresenceMask . . . 584
8.640.2.3 Tlvresult . . . 584
8.641 unpack_dms_GetCrashAction_t Struct Reference . . . 584
8.641.1 Detailed Description . . . 584
8.641.2 Field Documentation . . . 584
8.641.2.1 DevCrashState . . . 584
8.641.2.2 ParamPresenceMask . . . 585
8.641.2.3 Tlvresult . . . 585
8.642 unpack_dms_GetCustFeature_t Struct Reference . . . 585
8.642.1 Detailed Description . . . 585
8.642.2 Field Documentation . . . 587
8.642.2.1 DHCPRelayEnabled . . . 587
8.642.2.2 DisableIMSI . . . 587
8.642.2.3 GpsEnable . . . 587
8.642.2.4 GPSLPM . . . 587

8.642.2.5 GPSSel . . . 587
8.642.2.6 IPFamSupport . . . 587
8.642.2.7 IsVoiceEnabled . . . 587
8.642.2.8 ParamPresenceMask . . . 587
8.642.2.9 RMAutoConnect . . . 587
8.642.2.10 SMSSupport . . . 587
8.642.2.11 Tlvresult . . . 587
8.643 unpack_dms_GetCustFeaturesV2_t Struct Reference . . . 587
8.643.1 Detailed Description . . . 587
8.643.2 Field Documentation . . . 588
8.643.2.1 GetCustomFeatureV2 . . . 588
8.643.2.2 ParamPresenceMask . . . 588
8.643.2.3 Tlvresult . . . 588
8.644 unpack_dms_GetDeviceCap_t Struct Reference . . . 588
8.644.1 Detailed Description . . . 588
8.644.2 Field Documentation . . . 589
8.644.2.1 DataServiceCapability . . . 589
8.644.2.2 MaxRXChannelRate . . . 589
8.644.2.3 MaxTXChannelRate . . . 589
8.644.2.4 ParamPresenceMask . . . 589
8.644.2.5 RadioIfaces . . . 589
8.644.2.6 RadioIfacesSize . . . 589
8.644.2.7 SimCapability . . . 589
8.644.2.8 Tlvresult . . . 589
8.645 unpack_dms_GetDeviceCapabilities_t Struct Reference . . . 589
8.645.1 Detailed Description . . . 590
8.645.2 Field Documentation . . . 590
8.645.2.1 dataServiceCaCapability . . . 591
8.645.2.2 maxRxChannelRate . . . 591
8.645.2.3 maxTxChannelRate . . . 591
8.645.2.4 ParamPresenceMask . . . 591
8.645.2.5 RadioIfaces . . . 591
8.645.2.6 radioIfacesSize . . . 591
8.645.2.7 simCapability . . . 591
8.646 unpack_dms_GetDeviceHardwareRev_t Struct Reference . . . 591
8.646.1 Detailed Description . . . 591
8.646.2 Field Documentation . . . 591
8.646.2.1 ParamPresenceMask . . . 591
8.646.2.2 String . . . 591
8.646.2.3 stringSize . . . 591

8.646.2.4 Tlvresult . . . 592
8.647 unpack_dms_GetDeviceMfr_t Struct Reference . . . 592
8.647.1 Detailed Description . . . 592
8.647.2 Field Documentation . . . 592
8.647.2.1 ParamPresenceMask . . . 592
8.647.2.2 String . . . 592
8.647.2.3 stringSize . . . 592
8.647.2.4 Tlvresult . . . 592
8.648 unpack_dms_GetDeviceSerialNumbers_t Struct Reference . . . 592
8.648.1 Detailed Description . . . 593
8.648.2 Field Documentation . . . 594
8.648.2.1 esnSize . . . 594
8.648.2.2 ESNString . . . 594
8.648.2.3 imeiSize . . . 594
8.648.2.4 IMEIString . . . 594
8.648.2.5 imeiSvnSize . . . 594
8.648.2.6 ImeiSvnString . . . 594
8.648.2.7 meidSize . . . 594
8.648.2.8 MEIDString . . . 594
8.648.2.9 ParamPresenceMask . . . 594
8.648.2.10 Tlvresult . . . 594
8.649 unpack_dms_GetFirmwareInfo_t Struct Reference . . . 594
8.649.1 Detailed Description . . . 595
8.649.2 Field Documentation . . . 595
8.649.2.1 appversion_str . . . 595
8.649.2.2 bootversion_str . . . 595
8.649.2.3 carrier_str . . . 596
8.649.2.4 cur_carr_name . . . 596
8.649.2.5 cur_carr_rev . . . 596
8.649.2.6 modelid_str . . . 596
8.649.2.7 packageid_str . . . 596
8.649.2.8 ParamPresenceMask . . . 596
8.649.2.9 priversion_str . . . 596
8.649.2.10 sku_str . . . 596
8.649.2.11 Tlvresult . . . 596
8.650 unpack_dms_GetFirmwareRevision_t Struct Reference . . . 596
8.650.1 Detailed Description . . . 596
8.650.2 Field Documentation . . . 597
8.650.2.1 amssSize . . . 597
8.650.2.2 AMSSString . . . 597

8.650.2.3 ParamPresenceMask . . . 597
8.650.2.4 PRIString . . . 597
8.650.2.5 Tlvresult . . . 597
8.651 unpack_dms_GetFirmwareRevisions_t Struct Reference . . . 597
8.651.1 Detailed Description . . . 597
8.651.2 Field Documentation . . . 598
8.651.2.1 amssSize . . . 598
8.651.2.2 AMSSString . . . 598
8.651.2.3 bootSize . . . 598
8.651.2.4 BootString . . . 598
8.651.2.5 ParamPresenceMask . . . 598
8.651.2.6 priSize . . . 598
8.651.2.7 PRIString . . . 598
8.651.2.8 Tlvresult . . . 598
8.652 unpack_dms_GetFSN_t Struct Reference . . . 598
8.652.1 Detailed Description . . . 598
8.652.2 Field Documentation . . . 599
8.652.2.1 ParamPresenceMask . . . 599
8.652.2.2 String . . . 599
8.652.2.3 Tlvresult . . . 599
8.653 unpack_dms_GetHardwareRevision_t Struct Reference . . . 599
8.653.1 Detailed Description . . . 599
8.653.2 Field Documentation . . . 599
8.653.2.1 hwVer . . . 599
8.653.2.2 ParamPresenceMask . . . 599
8.654 unpack_dms_GetIMSI_t Struct Reference . . . 599
8.654.1 Detailed Description . . . 599
8.654.2 Field Documentation . . . 600
8.654.2.1 imsi . . . 600
8.654.2.2 ParamPresenceMask . . . 600
8.654.2.3 Tlvresult . . . 600
8.655 unpack_dms_GetManufacturer_t Struct Reference . . . 600
8.655.1 Detailed Description . . . 600
8.655.2 Field Documentation . . . 600
8.655.2.1 manufacturer . . . 600
8.655.2.2 ParamPresenceMask . . . 600
8.655.2.3 Tlvresult . . . 600
8.656 unpack_dms_GetModelID_t Struct Reference . . . 601
8.656.1 Detailed Description . . . 601
8.656.2 Field Documentation . . . 601

8.656.2.1 modelid . . . 601
8.656.2.2 ParamPresenceMask . . . 601
8.656.2.3 Tlvresult . . . 601
8.657 unpack_dms_GetNetworkTime_t Struct Reference . . . 601
8.657.1 Detailed Description . . . 601
8.657.2 Field Documentation . . . 602
8.657.2.1 ParamPresenceMask . . . 602
8.657.2.2 source . . . 602
8.657.2.3 timestamp . . . 602
8.657.2.4 Tlvresult . . . 602
8.658 unpack_dms_GetOfflineReason_t Struct Reference . . . 602
8.658.1 Detailed Description . . . 602
8.658.2 Field Documentation . . . 603
8.658.2.1 ParamPresenceMask . . . 603
8.658.2.2 pbPlatform . . . 603
8.658.2.3 pReasonMask . . . 603
8.658.2.4 Tlvresult . . . 603
8.659 unpack_dms_GetPower_t Struct Reference . . . 603
8.659.1 Detailed Description . . . 603
8.659.2 Field Documentation . . . 604
8.659.2.1 HardwareControlledMode . . . 604
8.659.2.2 OfflineReason . . . 604
8.659.2.3 OperationMode . . . 604
8.659.2.4 ParamPresenceMask . . . 604
8.659.2.5 Tlvresult . . . 604
8.660 unpack_dms_GetPRLVersion_t Struct Reference . . . 604
8.660.1 Detailed Description . . . 604
8.660.2 Field Documentation . . . 605
8.660.2.1 ParamPresenceMask . . . 605
8.660.2.2 Tlvresult . . . 605
8.660.2.3 u16PRLVersion . . . 605
8.660.2.4 u8PRLPreference . . . 605
8.661 unpack_dms_GetSerialNumbers_t Struct Reference . . . 605
8.661.1 Detailed Description . . . 605
8.661.2 Field Documentation . . . 606
8.661.2.1 esn . . . 606
8.661.2.2 imei_no . . . 606
8.661.2.3 imeisv_svn . . . 606
8.661.2.4 meid . . . 606
8.661.2.5 ParamPresenceMask . . . 606

8.662 unpack_dms_GetUSBComp_t Struct Reference . . . 606
8.662.1 Detailed Description . . . 606
8.662.2 Field Documentation . . . 608
8.662.2.1 NumSupUSBComps . . . 608
8.662.2.2 ParamPresenceMask . . . 608
8.662.2.3 SupUSBComps . . . 608
8.662.2.4 Tlvresult . . . 608
8.662.2.5 USBComp . . . 608
8.663 unpack_dms_GetVoiceNumber_t Struct Reference . . . 608
8.663.1 Detailed Description . . . 609
8.663.2 Field Documentation . . . 609
8.663.2.1 MIN . . . 609
8.663.2.2 minSize . . . 609
8.663.2.3 ParamPresenceMask . . . 609
8.663.2.4 Tlvresult . . . 609
8.663.2.5 VoiceNumber . . . 609
8.663.2.6 voiceNumberSize . . . 609
8.664 unpack_dms_PSMCfgChange_ind_t Struct Reference . . . 610
8.664.1 Detailed Description . . . 610
8.664.2 Field Documentation . . . 610
8.664.2.1 ActiveTimer . . . 610
8.664.2.2 EnableState . . . 610
8.664.2.3 ParamPresenceMask . . . 610
8.664.2.4 PeriodicUpdateTimer . . . 610
8.664.2.5 Tlvresult . . . 610
8.665 unpack_dms_ResetToFactoryDefaults_t Struct Reference . . . 610
8.665.1 Detailed Description . . . 611
8.665.2 Field Documentation . . . 611
8.665.2.1 ParamPresenceMask . . . 611
8.665.2.2 Tlvresult . . . 611
8.666 unpack_dms_SetActivationStatusCallback_t Struct Reference . . . 611
8.666.1 Detailed Description . . . 611
8.666.2 Field Documentation . . . 611
8.666.2.1 ParamPresenceMask . . . 611
8.666.2.2 Tlvresult . . . 611
8.667 unpack_dms_SetCrashAction_t Struct Reference . . . 611
8.667.1 Detailed Description . . . 612
8.667.2 Field Documentation . . . 612
8.667.2.1 notused . . . 612
8.667.2.2 ParamPresenceMask . . . 612

8.668 unpack_dms_SetCustFeature_t Struct Reference . . . 612
8.668.1 Detailed Description . . . 612
8.668.2 Field Documentation . . . 612
8.668.2.1 ParamPresenceMask . . . 612
8.668.2.2 Tlvresult . . . 612
8.669 unpack_dms_SetCustFeaturesV2_t Struct Reference . . . 612
8.669.1 Detailed Description . . . 613
8.669.2 Field Documentation . . . 613
8.669.2.1 ParamPresenceMask . . . 613
8.669.2.2 Tlvresult . . . 613
8.670 unpack_dms_SetEventReport_ind_t Struct Reference . . . 613
8.670.1 Detailed Description . . . 613
8.670.2 Field Documentation . . . 613
8.670.2.1 ActivationStatusTlv . . . 613
8.670.2.2 OperatingModeTlv . . . 614
8.670.2.3 ParamPresenceMask . . . 614
8.670.2.4 Tlvresult . . . 614
8.671 unpack_dms_SetEventReport_t Struct Reference . . . 614
8.671.1 Detailed Description . . . 614
8.671.2 Field Documentation . . . 614
8.671.2.1 ParamPresenceMask . . . 614
8.671.2.2 Tlvresult . . . 614
8.672 unpack_dms_SetFirmwarePreference_t Struct Reference . . . 614
8.672.1 Detailed Description . . . 614
8.672.2 Field Documentation . . . 615
8.672.2.1 ParamPresenceMask . . . 615
8.672.2.2 Tlvresult . . . 615
8.673 unpack_dms_SetIndicationRegister_t Struct Reference . . . 615
8.673.1 Detailed Description . . . 615
8.673.2 Field Documentation . . . 615
8.673.2.1 ParamPresenceMask . . . 615
8.673.2.2 Tlvresult . . . 615
8.674 unpack_dms_SetPower_t Struct Reference . . . 615
8.674.1 Detailed Description . . . 615
8.674.2 Field Documentation . . . 616
8.674.2.1 ParamPresenceMask . . . 616
8.674.2.2 Tlvresult . . . 616
8.675 unpack_dms_SetUSBComp_t Struct Reference . . . 616
8.675.1 Detailed Description . . . 616
8.675.2 Field Documentation . . . 616

8.675.2.1 ParamPresenceMask . . . 616
8.675.2.2 Tlvresult . . . 616
8.676 unpack_dms_SLQSDmsSwiGetPCInfo_t Struct Reference . . . 616
8.676.1 Detailed Description . . . 616
8.676.2 Field Documentation . . . 618
8.676.2.1 has_LpmFlag . . . 618
8.676.2.2 has_PersistentLpm . . . 618
8.676.2.3 has_PowerOffMode . . . 618
8.676.2.4 has_Wdisable . . . 619
8.676.2.5 LpmFlag . . . 619
8.676.2.6 opMode . . . 619
8.676.2.7 ParamPresenceMask . . . 619
8.676.2.8 PersistentLpm . . . 619
8.676.2.9 PowerOffMode . . . 619
8.676.2.10 Wdisable . . . 619
8.677 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference . . . 619
8.677.1 Detailed Description . . . 619
8.677.2 Field Documentation . . . 620
8.677.2.1 ParamPresenceMask . . . 620
8.677.2.2 source . . . 620
8.677.2.3 Tlvresult . . . 620
8.677.2.4 type . . . 620
8.678 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference . . . 620
8.678.1 Detailed Description . . . 620
8.678.2 Field Documentation . . . 621
8.678.2.1 ParamPresenceMask . . . 621
8.678.2.2 source . . . 621
8.678.2.3 Tlvresult . . . 621
8.678.2.4 type . . . 621
8.679 unpack_dms_SLQSDmsSwiGetUimSelection_t Struct Reference . . . 621
8.679.1 Detailed Description . . . 621
8.679.2 Field Documentation . . . 622
8.679.2.1 ParamPresenceMask . . . 622
8.679.2.2 pUimAutoSwitchActSlot . . . 622
8.679.2.3 uimSelect . . . 622
8.680 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 622
8.680.1 Detailed Description . . . 622
8.680.2 Field Documentation . . . 623
8.680.2.1 ParamPresenceMask . . . 623
8.680.2.2 Tlvresult . . . 623

8.681 unpack_dms_SLQSGetBandCapability_t Struct Reference . . . 623
8.681.1 Detailed Description . . . 623
8.681.2 Field Documentation . . . 627
8.681.2.1 bandCapability . . . 627
8.681.2.2 is_LteBandCapability_Available . . . 627
8.681.2.3 is_TdsBandCapability_Available . . . 627
8.681.2.4 LteBandCapability . . . 627
8.681.2.5 ParamPresenceMask . . . 627
8.681.2.6 TdsBandCapability . . . 627
8.682 unpack_dms_SLQSGetERIFile_t Struct Reference . . . 627
8.682.1 Detailed Description . . . 627
8.682.2 Field Documentation . . . 628
8.682.2.1 eriFile . . . 628
8.682.2.2 ParamPresenceMask . . . 628
8.682.2.3 Tlvresult . . . 628
8.683 unpack_dms_SLQSGetPowerSaveModeConfig_t Struct Reference . . . 628
8.683.1 Detailed Description . . . 628
8.683.2 Field Documentation . . . 629
8.683.2.1 pActiveTimer . . . 629
8.683.2.2 ParamPresenceMask . . . 629
8.683.2.3 pDurationDueToOOS . . . 629
8.683.2.4 pDurationThreshold . . . 629
8.683.2.5 pEarlyWakeupTime . . . 629
8.683.2.6 pPeriodicUpdateTimer . . . 629
8.683.2.7 pPsmEnableState . . . 629
8.683.2.8 pRandomizationWindow . . . 629
8.684 unpack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference . . . 629
8.684.1 Detailed Description . . . 629
8.684.2 Field Documentation . . . 630
8.684.2.1 ParamPresenceMask . . . 630
8.684.2.2 Tlvresult . . . 630
8.685 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference . . . 630
8.685.1 Detailed Description . . . 630
8.685.2 Field Documentation . . . 630
8.685.2.1 ParamPresenceMask . . . 630
8.685.2.2 Tlvresult . . . 630
8.686 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 630
8.686.1 Detailed Description . . . 630
8.686.2 Field Documentation . . . 631
8.686.2.1 crashInfoParam . . . 631

8.686.2.2 ParamPresenceMask . . . 631
8.686.2.3 Tlvresult . . . 631
8.687 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference . . . 631
8.687.1 Detailed Description . . . 631
8.687.2 Field Documentation . . . 632
8.687.2.1 ParamPresenceMask . . . 632
8.687.2.2 pGetDyingGaspCfg . . . 632
8.687.2.3 Tlvresult . . . 632
8.688 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference . . . 632
8.688.1 Detailed Description . . . 632
8.688.2 Field Documentation . . . 632
8.688.2.1 ParamPresenceMask . . . 632
8.688.2.2 pGetDyingGaspStatistics . . . 632
8.688.2.3 Tlvresult . . . 632
8.689 unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference . . . 632
8.689.1 Detailed Description . . . 633
8.689.2 Field Documentation . . . 633
8.689.2.1 carrier . . . 633
8.689.2.2 fwvers . . . 633
8.689.2.3 numEntries . . . 633
8.689.2.4 ParamPresenceMask . . . 633
8.689.2.5 pCurrImgInfo . . . 633
8.689.2.6 pkgver . . . 633
8.689.2.7 priver . . . 633
8.690 unpack_dms_SLQSSwiGetFwUpdateStatus_t Struct Reference . . . 633
8.690.1 Detailed Description . . . 634
8.690.2 Field Documentation . . . 635
8.690.2.1 imgType . . . 635
8.690.2.2 logString . . . 635
8.690.2.3 ParamPresenceMask . . . 635
8.690.2.4 refData . . . 635
8.690.2.5 refString . . . 635
8.690.2.6 ResCode . . . 635
8.690.2.7 Tlvresult . . . 635
8.691 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference . . . 635
8.691.1 Detailed Description . . . 635
8.691.2 Field Documentation . . . 636
8.691.2.1 hostID . . . 636
8.691.2.2 manString . . . 636
8.691.2.3 modelString . . . 636

8.691.2.4 ParamPresenceMask . . . 636
8.691.2.5 plasmaIDString . . . 636
8.691.2.6 swVerString . . . 636
8.691.2.7 Tlvresult . . . 636
8.692 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference . . . 636
8.692.1 Detailed Description . . . 637
8.692.2 Field Documentation . . . 637
8.692.2.1 nameString . . . 637
8.692.2.2 ParamPresenceMask . . . 637
8.692.2.3 Tlvresult . . . 637
8.692.2.4 versionString . . . 637
8.693 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference . . . 637
8.693.1 Detailed Description . . . 637
8.693.2 Field Documentation . . . 638
8.693.2.1 meidString . . . 638
8.693.2.2 ParamPresenceMask . . . 638
8.693.2.3 Tlvresult . . . 638
8.694 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 638
8.694.1 Detailed Description . . . 638
8.694.2 Field Documentation . . . 638
8.694.2.1 ParamPresenceMask . . . 638
8.694.2.2 Tlvresult . . . 638
8.695 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 638
8.695.1 Detailed Description . . . 638
8.695.2 Field Documentation . . . 639
8.695.2.1 ParamPresenceMask . . . 639
8.695.2.2 Tlvresult . . . 639
8.696 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 639
8.696.1 Detailed Description . . . 639
8.696.2 Field Documentation . . . 639
8.696.2.1 ParamPresenceMask . . . 639
8.696.2.2 Tlvresult . . . 639
8.697 unpack_dms_SLQSUIMGetState_t Struct Reference . . . 639
8.697.1 Detailed Description . . . 639
8.697.2 Field Documentation . . . 640
8.697.2.1 ParamPresenceMask . . . 640
8.697.2.2 state . . . 640
8.697.2.3 Tlvresult . . . 640
8.698 unpack_dms_SwiEventReportCallBack_ind_t Struct Reference . . . 640
8.698.1 Detailed Description . . . 640

8.698.2 Field Documentation . . . 641
8.698.2.1 ParamPresenceMask . . . 641
8.698.2.2 TempTlv . . . 641
8.698.2.3 UimStatusTlv . . . 641
8.698.2.4 VoltTlv . . . 641
8.699 unpack_dms_SwiSetEventReport_t Struct Reference . . . 641
8.699.1 Detailed Description . . . 641
8.699.2 Field Documentation . . . 641
8.699.2.1 ParamPresenceMask . . . 641
8.699.2.2 Tlvresult . . . 641
8.700 unpack_dms_SwiUimSelect_t Struct Reference . . . 641
8.700.1 Detailed Description . . . 641
8.700.2 Field Documentation . . . 642
8.700.2.1 ParamPresenceMask . . . 642
8.700.2.2 Tlvresult . . . 642
8.701 unpack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 642
8.701.1 Detailed Description . . . 642
8.701.2 Field Documentation . . . 643
8.701.2.1 facilityState . . . 643
8.701.2.2 ParamPresenceMask . . . 643
8.701.2.3 Tlvresult . . . 643
8.701.2.4 unblockRetriesLeft . . . 643
8.701.2.5 verifyRetriesLeft . . . 643
8.702 unpack_dms_UIMGetICCID_t Struct Reference . . . 643
8.702.1 Detailed Description . . . 643
8.702.2 Field Documentation . . . 643
8.702.2.1 ParamPresenceMask . . . 643
8.702.2.2 String . . . 643
8.702.2.3 stringSize . . . 643
8.702.2.4 Tlvresult . . . 643
8.703 unpack_dms_UIMGetPINStatus_t Struct Reference . . . 644
8.703.1 Detailed Description . . . 644
8.703.2 Field Documentation . . . 645
8.703.2.1 p1Status . . . 645
8.703.2.2 p1UnblockRetriesLeft . . . 645
8.703.2.3 p1VerifyRetriesLeft . . . 645
8.703.2.4 p2Status . . . 645
8.703.2.5 p2UnblockRetriesLeft . . . 645
8.703.2.6 p2VerifyRetriesLeft . . . 645
8.703.2.7 ParamPresenceMask . . . 645

8.703.2.8 TlvResult . . . 645
8.704 unpack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 645
8.704.1 Detailed Description . . . 645
8.704.2 Field Documentation . . . 646
8.704.2.1 ParamPresenceMask . . . 646
8.704.2.2 TlvResult . . . 646
8.704.2.3 verifyRetriesLeft . . . 646
8.705 unpack_dms_UIMSetPINProtection_t Struct Reference . . . 646
8.705.1 Detailed Description . . . 646
8.705.2 Field Documentation . . . 647
8.705.2.1 ParamPresenceMask . . . 647
8.705.2.2 TlvResult . . . 647
8.705.2.3 unblockRetriesLeft . . . 647
8.705.2.4 verifyRetriesLeft . . . 647
8.706 unpack_dms_UIMUnblockControlKey_t Struct Reference . . . 647
8.706.1 Detailed Description . . . 647
8.706.2 Field Documentation . . . 647
8.706.2.1 ParamPresenceMask . . . 647
8.706.2.2 TlvResult . . . 647
8.706.2.3 unblockRetriesLeft . . . 647
8.707 unpack_fms_GetImagesPreference_t Struct Reference . . . 647
8.707.1 Detailed Description . . . 648
8.707.2 Field Documentation . . . 648
8.707.2.1 ImageListSize . . . 648
8.707.2.2 ParamPresenceMask . . . 648
8.707.2.3 pImageList . . . 648
8.707.2.4 TlvResult . . . 648
8.708 unpack_fms_GetStoredImages_t Struct Reference . . . 648
8.708.1 Detailed Description . . . 648
8.708.2 Field Documentation . . . 649
8.708.2.1 imageList . . . 649
8.708.2.2 imagelistSize . . . 649
8.708.2.3 ParamPresenceMask . . . 649
8.708.2.4 TlvResult . . . 649
8.709 unpack_fms_SetImagesPreference_t Struct Reference . . . 649
8.709.1 Detailed Description . . . 649
8.709.2 Field Documentation . . . 649
8.709.2.1 ImageTypes . . . 649
8.709.2.2 ImageTypesSize . . . 649
8.709.2.3 ParamPresenceMask . . . 649

8.709.2.4 TlvResult . . . 649
8.710 unpack_ims_SLQSGetIMSSMSConfig_t Struct Reference . . . 649
8.710.1 Detailed Description . . . 650
8.710.2 Field Documentation . . . 650
8.710.2.1 ParamPresenceMask . . . 650
8.710.2.2 pPhoneCtxtURI . . . 650
8.710.2.3 pPhoneCtxtURILen . . . 650
8.710.2.4 pSettingResp . . . 650
8.710.2.5 pSMSFormat . . . 650
8.710.2.6 pSMSOverIPNwInd . . . 650
8.711 unpack_ims_SLQSGetIMSUserConfig_t Struct Reference . . . 650
8.711.1 Detailed Description . . . 651
8.711.2 Field Documentation . . . 651
8.711.2.1 ParamPresenceMask . . . 651
8.711.2.2 pIMSDomain . . . 651
8.711.2.3 pIMSDomainLen . . . 651
8.711.2.4 pSettingResp . . . 651
8.712 unpack_ims_SLQSGetIMSVoIPConfig_t Struct Reference . . . 651
8.712.1 Detailed Description . . . 652
8.712.2 Field Documentation . . . 653
8.712.2.1 pAmrMode . . . 653
8.712.2.2 pAmrOctetAligned . . . 653
8.712.2.3 pAmrWbEnable . . . 653
8.712.2.4 pAmrWBMode . . . 654
8.712.2.5 pAmrWBOctetAligned . . . 654
8.712.2.6 ParamPresenceMask . . . 654
8.712.2.7 pMinSessionExpiryTimer . . . 654
8.712.2.8 pRingBackTimer . . . 654
8.712.2.9 pRingingTimer . . . 654
8.712.2.10 pRTPRTCPInactTimer . . . 654
8.712.2.11 pScrAmrEnable . . . 654
8.712.2.12 pScrAmrWbEnable . . . 654
8.712.2.13 pSessionExpiryTimer . . . 654
8.712.2.14 pSettingResp . . . 654
8.713 unpack_ims_SLQSGetRegMgrConfig_t Struct Reference . . . 654
8.713.1 Detailed Description . . . 654
8.713.2 Field Documentation . . . 655
8.713.2.1 ParamPresenceMask . . . 655
8.713.2.2 pIMSTestMode . . . 655
8.713.2.3 pPCSCFPort . . . 655

8.713.2.4 pPriCSCFPortName . . . 655
8.713.2.5 pPriCSCFPortNameLen . . . 655
8.713.2.6 pSettingResp . . . 655
8.714 unpack_ims_SLQSGetSIPConfig_t Struct Reference . . . 655
8.714.1 Detailed Description . . . 655
8.714.2 Field Documentation . . . 656
8.714.2.1 ParamPresenceMask . . . 656
8.714.2.2 pSettingResp . . . 656
8.714.2.3 pSigCompEnabled . . . 656
8.714.2.4 pSIPLocalPort . . . 656
8.714.2.5 pSubscribeTimer . . . 656
8.714.2.6 pTimerSIPReg . . . 656
8.714.2.7 pTimerT1 . . . 656
8.714.2.8 pTimerT2 . . . 656
8.714.2.9 pTimerTf . . . 656
8.715 unpack_ims_SLQSRegMgrCfgCallBack_ind_t Struct Reference . . . 656
8.715.1 Detailed Description . . . 657
8.715.2 Field Documentation . . . 657
8.715.2.1 ParamPresenceMask . . . 657
8.715.2.2 PCTlv . . . 657
8.715.2.3 PNTlv . . . 657
8.715.2.4 TMTlv . . . 657
8.716 unpack_ims_SLQSSetIMSSMSConfig_t Struct Reference . . . 657
8.716.1 Detailed Description . . . 657
8.716.2 Field Documentation . . . 658
8.716.2.1 ParamPresenceMask . . . 658
8.716.2.2 pSettingResp . . . 658
8.717 unpack_ims_SLQSSetIMSUserConfig_t Struct Reference . . . 658
8.717.1 Detailed Description . . . 658
8.717.2 Field Documentation . . . 658
8.717.2.1 ParamPresenceMask . . . 658
8.717.2.2 pSettingResp . . . 658
8.718 unpack_ims_SLQSSetIMSVoIPConfig_t Struct Reference . . . 658
8.718.1 Detailed Description . . . 658
8.718.2 Field Documentation . . . 659
8.718.2.1 ParamPresenceMask . . . 659
8.718.2.2 pSettingResp . . . 659
8.719 unpack_ims_SLQSSetRegMgrConfig_t Struct Reference . . . 659
8.719.1 Detailed Description . . . 659
8.719.2 Field Documentation . . . 659

8.719.2.1 ParamPresenceMask . . . 659
8.719.2.2 pSettingResp . . . 659
8.720 unpack_ims_SLQSSetSIPConfig_t Struct Reference . . . 659
8.720.1 Detailed Description . . . 659
8.720.2 Field Documentation . . . 660
8.720.2.1 ParamPresenceMask . . . 660
8.720.2.2 pSettingResp . . . 660
8.721 unpack_ims_SLQSSIPCfgCallBack_ind_t Struct Reference . . . 660
8.721.1 Detailed Description . . . 660
8.721.2 Field Documentation . . . 661
8.721.2.1 ParamPresenceMask . . . 661
8.721.2.2 SCTlv . . . 661
8.721.2.3 SPTlv . . . 661
8.721.2.4 SRTlv . . . 661
8.721.2.5 STTlv . . . 661
8.721.2.6 TT1Tlv . . . 661
8.721.2.7 TT2Tlv . . . 661
8.721.2.8 TTfTlv . . . 661
8.722 unpack_ims_SLQSSMSCfgCallBack_ind_t Struct Reference . . . 661
8.722.1 Detailed Description . . . 661
8.722.2 Field Documentation . . . 662
8.722.2.1 ParamPresenceMask . . . 662
8.722.2.2 PCURTlv . . . 662
8.722.2.3 SFTlv . . . 662
8.722.2.4 SINTlv . . . 662
8.723 unpack_ims_SLQSUserCfgCallBack_ind_t Struct Reference . . . 662
8.723.1 Detailed Description . . . 662
8.723.2 Field Documentation . . . 663
8.723.2.1 IDTlv . . . 663
8.723.2.2 ParamPresenceMask . . . 663
8.724 unpack_ims_SLQSVoIPCfgCallBack_ind_t Struct Reference . . . 663
8.724.1 Detailed Description . . . 663
8.724.2 Field Documentation . . . 664
8.724.2.1 AMTlv . . . 664
8.724.2.2 AOATlv . . . 664
8.724.2.3 AWMTlv . . . 664
8.724.2.4 AWOATlv . . . 665
8.724.2.5 EAWTlv . . . 665
8.724.2.6 ESATlv . . . 665
8.724.2.7 ESAWTlv . . . 665

8.724.2.8 MSETlv . . . 665
8.724.2.9 ParamPresenceMask . . . 665
8.724.2.10 RBTTlv . . . 665
8.724.2.11 RTIDTlv . . . 665
8.724.2.12 RTTlv . . . 665
8.724.2.13 SDTlv . . . 665
8.725 unpack_imsa_SLQSGetIMSARegStatus_t Struct Reference . . . 665
8.725.1 Detailed Description . . . 665
8.725.2 Field Documentation . . . 666
8.725.2.1 ImsRegErrCode . . . 666
8.725.2.2 ImsRegStatus . . . 666
8.725.2.3 NewImsRegStatus . . . 666
8.725.2.4 ParamPresenceMask . . . 666
8.726 unpack_imsa_SLQSGetIMSAServiceStatus_t Struct Reference . . . 666
8.726.1 Detailed Description . . . 666
8.726.2 Field Documentation . . . 668
8.726.2.1 ParamPresenceMask . . . 668
8.726.2.2 SmsServiceRat . . . 668
8.726.2.3 SmsServiceStatus . . . 668
8.726.2.4 UtServiceRat . . . 668
8.726.2.5 UtServiceStatus . . . 668
8.726.2.6 VoipServiceRat . . . 668
8.726.2.7 VoipServiceStatus . . . 668
8.726.2.8 VsServiceRat . . . 668
8.726.2.9 VsServiceStatus . . . 668
8.726.2.10 VtServiceRat . . . 668
8.726.2.11 VtServiceStatus . . . 668
8.727 unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t Struct Reference . . . 668
8.727.1 Detailed Description . . . 669
8.727.2 Field Documentation . . . 669
8.727.2.1 FailErrCode . . . 669
8.727.2.2 ParamPresenceMask . . . 669
8.727.2.3 PdpConnState . . . 669
8.728 unpack_imsa_SLQSImsaRatStatusCallBack_ind_t Struct Reference . . . 669
8.728.1 Detailed Description . . . 669
8.728.2 Field Documentation . . . 670
8.728.2.1 ParamPresenceMask . . . 670
8.728.2.2 RatHandover . . . 670
8.729 unpack_imsa_SLQSImsaRegStatusCallBack_ind_t Struct Reference . . . 670
8.729.1 Detailed Description . . . 670

8.729.2 Field Documentation . . . 670
8.729.2.1 IMSRegistration . . . 670
8.729.2.2 IMSRegistrationError . . . 670
8.729.2.3 NewIMSRegistration . . . 670
8.729.2.4 ParamPresenceMask . . . 671
8.730 unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t Struct Reference . . . 671
8.730.1 Detailed Description . . . 671
8.730.2 Field Documentation . . . 672
8.730.2.1 ParamPresenceMask . . . 672
8.730.2.2 SmsRat . . . 672
8.730.2.3 SmsService . . . 672
8.730.2.4 UtRat . . . 672
8.730.2.5 UtService . . . 672
8.730.2.6 VoipRat . . . 672
8.730.2.7 VoipService . . . 672
8.730.2.8 VtRat . . . 672
8.730.2.9 VtService . . . 672
8.731 unpack_loc_BestAvailPos_Ind_t Struct Reference . . . 672
8.731.1 Detailed Description . . . 673
8.731.2 Field Documentation . . . 678
8.731.2.1 pAltitudeWrtEllipsoid . . . 678
8.731.2.2 pAltitudeWrtMeanSeaLevel . . . 678
8.731.2.3 ParamPresenceMask . . . 678
8.731.2.4 pGpsTime . . . 678
8.731.2.5 pHeading . . . 678
8.731.2.6 pHeadingUnc . . . 678
8.731.2.7 pHorCirConf . . . 678
8.731.2.8 pHorEllpConf . . . 678
8.731.2.9 pHorReliability . . . 678
8.731.2.10 pHorUncCircular . . . 678
8.731.2.11 pHorUncEllipseOrientAzimuth . . . 678
8.731.2.12 pHorUncEllipseSemiMajor . . . 678
8.731.2.13 pHorUncEllipseSemiMinor . . . 678
8.731.2.14 pLatitude . . . 678
8.731.2.15 pLongitude . . . 678
8.731.2.16 pMagneticDeviation . . . 678
8.731.2.17 pPrecisionDilution . . . 678
8.731.2.18 pSensorDataUsage . . . 678
8.731.2.19 pSpeedHorizontal . . . 678
8.731.2.20 pSpeedUnc . . . 678

8.731.2.21 pSpeedVertical . . . 678
8.731.2.22 pSpeedVerticalUnc . . . 678
8.731.2.23 pSvUsedforFix . . . 678
8.731.2.24 pTechnologyMask . . . 678
8.731.2.25 pTimeSrc . . . 678
8.731.2.26 pTimestampUtc . . . 678
8.731.2.27 pTimeUnc . . . 678
8.731.2.28 pVertConfidence . . . 679
8.731.2.29 pVertReliability . . . 679
8.731.2.30 pVertUnc . . . 679
8.731.2.31 pXid . . . 679
8.731.2.32 status . . . 679
8.731.2.33 Tlvresult . . . 679
8.732 unpack_loc_CradleMountCallback_Ind_t Struct Reference . . . 679
8.732.1 Detailed Description . . . 679
8.732.2 Field Documentation . . . 679
8.732.2.1 cradleMountConfigStatus . . . 679
8.732.2.2 ParamPresenceMask . . . 679
8.733 unpack_loc_Delete_Assist_Data_t Struct Reference . . . 679
8.733.1 Detailed Description . . . 680
8.733.2 Field Documentation . . . 680
8.733.2.1 ParamPresenceMask . . . 680
8.733.2.2 Tlvresult . . . 680
8.734 unpack_loc_DeleteAssistData_Ind_t Struct Reference . . . 680
8.734.1 Detailed Description . . . 680
8.734.2 Field Documentation . . . 681
8.734.2.1 ParamPresenceMask . . . 681
8.734.2.2 status . . . 681
8.734.2.3 Tlvresult . . . 681
8.735 unpack_loc_EngineState_Ind_t Struct Reference . . . 681
8.735.1 Detailed Description . . . 681
8.735.2 Field Documentation . . . 682
8.735.2.1 engineState . . . 682
8.735.2.2 ParamPresenceMask . . . 682
8.735.2.3 Tlvresult . . . 682
8.736 unpack_loc_EventNMEA_Ind_t Struct Reference . . . 682
8.736.1 Detailed Description . . . 682
8.736.2 Field Documentation . . . 682
8.736.2.1 NMEAData . . . 682
8.736.2.2 ParamPresenceMask . . . 682

8.736.2.3 TlvResult . . . 682
8.737 unpack_loc_EventRegister_t Struct Reference . . . 683
8.737.1 Detailed Description . . . 683
8.737.2 Field Documentation . . . 683
8.737.2.1 ParamPresenceMask . . . 683
8.737.2.2 TlvResult . . . 683
8.738 unpack_loc_EventTimeSyncCallback_Ind_t Struct Reference . . . 683
8.738.1 Detailed Description . . . 683
8.738.2 Field Documentation . . . 683
8.738.2.1 ParamPresenceMask . . . 683
8.738.2.2 timeSyncRefCounter . . . 683
8.739 unpack_loc_GetServer_Ind_t Struct Reference . . . 683
8.739.1 Detailed Description . . . 684
8.739.2 Field Documentation . . . 685
8.739.2.1 ParamPresenceMask . . . 685
8.739.2.2 pIPv4AddrInfo . . . 685
8.739.2.3 pIPv6AddrInfo . . . 685
8.739.2.4 pURL . . . 685
8.739.2.5 serverStatus . . . 685
8.739.2.6 serverType . . . 685
8.739.2.7 TlvResult . . . 685
8.740 unpack_loc_GnssSvInfo_Ind_t Struct Reference . . . 685
8.740.1 Detailed Description . . . 685
8.740.2 Field Documentation . . . 686
8.740.2.1 altitudeAssumed . . . 686
8.740.2.2 ParamPresenceMask . . . 686
8.740.2.3 pSatelliteInfo . . . 686
8.740.2.4 TlvResult . . . 686
8.741 unpack_loc_InjectPositionCallback_Ind_t Struct Reference . . . 686
8.741.1 Detailed Description . . . 686
8.741.2 Field Documentation . . . 686
8.741.2.1 ParamPresenceMask . . . 686
8.741.2.2 status . . . 686
8.742 unpack_loc_InjectSensorDataCallback_Ind_t Struct Reference . . . 686
8.742.1 Detailed Description . . . 687
8.742.2 Field Documentation . . . 688
8.742.2.1 injectSensorDataStatus . . . 688
8.742.2.2 pAccelSamplesAccepted . . . 688
8.742.2.3 pAccelTempSamplesAccepted . . . 688
8.742.2.4 ParamPresenceMask . . . 688

8.742.2.5 pGyroSamplesAccepted . . . 688
8.742.2.6 pGyroTempSamplesAccepted . . . 688
8.742.2.7 pOpaqueIdentifier . . . 688
8.743 unpack_loc_InjectTimeSyncDataCallback_Ind_t Struct Reference . . . 688
8.743.1 Detailed Description . . . 688
8.743.2 Field Documentation . . . 689
8.743.2.1 injectTimeSyncStatus . . . 689
8.743.2.2 ParamPresenceMask . . . 689
8.744 unpack_loc_InjectUTCTimeCallback_Ind_t Struct Reference . . . 689
8.744.1 Detailed Description . . . 689
8.744.2 Field Documentation . . . 689
8.744.2.1 ParamPresenceMask . . . 689
8.744.2.2 status . . . 689
8.745 unpack_loc_PositionRpt_Ind_t Struct Reference . . . 689
8.745.1 Detailed Description . . . 690
8.745.2 Field Documentation . . . 695
8.745.2.1 pAltitudeAssumed . . . 695
8.745.2.2 pAltitudeWrtEllipsoid . . . 695
8.745.2.3 pAltitudeWrtMeanSeaLevel . . . 695
8.745.2.4 ParamPresenceMask . . . 695
8.745.2.5 pFixId . . . 695
8.745.2.6 pGpsTime . . . 695
8.745.2.7 pHeading . . . 695
8.745.2.8 pHeadingUnc . . . 695
8.745.2.9 pHorConfidence . . . 695
8.745.2.10 pHorReliability . . . 695
8.745.2.11 pHorUncCircular . . . 695
8.745.2.12 pHorUncEllipseOrientAzimuth . . . 695
8.745.2.13 pHorUncEllipseSemiMajor . . . 695
8.745.2.14 pHorUncEllipseSemiMinor . . . 695
8.745.2.15 pLatitude . . . 695
8.745.2.16 pLeapSeconds . . . 695
8.745.2.17 pLongitude . . . 695
8.745.2.18 pMagneticDeviation . . . 696
8.745.2.19 pPrecisionDilution . . . 696
8.745.2.20 pSensorDataUsage . . . 696
8.745.2.21 pSpeedHorizontal . . . 696
8.745.2.22 pSpeedUnc . . . 696
8.745.2.23 pSpeedVertical . . . 696
8.745.2.24 pSvUsedforFix . . . 696

8.745.2.25 pTechnologyMask . . . 696
8.745.2.26 pTimeSrc . . . 696
8.745.2.27 pTimestampUtc . . . 696
8.745.2.28 pTimeUnc . . . 696
8.745.2.29 pVertConfidence . . . 696
8.745.2.30 pVertReliability . . . 696
8.745.2.31 pVertUnc . . . 696
8.745.2.32 sessionId . . . 696
8.745.2.33 sessionStatus . . . 696
8.745.2.34 TlvResult . . . 696
8.746 unpack_loc_SensorStreamingCallback_Ind_t Struct Reference . . . 696
8.746.1 Detailed Description . . . 696
8.746.2 Field Documentation . . . 697
8.746.2.1 pAccelAcceptReady . . . 697
8.746.2.2 pAccelTempAcceptReady . . . 697
8.746.2.3 ParamPresenceMask . . . 697
8.746.2.4 pGyroAcceptReady . . . 697
8.746.2.5 pGyroTempAcceptReady . . . 697
8.747 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference . . . 697
8.747.1 Detailed Description . . . 697
8.747.2 Field Documentation . . . 698
8.747.2.1 ParamPresenceMask . . . 698
8.747.2.2 status . . . 698
8.747.2.3 TlvResult . . . 698
8.748 unpack_loc_SetExtPowerState_t Struct Reference . . . 698
8.748.1 Detailed Description . . . 698
8.748.2 Field Documentation . . . 698
8.748.2.1 ParamPresenceMask . . . 698
8.748.2.2 TlvResult . . . 698
8.749 unpack_loc_SetOperationMode_Ind_t Struct Reference . . . 699
8.749.1 Detailed Description . . . 699
8.749.2 Field Documentation . . . 699
8.749.2.1 ParamPresenceMask . . . 699
8.749.2.2 status . . . 699
8.749.2.3 TlvResult . . . 699
8.750 unpack_loc_SetOperationMode_t Struct Reference . . . 699
8.750.1 Detailed Description . . . 699
8.750.2 Field Documentation . . . 700
8.750.2.1 ParamPresenceMask . . . 700
8.750.2.2 TlvResult . . . 700

8.751 unpack_loc_SetServer_Ind_t Struct Reference . . . 700
8.751.1 Detailed Description . . . 700
8.751.2 Field Documentation . . . 701
8.751.2.1 ParamPresenceMask . . . 701
8.751.2.2 serverStatus . . . 701
8.751.2.3 Tlvresult . . . 701
8.752 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 701
8.752.1 Detailed Description . . . 701
8.752.2 Field Documentation . . . 701
8.752.2.1 ParamPresenceMask . . . 701
8.752.2.2 Tlvresult . . . 701
8.753 unpack_loc_Start_t Struct Reference . . . 701
8.753.1 Detailed Description . . . 701
8.753.2 Field Documentation . . . 701
8.753.2.1 ParamPresenceMask . . . 702
8.753.2.2 Tlvresult . . . 702
8.754 unpack_loc_Stop_t Struct Reference . . . 702
8.754.1 Detailed Description . . . 702
8.754.2 Field Documentation . . . 702
8.754.2.1 ParamPresenceMask . . . 702
8.754.2.2 Tlvresult . . . 702
8.755 unpack_nas_GetACCOLC_t Struct Reference . . . 702
8.755.1 Detailed Description . . . 702
8.755.2 Field Documentation . . . 702
8.755.2.1 pAccolc . . . 703
8.755.2.2 ParamPresenceMask . . . 703
8.756 unpack_nas_GetANAAAAuthenticationStatus_t Struct Reference . . . 703
8.756.1 Detailed Description . . . 703
8.756.2 Field Documentation . . . 703
8.756.2.1 ParamPresenceMask . . . 703
8.756.2.2 pAuthStatus . . . 703
8.757 unpack_nas_GetCDMANetworkParameters_t Struct Reference . . . 703
8.757.1 Detailed Description . . . 704
8.757.2 Field Documentation . . . 705
8.757.2.1 Application . . . 705
8.757.2.2 Broadcast . . . 705
8.757.2.3 CustomSCP . . . 705
8.757.2.4 ForceRev0 . . . 705
8.757.2.5 ParamPresenceMask . . . 705
8.757.2.6 Protocol . . . 705

8.757.2.7 RegForeignNID . . . 705
8.757.2.8 RegForeignSID . . . 706
8.757.2.9 RegHomeSID . . . 706
8.757.2.10 Roaming . . . 706
8.757.2.11 SCI . . . 706
8.757.2.12 SCM . . . 706
8.758 unpack_nas_GetHomeNetwork3GPP2_t Struct Reference . . . 706
8.758.1 Detailed Description . . . 706
8.758.2 Field Documentation . . . 708
8.758.2.1 nameSize . . . 708
8.758.2.2 ParamPresenceMask . . . 708
8.758.2.3 pMCC . . . 708
8.758.2.4 pMNC . . . 708
8.758.2.5 pName . . . 708
8.758.2.6 pNID . . . 708
8.758.2.7 pNw2DescDisp . . . 708
8.758.2.8 pNw2DescEnc . . . 708
8.758.2.9 pNw2DescLen . . . 708
8.758.2.10 pNw2MCC . . . 708
8.758.2.11 pNw2MNC . . . 708
8.758.2.12 pNw2Name . . . 708
8.758.2.13 pSID . . . 708
8.759 unpack_nas_GetHomeNetwork_t Struct Reference . . . 708
8.759.1 Detailed Description . . . 708
8.759.2 Field Documentation . . . 709
8.759.2.1 mcc . . . 709
8.759.2.2 mnc . . . 709
8.759.2.3 name . . . 709
8.759.2.4 nid . . . 709
8.759.2.5 ParamPresenceMask . . . 709
8.759.2.6 sid . . . 709
8.760 unpack_nas_GetNetworkPreference_t Struct Reference . . . 709
8.760.1 Detailed Description . . . 709
8.760.2 Field Documentation . . . 710
8.760.2.1 ActiveTechPref . . . 710
8.760.2.2 Duration . . . 710
8.760.2.3 ParamPresenceMask . . . 710
8.760.2.4 PersistentTechPref . . . 710
8.760.2.5 Tlvresult . . . 710
8.761 unpack_nas_GetRFInfo_t Struct Reference . . . 710

8.761.1 Detailed Description . . . 711
8.761.2 Field Documentation . . . 711
8.761.2.1 instancesSize . . . 711
8.761.2.2 ParamPresenceMask . . . 711
8.761.2.3 RFBandInfoElements . . . 711
8.762 unpack_nas_GetServingNetwork_t Struct Reference . . . 711
8.762.1 Detailed Description . . . 712
8.762.2 Field Documentation . . . 713
8.762.2.1 CSDomain . . . 713
8.762.2.2 DataCaps . . . 713
8.762.2.3 DataCapsLen . . . 713
8.762.2.4 MCC . . . 713
8.762.2.5 MNC . . . 713
8.762.2.6 Name . . . 714
8.762.2.7 nameSize . . . 714
8.762.2.8 ParamPresenceMask . . . 714
8.762.2.9 PSDomain . . . 714
8.762.2.10 RadioIfaces . . . 714
8.762.2.11 RadioIfacesSize . . . 714
8.762.2.12 RAN . . . 714
8.762.2.13 RegistrationState . . . 714
8.762.2.14 Roaming . . . 714
8.763 unpack_nas_GetServingNetworkCapabilities_t Struct Reference . . . 714
8.763.1 Detailed Description . . . 714
8.763.2 Field Documentation . . . 715
8.763.2.1 DataCaps . . . 715
8.763.2.2 DataCapsLen . . . 715
8.763.2.3 ParamPresenceMask . . . 715
8.764 unpack_nas_GetSignalStrengths_t Struct Reference . . . 715
8.764.1 Detailed Description . . . 715
8.764.2 Field Documentation . . . 715
8.764.2.1 len . . . 715
8.764.2.2 ParamPresenceMask . . . 715
8.764.2.3 radio . . . 715
8.764.2.4 rssi . . . 715
8.765 unpack_nas_PerformNetworkScan_t Struct Reference . . . 716
8.765.1 Detailed Description . . . 716
8.765.2 Field Documentation . . . 717
8.765.2.1 p3GppNetworkInfoInstances . . . 717
8.765.2.2 p3GppNetworkInstanceSize . . . 717

8.765.2.3 ParamPresenceMask . . . 717
8.765.2.4 pLteOpModeTlv . . . 717
8.765.2.5 pPCIInfo . . . 717
8.765.2.6 pPCSInstance . . . 717
8.765.2.7 pPCSInstanceSize . . . 717
8.765.2.8 pRATINstance . . . 717
8.765.2.9 pRATInstanceSize . . . 717
8.765.2.10 pScanResult . . . 717
8.766 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference . . . 717
8.766.1 Detailed Description . . . 718
8.766.2 Field Documentation . . . 718
8.766.2.1 dataCaps . . . 718
8.766.2.2 dataCapsSize . . . 718
8.766.2.3 ParamPresenceMask . . . 718
8.767 unpack_nas_SetEventReportInd_t Struct Reference . . . 718
8.767.1 Detailed Description . . . 718
8.767.2 Field Documentation . . . 719
8.767.2.1 ParamPresenceMask . . . 719
8.767.2.2 RFTlv . . . 719
8.767.2.3 RRTlv . . . 719
8.767.2.4 SLQSSSTlv . . . 719
8.767.2.5 SSTlv . . . 719
8.768 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference . . . 719
8.768.1 Detailed Description . . . 719
8.768.2 Field Documentation . . . 720
8.768.2.1 ParamPresenceMask . . . 720
8.768.2.2 sPhyCaAggPcellInfo . . . 720
8.768.2.3 sPhyCaAggScellDlBw . . . 720
8.768.2.4 sPhyCaAggScellIndex . . . 720
8.768.2.5 sPhyCaAggScellIndType . . . 720
8.768.2.6 sPhyCaAggScellInfo . . . 720
8.769 unpack_nas_SetNetworkPreference_t Struct Reference . . . 720
8.769.1 Detailed Description . . . 720
8.769.2 Field Documentation . . . 720
8.769.2.1 ParamPresenceMask . . . 721
8.769.2.2 Tlvresult . . . 721
8.770 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference . . . 721
8.770.1 Detailed Description . . . 721
8.770.2 Field Documentation . . . 721
8.770.2.1 ParamPresenceMask . . . 721

8.770.2.2 roaming . . . 721
8.771 unpack_nas_SetServingSystemCallback_ind_t Struct Reference . . . 721
8.771.1 Detailed Description . . . 721
8.771.2 Field Documentation . . . 722
8.771.2.1 ParamPresenceMask . . . 722
8.771.2.2 SSInfo . . . 722
8.771.2.3 Tlvresult . . . 722
8.772 unpack_nas_SLQSGetErrorRate_t Struct Reference . . . 722
8.772.1 Detailed Description . . . 722
8.772.2 Field Documentation . . . 723
8.772.2.1 ParamPresenceMask . . . 723
8.772.2.2 pCDMAFrameErrRate . . . 723
8.772.2.3 pGSMBER . . . 723
8.772.2.4 pHDRPackErrRate . . . 723
8.772.2.5 pWCDMABER . . . 723
8.773 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference . . . 723
8.773.1 Detailed Description . . . 723
8.773.2 Field Documentation . . . 724
8.773.2.1 LTECphyCaInfo . . . 724
8.773.2.2 ParamPresenceMask . . . 724
8.773.2.3 Tlvresult . . . 724
8.774 unpack_nas_SLQSGetNetworkTime_t Struct Reference . . . 724
8.774.1 Detailed Description . . . 724
8.774.2 Field Documentation . . . 724
8.774.2.1 p3GPP2TimeInfo . . . 724
8.774.2.2 p3GPPTimeInfo . . . 724
8.774.2.3 ParamPresenceMask . . . 724
8.775 unpack_nas_SLQSGetOperatorNameData_t Struct Reference . . . 724
8.775.1 Detailed Description . . . 725
8.775.2 Field Documentation . . . 725
8.775.2.1 ParamPresenceMask . . . 725
8.775.2.2 pNITZInformation . . . 725
8.775.2.3 pOperatorNameString . . . 725
8.775.2.4 pOperatorPLMNList . . . 725
8.775.2.5 pPLMNNetworkName . . . 725
8.775.2.6 pSrvcProviderName . . . 725
8.776 unpack_nas_SLQSGetPLMNName_t Struct Reference . . . 726
8.776.1 Detailed Description . . . 726
8.776.2 Field Documentation . . . 728
8.776.2.1 longName . . . 728

8.776.2.2 longNameCI . . . 728
8.776.2.3 longNameEn . . . 728
8.776.2.4 longNameLen . . . 728
8.776.2.5 longNameSB . . . 728
8.776.2.6 ParamPresenceMask . . . 728
8.776.2.7 shortName . . . 728
8.776.2.8 shortNameCI . . . 728
8.776.2.9 shortNameEn . . . 728
8.776.2.10 shortNameLen . . . 728
8.776.2.11 shortNameSB . . . 728
8.776.2.12 spn . . . 728
8.776.2.13 spnEncoding . . . 729
8.776.2.14 spnLength . . . 729
8.777 unpack_nas_SLQSGetServingSystem_t Struct Reference . . . 729
8.777.1 Detailed Description . . . 729
8.777.2 Field Documentation . . . 732
8.777.2.1 BasestationID . . . 732
8.777.2.2 BasestationLatitude . . . 732
8.777.2.3 BasestationLongitude . . . 732
8.777.2.4 CallBarStatus . . . 732
8.777.2.5 CDMA_P_Rev . . . 732
8.777.2.6 CDMASystemInfoExt . . . 732
8.777.2.7 CellID . . . 733
8.777.2.8 ConcSvcInfo . . . 733
8.777.2.9 CurrentPLMN . . . 733
8.777.2.10 DataSrvCapabilities . . . 733
8.777.2.11 DefaultRoamInd . . . 733
8.777.2.12 DetailedSvcInfo . . . 733
8.777.2.13 DTMInd . . . 733
8.777.2.14 Gpp2TimeZone . . . 733
8.777.2.15 GppNetworkDSTAdjustment . . . 733
8.777.2.16 GppTimeZone . . . 733
8.777.2.17 HdrPersonality . . . 733
8.777.2.18 Lac . . . 733
8.777.2.19 NetworkID . . . 733
8.777.2.20 ParamPresenceMask . . . 733
8.777.2.21 PRLInd . . . 733
8.777.2.22 RoamIndicatorVal . . . 733
8.777.2.23 RoamingIndicatorList . . . 733
8.777.2.24 ServingSystem . . . 733

8.777.2.25 SystemID . . . 733
8.777.2.26 TrackAreaCode . . . 733
8.778 unpack_nas_SLQSGetSignalStrength_t Struct Reference . . . 733
8.778.1 Detailed Description . . . 734
8.778.2 Field Documentation . . . 735
8.778.2.1 ecioList . . . 735
8.778.2.2 ecioListLen . . . 735
8.778.2.3 errorRateList . . . 735
8.778.2.4 errorRateListLen . . . 735
8.778.2.5 Io . . . 736
8.778.2.6 ltersrp . . . 736
8.778.2.7 ltesnr . . . 736
8.778.2.8 ParamPresenceMask . . . 736
8.778.2.9 rsrqInfo . . . 736
8.778.2.10 rxSignalStrengthList . . . 736
8.778.2.11 rxSignalStrengthListLen . . . 736
8.778.2.12 signalStrengthReqMask . . . 736
8.778.2.13 sinr . . . 736
8.779 unpack_nas_SLQSGetSysInfo_t Struct Reference . . . 736
8.779.1 Detailed Description . . . 736
8.779.2 Field Documentation . . . 739
8.779.2.1 pAddCDMASysInfo . . . 739
8.779.2.2 pAddGSMSysInfo . . . 739
8.779.2.3 pAddHDRSysInfo . . . 739
8.779.2.4 pAddLTESysInfo . . . 739
8.779.2.5 pAddWCDMASysInfo . . . 739
8.779.2.6 ParamPresenceMask . . . 739
8.779.2.7 pCampedCiotLteOpMode . . . 739
8.779.2.8 pCDMASrvStatusInfo . . . 739
8.779.2.9 pCDMASysInfo . . . 739
8.779.2.10 pGSMCallBarringSysInfo . . . 739
8.779.2.11 pGSMCipherDomainSysInfo . . . 739
8.779.2.12 pGSMSrvStatusInfo . . . 739
8.779.2.13 pGSMSysInfo . . . 739
8.779.2.14 pHDRSrvStatusInfo . . . 739
8.779.2.15 pHDRSysInfo . . . 739
8.779.2.16 pLTESrvStatusInfo . . . 739
8.779.2.17 pLTESysInfo . . . 739
8.779.2.18 pLTEVoiceSupportSysInfo . . . 739
8.779.2.19 pWCDMACallBarringSysInfo . . . 739

8.779.2.20 pWCDMACipherDomainSysInfo . . . 740
8.779.2.21 pWCDMASrvStatusInfo . . . 740
8.779.2.22 pWCDMASysInfo . . . 740
8.780 unpack_nas_SLQSGetSysSelectionPref_t Struct Reference . . . 740
8.780.1 Detailed Description . . . 740
8.780.2 Field Documentation . . . 744
8.780.2.1 ParamPresenceMask . . . 744
8.780.2.2 pBandPref . . . 744
8.780.2.3 pEmerMode . . . 744
8.780.2.4 pGWAcqOrderPref . . . 744
8.780.2.5 pLTEBandPref . . . 744
8.780.2.6 pModePref . . . 744
8.780.2.7 pNetSelPref . . . 744
8.780.2.8 pPRLPref . . . 744
8.780.2.9 pRoamPref . . . 744
8.780.2.10 pSrvDomainPref . . . 744
8.781 unpack_nas_SLQSGetSysSelectionPrefExt_t Struct Reference . . . 744
8.781.1 Detailed Description . . . 745
8.781.2 Field Documentation . . . 746
8.781.2.1 pAcqOrderPref . . . 746
8.781.2.2 pBandPref . . . 746
8.781.2.3 pCiotAcqOrderPref . . . 746
8.781.2.4 pCiotLteOpModePref . . . 747
8.781.2.5 pEmerMode . . . 747
8.781.2.6 pGWAcqOrderPref . . . 747
8.781.2.7 pLTEBandPref . . . 747
8.781.2.8 pLTEBandPrefExt . . . 747
8.781.2.9 pLteM1BandPref . . . 747
8.781.2.10 pLteNb1BandPref . . . 747
8.781.2.11 pModePref . . . 747
8.781.2.12 pNetSelPref . . . 747
8.781.2.13 pNr5gBandPref . . . 747
8.781.2.14 pPRLPref . . . 747
8.781.2.15 pRatDisabledMask . . . 747
8.781.2.16 pRoamPref . . . 747
8.781.2.17 pSrvDomainPref . . . 747
8.782 unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t Struct Reference . . . 747
8.782.1 Detailed Description . . . 747
8.782.2 Field Documentation . . . 748
8.782.2.1 edrxCiotLteModeTlv . . . 748

8.782.2.2 edrxCycleLengthTlv . . . 748
8.782.2.3 edrxEnableTypeTlv . . . 748
8.782.2.4 edrxPagingTimeWindowTlv . . . 748
8.782.2.5 edrxRatTypeTlv . . . 748
8.782.2.6 ParamPresenceMask . . . 748
8.783 unpack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 748
8.783.1 Detailed Description . . . 748
8.783.2 Field Documentation . . . 749
8.783.2.1 ParamPresenceMask . . . 749
8.783.2.2 pCDMAChannel . . . 749
8.783.2.3 pDirNum . . . 749
8.783.2.4 pHomeSIDNID . . . 749
8.783.2.5 pMinBasedIMSI . . . 749
8.783.2.6 pNAMNameInfo . . . 749
8.783.2.7 pTrueIMSI . . . 749
8.784 unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference . . . 749
8.784.1 Detailed Description . . . 750
8.784.2 Field Documentation . . . 750
8.784.2.1 ParamPresenceMask . . . 751
8.784.2.2 pCDMAInfo . . . 751
8.784.2.3 pGERANInfo . . . 751
8.784.2.4 pLTEInfoInterfreq . . . 751
8.784.2.5 pLTEInfoIntrafreq . . . 751
8.784.2.6 pLTEInfoNeighboringGSM . . . 751
8.784.2.7 pLTEInfoNeighboringWCDMA . . . 751
8.784.2.8 pUMTSCellID . . . 751
8.784.2.9 pUMTSInfo . . . 751
8.784.2.10 pWCDMAInfoLTENeighborCell . . . 751
8.785 unpack_nas_SLQSNASGeteDRXParams_t Struct Reference . . . 751
8.785.1 Detailed Description . . . 751
8.785.2 Field Documentation . . . 752
8.785.2.1 ParamPresenceMask . . . 752
8.785.2.2 pCycleLen . . . 752
8.785.2.3 pEdrxEnable . . . 752
8.785.2.4 pPagingTimeWindow . . . 752
8.786 unpack_nas_SLQSNASGeteDRXParamsExt_t Struct Reference . . . 752
8.786.1 Detailed Description . . . 752
8.786.2 Field Documentation . . . 753
8.786.2.1 ParamPresenceMask . . . 753
8.786.2.2 pCycleLen . . . 753

8.786.2.3 pEdrxEnable . . . 753
8.786.2.4 pEdrxRAT . . . 753
8.786.2.5 pLteOpMode . . . 753
8.786.2.6 pPagingTimeWindow . . . 753
8.787 unpack_nas_SLQSNASGetForbiddenNetworks_t Struct Reference . . . 753
8.787.1 Detailed Description . . . 754
8.787.2 Field Documentation . . . 754
8.787.2.1 ParamPresenceMask . . . 754
8.787.2.2 pForbiddenNetworks3GPP . . . 754
8.788 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference . . . 754
8.788.1 Detailed Description . . . 754
8.788.2 Field Documentation . . . 754
8.788.2.1 ParamPresenceMask . . . 754
8.788.2.2 pColorCode . . . 754
8.789 unpack_nas_SLQSNasGetRFInfo_t Struct Reference . . . 754
8.789.1 Detailed Description . . . 755
8.789.2 Field Documentation . . . 755
8.789.2.1 pLTEOperationMode . . . 755
8.789.2.2 pRfBandInfoExtFormat . . . 755
8.789.2.3 pRfBandwidthInfo . . . 755
8.789.2.4 pRfDedicatedBandInfo . . . 755
8.789.2.5 rfbandInfoList . . . 755
8.789.2.6 Tlvresult . . . 755
8.790 unpack_nas_SLQSNasGetSigInfo_t Struct Reference . . . 755
8.790.1 Detailed Description . . . 756
8.790.2 Field Documentation . . . 756
8.790.2.1 CDMASSInfo . . . 756
8.790.2.2 GSMSSInfo . . . 756
8.790.2.3 HDRSSInfo . . . 756
8.790.2.4 LTESSInfo . . . 756
8.790.2.5 ParamPresenceMask . . . 756
8.790.2.6 WCDMASSInfo . . . 756
8.791 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 756
8.791.1 Detailed Description . . . 757
8.791.2 Field Documentation . . . 757
8.791.2.1 ParamPresenceMask . . . 757
8.791.2.2 pRXChain0Info . . . 757
8.791.2.3 pRXChain1Info . . . 757
8.791.2.4 pTXInfo . . . 757
8.792 unpack_nas_SLQSNasNetworkRejectCallback_Ind_t Struct Reference . . . 757

8.792.1 Detailed Description . . . 757
8.792.2 Field Documentation . . . 760
8.792.2.1 ParamPresenceMask . . . 760
8.792.2.2 pCsgId . . . 760
8.792.2.3 pLteOpMode . . . 760
8.792.2.4 pPlmnId . . . 760
8.792.2.5 radioIf . . . 760
8.792.2.6 rejectCause . . . 760
8.792.2.7 serviceDomain . . . 760
8.793 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference . . . 760
8.793.1 Detailed Description . . . 760
8.793.2 Field Documentation . . . 761
8.793.2.1 ParamPresenceMask . . . 761
8.793.2.2 pDayltSavAdj . . . 761
8.793.2.3 pRadioInterface . . . 761
8.793.2.4 pTimeZone . . . 761
8.793.2.5 universalTime . . . 761
8.794 unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference . . . 761
8.794.1 Detailed Description . . . 762
8.794.2 Field Documentation . . . 762
8.794.2.1 ParamPresenceMask . . . 762
8.794.2.2 pCDMASigInfo . . . 762
8.794.2.3 pGSMSigInfo . . . 762
8.794.2.4 pHDRSigInfo . . . 762
8.794.2.5 pLTESigInfo . . . 762
8.794.2.6 pRscp . . . 762
8.794.2.7 pTDSCDMASigInfoExt . . . 763
8.794.2.8 pWCDMASigInfo . . . 763
8.795 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference . . . 763
8.795.1 Detailed Description . . . 763
8.795.2 Field Documentation . . . 763
8.795.2.1 ParamPresenceMask . . . 763
8.795.2.2 pLteEARFCN . . . 763
8.795.2.3 pLtePCI . . . 763
8.795.2.4 pWcdmaUARFCN . . . 763
8.796 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference . . . 763
8.796.1 Detailed Description . . . 764
8.796.2 Field Documentation . . . 764
8.796.2.1 commonInfo . . . 764
8.796.2.2 ParamPresenceMask . . . 764

8.796.2.3 pLTEInfo . . . 764
8.797 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference . . . 764
8.797.1 Detailed Description . . . 764
8.797.2 Field Documentation . . . 765
8.797.2.1 Info . . . 765
8.797.2.2 ParamPresenceMask . . . 765
8.797.2.3 Tlvresult . . . 765
8.798 unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference . . . 765
8.798.1 Detailed Description . . . 765
8.798.2 Field Documentation . . . 765
8.798.2.1 ParamPresenceMask . . . 765
8.798.2.2 t3396_apn . . . 765
8.798.2.3 t3396_plmn_id . . . 765
8.798.2.4 t3396_val . . . 765
8.799 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference . . . 765
8.799.1 Detailed Description . . . 766
8.799.2 Field Documentation . . . 766
8.799.2.1 Info . . . 766
8.799.2.2 ParamPresenceMask . . . 766
8.799.2.3 Tlvresult . . . 766
8.800 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference . . . 766
8.800.1 Detailed Description . . . 766
8.800.2 Field Documentation . . . 767
8.800.2.1 ParamPresenceMask . . . 767
8.800.2.2 pCurrentPersonality . . . 767
8.800.2.3 pPersonalityListLength . . . 767
8.800.2.4 pProtocolSubtypeElement . . . 767
8.801 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference . . . 767
8.801.1 Detailed Description . . . 767
8.801.2 Field Documentation . . . 768
8.801.2.1 pAppSubType . . . 768
8.801.2.2 ParamPresenceMask . . . 768
8.801.2.3 pCurrentPrsnlty . . . 768
8.801.2.4 pPersonalityListLength . . . 768
8.801.2.5 pProtoSubTypElmnt . . . 768
8.802 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference . . . 768
8.802.1 Detailed Description . . . 768
8.802.2 Field Documentation . . . 769
8.802.2.1 ParamPresenceMask . . . 769
8.802.2.2 pDRCParams . . . 769

8.802.2.3 pPilotSetData . . . 769
8.802.2.4 pUATI . . . 769
8.803 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference . . . 769
8.803.1 Detailed Description . . . 769
8.803.2 Field Documentation . . . 770
8.803.2.1 CQIValueCW0 . . . 770
8.803.2.2 CQIValueCW1 . . . 770
8.803.2.3 ParamPresenceMask . . . 770
8.803.2.4 ValidityCW0 . . . 770
8.803.2.5 ValidityCW1 . . . 770
8.804 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference . . . 770
8.804.1 Detailed Description . . . 770
8.804.2 Field Documentation . . . 770
8.804.2.1 ParamPresenceMask . . . 770
8.804.2.2 pSccRxInfo . . . 770
8.805 unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t Struct Reference . . . 770
8.805.1 Detailed Description . . . 770
8.805.2 Field Documentation . . . 771
8.805.2.1 ParamPresenceMask . . . 771
8.805.2.2 pHDRPersInd . . . 771
8.806 unpack_nas_SLQSSwiNetworkDebug_t Struct Reference . . . 771
8.806.1 Detailed Description . . . 771
8.806.2 Field Documentation . . . 772
8.806.2.1 ParamPresenceMask . . . 772
8.806.2.2 pDataStatusDetail . . . 772
8.806.2.3 pDeviceConfigDetail . . . 772
8.806.2.4 pNetworkStat1x . . . 772
8.806.2.5 pNetworkStatEVDO . . . 772
8.806.2.6 pObjectVer . . . 772
8.807 unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t Struct Reference . . . 772
8.807.1 Detailed Description . . . 772
8.807.2 Field Documentation . . . 772
8.807.2.1 ParamPresenceMask . . . 772
8.807.2.2 rankIndicatorTlv . . . 772
8.808 unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference . . . 772
8.808.1 Detailed Description . . . 773
8.808.2 Field Documentation . . . 775
8.808.2.1 pAddCDMASysInfo . . . 776
8.808.2.2 pAddGSMSysInfo . . . 776
8.808.2.3 pAddHDRSysInfo . . . 776

8.808.2.4 pAddLTESysInfo . . . 776
8.808.2.5 pAddWCDMASysInfo . . . 776
8.808.2.6 ParamPresenceMask . . . 776
8.808.2.7 pCDMASrvStatusInfo . . . 776
8.808.2.8 pCDMASysInfo . . . 776
8.808.2.9 pGSMCallBarringSysInfo . . . 776
8.808.2.10 pGSMCipherDomainSysInfo . . . 776
8.808.2.11 pGSMSrvStatusInfo . . . 776
8.808.2.12 pGSMSysInfo . . . 776
8.808.2.13 pHDRSrvStatusInfo . . . 776
8.808.2.14 pHDRSysInfo . . . 776
8.808.2.15 pLteCiotOpModeTlv . . . 776
8.808.2.16 pLTESrvStatusInfo . . . 776
8.808.2.17 pLTESysInfo . . . 776
8.808.2.18 pLTEVoiceSupportSysInfo . . . 776
8.808.2.19 pNR5GCellStatus . . . 776
8.808.2.20 pNR5GSerStatTlv . . . 776
8.808.2.21 pNR5GSystemInfoTlv . . . 776
8.808.2.22 pSysInfoNoChange . . . 776
8.808.2.23 pWCDMACallBarringSysInfo . . . 776
8.808.2.24 pWCDMACipherDomainSysInfo . . . 776
8.808.2.25 pWCDMASrvStatusInfo . . . 776
8.808.2.26 pWCDMASysInfo . . . 776
8.809 unpack_omaDmConfigTlv_t Struct Reference . . . 776
8.809.1 Detailed Description . . . 777
8.809.2 Field Documentation . . . 777
8.809.2.1 alertmsg . . . 777
8.809.2.2 alertmsglength . . . 777
8.809.2.3 state . . . 777
8.809.2.4 userInputReq . . . 777
8.809.2.5 userInputTimeout . . . 777
8.810 unpack_omaDmFotaTlv_t Struct Reference . . . 777
8.810.1 Detailed Description . . . 778
8.810.2 Field Documentation . . . 779
8.810.2.1 description . . . 779
8.810.2.2 descriptionlength . . . 779
8.810.2.3 fwdloadsize . . . 779
8.810.2.4 fwloadComplete . . . 779
8.810.2.5 namelength . . . 779
8.810.2.6 package_name . . . 779

8.810.2.7 sessionType . . . 779
8.810.2.8 severity . . . 779
8.810.2.9 state . . . 779
8.810.2.10 updateCompleteStatus . . . 779
8.810.2.11 userInputReq . . . 779
8.810.2.12 userInputTimeout . . . 779
8.810.2.13 version . . . 779
8.810.2.14 versionLength . . . 779
8.811 unpack_omaDmNotificationsTlv_t Struct Reference . . . 779
8.811.1 Detailed Description . . . 780
8.811.2 Field Documentation . . . 780
8.811.2.1 notification . . . 780
8.811.2.2 sessionStatus . . . 780
8.812 unpack_pds_GetPDSDefaults_t Struct Reference . . . 780
8.812.1 Detailed Description . . . 780
8.812.2 Field Documentation . . . 781
8.812.2.1 pAccuracy . . . 781
8.812.2.2 ParamPresenceMask . . . 781
8.812.2.3 pInterval . . . 781
8.812.2.4 pOperation . . . 781
8.812.2.5 pTimeout . . . 781
8.813 unpack_pds_GetPDSState_t Struct Reference . . . 781
8.813.1 Detailed Description . . . 781
8.813.2 Field Documentation . . . 782
8.813.2.1 ParamPresenceMask . . . 782
8.813.2.2 pEnabledStatus . . . 782
8.813.2.3 pTrackingStatus . . . 782
8.814 unpack_pds_GetPortAutomaticTracking_t Struct Reference . . . 782
8.814.1 Detailed Description . . . 782
8.814.2 Field Documentation . . . 782
8.814.2.1 ParamPresenceMask . . . 783
8.814.2.2 pbAuto . . . 783
8.815 unpack_pds_GetServiceAutomaticTracking_t Struct Reference . . . 783
8.815.1 Detailed Description . . . 783
8.815.2 Field Documentation . . . 783
8.815.2.1 ParamPresenceMask . . . 783
8.815.2.2 pbAuto . . . 783
8.816 unpack_pds_GetXTRAAutomaticDownload_t Struct Reference . . . 783
8.816.1 Detailed Description . . . 783
8.816.2 Field Documentation . . . 784

8.816.2.1 ParamPresenceMask . . . 784
8.816.2.2 pbEnabled . . . 784
8.816.2.3 pInterval . . . 784
8.817 unpack_pds_GetXTRANetwork_t Struct Reference . . . 784
8.817.1 Detailed Description . . . 784
8.817.2 Field Documentation . . . 784
8.817.2.1 ParamPresenceMask . . . 784
8.817.2.2 pPreference . . . 784
8.818 unpack_pds_GetXTRAValidity_t Struct Reference . . . 784
8.818.1 Detailed Description . . . 785
8.818.2 Field Documentation . . . 785
8.818.2.1 ParamPresenceMask . . . 785
8.818.2.2 pDuration . . . 785
8.818.2.3 pGPSWeek . . . 785
8.818.2.4 pGPSWeekOffset . . . 785
8.819 unpack_pds_SetEventReport_Ind_t Struct Reference . . . 785
8.819.1 Detailed Description . . . 786
8.819.2 Field Documentation . . . 787
8.819.2.1 dLatitude . . . 787
8.819.2.2 dLongitude . . . 787
8.819.2.3 has_dLatitude . . . 787
8.819.2.4 has_dLongitude . . . 787
8.819.2.5 has_PositionDataNMEA . . . 787
8.819.2.6 has_posSrc . . . 787
8.819.2.7 has_SessionStatus . . . 787
8.819.2.8 ParamPresenceMask . . . 787
8.819.2.9 PositionDataNMEA . . . 787
8.819.2.10 posSrc . . . 787
8.819.2.11 SessionStatus . . . 787
8.820 unpack_pds_SetPdsState_Ind_t Struct Reference . . . 787
8.820.1 Detailed Description . . . 787
8.820.2 Field Documentation . . . 788
8.820.2.1 EnabledStatus . . . 788
8.820.2.2 ParamPresenceMask . . . 788
8.820.2.3 TlvPresent . . . 788
8.820.2.4 TrackingStatus . . . 788
8.821 unpack_pds_SLQSGetAGPSConfig_t Struct Reference . . . 788
8.821.1 Detailed Description . . . 788
8.821.2 Field Documentation . . . 789
8.821.2.1 ParamPresenceMask . . . 789

8.821.2.2 pServerAddress . . . 789
8.821.2.3 pServerPort . . . 789
8.821.2.4 pServerURL . . . 789
8.821.2.5 pServerURLLength . . . 789
8.822 unpack_pds_SLQSGetGPSStateInfo_t Struct Reference . . . 789
8.822.1 Detailed Description . . . 790
8.822.2 Field Documentation . . . 793
8.822.2.1 Altitude . . . 793
8.822.2.2 EngineState . . . 793
8.822.2.3 glo_almanac_sv_msk . . . 793
8.822.2.4 glo_ephemeris_sv_msk . . . 793
8.822.2.5 glo_health_sv_msk . . . 793
8.822.2.6 glo_visible_sv_msk . . . 793
8.822.2.7 gps_almanac_sv_msk . . . 793
8.822.2.8 gps_ephemeris_sv_msk . . . 793
8.822.2.9 gps_health_sv_msk . . . 793
8.822.2.10 gps_visible_sv_msk . . . 793
8.822.2.11 HorizontalUncertainty . . . 793
8.822.2.12 iono_valid . . . 793
8.822.2.13 Latitude . . . 793
8.822.2.14 Longitude . . . 793
8.822.2.15 ParamPresenceMask . . . 793
8.822.2.16 sbas_almanac_sv_msk . . . 793
8.822.2.17 sbas_ephemeris_sv_msk . . . 793
8.822.2.18 sbas_health_sv_msk . . . 793
8.822.2.19 sbas_visible_sv_msk . . . 793
8.822.2.20 Time_uncert_ms . . . 793
8.822.2.21 TimeStmp_gps_week . . . 793
8.822.2.22 TimeStmp_tow_ms . . . 793
8.822.2.23 ValidMask . . . 793
8.822.2.24 VerticalUncertainty . . . 793
8.822.2.25 xtra_start_gps_minutes . . . 794
8.822.2.26 xtra_start_gps_week . . . 794
8.822.2.27 xtra_valid_duration_hours . . . 794
8.823 unpack_qmi_t Struct Reference . . . 794
8.823.1 Detailed Description . . . 794
8.823.2 Field Documentation . . . 794
8.823.2.1 msgid . . . 794
8.823.2.2 type . . . 794
8.823.2.3 xid . . . 794

8.824 unpack_qos_BindDataPort_t Struct Reference . . . 794
8.824.1 Detailed Description . . . 794
8.824.2 Field Documentation . . . 794
8.824.2.1 ParamPresenceMask . . . 794
8.825 unpack_qos_dataRate_t Struct Reference . . . 795
8.825.1 Detailed Description . . . 795
8.825.2 Field Documentation . . . 795
8.825.2.1 dataRateMax . . . 795
8.825.2.2 guaranteedRate . . . 795
8.826 unpack_qos_IPv4Addr_t Struct Reference . . . 795
8.826.1 Detailed Description . . . 795
8.826.2 Field Documentation . . . 795
8.826.2.1 addr . . . 795
8.826.2.2 subnetMask . . . 795
8.827 unpack_qos_IPv6Addr_t Struct Reference . . . 795
8.827.1 Detailed Description . . . 796
8.827.2 Field Documentation . . . 796
8.827.2.1 addr . . . 796
8.827.2.2 prefixLen . . . 796
8.828 unpack_qos_IPv6TrafCls_t Struct Reference . . . 796
8.828.1 Detailed Description . . . 796
8.828.2 Field Documentation . . . 796
8.828.2.1 mask . . . 796
8.828.2.2 val . . . 796
8.829 unpack_qos_pktErrRate_t Struct Reference . . . 796
8.829.1 Detailed Description . . . 797
8.829.2 Field Documentation . . . 797
8.829.2.1 exponent . . . 797
8.829.2.2 multiplier . . . 797
8.830 unpack_qos_Port_t Struct Reference . . . 797
8.830.1 Detailed Description . . . 797
8.830.2 Field Documentation . . . 797
8.830.2.1 port . . . 797
8.830.2.2 range . . . 797
8.831 unpack_qos_QosFlowInfo_t Struct Reference . . . 797
8.831.1 Detailed Description . . . 798
8.831.2 Field Documentation . . . 798
8.831.2.1 BearerID . . . 798
8.831.2.2 is_RxQFlowGranted_Available . . . 798
8.831.2.3 is_TxQFlowGranted_Available . . . 798

8.831.2.4 NumRxFilters . . . 799
8.831.2.5 NumTxFilters . . . 799
8.831.2.6 QFlowState . . . 799
8.831.2.7 RxQFilter . . . 799
8.831.2.8 RxQFlowGranted . . . 799
8.831.2.9 TxQFilter . . . 799
8.831.2.10 TxQFlowGranted . . . 799
8.832 unpack_qos_QosFlowInfoState_t Struct Reference . . . 799
8.832.1 Detailed Description . . . 799
8.832.2 Field Documentation . . . 799
8.832.2.1 id . . . 799
8.832.2.2 isNewFlow . . . 799
8.832.2.3 state . . . 799
8.833 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference . . . 799
8.833.1 Detailed Description . . . 800
8.833.2 Field Documentation . . . 800
8.833.2.1 NWQoSStatus . . . 800
8.833.2.2 ParamPresenceMask . . . 800
8.834 unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 800
8.834.1 Detailed Description . . . 800
8.834.2 Field Documentation . . . 801
8.834.2.1 ambr_dl . . . 801
8.834.2.2 ambr_dl_ext . . . 801
8.834.2.3 ambr_dl_ext2 . . . 801
8.834.2.4 ambr_ul . . . 801
8.834.2.5 ambr_ul_ext . . . 801
8.834.2.6 ambr_ul_ext2 . . . 801
8.834.2.7 apnId . . . 801
8.834.2.8 ParamPresenceMask . . . 801
8.835 unpack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 802
8.835.1 Detailed Description . . . 802
8.835.2 Field Documentation . . . 803
8.835.2.1 apnId . . . 803
8.835.2.2 numQosFlow . . . 803
8.835.2.3 ParamPresenceMask . . . 803
8.835.2.4 qosFlow . . . 803
8.835.2.5 total_rx_bytes . . . 803
8.835.2.6 total_rx_pkt . . . 803
8.835.2.7 total_tx_bytes . . . 803
8.835.2.8 total_tx_bytes_drp . . . 803

8.835.2.9 total_tx_pkt . . . 803
8.835.2.10 total_tx_pkt_drp . . . 803
8.836 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference . . . 803
8.836.1 Detailed Description . . . 803
8.836.2 Field Documentation . . . 804
8.836.2.1 NumFlows . . . 804
8.836.2.2 ParamPresenceMask . . . 804
8.836.2.3 QosFlowInfo . . . 804
8.837 unpack_qos_SLQSSetQosEventCallback_t Struct Reference . . . 804
8.837.1 Detailed Description . . . 804
8.837.2 Field Documentation . . . 804
8.837.2.1 ParamPresenceMask . . . 804
8.838 unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference . . . 804
8.838.1 Detailed Description . . . 804
8.838.2 Field Documentation . . . 805
8.838.2.1 ParamPresenceMask . . . 805
8.838.2.2 status . . . 805
8.839 unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference . . . 805
8.839.1 Detailed Description . . . 805
8.839.2 Field Documentation . . . 805
8.839.2.1 event . . . 805
8.839.2.2 ParamPresenceMask . . . 805
8.840 unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference . . . 805
8.840.1 Detailed Description . . . 806
8.840.2 Field Documentation . . . 807
8.840.2.1 event . . . 807
8.840.2.2 id . . . 807
8.840.2.3 ParamPresenceMask . . . 807
8.840.2.4 reason . . . 807
8.840.2.5 status . . . 807
8.841 unpack_qos_swiQosFilter_t Struct Reference . . . 807
8.841.1 Detailed Description . . . 808
8.841.2 Field Documentation . . . 809
8.841.2.1 EspSpi . . . 809
8.841.2.2 Id . . . 809
8.841.2.3 index . . . 809
8.841.2.4 IPv4DstAddr . . . 809
8.841.2.5 IPv4SrcAddr . . . 809
8.841.2.6 IPv4Tos . . . 809
8.841.2.7 IPv6DstAddr . . . 809

8.841.2.8 IPv6Label . . . 809
8.841.2.9 IPv6SrcAddr . . . 809
8.841.2.10 IPv6TrafCls . . . 809
8.841.2.11 is_EspSpi_Available . . . 809
8.841.2.12 is_Id_Available . . . 809
8.841.2.13 is_IPv4DstAddr_Available . . . 809
8.841.2.14 is_IPv4SrcAddr_Available . . . 809
8.841.2.15 is_IPv4Tos_Available . . . 809
8.841.2.16 is_IPv6DstAddr_Available . . . 809
8.841.2.17 is_IPv6Label_Available . . . 809
8.841.2.18 is_IPv6SrcAddr_Available . . . 809
8.841.2.19 is_IPv6TrafCls_Available . . . 810
8.841.2.20 is_NxtHdrProto_Available . . . 810
8.841.2.21 is_Precedence_Available . . . 810
8.841.2.22 is_TCPDstPort_Available . . . 810
8.841.2.23 is_TCPSrcPort_Available . . . 810
8.841.2.24 is_TranDstPort_Available . . . 810
8.841.2.25 is_TranSrcPort_Available . . . 810
8.841.2.26 is_UDPDstPort_Available . . . 810
8.841.2.27 is_UDPSrcPort_Available . . . 810
8.841.2.28 NxtHdrProto . . . 810
8.841.2.29 Precedence . . . 810
8.841.2.30 TCPDstPort . . . 810
8.841.2.31 TCPSrcPort . . . 810
8.841.2.32 TranDstPort . . . 810
8.841.2.33 TranSrcPort . . . 810
8.841.2.34 UDPDstPort . . . 810
8.841.2.35 UDPSrcPort . . . 810
8.841.2.36 version . . . 810
8.842 unpack_qos_swiQosFlow_t Struct Reference . . . 810
8.842.1 Detailed Description . . . 811
8.842.2 Field Documentation . . . 813
8.842.2.1 DataRate . . . 813
8.842.2.2 index . . . 813
8.842.2.3 is_DataRate_Available . . . 813
8.842.2.4 is_Jitter_Available . . . 813
8.842.2.5 is_Latency_Available . . . 813
8.842.2.6 is_LteQci_Available . . . 813
8.842.2.7 is_MaxAllowedPktSz_Available . . . 813
8.842.2.8 is_MinPolicedPktSz_Available . . . 813

8.842.2.9 is_PktErrRate_Available . . . 813
8.842.2.10 is_ProfileId3GPP2_Available . . . 813
8.842.2.11 is_TokenBucket_Available . . . 813
8.842.2.12 is_TrafficClass_Available . . . 813
8.842.2.13 is_val_3GPP2Pri_Available . . . 813
8.842.2.14 is_val_3GPPImCn_Available . . . 813
8.842.2.15 is_val_3GPPResResidualBER_Available . . . 814
8.842.2.16 is_val_3GPPSigInd_Available . . . 814
8.842.2.17 is_val_3GPPTraHdlPri_Available . . . 814
8.842.2.18 Jitter . . . 814
8.842.2.19 Latency . . . 814
8.842.2.20 LteQci . . . 814
8.842.2.21 MaxAllowedPktSz . . . 814
8.842.2.22 MinPolicedPktSz . . . 814
8.842.2.23 PktErrRate . . . 814
8.842.2.24 ProfileId3GPP2 . . . 814
8.842.2.25 TokenBucket . . . 814
8.842.2.26 TrafficClass . . . 814
8.842.2.27 val_3GPP2Pri . . . 814
8.842.2.28 val_3GPPImCn . . . 814
8.842.2.29 val_3GPPResResidualBER . . . 814
8.842.2.30 val_3GPPSigInd . . . 814
8.842.2.31 val_3GPPTraHdlPri . . . 814
8.843 unpack_qos_tokenBucket_t Struct Reference . . . 814
8.843.1 Detailed Description . . . 814
8.843.2 Field Documentation . . . 815
8.843.2.1 bucketSz . . . 815
8.843.2.2 peakRate . . . 815
8.843.2.3 tokenRate . . . 815
8.844 unpack_qos_Tos_t Struct Reference . . . 815
8.844.1 Detailed Description . . . 815
8.844.2 Field Documentation . . . 815
8.844.2.1 mask . . . 815
8.844.2.2 val . . . 815
8.845 unpack_QosFlowStat_t Struct Reference . . . 815
8.845.1 Detailed Description . . . 815
8.845.2 Field Documentation . . . 816
8.845.2.1 bearerId . . . 816
8.845.2.2 tx_bytes . . . 816
8.845.2.3 tx_bytes_drp . . . 816

8.845.2.4 tx_pkt . . . 816
8.845.2.5 tx_pkt_drp . . . 816
8.846 unpack_result_t Struct Reference . . . 816
8.846.1 Detailed Description . . . 816
8.846.2 Field Documentation . . . 816
8.846.2.1 ParamPresenceMask . . . 816
8.846.2.2 Tlvresult . . . 816
8.847 unpack_rms_GetSMSWake_t Struct Reference . . . 817
8.847.1 Detailed Description . . . 817
8.847.2 Field Documentation . . . 817
8.847.2.1 enabled . . . 817
8.847.2.2 ParamPresenceMask . . . 817
8.847.2.3 wake_mask . . . 817
8.848 unpack_rms_SetSMSWake_t Struct Reference . . . 817
8.848.1 Detailed Description . . . 817
8.848.2 Field Documentation . . . 818
8.848.2.1 ParamPresenceMask . . . 818
8.849 unpack_RMTransferStatistics_ind_t Struct Reference . . . 818
8.849.1 Detailed Description . . . 818
8.849.2 Field Documentation . . . 818
8.849.2.1 ParamPresenceMask . . . 818
8.849.2.2 RxDropConutTlv . . . 818
8.849.2.3 RxOkByteCountTlv . . . 819
8.849.2.4 RxOkConutTlv . . . 819
8.849.2.5 TxDropConutTlv . . . 819
8.849.2.6 TxOkByteCountTlv . . . 819
8.849.2.7 TxOkConutTlv . . . 819
8.850 unpack_sar_SLQSGetRfSarState_t Struct Reference . . . 819
8.850.1 Detailed Description . . . 819
8.850.2 Field Documentation . . . 819
8.850.2.1 ParamPresenceMask . . . 819
8.850.2.2 pSarRFState . . . 819
8.851 unpack_sms_GetSMSCAddress_t Struct Reference . . . 819
8.851.1 Detailed Description . . . 820
8.851.2 Field Documentation . . . 820
8.851.2.1 addressSize . . . 820
8.851.2.2 ParamPresenceMask . . . 820
8.851.2.3 pSMSCAddress . . . 820
8.851.2.4 pSMSCType . . . 820
8.851.2.5 typeSize . . . 820

8.852 unpack_sms_SaveSMS_t Struct Reference . . . 820
8.852.1 Detailed Description . . . 821
8.852.2 Field Documentation . . . 821
8.852.2.1 ParamPresenceMask . . . 821
8.852.2.2 pMessageIndex . . . 821
8.853 unpack_sms_SendSMS_t Struct Reference . . . 821
8.853.1 Detailed Description . . . 821
8.853.2 Field Documentation . . . 821
8.853.2.1 messageFailureCode . . . 821
8.853.2.2 messageID . . . 821
8.853.2.3 ParamPresenceMask . . . 821
8.854 unpack_sms_SetNewSMSCallback_ind_t Struct Reference . . . 822
8.854.1 Detailed Description . . . 822
8.854.2 Field Documentation . . . 823
8.854.2.1 ETWSPLMNTlv . . . 823
8.854.2.2 ETWSTlv . . . 823
8.854.2.3 IMSTlv . . . 823
8.854.2.4 MMTlv . . . 823
8.854.2.5 NewMMTlv . . . 823
8.854.2.6 ParamPresenceMask . . . 823
8.854.2.7 SMSCTlv . . . 823
8.854.2.8 TRMessageTlv . . . 823
8.855 unpack_sms_SetNewSMSCallback_t Struct Reference . . . 823
8.855.1 Detailed Description . . . 823
8.855.2 Field Documentation . . . 823
8.855.2.1 ParamPresenceMask . . . 823
8.856 unpack_sms_SetSMSCAddress_t Struct Reference . . . 823
8.856.1 Detailed Description . . . 823
8.856.2 Field Documentation . . . 824
8.856.2.1 ParamPresenceMask . . . 824
8.857 unpack_sms_SLQSDeleteSMS_t Struct Reference . . . 824
8.857.1 Detailed Description . . . 824
8.857.2 Field Documentation . . . 824
8.857.2.1 ParamPresenceMask . . . 824
8.858 unpack_sms_SLQSGetIndicationRegister_t Struct Reference . . . 824
8.858.1 Detailed Description . . . 824
8.858.2 Field Documentation . . . 825
8.858.2.1 ParamPresenceMask . . . 825
8.858.2.2 pGetIndicationRegInfo . . . 825
8.859 unpack_sms_SLQSGetMessageWaiting_t Struct Reference . . . 825

8.859.1 Detailed Description . . . 825
8.859.2 Field Documentation . . . 825
8.859.2.1 ParamPresenceMask . . . 825
8.859.2.2 pGetMsgWaitingInfoResp . . . 825
8.860 unpack_sms_SLQSGetSMS_t Struct Reference . . . 825
8.860.1 Detailed Description . . . 825
8.860.2 Field Documentation . . . 826
8.860.2.1 message . . . 826
8.860.2.2 messageFormat . . . 826
8.860.2.3 messageSize . . . 826
8.860.2.4 messageTag . . . 826
8.860.2.5 ParamPresenceMask . . . 826
8.861 unpack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 826
8.861.1 Detailed Description . . . 827
8.861.2 Field Documentation . . . 827
8.861.2.1 ParamPresenceMask . . . 827
8.861.2.2 pBroadcastConfig . . . 827
8.861.2.3 pCDMABroadcastConfig . . . 827
8.862 unpack_sms_SLQSGetSMSList_t Struct Reference . . . 827
8.862.1 Detailed Description . . . 827
8.862.2 Field Documentation . . . 828
8.862.2.1 messageList . . . 828
8.862.2.2 messageListSize . . . 828
8.862.2.3 ParamPresenceMask . . . 828
8.863 unpack_sms_SLQSGetTransLayerInfo_t Struct Reference . . . 828
8.863.1 Detailed Description . . . 828
8.863.2 Field Documentation . . . 828
8.863.2.1 ParamPresenceMask . . . 828
8.863.2.2 pGetTransLayerInfo . . . 828
8.864 unpack_sms_SLQSGetTransNWRegInfo_t Struct Reference . . . 828
8.864.1 Detailed Description . . . 829
8.864.2 Field Documentation . . . 829
8.864.2.1 ParamPresenceMask . . . 829
8.864.2.2 pGetTransNWRegInfo . . . 829
8.865 unpack_sms_SLQSModifySMSStatus_t Struct Reference . . . 829
8.865.1 Detailed Description . . . 829
8.865.2 Field Documentation . . . 829
8.865.2.1 ParamPresenceMask . . . 829
8.866 unpack_sms_SLQSNWRegInfoCallback_ind_t Struct Reference . . . 829
8.866.1 Detailed Description . . . 830

8.866.2 Field Documentation . . . 830
8.866.2.1 NWRegStat . . . 830
8.866.2.2 ParamPresenceMask . . . 830
8.867 unpack_sms_SLQSSendAsyncSMS_t Struct Reference . . . 830
8.867.1 Detailed Description . . . 830
8.867.2 Field Documentation . . . 830
8.867.2.1 ParamPresenceMask . . . 830
8.868 unpack_sms_SLQSSetIndicationRegister_t Struct Reference . . . 830
8.868.1 Detailed Description . . . 831
8.868.2 Field Documentation . . . 831
8.868.2.1 ParamPresenceMask . . . 831
8.869 unpack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference . . . 831
8.869.1 Detailed Description . . . 831
8.869.2 Field Documentation . . . 831
8.869.2.1 ParamPresenceMask . . . 831
8.870 unpack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference . . . 831
8.870.1 Detailed Description . . . 831
8.870.2 Field Documentation . . . 832
8.870.2.1 ParamPresenceMask . . . 832
8.871 unpack_sms_SLQSSetSmsStorage_t Struct Reference . . . 832
8.871.1 Detailed Description . . . 832
8.871.2 Field Documentation . . . 832
8.871.2.1 ParamPresenceMask . . . 832
8.872 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 832
8.872.1 Detailed Description . . . 832
8.872.2 Field Documentation . . . 832
8.872.2.1 ParamPresenceMask . . . 833
8.872.2.2 pMaxStorageSizeResp . . . 833
8.873 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference . . . 833
8.873.1 Detailed Description . . . 833
8.873.2 Field Documentation . . . 833
8.873.2.1 ParamPresenceMask . . . 833
8.873.2.2 pMessageProtocol . . . 833
8.874 unpack_sms_SLQSSmsSetRoutes_t Struct Reference . . . 833
8.874.1 Detailed Description . . . 833
8.874.2 Field Documentation . . . 833
8.874.2.1 ParamPresenceMask . . . 833
8.875 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference . . . 834
8.875.1 Detailed Description . . . 834
8.875.2 Field Documentation . . . 834

8.875.2.1 ParamPresenceMask . . . 834
8.875.2.2 pSmsStorage . . . 834
8.876 unpack_sms_SLQSTransLayerInfoCallback_ind_t Struct Reference . . . 834
8.876.1 Detailed Description . . . 834
8.876.2 Field Documentation . . . 835
8.876.2.1 ParamPresenceMask . . . 835
8.876.2.2 pTransLayerInfo . . . 835
8.876.2.3 regInd . . . 835
8.877 unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t Struct Reference . . . 835
8.877.1 Detailed Description . . . 835
8.877.2 Field Documentation . . . 837
8.877.2.1 alphaIDLen . . . 837
8.877.2.2 causeCode . . . 837
8.877.2.3 errorClass . . . 837
8.877.2.4 messageID . . . 837
8.877.2.5 msgDelFailureCause . . . 837
8.877.2.6 msgDelFailureType . . . 837
8.877.2.7 pAlphaID . . . 837
8.877.2.8 ParamPresenceMask . . . 837
8.877.2.9 RPCause . . . 837
8.877.2.10 sendStatus . . . 837
8.877.2.11 TPCause . . . 837
8.877.2.12 userData . . . 837
8.878 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference . . . 837
8.878.1 Detailed Description . . . 837
8.878.2 Field Documentation . . . 838
8.878.2.1 messageMode . . . 838
8.878.2.2 ParamPresenceMask . . . 838
8.878.2.3 storageType . . . 838
8.879 unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t Struct Reference . . . 838
8.879.1 Detailed Description . . . 838
8.879.2 Field Documentation . . . 838
8.879.2.1 msgWaitInfo . . . 838
8.879.2.2 numInstances . . . 838
8.879.2.3 ParamPresenceMask . . . 838
8.880 unpack_swiaudio_SLQSGetM2MAudioProfile_t Struct Reference . . . 838
8.880.1 Detailed Description . . . 839
8.880.2 Field Documentation . . . 839
8.880.2.1 CwtMute . . . 839
8.880.2.2 EarMute . . . 839

8.880.2.3 Generator . . . 839
8.880.2.4 MicMute . . . 840
8.880.2.5 ParamPresenceMask . . . 840
8.880.2.6 Profile . . . 840
8.880.2.7 Volume . . . 840
8.881 unpack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference . . . 840
8.881.1 Detailed Description . . . 840
8.881.2 Field Documentation . . . 840
8.881.2.1 Level . . . 840
8.881.2.2 ParamPresenceMask . . . 840
8.882 unpack_swiaudio_SLQSGetM2MAVMute_t Struct Reference . . . 840
8.882.1 Detailed Description . . . 840
8.882.2 Field Documentation . . . 841
8.882.2.1 CwtMute . . . 841
8.882.2.2 EarMute . . . 841
8.882.2.3 MicMute . . . 841
8.882.2.4 ParamPresenceMask . . . 841
8.883 unpack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference . . . 841
8.883.1 Detailed Description . . . 841
8.883.2 Field Documentation . . . 842
8.883.2.1 ParamPresenceMask . . . 842
8.883.2.2 Value . . . 842
8.884 unpack_swiavms_SLQSAVMSEventReportInd_t Struct Reference . . . 842
8.884.1 Detailed Description . . . 842
8.884.2 Field Documentation . . . 843
8.884.2.1 ParamPresenceMask . . . 843
8.884.2.2 pBinaryUpdateSessionInfo . . . 843
8.884.2.3 pConfig . . . 843
8.884.2.4 pConnectionRequest . . . 843
8.884.2.5 pDataSessionStatus . . . 843
8.884.2.6 pHTTPStatus . . . 843
8.884.2.7 pNotification . . . 843
8.884.2.8 pPackageID . . . 843
8.884.2.9 pRegStatus . . . 843
8.884.2.10 pSessionType . . . 843
8.884.2.11 pWAMSParaChanged . . . 843
8.884.2.12 resultcode . . . 843
8.885 unpack_swiavms_SLQSAVMSGetSettings_t Struct Reference . . . 843
8.885.1 Detailed Description . . . 844
8.885.2 Field Documentation . . . 845

8.885.2.1 AutoConnect . . . 845
8.885.2.2 AutoReboot . . . 845
8.885.2.3 FwAutoSDM . . . 845
8.885.2.4 FwPromptdownload . . . 846
8.885.2.5 FwPromptUpdate . . . 846
8.885.2.6 OMADMEnabled . . . 846
8.885.2.7 pAPNInfo . . . 846
8.885.2.8 ParamPresenceMask . . . 846
8.885.2.9 pConnectionRetryTimers . . . 846
8.885.2.10 pNotificationStore . . . 846
8.885.2.11 pPeroidsInfo . . . 846
8.885.2.12 pPollingTimer . . . 846
8.885.2.13 resultcode . . . 846
8.886 unpack_swiavms_SLQSAVMSGetSettings_v2_t Struct Reference . . . 846
8.886.1 Detailed Description . . . 846
8.886.2 Field Documentation . . . 848
8.886.2.1 AutoConnect . . . 848
8.886.2.2 FwAutoSDM . . . 848
8.886.2.3 FwPromptdownload . . . 848
8.886.2.4 FwPromptUpdate . . . 848
8.886.2.5 OMADMEnabled . . . 848
8.886.2.6 pAPNInfo . . . 848
8.886.2.7 ParamPresenceMask . . . 848
8.886.2.8 pAutoReboot . . . 848
8.886.2.9 pConnectionRetryTimers . . . 848
8.886.2.10 pNotificationStore . . . 848
8.886.2.11 pPeroidsInfo . . . 848
8.886.2.12 pPollingTimer . . . 848
8.886.2.13 resultcode . . . 848
8.887 unpack_swiavms_SLQSAVMSSendSelection_t Struct Reference . . . 848
8.887.1 Detailed Description . . . 848
8.887.2 Field Documentation . . . 849
8.887.2.1 ParamPresenceMask . . . 849
8.887.2.2 resultcode . . . 849
8.888 unpack_swiavms_SLQSAVMSSessionGetInfo_t Struct Reference . . . 849
8.888.1 Detailed Description . . . 849
8.888.2 Field Documentation . . . 849
8.888.2.1 ParamPresenceMask . . . 849
8.888.2.2 pBinaryUpdateSessionInfo . . . 849
8.888.2.3 pConfig . . . 850

8.888.2.4 pNotification . . . 850
8.888.2.5 pPackageID . . . 850
8.888.2.6 resultcode . . . 850
8.889 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference . . . 850
8.889.1 Detailed Description . . . 850
8.889.2 Field Documentation . . . 850
8.889.2.1 ParamPresenceMask . . . 850
8.889.2.2 resultcode . . . 850
8.890 unpack_swiavms_SLQSAVMSSetSettings_t Struct Reference . . . 850
8.890.1 Detailed Description . . . 850
8.890.2 Field Documentation . . . 850
8.890.2.1 ParamPresenceMask . . . 850
8.890.2.2 resultcode . . . 851
8.891 unpack_swiavms_SLQSAVMSSetSettings_v2_t Struct Reference . . . 851
8.891.1 Detailed Description . . . 851
8.891.2 Field Documentation . . . 851
8.891.2.1 ParamPresenceMask . . . 851
8.891.2.2 resultcode . . . 851
8.892 unpack_swiavms_SLQSAVMSStartSession_t Struct Reference . . . 851
8.892.1 Detailed Description . . . 851
8.892.2 Field Documentation . . . 851
8.892.2.1 ParamPresenceMask . . . 851
8.892.2.2 resultcode . . . 852
8.892.2.3 sessionResponse . . . 852
8.893 unpack_swiavms_SLQSAVMSStopSession_t Struct Reference . . . 852
8.893.1 Detailed Description . . . 852
8.893.2 Field Documentation . . . 852
8.893.2.1 ParamPresenceMask . . . 852
8.893.2.2 resultcode . . . 852
8.894 unpack_swidms_SLQSSwiDmsGetMTU_t Struct Reference . . . 852
8.894.1 Detailed Description . . . 852
8.894.2 Field Documentation . . . 853
8.894.2.1 ParamPresenceMask . . . 853
8.894.2.2 pEhrpdMTUSize . . . 853
8.894.2.3 pHrpdMTUSize . . . 853
8.894.2.4 pMTUSize3gpp . . . 853
8.894.2.5 pUsbMTUSize . . . 853
8.895 unpack_swidms_SLQSSwiDmsGetUsbComp_t Struct Reference . . . 853
8.895.1 Detailed Description . . . 853
8.895.2 Field Documentation . . . 853

8.895.2.1 ParamPresenceMask . . . 854
8.895.2.2 pInterfaceCfg . . . 854
8.895.2.3 pSupportedBitmasks . . . 854
8.896 unpack_swidms_SLQSSwiDmsGetUsbNetNum_t Struct Reference . . . 854
8.896.1 Detailed Description . . . 854
8.896.2 Field Documentation . . . 854
8.896.2.1 ParamPresenceMask . . . 854
8.896.2.2 usbNetNum . . . 854
8.897 unpack_swidms_SLQSSwiDmsSetMTU_t Struct Reference . . . 854
8.897.1 Detailed Description . . . 854
8.897.2 Field Documentation . . . 855
8.897.2.1 ParamPresenceMask . . . 855
8.897.2.2 TlvResult . . . 855
8.898 unpack_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference . . . 855
8.898.1 Detailed Description . . . 855
8.898.2 Field Documentation . . . 855
8.898.2.1 ParamPresenceMask . . . 855
8.898.2.2 TlvResult . . . 855
8.899 unpack_swiloc_SwiLocGetAutoStart_t Struct Reference . . . 855
8.899.1 Detailed Description . . . 855
8.899.2 Field Documentation . . . 857
8.899.2.1 fix_rate . . . 857
8.899.2.2 fix_rate_reported . . . 857
8.899.2.3 fix_type . . . 857
8.899.2.4 fix_type_reported . . . 857
8.899.2.5 function . . . 857
8.899.2.6 function_reported . . . 857
8.899.2.7 max_dist . . . 857
8.899.2.8 max_dist_reported . . . 857
8.899.2.9 max_time . . . 857
8.899.2.10 max_time_reported . . . 857
8.899.2.11 ParamPresenceMask . . . 857
8.900 unpack_swioma_SLQSOMADMAlertCallback_ind_t Struct Reference . . . 857
8.900.1 Detailed Description . . . 857
8.900.2 Field Documentation . . . 858
8.900.2.1 eventType . . . 858
8.900.2.2 ParamPresenceMask . . . 858
8.900.2.3 SessionInfoConfig . . . 858
8.900.2.4 SessionInfoFota . . . 858
8.900.2.5 SessionInfoNotification . . . 858

8.901 unpack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 858
8.901.1 Detailed Description . . . 858
8.901.2 Field Documentation . . . 860
8.901.2.1 Date . . . 860
8.901.2.2 DateLength . . . 861
8.901.2.3 ParamPresenceMask . . . 861
8.901.2.4 PkgDescLength . . . 861
8.901.2.5 PkgDescription . . . 861
8.901.2.6 PkgName . . . 861
8.901.2.7 PkgNameLength . . . 861
8.901.2.8 RetryCount . . . 861
8.901.2.9 SessionState . . . 861
8.901.2.10 SessionType . . . 861
8.901.2.11 Severity . . . 861
8.901.2.12 Source . . . 861
8.901.2.13 SourceLength . . . 861
8.901.2.14 Status . . . 861
8.901.2.15 Time . . . 861
8.901.2.16 TimeLength . . . 861
8.901.2.17 UpdateCompleteStatus . . . 861
8.902 unpack_swioma_SLQSOMADMGetSettings_t Struct Reference . . . 861
8.902.1 Detailed Description . . . 861
8.902.2 Field Documentation . . . 862
8.902.2.1 Autosdm . . . 862
8.902.2.2 FOTAdownload . . . 862
8.902.2.3 FOTAUpdate . . . 862
8.902.2.4 FwAutoCheck . . . 863
8.902.2.5 OMADMEnabled . . . 863
8.902.2.6 ParamPresenceMask . . . 863
8.903 unpack_swioma_SLQSOMADMStartSession_t Struct Reference . . . 863
8.903.1 Detailed Description . . . 863
8.903.2 Field Documentation . . . 863
8.903.2.1 FwAvailability . . . 863
8.903.2.2 ParamPresenceMask . . . 863
8.904 unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference . . . 863
8.904.1 Detailed Description . . . 863
8.904.2 Field Documentation . . . 864
8.904.2.1 ParamPresenceMask . . . 864
8.904.2.2 Tlvresult . . . 864
8.905 unpack_tmd_SLQSTmdGetMitigationDevList_t Struct Reference . . . 864

8.905.1 Detailed Description . . . 864
8.905.2 Field Documentation . . . 864
8.905.2.1 MitigationDevList . . . 864
8.905.2.2 MitigationDevListLen . . . 865
8.905.2.3 ParamPresenceMask . . . 865
8.905.2.4 Tlvresult . . . 865
8.906 unpack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference . . . 865
8.906.1 Detailed Description . . . 865
8.906.2 Field Documentation . . . 865
8.906.2.1 CurrentmitigationLvl . . . 865
8.906.2.2 ParamPresenceMask . . . 865
8.906.2.3 ReqMitigationLvl . . . 865
8.906.2.4 Tlvresult . . . 865
8.907 unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t Struct Reference . . . 865
8.907.1 Detailed Description . . . 866
8.907.2 Field Documentation . . . 866
8.907.2.1 deviceID . . . 866
8.907.2.2 deviceIdLen . . . 866
8.907.2.3 lvl . . . 866
8.907.2.4 ParamPresenceMask . . . 866
8.908 unpack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference . . . 866
8.908.1 Detailed Description . . . 866
8.908.2 Field Documentation . . . 867
8.908.2.1 ParamPresenceMask . . . 867
8.908.2.2 Tlvresult . . . 867
8.909 unpack_uim_ChangePin_t Struct Reference . . . 867
8.909.1 Detailed Description . . . 867
8.909.2 Field Documentation . . . 867
8.909.2.1 ParamPresenceMask . . . 867
8.909.2.2 pEncryptedPIN1 . . . 867
8.909.2.3 pIndicationToken . . . 867
8.909.2.4 pRemainingRetries . . . 868
8.909.2.5 Tlvresult . . . 868
8.910 unpack_uim_GetCardStatus_t Struct Reference . . . 868
8.910.1 Detailed Description . . . 868
8.910.2 Field Documentation . . . 868
8.910.2.1 ParamPresenceMask . . . 868
8.910.2.2 pCardStatus . . . 868
8.910.2.3 pHotSwapStatus . . . 868
8.910.2.4 Tlvresult . . . 868

8.911 unpack_uim_ReadTransparent_t Struct Reference . . . 868
8.911.1 Detailed Description . . . 869
8.911.2 Field Documentation . . . 869
8.911.2.1 ParamPresenceMask . . . 869
8.911.2.2 pCardResult . . . 869
8.911.2.3 pEncryptedData . . . 869
8.911.2.4 pIndicationToken . . . 869
8.911.2.5 pReadResult . . . 869
8.911.2.6 Tlvresult . . . 869
8.912 unpack_uim_SetPinProtection_t Struct Reference . . . 869
8.912.1 Detailed Description . . . 870
8.912.2 Field Documentation . . . 870
8.912.2.1 ParamPresenceMask . . . 870
8.912.2.2 pEncryptedPIN1 . . . 870
8.912.2.3 pIndicationToken . . . 870
8.912.2.4 pRemainingRetries . . . 870
8.912.2.5 Tlvresult . . . 870
8.913 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference . . . 870
8.913.1 Detailed Description . . . 870
8.913.2 Field Documentation . . . 871
8.913.2.1 bNumberOfPhySlots . . . 871
8.913.2.2 ParamPresenceMask . . . 871
8.913.2.3 slotsstatusChange . . . 871
8.914 unpack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 871
8.914.1 Detailed Description . . . 871
8.914.2 Field Documentation . . . 871
8.914.2.1 ParamPresenceMask . . . 871
8.914.2.2 pAuthenticateResult . . . 872
8.914.2.3 pCardResult . . . 872
8.914.2.4 pIndicationToken . . . 872
8.915 unpack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 872
8.915.1 Detailed Description . . . 872
8.915.2 Field Documentation . . . 872
8.915.2.1 ParamPresenceMask . . . 872
8.915.2.2 pRemainingRetries . . . 872
8.916 unpack_uim_SLQSUIMEventRegister_t Struct Reference . . . 872
8.916.1 Detailed Description . . . 872
8.916.2 Field Documentation . . . 873
8.916.2.1 eventMask . . . 873
8.916.2.2 ParamPresenceMask . . . 873

8.917 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 873
8.917.1 Detailed Description . . . 873
8.917.2 Field Documentation . . . 873
8.917.2.1 ParamPresenceMask . . . 874
8.917.2.2 pAutoSelection . . . 874
8.917.2.3 pHaltSubscription . . . 874
8.917.2.4 pPersonalizationStatus . . . 874
8.918 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 874
8.918.1 Detailed Description . . . 874
8.918.2 Field Documentation . . . 874
8.918.2.1 ParamPresenceMask . . . 874
8.918.2.2 pCardResult . . . 874
8.918.2.3 pFileAttributes . . . 874
8.918.2.4 pIndicationToken . . . 874
8.919 unpack_uim_SLQSUIMGetSlotsStatus_t Struct Reference . . . 874
8.919.1 Detailed Description . . . 875
8.919.2 Field Documentation . . . 875
8.919.2.1 ParamPresenceMask . . . 875
8.919.2.2 pNumberOfPhySlot . . . 875
8.919.2.3 pUimSlotsStatus . . . 875
8.920 unpack_uim_SLQSUIMRefreshCallback_Ind_t Struct Reference . . . 875
8.920.1 Detailed Description . . . 875
8.920.2 Field Documentation . . . 876
8.920.2.1 ParamPresenceMask . . . 876
8.920.2.2 refreshEvent . . . 876
8.920.2.3 TlvPresent . . . 876
8.921 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 876
8.921.1 Detailed Description . . . 876
8.921.2 Field Documentation . . . 876
8.921.2.1 ParamPresenceMask . . . 876
8.921.2.2 pRefreshEvent . . . 876
8.922 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference . . . 876
8.922.1 Detailed Description . . . 876
8.922.2 Field Documentation . . . 877
8.922.2.1 ParamPresenceMask . . . 877
8.922.2.2 pCardStatus . . . 877
8.923 unpack_uim_UnblockPin_t Struct Reference . . . 877
8.923.1 Detailed Description . . . 877
8.923.2 Field Documentation . . . 877
8.923.2.1 ParamPresenceMask . . . 877

8.923.2.2 pEncryptedPIN1 . . . 878
8.923.2.3 pIndicationToken . . . 878
8.923.2.4 pRemainingRetries . . . 878
8.923.2.5 Tlvresult . . . 878
8.924 unpack_uim_VerifyPin_t Struct Reference . . . 878
8.924.1 Detailed Description . . . 878
8.924.2 Field Documentation . . . 878
8.924.2.1 ParamPresenceMask . . . 878
8.924.2.2 pEncryptedPIN1 . . . 878
8.924.2.3 pIndicationToken . . . 878
8.924.2.4 pRemainingRetries . . . 878
8.924.2.5 Tlvresult . . . 878
8.925 unpack_voice_allCallStatusCallback_ind_t Struct Reference . . . 879
8.925.1 Detailed Description . . . 879
8.925.2 Field Documentation . . . 880
8.925.2.1 arrCallInfomation . . . 880
8.925.2.2 ParamPresenceMask . . . 880
8.925.2.3 pArrAlertingPattern . . . 880
8.925.2.4 pArrAlertingType . . . 880
8.925.2.5 pArrAlphaID . . . 880
8.925.2.6 pArrCalledPartyNum . . . 880
8.925.2.7 pArrCallEndReason . . . 880
8.925.2.8 pArrConnectPartyNum . . . 880
8.925.2.9 pArrDiagInfo . . . 880
8.925.2.10 pArrRedirPartyNum . . . 880
8.925.2.11 pArrRemotePartyName . . . 880
8.925.2.12 pArrRemotePartyNum . . . 880
8.925.2.13 pArrSvcOption . . . 880
8.926 unpack_voice_DTMFEventCallback_ind_t Struct Reference . . . 881
8.926.1 Detailed Description . . . 881
8.926.2 Field Documentation . . . 881
8.926.2.1 DTMFInformation . . . 881
8.926.2.2 ParamPresenceMask . . . 881
8.926.2.3 pOffLength . . . 881
8.926.2.4 pOnLength . . . 882
8.927 unpack_voice_OTASPStatusCallback_ind_t Struct Reference . . . 882
8.927.1 Detailed Description . . . 882
8.927.2 Field Documentation . . . 883
8.927.2.1 callID . . . 883
8.927.2.2 OTASPStatus . . . 883

8.927.2.3 ParamPresenceMask . . . 883
8.928 unpack_voice_SLQSOriginateUSSD_t Struct Reference . . . 883
8.928.1 Detailed Description . . . 883
8.928.2 Field Documentation . . . 885
8.928.2.1 pAlphaIDInfo . . . 885
8.928.2.2 ParamPresenceMask . . . 885
8.928.2.3 pCallId . . . 885
8.928.2.4 pCcResultType . . . 885
8.928.2.5 pCCSuppsType . . . 885
8.928.2.6 pfailureCause . . . 885
8.928.2.7 pUSSDInfo . . . 885
8.929 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference . . . 885
8.929.1 Detailed Description . . . 885
8.929.2 Field Documentation . . . 886
8.929.2.1 ParamPresenceMask . . . 886
8.929.2.2 pCallId . . . 886
8.930 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference . . . 886
8.930.1 Detailed Description . . . 886
8.930.2 Field Documentation . . . 886
8.930.2.1 ParamPresenceMask . . . 886
8.930.2.2 pCallID . . . 886
8.931 unpack_voice_SLQSVoiceDialCall_t Struct Reference . . . 886
8.931.1 Detailed Description . . . 887
8.931.2 Field Documentation . . . 887
8.931.2.1 pAlphaIDInfo . . . 887
8.931.2.2 ParamPresenceMask . . . 887
8.931.2.3 pCallID . . . 887
8.931.2.4 pCCResultType . . . 887
8.931.2.5 pCCSUPSType . . . 887
8.932 unpack_voice_SLQSVoiceEndCall_t Struct Reference . . . 887
8.932.1 Detailed Description . . . 888
8.932.2 Field Documentation . . . 888
8.932.2.1 ParamPresenceMask . . . 888
8.932.2.2 pCallId . . . 888
8.933 unpack_voice_SLQSVoiceGetAllCallInfo_t Struct Reference . . . 888
8.933.1 Detailed Description . . . 888
8.933.2 Field Documentation . . . 891
8.933.2.1 ParamPresenceMask . . . 891
8.933.2.2 pArrAlertingPattern . . . 891
8.933.2.3 pArrAlertingType . . . 891

8.933.2.4 pArrAlphaID . . . 891
8.933.2.5 pArrCalledPartyNum . . . 891
8.933.2.6 pArrCallEndReason . . . 891
8.933.2.7 pArrCallInfo . . . 891
8.933.2.8 pArrConnectPartyNum . . . 891
8.933.2.9 pArrDiagInfo . . . 891
8.933.2.10 pArrRedirPartyNum . . . 891
8.933.2.11 pArrRemotePartyName . . . 891
8.933.2.12 pArrRemotePartyNum . . . 891
8.933.2.13 pArrSvcOption . . . 891
8.933.2.14 pArrUUSInfo . . . 891
8.933.2.15 pOTASPStatus . . . 891
8.933.2.16 pVoicePrivacy . . . 891
8.934 unpack_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 892
8.934.1 Detailed Description . . . 892
8.934.2 Field Documentation . . . 893
8.934.2.1 pAlphaIDInfo . . . 893
8.934.2.2 ParamPresenceMask . . . 893
8.934.2.3 pCallID . . . 893
8.934.2.4 pCCResType . . . 893
8.934.2.5 pCCSUPSType . . . 893
8.934.2.6 pFailCause . . . 893
8.934.2.7 pSvcClass . . . 893
8.935 unpack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 893
8.935.1 Detailed Description . . . 893
8.935.2 Field Documentation . . . 894
8.935.2.1 pAlphaIDInfo . . . 894
8.935.2.2 ParamPresenceMask . . . 894
8.935.2.3 pCallID . . . 894
8.935.2.4 pCCResType . . . 894
8.935.2.5 pCCSUPSType . . . 894
8.935.2.6 pFailCause . . . 895
8.935.2.7 pGetCallFWExtInfo . . . 895
8.935.2.8 pGetCallFWInfo . . . 895
8.936 unpack_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 895
8.936.1 Detailed Description . . . 895
8.936.2 Field Documentation . . . 897
8.936.2.1 pAlertingPattern . . . 897
8.936.2.2 pAlertType . . . 897
8.936.2.3 pAlphaIDInfo . . . 897

8.936.2.4 ParamPresenceMask . . . 897
8.936.2.5 pCallInfo . . . 897
8.936.2.6 pConnectNumInfo . . . 897
8.936.2.7 pDiagInfo . . . 897
8.936.2.8 pOTASPStatus . . . 897
8.936.2.9 pRemotePartyName . . . 897
8.936.2.10 pRemotePartyNum . . . 897
8.936.2.11 pSrvOpt . . . 898
8.936.2.12 pUUSInfo . . . 898
8.936.2.13 pVoicePrivacy . . . 898
8.937 unpack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 898
8.937.1 Detailed Description . . . 898
8.937.2 Field Documentation . . . 899
8.937.2.1 pAlphaIDInfo . . . 899
8.937.2.2 ParamPresenceMask . . . 899
8.937.2.3 pCallID . . . 899
8.937.2.4 pCCResType . . . 899
8.937.2.5 pCCSUPSType . . . 899
8.937.2.6 pFailCause . . . 899
8.937.2.7 pSvcClass . . . 899
8.938 unpack_voice_SLQSVoiceGetCLIP_t Struct Reference . . . 899
8.938.1 Detailed Description . . . 899
8.938.2 Field Documentation . . . 900
8.938.2.1 pAlphaIDInfo . . . 900
8.938.2.2 ParamPresenceMask . . . 900
8.938.2.3 pCallID . . . 900
8.938.2.4 pCCResType . . . 900
8.938.2.5 pCCSUPSType . . . 900
8.938.2.6 pCLIPResp . . . 901
8.938.2.7 pFailCause . . . 901
8.939 unpack_voice_SLQSVoiceGetCLIR_t Struct Reference . . . 901
8.939.1 Detailed Description . . . 901
8.939.2 Field Documentation . . . 902
8.939.2.1 pAlphaIDInfo . . . 902
8.939.2.2 ParamPresenceMask . . . 902
8.939.2.3 pCallID . . . 902
8.939.2.4 pCCResType . . . 902
8.939.2.5 pCCSUPSType . . . 902
8.939.2.6 pCLIRResp . . . 902
8.939.2.7 pFailCause . . . 902

8.940 unpack_voice_SLQSVoiceGetCNAP_t Struct Reference . . . 902
8.940.1 Detailed Description . . . 902
8.940.2 Field Documentation . . . 903
8.940.2.1 pAlphaIDInfo . . . 903
8.940.2.2 ParamPresenceMask . . . 903
8.940.2.3 pCallID . . . 903
8.940.2.4 pCCResType . . . 903
8.940.2.5 pCCSUPSType . . . 903
8.940.2.6 pCNAPResp . . . 903
8.940.2.7 pFailCause . . . 903
8.941 unpack_voice_SLQSVoiceGetCOLP_t Struct Reference . . . 904
8.941.1 Detailed Description . . . 904
8.941.2 Field Documentation . . . 905
8.941.2.1 pAlphaIDInfo . . . 905
8.941.2.2 ParamPresenceMask . . . 905
8.941.2.3 pCallID . . . 905
8.941.2.4 pCCResType . . . 905
8.941.2.5 pCCSUPSType . . . 905
8.941.2.6 pCOLPResp . . . 905
8.941.2.7 pFailCause . . . 905
8.942 unpack_voice_SLQSVoiceGetCOLR_t Struct Reference . . . 905
8.942.1 Detailed Description . . . 905
8.942.2 Field Documentation . . . 906
8.942.2.1 pAlphaIDInfo . . . 906
8.942.2.2 ParamPresenceMask . . . 906
8.942.2.3 pCallID . . . 906
8.942.2.4 pCCResType . . . 906
8.942.2.5 pCCSUPSType . . . 906
8.942.2.6 pCOLRResp . . . 906
8.942.2.7 pFailCause . . . 906
8.943 unpack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 906
8.943.1 Detailed Description . . . 907
8.943.2 Field Documentation . . . 908
8.943.2.1 pAirTimerCnt . . . 908
8.943.2.2 ParamPresenceMask . . . 908
8.943.2.3 pAutoAnswerStat . . . 908
8.943.2.4 pCurAMRConfig . . . 908
8.943.2.5 pCurPrefVoiceSO . . . 908
8.943.2.6 pCurrTTYMode . . . 908
8.943.2.7 pCurVoiceDomainPref . . . 908

8.943.2.8 pCurVoicePrivacyPref . . . 908
8.943.2.9 pRoamTimerCnt . . . 908
8.944 unpack_voice_SLQSVoiceManageCalls_t Struct Reference . . . 908
8.944.1 Detailed Description . . . 909
8.944.2 Field Documentation . . . 909
8.944.2.1 ParamPresenceMask . . . 909
8.944.2.2 pFailCause . . . 909
8.945 unpack_voice_SLQSVoiceSendFlash_t Struct Reference . . . 909
8.945.1 Detailed Description . . . 909
8.945.2 Field Documentation . . . 909
8.945.2.1 ParamPresenceMask . . . 909
8.945.2.2 pCallID . . . 910
8.946 unpack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 910
8.946.1 Detailed Description . . . 910
8.946.2 Field Documentation . . . 911
8.946.2.1 pAlphaIDInfo . . . 911
8.946.2.2 ParamPresenceMask . . . 911
8.946.2.3 pCallID . . . 911
8.946.2.4 pCCResType . . . 911
8.946.2.5 pCCSUPSType . . . 911
8.946.2.6 pFailCause . . . 911
8.947 unpack_voice_SLQSVoiceSetConfig_t Struct Reference . . . 911
8.947.1 Detailed Description . . . 911
8.947.2 Field Documentation . . . 912
8.947.2.1 pAirTimerStatus . . . 912
8.947.2.2 ParamPresenceMask . . . 912
8.947.2.3 pAutoAnsStatus . . . 912
8.947.2.4 pPrefVoiceSOStatus . . . 912
8.947.2.5 pRoamTimerStatus . . . 912
8.947.2.6 pTTYConfigStatus . . . 912
8.947.2.7 pVoiceDomainPrefStatus . . . 912
8.948 unpack_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 913
8.948.1 Detailed Description . . . 913
8.948.2 Field Documentation . . . 914
8.948.2.1 pAlphaIDInfo . . . 914
8.948.2.2 ParamPresenceMask . . . 914
8.948.2.3 pCallID . . . 914
8.948.2.4 pCCResultType . . . 914
8.948.2.5 pCCSUPSType . . . 914
8.948.2.6 pFailCause . . . 914

8.949 unpack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 914
8.949.1 Detailed Description . . . 914
8.949.2 Field Documentation . . . 914
8.949.2.1 ParamPresenceMask . . . 914
8.949.2.2 pCallID . . . 914
8.950 unpack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 914
8.950.1 Detailed Description . . . 915
8.950.2 Field Documentation . . . 915
8.950.2.1 callID . . . 915
8.950.2.2 ParamPresenceMask . . . 915
8.951 unpack_voice_SLQSVoiceSUPSCallback_ind_t Struct Reference . . . 915
8.951.1 Detailed Description . . . 915
8.951.2 Field Documentation . . . 917
8.951.2.1 pAlphaIDInfo . . . 917
8.951.2.2 ParamPresenceMask . . . 917
8.951.2.3 pCallBarPasswd . . . 917
8.951.2.4 pCallFwdInfo . . . 918
8.951.2.5 pCallFWNum . . . 918
8.951.2.6 pCallFWTimerVal . . . 918
8.951.2.7 pCallID . . . 918
8.951.2.8 pCLIPstatus . . . 918
8.951.2.9 pCLIRstatus . . . 918
8.951.2.10 pCNAPstatus . . . 918
8.951.2.11 pCOLPstatus . . . 918
8.951.2.12 pCOLRstatus . . . 918
8.951.2.13 pDataSrc . . . 918
8.951.2.14 pFailCause . . . 918
8.951.2.15 pNewPwdData . . . 918
8.951.2.16 pReason . . . 918
8.951.2.17 pSvcClass . . . 918
8.951.2.18 pUSSInfo . . . 918
8.951.2.19 SUPSInformation . . . 918
8.952 unpack_voice_SUPSNotificationCallback_ind_t Struct Reference . . . 918
8.952.1 Detailed Description . . . 918
8.952.2 Field Documentation . . . 920
8.952.2.1 callID . . . 920
8.952.2.2 notifType . . . 920
8.952.2.3 ParamPresenceMask . . . 920
8.952.2.4 pCUGIndex . . . 920
8.952.2.5 pECTNum . . . 920

8.953 unpack_voice_USSDNotificationCallback_ind_t Struct Reference . . . 920
8.953.1 Detailed Description . . . 920
8.953.2 Field Documentation . . . 920
8.953.2.1 notification_Type . . . 920
8.953.2.2 ParamPresenceMask . . . 920
8.953.2.3 USSDNotificationNetworkInfo . . . 920
8.954 unpack_voice_VoiceInfoRecCallback_ind_t Struct Reference . . . 920
8.954.1 Detailed Description . . . 921
8.954.2 Field Documentation . . . 923
8.954.2.1 callID . . . 923
8.954.2.2 ParamPresenceMask . . . 923
8.954.2.3 pCalledPartyInfo . . . 923
8.954.2.4 pCallerIDInfo . . . 923
8.954.2.5 pCallerNameInfo . . . 923
8.954.2.6 pCallingPartyInfo . . . 923
8.954.2.7 pCallWaitInd . . . 923
8.954.2.8 pCLIRCause . . . 923
8.954.2.9 pConnectNumInfo . . . 923
8.954.2.10 pDispInfo . . . 923
8.954.2.11 pExtDispInfo . . . 923
8.954.2.12 pExtDispRecInfo . . . 923
8.954.2.13 pLineCtrlInfo . . . 923
8.954.2.14 pNSSAudioCtrl . . . 923
8.954.2.15 pNSSRelease . . . 923
8.954.2.16 pRedirNumInfo . . . 923
8.954.2.17 pSignalInfo . . . 923
8.955 unpack_voice_voicePrivacyChangeCallback_ind_t Struct Reference . . . 923
8.955.1 Detailed Description . . . 923
8.955.2 Field Documentation . . . 924
8.955.2.1 callID . . . 924
8.955.2.2 ParamPresenceMask . . . 924
8.955.2.3 voicePrivacy . . . 924
8.956 unpack_wds_DHCPv4ClientLease_ind_t Struct Reference . . . 924
8.956.1 Detailed Description . . . 924
8.956.2 Field Documentation . . . 925
8.956.2.1 DHCPv4LeaseOptTlv . . . 925
8.956.2.2 DHCPv4LeaseStateTlv . . . 925
8.956.2.3 IPv4AddrTlv . . . 925
8.956.2.4 ParamPresenceMask . . . 925
8.956.2.5 ProfileIdTlv . . . 925

8.957 unpack_wds_GetAutoconnect_t Struct Reference . . . 925
8.957.1 Detailed Description . . . 925
8.957.2 Field Documentation . . . 925
8.957.2.1 ParamPresenceMask . . . 925
8.957.2.2 psetting . . . 925
8.958 unpack_wds_GetByteTotals_t Struct Reference . . . 925
8.958.1 Detailed Description . . . 925
8.958.2 Field Documentation . . . 926
8.958.2.1 ParamPresenceMask . . . 926
8.958.2.2 pRXTotalBytes . . . 926
8.958.2.3 pTXTotalBytes . . . 926
8.959 unpack_wds_GetConnectionRate_t Struct Reference . . . 926
8.959.1 Detailed Description . . . 926
8.959.2 Field Documentation . . . 927
8.959.2.1 currentChannelRXRate . . . 927
8.959.2.2 currentChannelTXRate . . . 927
8.959.2.3 maxChannelRXRate . . . 927
8.959.2.4 maxChannelTXRate . . . 927
8.959.2.5 ParamPresenceMask . . . 927
8.960 unpack_wds_GetDataBearerTechnology_t Struct Reference . . . 927
8.960.1 Detailed Description . . . 927
8.960.2 Field Documentation . . . 928
8.960.2.1 ParamPresenceMask . . . 928
8.960.2.2 pDataBearer . . . 928
8.961 unpack_wds_GetDefaultProfile_t Struct Reference . . . 928
8.961.1 Detailed Description . . . 929
8.961.2 Field Documentation . . . 930
8.961.2.1 apnname . . . 930
8.961.2.2 apnsize . . . 930
8.961.2.3 auth . . . 930
8.961.2.4 ipaddr . . . 931
8.961.2.5 ipaddrv6 . . . 931
8.961.2.6 name . . . 931
8.961.2.7 namesize . . . 931
8.961.2.8 ParamPresenceMask . . . 931
8.961.2.9 pdptype . . . 931
8.961.2.10 pridns . . . 931
8.961.2.11 pridnsv6 . . . 931
8.961.2.12 secdns . . . 931
8.961.2.13 secdnsv6 . . . 931

8.961.2.14 username . . . 931
8.961.2.15 usersize . . . 931
8.962 unpack_wds_GetDefaultProfileNum_t Struct Reference . . . 931
8.962.1 Detailed Description . . . 931
8.962.2 Field Documentation . . . 931
8.962.2.1 index . . . 931
8.962.2.2 ParamPresenceMask . . . 931
8.963 unpack_wds_GetDormancyState_t Struct Reference . . . 931
8.963.1 Detailed Description . . . 932
8.963.2 Field Documentation . . . 932
8.963.2.1 dormancyState . . . 932
8.963.2.2 ParamPresenceMask . . . 932
8.964 unpack_wds_GetLastMobileIPError_t Struct Reference . . . 932
8.964.1 Detailed Description . . . 932
8.964.2 Field Documentation . . . 932
8.964.2.1 error . . . 933
8.964.2.2 ParamPresenceMask . . . 933
8.965 unpack_wds_GetMobileIP_t Struct Reference . . . 933
8.965.1 Detailed Description . . . 933
8.965.2 Field Documentation . . . 933
8.965.2.1 mipMode . . . 933
8.965.2.2 ParamPresenceMask . . . 933
8.966 unpack_wds_GetMobileIPProfile_t Struct Reference . . . 933
8.966.1 Detailed Description . . . 934
8.966.2 Field Documentation . . . 935
8.966.2.1 AAASPI . . . 935
8.966.2.2 AAAState . . . 935
8.966.2.3 address . . . 935
8.966.2.4 enabled . . . 935
8.966.2.5 HASPI . . . 935
8.966.2.6 HAState . . . 935
8.966.2.7 NAI . . . 935
8.966.2.8 naiSize . . . 935
8.966.2.9 ParamPresenceMask . . . 935
8.966.2.10 primaryHA . . . 935
8.966.2.11 revTunneling . . . 935
8.966.2.12 secondaryHA . . . 935
8.967 unpack_wds_GetPacketStatistics_t Struct Reference . . . 935
8.967.1 Detailed Description . . . 936
8.967.2 Field Documentation . . . 937

8.967.2.1 ParamPresenceMask . . . 937
8.967.2.2 pRXDroppedCount . . . 937
8.967.2.3 pRXOkBytesCount . . . 937
8.967.2.4 pRXOKBytesLastCall . . . 937
8.967.2.5 pRXPacketErrors . . . 937
8.967.2.6 pRXPacketOverflows . . . 937
8.967.2.7 pRXPacketSuccesses . . . 937
8.967.2.8 pTXDroppedCount . . . 937
8.967.2.9 pTXOkBytesCount . . . 937
8.967.2.10 pTXOKBytesLastCall . . . 937
8.967.2.11 pTXPacketErrors . . . 937
8.967.2.12 pTXPacketOverflows . . . 937
8.967.2.13 pTXPacketSuccesses . . . 937
8.968 unpack_wds_GetPacketStatus_t Struct Reference . . . 937
8.968.1 Detailed Description . . . 938
8.968.2 Field Documentation . . . 939
8.968.2.1 ParamPresenceMask . . . 939
8.968.2.2 rXDroppedCount . . . 939
8.968.2.3 rXOkBytesCount . . . 939
8.968.2.4 rXOKBytesLastCall . . . 939
8.968.2.5 rXPacketErrors . . . 939
8.968.2.6 rXPacketOverflows . . . 939
8.968.2.7 rXPacketSuccesses . . . 939
8.968.2.8 tXDroppedCount . . . 939
8.968.2.9 tXOkBytesCount . . . 939
8.968.2.10 tXOKBytesLastCall . . . 939
8.968.2.11 tXPacketErrors . . . 939
8.968.2.12 tXPacketOverflows . . . 939
8.968.2.13 tXPacketSuccesses . . . 939
8.969 unpack_wds_GetSessionDuration_t Struct Reference . . . 939
8.969.1 Detailed Description . . . 939
8.969.2 Field Documentation . . . 940
8.969.2.1 callDuration . . . 940
8.969.2.2 ParamPresenceMask . . . 940
8.970 unpack_wds_GetSessionState_t Struct Reference . . . 940
8.970.1 Detailed Description . . . 940
8.970.2 Field Documentation . . . 940
8.970.2.1 connectionStatus . . . 940
8.970.2.2 ParamPresenceMask . . . 940
8.971 unpack_wds_RMSetTransferStatistics_t Struct Reference . . . 940

8.971.1 Detailed Description . . . 941
8.971.2 Field Documentation . . . 941
8.971.2.1 ParamPresenceMask . . . 941
8.972 unpack_wds_SetMobileIPProfile_t Struct Reference . . . 941
8.972.1 Detailed Description . . . 941
8.972.2 Field Documentation . . . 941
8.972.2.1 ParamPresenceMask . . . 941
8.973 unpack_wds_SLQSCreateProfile_t Struct Reference . . . 941
8.973.1 Detailed Description . . . 941
8.973.2 Field Documentation . . . 942
8.973.2.1 ParamPresenceMask . . . 942
8.973.2.2 pCreateProfileOut . . . 942
8.973.2.3 pProfileID . . . 942
8.973.2.4 Tlvresult . . . 942
8.974 unpack_wds_SLQSDeleteProfile_t Struct Reference . . . 942
8.974.1 Detailed Description . . . 942
8.974.2 Field Documentation . . . 942
8.974.2.1 extendedErrorCode . . . 942
8.974.2.2 ParamPresenceMask . . . 942
8.975 unpack_wds_SLQSDUNCallInfoCallBack_ind_t Struct Reference . . . 943
8.975.1 Detailed Description . . . 943
8.975.2 Field Documentation . . . 944
8.975.2.1 CRTlv . . . 944
8.975.2.2 CSTlv . . . 944
8.975.2.3 DBTTlv . . . 944
8.975.2.4 DSTlv . . . 944
8.975.2.5 LMCERTlv . . . 944
8.975.2.6 ParamPresenceMask . . . 944
8.975.2.7 RXBOTlv . . . 944
8.975.2.8 TXBOTlv . . . 944
8.976 unpack_wds_SLQSGet3GPPConfigItem_t Struct Reference . . . 944
8.976.1 Detailed Description . . . 944
8.976.2 Field Documentation . . . 945
8.976.2.1 _3gppRelease . . . 945
8.976.2.2 defaultPDNEnabled . . . 945
8.976.2.3 LTEAttachProfile . . . 946
8.976.2.4 LTEAttachProfileList . . . 946
8.976.2.5 LTEAttachProfileListLen . . . 946
8.976.2.6 ParamPresenceMask . . . 946
8.976.2.7 profileList . . . 946

8.977 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 946
8.977.1 Detailed Description . . . 946
8.977.2 Field Documentation . . . 946
8.977.2.1 currNetworkInfo . . . 946
8.977.2.2 networkInfoLen . . . 946
8.977.2.3 ParamPresenceMask . . . 947
8.977.2.4 prefNetwork . . . 947
8.978 unpack_wds_SLQSGetCurrentChannelRate_t Struct Reference . . . 947
8.978.1 Detailed Description . . . 947
8.978.2 Field Documentation . . . 948
8.978.2.1 current_channel_rx_rate . . . 948
8.978.2.2 current_channel_tx_rate . . . 948
8.978.2.3 max_channel_rx_rate . . . 948
8.978.2.4 max_channel_tx_rate . . . 948
8.978.2.5 ParamPresenceMask . . . 948
8.979 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 948
8.979.1 Detailed Description . . . 948
8.979.2 Field Documentation . . . 948
8.979.2.1 curDataBearerTechnology . . . 948
8.979.2.2 dataBearerMask . . . 949
8.979.2.3 lastCallDataBearerTechnology . . . 949
8.979.2.4 ParamPresenceMask . . . 949
8.980 unpack_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 949
8.980.1 Detailed Description . . . 949
8.980.2 Field Documentation . . . 951
8.980.2.1 callEndReason . . . 951
8.980.2.2 channelRate . . . 951
8.980.2.3 connectionStatus . . . 951
8.980.2.4 dataBearerTech . . . 951
8.980.2.5 dormancyStatus . . . 951
8.980.2.6 lastCallDataBearerTech . . . 951
8.980.2.7 lastCallRXOKBytesCnt . . . 951
8.980.2.8 lastCallTXOKBytesCnt . . . 952
8.980.2.9 mdmCallDurationActive . . . 952
8.980.2.10 ParamPresenceMask . . . 952
8.980.2.11 rxOKBytesCount . . . 952
8.980.2.12 txOKBytesCount . . . 952
8.981 unpack_wds_SLQSGetProfileSettings_t Struct Reference . . . 952
8.981.1 Detailed Description . . . 952
8.981.2 Field Documentation . . . 952

8.981.2.1 ParamPresenceMask . . . 952
8.981.2.2 pProfileSettings . . . 952
8.981.2.3 ProfileType . . . 952
8.981.2.4 Tlvresult . . . 952
8.982 unpack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 952
8.982.1 Detailed Description . . . 953
8.982.2 Field Documentation . . . 955
8.982.2.1 APNName . . . 955
8.982.2.2 Authentication . . . 955
8.982.2.3 DomainList . . . 955
8.982.2.4 GPRSGrantedQoS . . . 955
8.982.2.5 GWAddressV4 . . . 955
8.982.2.6 IMCNflag . . . 955
8.982.2.7 IPFamilyPreference . . . 955
8.982.2.8 IPv4 . . . 955
8.982.2.9 IPV6AddrInfo . . . 955
8.982.2.10 IPV6GWAddrInfo . . . 955
8.982.2.11 Mtu . . . 955
8.982.2.12 ParamPresenceMask . . . 955
8.982.2.13 PCSCFAddrPCO . . . 955
8.982.2.14 PCSCFFQDNAddrList . . . 955
8.982.2.15 PDPType . . . 955
8.982.2.16 PrimaryDNSV4 . . . 955
8.982.2.17 PrimaryDNSV6 . . . 955
8.982.2.18 ProfileID . . . 955
8.982.2.19 ProfileName . . . 955
8.982.2.20 SecondaryDNSV4 . . . 955
8.982.2.21 SecondaryDNSV6 . . . 955
8.982.2.22 ServerAddrList . . . 955
8.982.2.23 SubnetMaskV4 . . . 955
8.982.2.24 Technology . . . 956
8.982.2.25 UMTSGrantedQoS . . . 956
8.982.2.26 Username . . . 956
8.983 unpack_wds_SLQSModifyProfile_t Struct Reference . . . 956
8.983.1 Detailed Description . . . 956
8.983.2 Field Documentation . . . 956
8.983.2.1 ParamPresenceMask . . . 956
8.983.2.2 pExtErrorCode . . . 956
8.984 unpack_wds_SLQSSetIPFamilyPreference_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 TlvResult . . . 957
8.985 unpack_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference . . . 957
8.985.1 Detailed Description . . . 957
8.985.2 Field Documentation . . . 958
8.985.2.1 bearerID . . . 958
8.985.2.2 conn_status . . . 959
8.985.2.3 ipFamily . . . 959
8.985.2.4 ParamPresenceMask . . . 959
8.985.2.5 reconfigReqd . . . 959
8.985.2.6 sessionEndReason . . . 959
8.985.2.7 techName . . . 959
8.985.2.8 verboseSessnEndReason . . . 959
8.985.2.9 verboseSessnEndReasonType . . . 959
8.986 unpack_wds_SLQSSetWdsEventCallback_ind_t Struct Reference . . . 959
8.986.1 Detailed Description . . . 959
8.986.2 Field Documentation . . . 963
8.986.2.1 currDBTechAvail . . . 963
8.986.2.2 currNWInfo . . . 963
8.986.2.3 dataSysStatAvail . . . 963
8.986.2.4 dBTechAvail . . . 963
8.986.2.5 dBtechExtAvail . . . 964
8.986.2.6 dBTechExtRatValue . . . 964
8.986.2.7 dBTechExtSoMask . . . 964
8.986.2.8 dBTechnology . . . 964
8.986.2.9 dBtechnologyExt . . . 964
8.986.2.10 dormancyStatAvail . . . 964
8.986.2.11 dormancyStatus . . . 964
8.986.2.12 mipstatAvail . . . 964
8.986.2.13 mipStatus . . . 964
8.986.2.14 netInfoLen . . . 964
8.986.2.15 ParamPresenceMask . . . 964
8.986.2.16 prefNetwork . . . 964
8.986.2.17 ratMask . . . 964
8.986.2.18 rx_bytes . . . 964
8.986.2.19 rx_pkts . . . 964
8.986.2.20 soMask . . . 964
8.986.2.21 tx_bytes . . . 964
8.986.2.22 tx_pkts . . . 964

8.986.2.23 xferStatAvail . . . 964
8.987 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 964
8.987.1 Detailed Description . . . 964
8.987.2 Field Documentation . . . 965
8.987.2.1 ParamPresenceMask . . . 965
8.987.2.2 pHwConfig . . . 965
8.987.2.3 pRequestOptionList . . . 965
8.988 unpack_wds_SLQSSGetLoopback_t Struct Reference . . . 965
8.988.1 Detailed Description . . . 965
8.988.2 Field Documentation . . . 966
8.988.2.1 ByteLoopbackMode . . . 966
8.988.2.2 ByteLoopbackMultiplier . . . 966
8.988.2.3 ParamPresenceMask . . . 966
8.989 unpack_wds_SLQSStartDataSession_t Struct Reference . . . 966
8.989.1 Detailed Description . . . 966
8.989.2 Field Documentation . . . 966
8.989.2.1 ParamPresenceMask . . . 966
8.989.2.2 pFailureReason . . . 966
8.989.2.3 psid . . . 967
8.989.2.4 pVerboseFailReasonType . . . 967
8.989.2.5 pVerboseFailureReason . . . 967
8.990 unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 967
8.990.1 Detailed Description . . . 967
8.990.2 Field Documentation . . . 969
8.990.2.1 apnName . . . 969
8.990.2.2 bearerId . . . 969
8.990.2.3 contextId . . . 969
8.990.2.4 ipv4Address . . . 969
8.990.2.5 ipv4GWAddress . . . 969
8.990.2.6 ipv6Address . . . 969
8.990.2.7 ipv6GWAddress . . . 969
8.990.2.8 ParamPresenceMask . . . 969
8.990.2.9 prDNSIPv4Address . . . 969
8.990.2.10 prDNSIPv6Address . . . 969
8.990.2.11 prPCSCFIPv4Address . . . 969
8.990.2.12 prPCSCFIPv6Address . . . 969
8.990.2.13 seDNSIPv4Address . . . 969
8.990.2.14 seDNSIPv6Address . . . 969
8.990.2.15 sePCSCFIPv4Address . . . 969
8.990.2.16 sePCSCFIPv6Address . . . 969

8.991 UnPackGetProfileSettingOut Struct Reference . . . 969
8.991.1 Detailed Description . . . 969
8.991.2 Field Documentation . . . 970
8.991.2.1 curProfile . . . 970
8.991.2.2 pExtErrCode . . . 970
8.992 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo Struct Reference . . . 970
8.992.1 Detailed Description . . . 970
8.992.2 Field Documentation . . . 971
8.992.2.1 bBinaryType . . . 971
8.992.2.2 bSerity . . . 971
8.992.2.3 bState . . . 971
8.992.2.4 bUserInputRequest . . . 971
8.992.2.5 szDescription . . . 971
8.992.2.6 szName . . . 971
8.992.2.7 szVersion . . . 971
8.992.2.8 TlvPresent . . . 971
8.992.2.9 ulPkgDownloadComplete . . . 972
8.992.2.10 ulPkgDownloadSize . . . 972
8.992.2.11 wDescriptionLength . . . 972
8.992.2.12 wNameLength . . . 972
8.992.2.13 wUpdateCompeteStatus . . . 972
8.992.2.14 wUserInputTimeout . . . 972
8.992.2.15 wVersionLength . . . 972
8.993 UnpackSwiAvmsEventReportConfig Struct Reference . . . 972
8.993.1 Detailed Description . . . 972
8.993.2 Field Documentation . . . 972
8.993.2.1 bState . . . 972
8.993.2.2 bUserInputRequest . . . 972
8.993.2.3 szAlertMsg . . . 973
8.993.2.4 TlvPresent . . . 973
8.993.2.5 wAlertMsgLength . . . 973
8.993.2.6 wUserInputTimeout . . . 973
8.994 UnpackSwiAvmsEventReportConnectionRequest Struct Reference . . . 973
8.994.1 Detailed Description . . . 973
8.994.2 Field Documentation . . . 973
8.994.2.1 bUserInputRequest . . . 973
8.994.2.2 TlvPresent . . . 973
8.994.2.3 wUserInputTimeout . . . 973
8.995 UnpackSwiAvmsEventReportDataSessionStatus Struct Reference . . . 973
8.995.1 Detailed Description . . . 973

8.995.2 Field Documentation . . . 974
8.995.2.1 bType . . . 974
8.995.2.2 TlvPresent . . . 974
8.995.2.3 wErrorCode . . . 974
8.996 UnpackSwiAvmsEventReportHTTPStatus Struct Reference . . . 974
8.996.1 Detailed Description . . . 974
8.996.2 Field Documentation . . . 975
8.996.2.1 TlvPresent . . . 975
8.996.2.2 wHTTPStatus . . . 976
8.997 UnpackSwiAvmsEventReportNotification Struct Reference . . . 976
8.997.1 Detailed Description . . . 976
8.997.2 Field Documentation . . . 976
8.997.2.1 bNotification . . . 977
8.997.2.2 TlvPresent . . . 977
8.997.2.3 wSessionStatus . . . 977
8.998 UnpackSwiAvmsEventReportPackageID Struct Reference . . . 977
8.998.1 Detailed Description . . . 977
8.998.2 Field Documentation . . . 977
8.998.2.1 bPackageID . . . 977
8.998.2.2 TlvPresent . . . 977
8.999 UnpackSwiAvmsEventReportRegStatus Struct Reference . . . 977
8.999.1 Detailed Description . . . 977
8.999.2 Field Documentation . . . 977
8.999.2.1 bRegStatus . . . 977
8.999.2.2 TlvPresent . . . 978
8.1000 UnpackSwiAvmsEventReportSessionType Struct Reference . . . 978
8.1000.1 Detailed Description . . . 978
8.1000.2 Field Documentation . . . 978
8.1000.2.1 bType . . . 978
8.1000.2.2 TlvPresent . . . 978
8.1001 UnpackSwiAvmsEventReportWAMSParamChange Struct Reference . . . 978
8.1001.1 Detailed Description . . . 978
8.1001.2 Field Documentation . . . 979
8.1001.2.1 TlvPresent . . . 979
8.1001.2.2 wWamsChangeMask . . . 979
8.1002 UnpackWdsProfileParam Union Reference . . . 979
8.1002.1 Detailed Description . . . 979
8.1002.2 Field Documentation . . . 979
8.1002.2.1 SlqsProfile3GPP . . . 979
8.1002.2.2 SlqsProfile3GPP2 . . . 979

8.1003 voice_airTimer Struct Reference . . . 979
8.1003.1 Detailed Description . . . 979
8.1003.2 Field Documentation . . . 980
8.1003.2.1 airTimerValue . . . 980
8.1003.2.2 camID . . . 980
8.1004 voice_allCallsAlphaIDInfo Struct Reference . . . 980
8.1004.1 Detailed Description . . . 980
8.1004.2 Field Documentation . . . 980
8.1004.2.1 AlphaIDInfo . . . 980
8.1004.2.2 callID . . . 980
8.1005 voice_allCallsDiagInfo Struct Reference . . . 980
8.1005.1 Detailed Description . . . 980
8.1005.2 Field Documentation . . . 981
8.1005.2.1 callID . . . 981
8.1005.2.2 DiagInfo . . . 981
8.1006 voice_allCallsUUSInfo Struct Reference . . . 981
8.1006.1 Detailed Description . . . 981
8.1006.2 Field Documentation . . . 981
8.1006.2.1 callID . . . 981
8.1006.2.2 uusInfo . . . 981
8.1007 voice_alphaIDInfo Struct Reference . . . 981
8.1007.1 Detailed Description . . . 981
8.1007.2 Field Documentation . . . 982
8.1007.2.1 alphaDcs . . . 982
8.1007.2.2 alphaLen . . . 982
8.1007.2.3 alphaText . . . 982
8.1008 voice_arrAlertingPattern Struct Reference . . . 982
8.1008.1 Detailed Description . . . 982
8.1008.2 Field Documentation . . . 983
8.1008.2.1 alertingPattern . . . 983
8.1008.2.2 callID . . . 983
8.1008.2.3 numInstances . . . 983
8.1009 voice_arrAlertingType Struct Reference . . . 983
8.1009.1 Detailed Description . . . 983
8.1009.2 Field Documentation . . . 984
8.1009.2.1 AlertingType . . . 984
8.1009.2.2 callID . . . 984
8.1009.2.3 numInstances . . . 984
8.1010 voice_arrAlphaID Struct Reference . . . 984
8.1010.1 Detailed Description . . . 984

8.1010.2 Field Documentation . . . 984
8.1010.2.1 allCallsAlphaIDInfoArr . . . 984
8.1010.2.2 numInstances . . . 984
8.1011 voice_arrCalledPartyNum Struct Reference . . . 984
8.1011.1 Detailed Description . . . 985
8.1011.2 Field Documentation . . . 985
8.1011.2.1 CalledPartyNum . . . 985
8.1011.2.2 numInstances . . . 985
8.1012 voice_arrCallEndReason Struct Reference . . . 985
8.1012.1 Detailed Description . . . 985
8.1012.2 Field Documentation . . . 985
8.1012.2.1 callEndReason . . . 985
8.1012.2.2 callID . . . 986
8.1012.2.3 numInstances . . . 986
8.1013 voice_arrCallInfo Struct Reference . . . 986
8.1013.1 Detailed Description . . . 986
8.1013.2 Field Documentation . . . 986
8.1013.2.1 getAllCallInfo . . . 986
8.1013.2.2 numInstances . . . 986
8.1014 voice_arrConnectPartyNum Struct Reference . . . 986
8.1014.1 Detailed Description . . . 986
8.1014.2 Field Documentation . . . 987
8.1014.2.1 ConnectedPartyNum . . . 987
8.1014.2.2 numInstances . . . 987
8.1015 voice_arrDiagInfo Struct Reference . . . 987
8.1015.1 Detailed Description . . . 987
8.1015.2 Field Documentation . . . 987
8.1015.2.1 DiagInfo . . . 987
8.1015.2.2 numInstances . . . 987
8.1016 voice_arrRedirPartyNum Struct Reference . . . 987
8.1016.1 Detailed Description . . . 987
8.1016.2 Field Documentation . . . 988
8.1016.2.1 numInstances . . . 988
8.1016.2.2 RedirPartyNum . . . 988
8.1017 voice_arrRemotePartyName Struct Reference . . . 988
8.1017.1 Detailed Description . . . 988
8.1017.2 Field Documentation . . . 988
8.1017.2.1 GetAllCallRmtPtyName . . . 988
8.1017.2.2 numInstances . . . 988
8.1018 voice_arrRemotePartyNum Struct Reference . . . 988

8.1018.1 Detailed Description . . . 988
8.1018.2 Field Documentation . . . 989
8.1018.2.1 numInstances . . . 989
8.1018.2.2 RmtPtyNum . . . 989
8.1019 voice_arrSvcOption Struct Reference . . . 989
8.1019.1 Detailed Description . . . 989
8.1019.2 Field Documentation . . . 989
8.1019.2.1 callID . . . 989
8.1019.2.2 numInstances . . . 989
8.1019.2.3 srvOption . . . 989
8.1020 voice_arrUUSInfo Struct Reference . . . 989
8.1020.1 Detailed Description . . . 990
8.1020.2 Field Documentation . . . 990
8.1020.2.1 AllCallsUUSInfo . . . 990
8.1020.2.2 numInstances . . . 990
8.1021 voice_burstDTMFInfo Struct Reference . . . 990
8.1021.1 Detailed Description . . . 990
8.1021.2 Field Documentation . . . 991
8.1021.2.1 digitCnt . . . 991
8.1021.2.2 pCallID . . . 991
8.1021.2.3 pDigitBuff . . . 991
8.1022 voice_calledPartyInfo Struct Reference . . . 991
8.1022.1 Detailed Description . . . 991
8.1022.2 Field Documentation . . . 992
8.1022.2.1 number . . . 992
8.1022.2.2 numLen . . . 992
8.1022.2.3 numPlan . . . 992
8.1022.2.4 numType . . . 992
8.1022.2.5 PI . . . 992
8.1022.2.6 SI . . . 992
8.1023 voice_calledPartySubAdd Struct Reference . . . 992
8.1023.1 Detailed Description . . . 993
8.1023.2 Field Documentation . . . 993
8.1023.2.1 extBit . . . 993
8.1023.2.2 oddEvenInd . . . 993
8.1023.2.3 subAddr . . . 993
8.1023.2.4 subAddrLen . . . 993
8.1023.2.5 subAddrType . . . 993
8.1024 voice_callerIDInfo Struct Reference . . . 993
8.1024.1 Detailed Description . . . 994

8.1024.2 Field Documentation . . . 994
8.1024.2.1 callerID . . . 994
8.1024.2.2 callerIDLen . . . 994
8.1024.2.3 PI . . . 994
8.1025 voice_callFwdTypeAndPlan Struct Reference . . . 994
8.1025.1 Detailed Description . . . 994
8.1025.2 Field Documentation . . . 995
8.1025.2.1 numberPlan . . . 995
8.1025.2.2 numberType . . . 995
8.1026 voice_callFWExtInfo Struct Reference . . . 995
8.1026.1 Detailed Description . . . 995
8.1026.2 Field Documentation . . . 996
8.1026.2.1 noReplyTimer . . . 996
8.1026.2.2 number . . . 996
8.1026.2.3 numLen . . . 997
8.1026.2.4 numPlan . . . 997
8.1026.2.5 numType . . . 997
8.1026.2.6 PI . . . 997
8.1026.2.7 SI . . . 997
8.1026.2.8 SvcClass . . . 997
8.1026.2.9 SvcStatus . . . 997
8.1027 voice_callFWInfo Struct Reference . . . 997
8.1027.1 Detailed Description . . . 997
8.1027.2 Field Documentation . . . 997
8.1027.2.1 noReplyTimer . . . 997
8.1027.2.2 number . . . 998
8.1027.2.3 numLen . . . 998
8.1027.2.4 SvcClass . . . 998
8.1027.2.5 SvcStatus . . . 998
8.1028 voice_callInfo Struct Reference . . . 998
8.1028.1 Detailed Description . . . 998
8.1028.2 Field Documentation . . . 999
8.1028.2.1 callID . . . 999
8.1028.2.2 callState . . . 999
8.1028.2.3 callType . . . 999
8.1028.2.4 direction . . . 999
8.1028.2.5 mode . . . 999
8.1029 voice_callingPartyInfo Struct Reference . . . 999
8.1029.1 Detailed Description . . . 1000
8.1029.2 Field Documentation . . . 1001

8.1029.2.1 number . . . 1001
8.1029.2.2 numLen . . . 1001
8.1029.2.3 numPlan . . . 1001
8.1029.2.4 numType . . . 1001
8.1029.2.5 PI . . . 1001
8.1029.2.6 SI . . . 1001
8.1030 voice_ccSUPSType Struct Reference . . . 1001
8.1030.1 Detailed Description . . . 1001
8.1030.2 Field Documentation . . . 1001
8.1030.2.1 reason . . . 1002
8.1030.2.2 svcType . . . 1002
8.1031 voice_CLIPResp Struct Reference . . . 1002
8.1031.1 Detailed Description . . . 1002
8.1031.2 Field Documentation . . . 1002
8.1031.2.1 ActiveStatus . . . 1002
8.1031.2.2 ProvisionStatus . . . 1002
8.1032 voice_CLIRResp Struct Reference . . . 1002
8.1032.1 Detailed Description . . . 1002
8.1032.2 Field Documentation . . . 1003
8.1032.2.1 ActiveStatus . . . 1003
8.1032.2.2 ProvisionStatus . . . 1003
8.1033 voice_CNAPResp Struct Reference . . . 1003
8.1033.1 Detailed Description . . . 1003
8.1033.2 Field Documentation . . . 1004
8.1033.2.1 ActiveStatus . . . 1004
8.1033.2.2 ProvisionStatus . . . 1004
8.1034 voice_COLPResp Struct Reference . . . 1004
8.1034.1 Detailed Description . . . 1004
8.1034.2 Field Documentation . . . 1004
8.1034.2.1 ActiveStatus . . . 1004
8.1034.2.2 ProvisionStatus . . . 1004
8.1035 voice_COLRResp Struct Reference . . . 1004
8.1035.1 Detailed Description . . . 1004
8.1035.2 Field Documentation . . . 1005
8.1035.2.1 ActiveStatus . . . 1005
8.1035.2.2 ProvisionStatus . . . 1005
8.1036 voice_connectNumInfo Struct Reference . . . 1005
8.1036.1 Detailed Description . . . 1005
8.1036.2 Field Documentation . . . 1006
8.1036.2.1 callerID . . . 1006

8.1036.2.2 callerIDLen . . . 1006
8.1036.2.3 numPlan . . . 1006
8.1036.2.4 numPresInd . . . 1007
8.1036.2.5 numType . . . 1007
8.1036.2.6 screeningInd . . . 1007
8.1037 voice_CUGInfo Struct Reference . . . 1007
8.1037.1 Detailed Description . . . 1007
8.1037.2 Field Documentation . . . 1007
8.1037.2.1 CUGIndex . . . 1007
8.1037.2.2 SuppOA . . . 1007
8.1037.2.3 SuppPrefCUG . . . 1007
8.1038 voice_curAMRConfig Struct Reference . . . 1007
8.1038.1 Detailed Description . . . 1007
8.1038.2 Field Documentation . . . 1008
8.1038.2.1 gsmAmrStat . . . 1008
8.1038.2.2 wcdmaAmrStat . . . 1008
8.1039 voice_diagInfo Struct Reference . . . 1008
8.1039.1 Detailed Description . . . 1008
8.1039.2 Field Documentation . . . 1008
8.1039.2.1 diagInfoLen . . . 1008
8.1039.2.2 diagnosticInfo . . . 1008
8.1040 voice_DTMFInfo Struct Reference . . . 1009
8.1040.1 Detailed Description . . . 1009
8.1040.2 Field Documentation . . . 1009
8.1040.2.1 callID . . . 1009
8.1040.2.2 digitBuff . . . 1009
8.1040.2.3 digitCnt . . . 1009
8.1040.2.4 DTMFEvent . . . 1009
8.1041 voice_DTMFLengths Struct Reference . . . 1009
8.1041.1 Detailed Description . . . 1010
8.1041.2 Field Documentation . . . 1010
8.1041.2.1 DTMFInterdigitInterval . . . 1010
8.1041.2.2 DTMFPulseWidth . . . 1010
8.1042 voice_ECTNum Struct Reference . . . 1010
8.1042.1 Detailed Description . . . 1010
8.1042.2 Field Documentation . . . 1011
8.1042.2.1 ECTCallState . . . 1011
8.1042.2.2 number . . . 1011
8.1042.2.3 presentationInd . . . 1011
8.1043 voice_extDispRecInfo Struct Reference . . . 1011

8.1043.1 Detailed Description . . . 1011
8.1043.2 Field Documentation . . . 1011
8.1043.2.1 dispType . . . 1011
8.1043.2.2 extDispInfo . . . 1011
8.1043.2.3 extDispInfoLen . . . 1012
8.1044 voice_getAllCallInformation Struct Reference . . . 1012
8.1044.1 Detailed Description . . . 1012
8.1044.2 Field Documentation . . . 1012
8.1044.2.1 ALS . . . 1012
8.1044.2.2 CallInfo . . . 1012
8.1044.2.3 isEmpty . . . 1012
8.1045 voice_getAllCallRmtPtyName Struct Reference . . . 1012
8.1045.1 Detailed Description . . . 1012
8.1045.2 Field Documentation . . . 1013
8.1045.2.1 callID . . . 1013
8.1045.2.2 RemotePartyName . . . 1013
8.1046 voice_getAllCallRmtPtyNum Struct Reference . . . 1013
8.1046.1 Detailed Description . . . 1013
8.1046.2 Field Documentation . . . 1013
8.1046.2.1 callID . . . 1013
8.1046.2.2 RemotePartyNum . . . 1013
8.1047 voice_getCallFWExtInfo Struct Reference . . . 1013
8.1047.1 Detailed Description . . . 1013
8.1047.2 Field Documentation . . . 1014
8.1047.2.1 CallFWExtInfo . . . 1014
8.1047.2.2 numInstances . . . 1014
8.1048 voice_getCallFWInfo Struct Reference . . . 1014
8.1048.1 Detailed Description . . . 1014
8.1048.2 Field Documentation . . . 1014
8.1048.2.1 CallFWInfo . . . 1014
8.1048.2.2 numInstances . . . 1014
8.1049 voice_lineCtrlInfo Struct Reference . . . 1014
8.1049.1 Detailed Description . . . 1015
8.1049.2 Field Documentation . . . 1015
8.1049.2.1 polarityIncluded . . . 1015
8.1049.2.2 pwrDenialTime . . . 1015
8.1049.2.3 revPolarity . . . 1015
8.1049.2.4 toggleMode . . . 1015
8.1050 voice_newPwdData Struct Reference . . . 1015
8.1050.1 Detailed Description . . . 1015

8.1050.2 Field Documentation . . . 1016
8.1050.2.1 newPwd . . . 1016
8.1050.2.2 newPwdAgain . . . 1016
8.1051 voice_NSSAudioCtrl Struct Reference . . . 1016
8.1051.1 Detailed Description . . . 1016
8.1051.2 Field Documentation . . . 1016
8.1051.2.1 downLink . . . 1016
8.1051.2.2 upLink . . . 1016
8.1052 voice_peerNumberInfo Struct Reference . . . 1016
8.1052.1 Detailed Description . . . 1016
8.1052.2 Field Documentation . . . 1017
8.1052.2.1 callID . . . 1018
8.1052.2.2 number . . . 1018
8.1052.2.3 numLen . . . 1018
8.1052.2.4 numPI . . . 1018
8.1052.2.5 numPlan . . . 1018
8.1052.2.6 numSI . . . 1018
8.1052.2.7 numType . . . 1018
8.1053 voice_prefVoiceSO Struct Reference . . . 1018
8.1053.1 Detailed Description . . . 1018
8.1053.2 Field Documentation . . . 1020
8.1053.2.1 evrcCapability . . . 1020
8.1053.2.2 homeOrigVoiceSO . . . 1020
8.1053.2.3 homePageVoiceSO . . . 1020
8.1053.2.4 namID . . . 1020
8.1053.2.5 roamOrigVoiceSO . . . 1020
8.1054 voice_redirNumInfo Struct Reference . . . 1020
8.1054.1 Detailed Description . . . 1020
8.1054.2 Field Documentation . . . 1021
8.1054.2.1 number . . . 1021
8.1054.2.2 numLen . . . 1021
8.1054.2.3 numPlan . . . 1021
8.1054.2.4 numType . . . 1021
8.1054.2.5 PI . . . 1021
8.1054.2.6 reason . . . 1021
8.1054.2.7 SI . . . 1021
8.1055 voice_remotePartyName Struct Reference . . . 1022
8.1055.1 Detailed Description . . . 1022
8.1055.2 Field Documentation . . . 1022
8.1055.2.1 callerName . . . 1022

8.1055.2.2 codingScheme . . . 1022
8.1055.2.3 nameLen . . . 1022
8.1055.2.4 namePI . . . 1022
8.1056 voice_remotePartyNum Struct Reference . . . 1022
8.1056.1 Detailed Description . . . 1023
8.1056.2 Field Documentation . . . 1023
8.1056.2.1 numLen . . . 1023
8.1056.2.2 presentationInd . . . 1023
8.1056.2.3 remPartyNumber . . . 1023
8.1057 voice_roamTimer Struct Reference . . . 1023
8.1057.1 Detailed Description . . . 1023
8.1057.2 Field Documentation . . . 1024
8.1057.2.1 namID . . . 1024
8.1057.2.2 roamTimerValue . . . 1024
8.1058 voice_signalInfo Struct Reference . . . 1024
8.1058.1 Detailed Description . . . 1024
8.1058.2 Field Documentation . . . 1024
8.1058.2.1 alertPitch . . . 1024
8.1058.2.2 signal . . . 1024
8.1058.2.3 signalType . . . 1024
8.1059 voice_SUPSInfo Struct Reference . . . 1025
8.1059.1 Detailed Description . . . 1025
8.1059.2 Field Documentation . . . 1025
8.1059.2.1 isModByCC . . . 1025
8.1059.2.2 svcType . . . 1025
8.1060 voice_USSDNotificationNetworkInfo Struct Reference . . . 1025
8.1060.1 Detailed Description . . . 1025
8.1060.2 Field Documentation . . . 1026
8.1060.2.1 networkInfo . . . 1026
8.1060.2.2 lvPresent . . . 1026
8.1061 voice_USSInfo Struct Reference . . . 1026
8.1061.1 Detailed Description . . . 1026
8.1061.2 Field Documentation . . . 1026
8.1061.2.1 ussData . . . 1026
8.1061.2.2 ussDCS . . . 1026
8.1061.2.3 ussLen . . . 1026
8.1062 voice_UUSInfo Struct Reference . . . 1026
8.1062.1 Detailed Description . . . 1027
8.1062.2 Field Documentation . . . 1027
8.1062.2.1 UUSData . . . 1027

8.1062.2.2 UUSDatalen . . . 1027
8.1062.2.3 UUSDcs . . . 1027
8.1062.2.4 UUSType . . . 1027
8.1063 wds_channelRate Struct Reference . . . 1028
8.1063.1 Detailed Description . . . 1028
8.1063.2 Field Documentation . . . 1028
8.1063.2.1 CurrChanRxRate . . . 1028
8.1063.2.2 CurrChanTxRate . . . 1028
8.1064 wds_ChannelRateTlv Struct Reference . . . 1028
8.1064.1 Detailed Description . . . 1028
8.1064.2 Field Documentation . . . 1028
8.1064.2.1 ChannelRate . . . 1028
8.1064.2.2 TlvPresent . . . 1028
8.1065 wds_ConnStatusTlv Struct Reference . . . 1029
8.1065.1 Detailed Description . . . 1029
8.1065.2 Field Documentation . . . 1029
8.1065.2.1 MDMConnStatus . . . 1029
8.1065.2.2 TlvPresent . . . 1029
8.1066 wds_currNetworkInfo Struct Reference . . . 1029
8.1066.1 Detailed Description . . . 1029
8.1066.2 Field Documentation . . . 1030
8.1066.2.1 NetworkType . . . 1030
8.1066.2.2 RATMask . . . 1030
8.1066.2.3 SOMask . . . 1030
8.1067 wds_DataBearTechTlv Struct Reference . . . 1030
8.1067.1 Detailed Description . . . 1030
8.1067.2 Field Documentation . . . 1031
8.1067.2.1 DataBearerTech . . . 1031
8.1067.2.2 TlvPresent . . . 1031
8.1068 wds_DataULongLongTlv Struct Reference . . . 1031
8.1068.1 Detailed Description . . . 1031
8.1068.2 Field Documentation . . . 1032
8.1068.2.1 TlvPresent . . . 1032
8.1068.2.2 ullData . . . 1032
8.1069 wds_DataULongTlv Struct Reference . . . 1032
8.1069.1 Detailed Description . . . 1032
8.1069.2 Field Documentation . . . 1032
8.1069.2.1 TlvPresent . . . 1032
8.1069.2.2 ulData . . . 1032
8.1070 wds_DHCPLeaseOptTlv Struct Reference . . . 1032

8.1070.1 Detailed Description . . . 1032
8.1070.2 Field Documentation . . . 1032
8.1070.2.1 numOpt . . . 1032
8.1070.2.2 optList . . . 1033
8.1070.2.3 optListData . . . 1033
8.1070.2.4 TlvPresent . . . 1033
8.1071 wds_DHCPLeaseStateTlv Struct Reference . . . 1033
8.1071.1 Detailed Description . . . 1033
8.1071.2 Field Documentation . . . 1033
8.1071.2.1 leaseState . . . 1033
8.1071.2.2 TlvPresent . . . 1033
8.1072 wds_DHCPOpt Struct Reference . . . 1033
8.1072.1 Detailed Description . . . 1033
8.1072.2 Field Documentation . . . 1034
8.1072.2.1 optCode . . . 1034
8.1072.2.2 optValLen . . . 1034
8.1072.2.3 pOptVal . . . 1034
8.1073 wds_DHCPProfileIdTlv Struct Reference . . . 1034
8.1073.1 Detailed Description . . . 1034
8.1073.2 Field Documentation . . . 1034
8.1073.2.1 profileId . . . 1034
8.1073.2.2 profileType . . . 1034
8.1073.2.3 TlvPresent . . . 1034
8.1074 wds_DHCPv4HWConfig Struct Reference . . . 1034
8.1074.1 Detailed Description . . . 1035
8.1074.2 Field Documentation . . . 1035
8.1074.2.1 chaddr . . . 1035
8.1074.2.2 chaddrLen . . . 1035
8.1074.2.3 hwType . . . 1035
8.1075 wds_DHCPv4Option Struct Reference . . . 1035
8.1075.1 Detailed Description . . . 1035
8.1075.2 Field Documentation . . . 1036
8.1075.2.1 optCode . . . 1036
8.1075.2.2 optVal . . . 1036
8.1075.2.3 optValLen . . . 1036
8.1076 wds_DHCPv4OptionList Struct Reference . . . 1036
8.1076.1 Detailed Description . . . 1036
8.1076.2 Field Documentation . . . 1036
8.1076.2.1 numOpt . . . 1036
8.1076.2.2 pOptList . . . 1036

8.1077 wds_DHCPv4ProfileId Struct Reference . . . 1036
8.1077.1 Detailed Description . . . 1036
8.1077.2 Field Documentation . . . 1037
8.1077.2.1 profileId . . . 1037
8.1077.2.2 profileType . . . 1037
8.1078 wds_Domain Struct Reference . . . 1037
8.1078.1 Detailed Description . . . 1037
8.1078.2 Field Documentation . . . 1037
8.1078.2.1 domainLen . . . 1037
8.1078.2.2 domainName . . . 1037
8.1079 wds_DomainNameList Struct Reference . . . 1037
8.1079.1 Detailed Description . . . 1037
8.1079.2 Field Documentation . . . 1038
8.1079.2.1 domain . . . 1038
8.1079.2.2 numInstances . . . 1038
8.1080 wds_DormStatTlv Struct Reference . . . 1038
8.1080.1 Detailed Description . . . 1038
8.1080.2 Field Documentation . . . 1038
8.1080.2.1 DormancyStat . . . 1038
8.1080.2.2 TlvPresent . . . 1038
8.1081 wds_GPRSQoS Struct Reference . . . 1038
8.1081.1 Detailed Description . . . 1039
8.1081.2 Field Documentation . . . 1039
8.1081.2.1 delayClass . . . 1039
8.1081.2.2 meanThroughputClass . . . 1039
8.1081.2.3 peakThroughputClass . . . 1039
8.1081.2.4 precedenceClass . . . 1039
8.1081.2.5 reliabilityClass . . . 1039
8.1082 wds_IPv4AdTlv Struct Reference . . . 1039
8.1082.1 Detailed Description . . . 1039
8.1082.2 Field Documentation . . . 1040
8.1082.2.1 IPv4Addr . . . 1040
8.1082.2.2 TlvPresent . . . 1040
8.1083 wds_IPV6AddressInfo Struct Reference . . . 1040
8.1083.1 Detailed Description . . . 1040
8.1083.2 Field Documentation . . . 1040
8.1083.2.1 IPAddressV6 . . . 1040
8.1083.2.2 IPV6PrefixLen . . . 1040
8.1084 wds_IPV6GWAddressInfo Struct Reference . . . 1040
8.1084.1 Detailed Description . . . 1040

8.1084.2 Field Documentation . . . 1041
8.1084.2.1 gwAddressV6 . . . 1041
8.1084.2.2 gwV6PrefixLen . . . 1041
8.1085 wds_LastMdmCallEndRsnTlv Struct Reference . . . 1041
8.1085.1 Detailed Description . . . 1041
8.1085.2 Field Documentation . . . 1041
8.1085.2.1 CallEndReason . . . 1041
8.1085.2.2 TlvPresent . . . 1041
8.1086 wds_PCSCFFQDNAddress Struct Reference . . . 1041
8.1086.1 Detailed Description . . . 1042
8.1086.2 Field Documentation . . . 1042
8.1086.2.1 fqdnAddr . . . 1042
8.1086.2.2 fqdnLen . . . 1042
8.1087 wds_PCSCFFQDNAddressList Struct Reference . . . 1042
8.1087.1 Detailed Description . . . 1042
8.1087.2 Field Documentation . . . 1042
8.1087.2.1 numInstances . . . 1042
8.1087.2.2 pcsfFQDNAddress . . . 1042
8.1088 wds_PCSCFIPv4ServerAddressList Struct Reference . . . 1042
8.1088.1 Detailed Description . . . 1043
8.1088.2 Field Documentation . . . 1043
8.1088.2.1 numInstances . . . 1043
8.1088.2.2 pcscfIPv4Addr . . . 1043
8.1089 wds_ProfileIdentifier Struct Reference . . . 1043
8.1089.1 Detailed Description . . . 1043
8.1089.2 Field Documentation . . . 1043
8.1089.2.1 profileIndex . . . 1044
8.1089.2.2 profileType . . . 1044
8.1090 wds_profileInfo Union Reference . . . 1044
8.1090.1 Detailed Description . . . 1044
8.1090.2 Field Documentation . . . 1044
8.1090.2.1 SlqsProfile3GPP . . . 1044
8.1090.2.2 SlqsProfile3GPP2 . . . 1044
8.1091 wds_RXBytesOKTlv Struct Reference . . . 1044
8.1091.1 Detailed Description . . . 1044
8.1091.2 Field Documentation . . . 1044
8.1091.2.1 RxByteOKCnt . . . 1044
8.1091.2.2 TlvPresent . . . 1044
8.1092 wds_transferStatInd Struct Reference . . . 1045
8.1092.1 Detailed Description . . . 1045

8.1092.2 Field Documentation . . . 1045
8.1092.2.1 StatsMask . . . 1045
8.1092.2.2 StatsPeriod . . . 1045
8.1093 wds_TrStatInd Struct Reference . . . 1045
8.1093.1 Detailed Description . . . 1045
8.1093.2 Field Documentation . . . 1046
8.1093.2.1 statsMask . . . 1046
8.1093.2.2 statsPeriod . . . 1046
8.1094 wds_TXBytesOKTlv Struct Reference . . . 1046
8.1094.1 Detailed Description . . . 1046
8.1094.2 Field Documentation . . . 1046
8.1094.2.1 TlvPresent . . . 1046
8.1094.2.2 TxByteOKCnt . . . 1047
8.1095 wds_UMTSMinQoS Struct Reference . . . 1047
8.1095.1 Detailed Description . . . 1047
8.1095.2 Field Documentation . . . 1048
8.1095.2.1 deliveryErrSDU . . . 1048
8.1095.2.2 grntDownlinkBitrate . . . 1049
8.1095.2.3 grntUplinkBitrate . . . 1049
8.1095.2.4 maxDownlinkBitrate . . . 1049
8.1095.2.5 maxSDUSize . . . 1049
8.1095.2.6 maxUplinkBitrate . . . 1049
8.1095.2.7 qosDeliveryOrder . . . 1049
8.1095.2.8 resBerRatio . . . 1049
8.1095.2.9 sduErrorRatio . . . 1049
8.1095.2.10 trafficClass . . . 1049
8.1095.2.11 trafficPriority . . . 1049
8.1095.2.12 transferDelay . . . 1049
8.1096 wdsDhcpv4HwConfig Struct Reference . . . 1049
8.1096.1 Detailed Description . . . 1049
8.1096.2 Field Documentation . . . 1049
8.1096.2.1 chaddr . . . 1049
8.1096.2.2 chaddrLen . . . 1049
8.1096.2.3 hwType . . . 1049
8.1097 wdsDhcpv4Option Struct Reference . . . 1050
8.1097.1 Detailed Description . . . 1050
8.1097.2 Field Documentation . . . 1050
8.1097.2.1 optCode . . . 1050
8.1097.2.2 optVal . . . 1050
8.1097.2.3 optValLen . . . 1050

8.1098 wdsDhcpv4OptionList Struct Reference . . . 1050
8.1098.1 Detailed Description . . . 1050
8.1098.2 Field Documentation . . . 1050
8.1098.2.1 numOpt . . . 1051
8.1098.2.2 pOptList . . . 1051
8.1099 wdsDhcpv4ProfileId Struct Reference . . . 1051
8.1099.1 Detailed Description . . . 1051
8.1099.2 Field Documentation . . . 1051
8.1099.2.1 profileId . . . 1051
8.1099.2.2 profileType . . . 1051

9 File Documentation **1053**

9.1 apdoxypages.c File Reference . . . 1053
9.1.1 Detailed Description . . . 1053
9.2 audio.h File Reference . . . 1053
9.2.1 Function Documentation . . . 1054
9.2.1.1 pack_audio_SLQSGetAudioPathConfig . . . 1054
9.2.1.2 pack_audio_SLQSGetAudioProfile . . . 1054
9.2.1.3 pack_audio_SLQSGetAudioVolTLBConfig . . . 1055
9.2.1.4 pack_audio_SLQSSetAudioPathConfig . . . 1055
9.2.1.5 pack_audio_SLQSSetAudioProfile . . . 1056
9.2.1.6 pack_audio_SLQSSetAudioVolTLBConfig . . . 1056
9.2.1.7 unpack_audio_SLQSGetAudioPathConfig . . . 1057
9.2.1.8 unpack_audio_SLQSGetAudioProfile . . . 1057
9.2.1.9 unpack_audio_SLQSGetAudioVolTLBConfig . . . 1057
9.2.1.10 unpack_audio_SLQSSetAudioPathConfig . . . 1058
9.2.1.11 unpack_audio_SLQSSetAudioProfile . . . 1058
9.2.1.12 unpack_audio_SLQSSetAudioVolTLBConfig . . . 1058
9.3 cat.h File Reference . . . 1059
9.3.1 Macro Definition Documentation . . . 1060
9.3.1.1 CAN_COMMON_EVENT_TLV_NUMBER . . . 1060
9.3.1.2 CAT_EVENT_DATA_MAX_LENGTH . . . 1060
9.3.2 Typedef Documentation . . . 1060
9.3.2.1 unpack_cat_CATSendEnvelopeCommand_t . . . 1060
9.3.2.2 unpack_cat_CATSendTerminalResponse_t . . . 1060
9.3.3 Function Documentation . . . 1060
9.3.3.1 pack_cat_CATSendEnvelopeCommand . . . 1060
9.3.3.2 pack_cat_CATSendTerminalResponse . . . 1060
9.3.3.3 pack_cat_SetCATEventCallback . . . 1061
9.3.3.4 unpack_cat_CATSendEnvelopeCommand . . . 1061

9.3.3.5 unpack_cat_CATSendTerminalResponse . . . 1061
9.3.3.6 unpack_cat_SetCATEventCallback . . . 1062
9.3.3.7 unpack_cat_SetCatEventCallback_ind . . . 1062
9.4 common.h File Reference . . . 1062
9.4.1 Macro Definition Documentation . . . 1064
9.4.1.1 DEAULT_LOC_TIMEOUT_IN_SEC . . . 1064
9.4.1.2 MINREQBKLEN . . . 1064
9.4.1.3 MSGID_AND_LEN . . . 1064
9.4.1.4 MSGID_DONT_CARE . . . 1064
9.4.1.5 SDK_VALIDATE_INPUT_PACK_PARAM . . . 1064
9.4.1.6 SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID . . . 1065
9.4.1.7 SDK_VALIDATE_INPUT_UNPACK_PARAM . . . 1065
9.4.1.8 SDU_HDR_LEN . . . 1065
9.4.1.9 SWI_INIT_UNPACK_RESULT_VALUE . . . 1065
9.4.1.10 UNUSEDPARAM . . . 1065
9.4.2 Typedef Documentation . . . 1065
9.4.2.1 logger . . . 1065
9.4.3 Enumeration Type Documentation . . . 1065
9.4.3.1 eLOG_LEVEL . . . 1065
9.4.3.2 eQMI_SVC . . . 1065
9.4.3.3 eTimeout . . . 1066
9.4.3.4 msgtype . . . 1066
9.4.4 Function Documentation . . . 1067
9.4.4.1 fill_pack_ctx . . . 1067
9.4.4.2 fill_sdu_hdr . . . 1067
9.4.4.3 get_version . . . 1067
9.4.4.4 helper_get_error_code . . . 1067
9.4.4.5 helper_get_error_reason . . . 1067
9.4.4.6 helper_get_req_str . . . 1067
9.4.4.7 helper_get_resp_ctx . . . 1067
9.4.4.8 helper_get_xid . . . 1067
9.4.4.9 helper_isBootLoader_DebugEnabled . . . 1067
9.4.4.10 helper_set_log_func . . . 1068
9.4.4.11 helper_set_log_lvl . . . 1068
9.4.4.12 liteqmi_GetVersion . . . 1068
9.4.4.13 liteqmi_helper_decode7bitAsciiEncString . . . 1068
9.4.4.14 liteqmi_log . . . 1068
9.4.4.15 unpack_result_code_only . . . 1068
9.4.5 Variable Documentation . . . 1068
9.4.5.1 glog . . . 1068

9.4.5.2 gloglvl . . . 1068
9.5 dms.h File Reference . . . 1068
9.5.1 Macro Definition Documentation . . . 1076
9.5.1.1 ACT_CODE_MAX_SIZE . . . 1076
9.5.1.2 CK_MAX_SIZE . . . 1076
9.5.1.3 DMS_IMGDETAILS_LEN . . . 1076
9.5.1.4 DMS_MAX_CUST_ID_LEN . . . 1076
9.5.1.5 DMS_MAX_CUST_VALUE_LEN . . . 1076
9.5.1.6 DMS_MAX_FWUPDATE_LOG_STR_SZ . . . 1076
9.5.1.7 DMS_MAX_FWUPDATE_REF_STR_SZ . . . 1076
9.5.1.8 DMS_MAX_RADIO_IFCS_SIZE . . . 1076
9.5.1.9 DMS_PM_FACTORY . . . 1076
9.5.1.10 DMS_PM_LOW . . . 1076
9.5.1.11 DMS_PM_OFFLINE . . . 1076
9.5.1.12 DMS_PM_ONLINE . . . 1076
9.5.1.13 DMS_PM_PERSISTENT_LOW . . . 1076
9.5.1.14 DMS_PM_RESET . . . 1076
9.5.1.15 DMS_PM_SHUT_DOWN . . . 1076
9.5.1.16 DMS_SET_REG_IND_DISABLE . . . 1076
9.5.1.17 DMS_SET_REG_IND_ENABLE . . . 1076
9.5.1.18 DMS_SET_REG_IND_NO_CHANGE . . . 1076
9.5.1.19 DMS_SET_REPORT_DISABLE . . . 1076
9.5.1.20 DMS_SET_REPORT_ENABLE . . . 1076
9.5.1.21 DMS_SLQSFWINFO_APPVERSION_SZ . . . 1077
9.5.1.22 DMS_SLQSFWINFO_BOOTVERSION_SZ . . . 1077
9.5.1.23 DMS_SLQSFWINFO_CARRIER_SZ . . . 1077
9.5.1.24 DMS_SLQSFWINFO_CUR_CARR_NAME . . . 1077
9.5.1.25 DMS_SLQSFWINFO_CUR_CARR_REV . . . 1077
9.5.1.26 DMS_SLQSFWINFO_MODELID_SZ . . . 1077
9.5.1.27 DMS_SLQSFWINFO_PACKAGEID_SZ . . . 1077
9.5.1.28 DMS_SLQSFWINFO_PRIVERSION_SZ . . . 1077
9.5.1.29 DMS_SLQSFWINFO_SKU_SZ . . . 1077
9.5.1.30 DMS_SWI_SET_IND_DISABLE . . . 1077
9.5.1.31 DMS_SWI_SET_IND_ENABLE . . . 1077
9.5.1.32 DMS_UINT8_MAX_STRING_SZ . . . 1077
9.5.1.33 DMS_VALID_FSN_LEN . . . 1077
9.5.1.34 ERI_DATA_MAX_SIZE . . . 1077
9.5.1.35 MAX_BUILD_ID_LEN . . . 1077
9.5.1.36 MEID_MAX_SIZE . . . 1077
9.5.1.37 SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH . . . 1077

9.5.1.38 SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH . . . 1077
9.5.1.39 SPC_SIZE . . . 1077
9.5.1.40 UNIQUE_ID_LEN . . . 1077
9.5.2 Function Documentation . . . 1077
9.5.2.1 pack_dms_ActivateAutomatic . . . 1077
9.5.2.2 pack_dms_GetActivationState . . . 1078
9.5.2.3 pack_dms_GetBandCapability . . . 1078
9.5.2.4 pack_dms_GetCrashAction . . . 1078
9.5.2.5 pack_dms_GetCustFeature . . . 1079
9.5.2.6 pack_dms_GetCustFeaturesV2 . . . 1079
9.5.2.7 pack_dms_GetDeviceCap . . . 1080
9.5.2.8 pack_dms_GetDeviceCapabilities . . . 1080
9.5.2.9 pack_dms_GetDeviceHardwareRev . . . 1080
9.5.2.10 pack_dms_GetDeviceMfr . . . 1081
9.5.2.11 pack_dms_GetDeviceSerialNumbers . . . 1081
9.5.2.12 pack_dms_GetFirmwareInfo . . . 1081
9.5.2.13 pack_dms_GetFirmwareRevision . . . 1082
9.5.2.14 pack_dms_GetFirmwareRevisions . . . 1082
9.5.2.15 pack_dms_GetFSN . . . 1082
9.5.2.16 pack_dms_GetHardwareRevision . . . 1083
9.5.2.17 pack_dms_GetIMSI . . . 1083
9.5.2.18 pack_dms_GetManufacturer . . . 1083
9.5.2.19 pack_dms_GetModelID . . . 1084
9.5.2.20 pack_dms_GetNetworkTime . . . 1084
9.5.2.21 pack_dms_GetOfflineReason . . . 1084
9.5.2.22 pack_dms_GetPower . . . 1085
9.5.2.23 pack_dms_GetPRLVersion . . . 1085
9.5.2.24 pack_dms_GetSerialNumbers . . . 1085
9.5.2.25 pack_dms_GetUSBComp . . . 1086
9.5.2.26 pack_dms_GetVoiceNumber . . . 1086
9.5.2.27 pack_dms_ResetToFactoryDefaults . . . 1087
9.5.2.28 pack_dms_SetActivationStatusCallback . . . 1087
9.5.2.29 pack_dms_SetCrashAction . . . 1087
9.5.2.30 pack_dms_SetCustFeature . . . 1088
9.5.2.31 pack_dms_SetCustFeaturesV2 . . . 1088
9.5.2.32 pack_dms_SetEventReport . . . 1088
9.5.2.33 pack_dms_SetFirmwarePreference . . . 1089
9.5.2.34 pack_dms_SetIndicationRegister . . . 1089
9.5.2.35 pack_dms_SetPower . . . 1090
9.5.2.36 pack_dms_SetUSBComp . . . 1090

9.5.2.37 pack_dms_SLQSDmsSwiGetPCInfo . . . 1090
9.5.2.38 pack_dms_SLQSDmsSwiGetResetInfo . . . 1091
9.5.2.39 pack_dms_SLQSDmsSwiGetUimSelection . . . 1091
9.5.2.40 pack_dms_SLQSDmsSwiIndicationRegister . . . 1091
9.5.2.41 pack_dms_SLQSGetBandCapability . . . 1092
9.5.2.42 pack_dms_SLQSGetERIFile . . . 1092
9.5.2.43 pack_dms_SLQSGetPowerSaveModeConfig . . . 1093
9.5.2.44 pack_dms_SLQSSetPowerSaveModeConfig . . . 1093
9.5.2.45 pack_dms_SLQSSwiClearDyingGaspStatistics . . . 1093
9.5.2.46 pack_dms_SLQSSwiGetCrashInfo . . . 1094
9.5.2.47 pack_dms_SLQSSwiGetDyingGaspCfg . . . 1094
9.5.2.48 pack_dms_SLQSSwiGetDyingGaspStatistics . . . 1094
9.5.2.49 pack_dms_SLQSSwiGetFirmwareCurr . . . 1095
9.5.2.50 pack_dms_SLQSSwiGetFwUpdateStatus . . . 1095
9.5.2.51 pack_dms_SLQSSwiGetHostDevInfo . . . 1095
9.5.2.52 pack_dms_SLQSSwiGetOSInfo . . . 1096
9.5.2.53 pack_dms_SLQSSwiGetSerialNoExt . . . 1096
9.5.2.54 pack_dms_SLQSSwiSetDyingGaspCfg . . . 1096
9.5.2.55 pack_dms_SLQSSwiSetHostDevInfo . . . 1097
9.5.2.56 pack_dms_SLQSSwiSetOSInfo . . . 1097
9.5.2.57 pack_dms_SLQSUIMGetState . . . 1097
9.5.2.58 pack_dms_SwiSetEventReport . . . 1098
9.5.2.59 pack_dms_SwiUimSelect . . . 1098
9.5.2.60 pack_dms_UIMChangePIN . . . 1099
9.5.2.61 pack_dms_UIMGetControlKeyStatus . . . 1099
9.5.2.62 pack_dms_UIMGetICCID . . . 1099
9.5.2.63 pack_dms_UIMGetPINStatus . . . 1100
9.5.2.64 pack_dms_UIMSetControlKeyProtection . . . 1100
9.5.2.65 pack_dms_UIMSetPINProtection . . . 1100
9.5.2.66 pack_dms_UIMUnblockControlKey . . . 1101
9.5.2.67 pack_dms_UIMUnblockPIN . . . 1101
9.5.2.68 pack_dms_UIMVerifyPIN . . . 1102
9.5.2.69 pack_dms_ValidateSPC . . . 1102
9.5.2.70 unpack_dms_ActivateAutomatic . . . 1102
9.5.2.71 unpack_dms_GetActivationState . . . 1103
9.5.2.72 unpack_dms_GetBandCapability . . . 1103
9.5.2.73 unpack_dms_GetCrashAction . . . 1103
9.5.2.74 unpack_dms_GetCustFeature . . . 1104
9.5.2.75 unpack_dms_GetCustFeaturesV2 . . . 1104
9.5.2.76 unpack_dms_GetDeviceCap . . . 1104

9.5.2.77 unpack_dms_GetDeviceCapabilities . . . 1105
9.5.2.78 unpack_dms_GetDeviceHardwareRev . . . 1105
9.5.2.79 unpack_dms_GetDeviceMfr . . . 1105
9.5.2.80 unpack_dms_GetDeviceSerialNumbers . . . 1106
9.5.2.81 unpack_dms_GetFirmwareInfo . . . 1106
9.5.2.82 unpack_dms_GetFirmwareRevision . . . 1106
9.5.2.83 unpack_dms_GetFirmwareRevisions . . . 1107
9.5.2.84 unpack_dms_GetFSN . . . 1107
9.5.2.85 unpack_dms_GetHardwareRevision . . . 1107
9.5.2.86 unpack_dms_GetIMSI . . . 1108
9.5.2.87 unpack_dms_GetManufacturer . . . 1108
9.5.2.88 unpack_dms_GetModelID . . . 1108
9.5.2.89 unpack_dms_GetNetworkTime . . . 1109
9.5.2.90 unpack_dms_GetOfflineReason . . . 1109
9.5.2.91 unpack_dms_GetPower . . . 1109
9.5.2.92 unpack_dms_GetPRLVersion . . . 1110
9.5.2.93 unpack_dms_GetSerialNumbers . . . 1110
9.5.2.94 unpack_dms_GetUSBComp . . . 1110
9.5.2.95 unpack_dms_GetVoiceNumber . . . 1111
9.5.2.96 unpack_dms_PSMCfgChange_ind . . . 1111
9.5.2.97 unpack_dms_ResetToFactoryDefaults . . . 1111
9.5.2.98 unpack_dms_SetActivationStatusCallback . . . 1112
9.5.2.99 unpack_dms_SetCrashAction . . . 1112
9.5.2.100 unpack_dms_SetCustFeature . . . 1112
9.5.2.101 unpack_dms_SetCustFeaturesV2 . . . 1113
9.5.2.102 unpack_dms_SetEventReport . . . 1113
9.5.2.103 unpack_dms_SetEventReport_ind . . . 1114
9.5.2.104 unpack_dms_SetFirmwarePreference . . . 1114
9.5.2.105 unpack_dms_SetIndicationRegister . . . 1114
9.5.2.106 unpack_dms_SetPower . . . 1115
9.5.2.107 unpack_dms_SetUSBComp . . . 1115
9.5.2.108 unpack_dms_SLQSDmsSwiGetPCInfo . . . 1115
9.5.2.109 unpack_dms_SLQSDmsSwiGetResetInfo . . . 1116
9.5.2.110 unpack_dms_SLQSDmsSwiGetResetInfo_Ind . . . 1116
9.5.2.111 unpack_dms_SLQSDmsSwiGetUimSelection . . . 1116
9.5.2.112 unpack_dms_SLQSDmsSwiIndicationRegister . . . 1117
9.5.2.113 unpack_dms_SLQSGetBandCapability . . . 1117
9.5.2.114 unpack_dms_SLQSGetERIFile . . . 1118
9.5.2.115 unpack_dms_SLQSGetPowerSaveModeConfig . . . 1118
9.5.2.116 unpack_dms_SLQSSetPowerSaveModeConfig . . . 1118

9.5.2.117 unpack_dms_SLQSSwiClearDyingGaspStatistics . . . 1119
9.5.2.118 unpack_dms_SLQSSwiGetCrashInfo . . . 1119
9.5.2.119 unpack_dms_SLQSSwiGetDyingGaspCfg . . . 1119
9.5.2.120 unpack_dms_SLQSSwiGetDyingGaspStatistics . . . 1120
9.5.2.121 unpack_dms_SLQSSwiGetFirmwareCurr . . . 1120
9.5.2.122 unpack_dms_SLQSSwiGetFwUpdateStatus . . . 1120
9.5.2.123 unpack_dms_SLQSSwiGetHostDevInfo . . . 1121
9.5.2.124 unpack_dms_SLQSSwiGetOSInfo . . . 1121
9.5.2.125 unpack_dms_SLQSSwiGetSerialNoExt . . . 1121
9.5.2.126 unpack_dms_SLQSSwiSetDyingGaspCfg . . . 1122
9.5.2.127 unpack_dms_SLQSSwiSetHostDevInfo . . . 1122
9.5.2.128 unpack_dms_SLQSSwiSetOSInfo . . . 1122
9.5.2.129 unpack_dms_SLQSUIMGetState . . . 1123
9.5.2.130 unpack_dms_SwiEventReportCallBack_ind . . . 1123
9.5.2.131 unpack_dms_SwiSetEventReport . . . 1123
9.5.2.132 unpack_dms_SwiUimSelect . . . 1124
9.5.2.133 unpack_dms_UIMChangePIN . . . 1124
9.5.2.134 unpack_dms_UIMGetControlKeyStatus . . . 1124
9.5.2.135 unpack_dms_UIMGetICCID . . . 1125
9.5.2.136 unpack_dms_UIMGetPINStatus . . . 1125
9.5.2.137 unpack_dms_UIMSetControlKeyProtection . . . 1125
9.5.2.138 unpack_dms_UIMSetPINProtection . . . 1126
9.5.2.139 unpack_dms_UIMUnblockControlKey . . . 1126
9.5.2.140 unpack_dms_UIMUnblockPIN . . . 1127
9.5.2.141 unpack_dms_UIMVerifyPIN . . . 1127
9.5.2.142 unpack_dms_ValidateSPC . . . 1127
9.6 fms.h File Reference . . . 1128
9.6.1 Macro Definition Documentation . . . 1129
9.6.1.1 FMS_FW_PRI_BUILD_MATCH_LEN . . . 1129
9.6.1.2 FMS_GOBI_LISTENTRIES_MAX . . . 1129
9.6.1.3 FMS_GOBI_MBN_BUILD_ID_STR_LEN . . . 1129
9.6.1.4 FMS_GOBI_MBN_IMG_ID_SIZE . . . 1129
9.6.1.5 FMS_GOBI_MBN_IMG_ID_STR_LEN . . . 1129
9.6.1.6 FMS_IMAGE_ID_BUILD_ID_LEN . . . 1129
9.6.1.7 FMS_IMAGE_ID_IMG_ID_LEN . . . 1129
9.6.1.8 FMS_IMAGE_ID_MAX_ENTRIES . . . 1129
9.6.1.9 FMS_IMAGE_ID_PRI_IMGTYPE . . . 1129
9.6.1.10 FMS_MAX_IMAGE_ID_ELEMENT . . . 1129
9.6.1.11 FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE . . . 1129
9.6.1.12 LITE_TOTAL_IMAGE_ID_STRING_SIZE . . . 1129

9.6.2 Function Documentation . . . 1129
9.6.2.1 GetValidFwPriCombinations . . . 1129
9.6.2.2 pack_fms_GetImagesPreference . . . 1130
9.6.2.3 pack_fms_GetStoredImages . . . 1130
9.6.2.4 pack_fms_SetImagesPreference . . . 1130
9.6.2.5 unpack_fms_GetImagesPreference . . . 1130
9.6.2.6 unpack_fms_GetStoredImages . . . 1130
9.6.2.7 unpack_fms_SetImagesPreference . . . 1131
9.7 ims.h File Reference . . . 1131
9.7.1 Macro Definition Documentation . . . 1133
9.7.1.1 MAX_NAME_LEN . . . 1133
9.7.2 Typedef Documentation . . . 1133
9.7.2.1 unpack_ims_SLQSImsConfigIndicationRegister_t . . . 1133
9.7.3 Function Documentation . . . 1133
9.7.3.1 pack_ims_SLQSGetIMSSMSConfig . . . 1133
9.7.3.2 pack_ims_SLQSGetIMSUserConfig . . . 1133
9.7.3.3 pack_ims_SLQSGetIMSVoIPConfig . . . 1134
9.7.3.4 pack_ims_SLQSGetRegMgrConfig . . . 1134
9.7.3.5 pack_ims_SLQSGetSIPConfig . . . 1135
9.7.3.6 pack_ims_SLQSImsConfigIndicationRegister . . . 1135
9.7.3.7 pack_ims_SLQSSetIMSSMSConfig . . . 1136
9.7.3.8 pack_ims_SLQSSetIMSUserConfig . . . 1136
9.7.3.9 pack_ims_SLQSSetIMSVoIPConfig . . . 1137
9.7.3.10 pack_ims_SLQSSetRegMgrConfig . . . 1137
9.7.3.11 pack_ims_SLQSSetSIPConfig . . . 1138
9.7.3.12 unpack_ims_SLQSGetIMSSMSConfig . . . 1138
9.7.3.13 unpack_ims_SLQSGetIMSUserConfig . . . 1138
9.7.3.14 unpack_ims_SLQSGetIMSVoIPConfig . . . 1139
9.7.3.15 unpack_ims_SLQSGetRegMgrConfig . . . 1139
9.7.3.16 unpack_ims_SLQSGetSIPConfig . . . 1139
9.7.3.17 unpack_ims_SLQSImsConfigIndicationRegister . . . 1140
9.7.3.18 unpack_ims_SLQSRegMgrCfgCallBack_ind . . . 1140
9.7.3.19 unpack_ims_SLQSSetIMSSMSConfig . . . 1140
9.7.3.20 unpack_ims_SLQSSetIMSUserConfig . . . 1141
9.7.3.21 unpack_ims_SLQSSetIMSVoIPConfig . . . 1141
9.7.3.22 unpack_ims_SLQSSetRegMgrConfig . . . 1141
9.7.3.23 unpack_ims_SLQSSetSIPConfig . . . 1142
9.7.3.24 unpack_ims_SLQSSIPCfgCallBack_ind . . . 1142
9.7.3.25 unpack_ims_SLQSSMSCfgCallBack_ind . . . 1142
9.7.3.26 unpack_ims_SLQSUserCfgCallBack_ind . . . 1143

9.7.3.27 unpack_ims_SLQSVoIPCfgCallBack_ind . . . 1143
9.8 imsa.h File Reference . . . 1143
9.8.1 Macro Definition Documentation . . . 1145
9.8.1.1 MAX_ERROR_CODE_LEN . . . 1145
9.8.2 Typedef Documentation . . . 1145
9.8.2.1 unpack_imsa_SLQSRegisterIMSAIndication_t . . . 1145
9.8.3 Function Documentation . . . 1145
9.8.3.1 pack_imsa_SLQSGetIMSARegStatus . . . 1145
9.8.3.2 pack_imsa_SLQSGetIMSAServiceStatus . . . 1145
9.8.3.3 pack_imsa_SLQSRegisterIMSAIndication . . . 1146
9.8.3.4 unpack_imsa_SLQSGetIMSARegStatus . . . 1146
9.8.3.5 unpack_imsa_SLQSGetIMSAServiceStatus . . . 1147
9.8.3.6 unpack_imsa_SLQSImsaPdpStatusCallBack_ind . . . 1147
9.8.3.7 unpack_imsa_SLQSImsaRatStatusCallBack_ind . . . 1148
9.8.3.8 unpack_imsa_SLQSImsaRegStatusCallBack_ind . . . 1148
9.8.3.9 unpack_imsa_SLQSImsaSvcStatusCallBack_ind . . . 1148
9.8.3.10 unpack_imsa_SLQSRegisterIMSAIndication . . . 1149
9.9 lite-fw.h File Reference . . . 1149
9.9.1 Detailed Description . . . 1151
9.9.2 Macro Definition Documentation . . . 1151
9.9.2.1 FIRMWARE_INFO_STRING_SIZE . . . 1151
9.9.2.2 IMG_MASK_CLEAR . . . 1151
9.9.2.3 IMG_MASK_GENERIC . . . 1151
9.9.2.4 IMG_MASK_MDM . . . 1151
9.9.2.5 IMG_MASK_PRI . . . 1151
9.9.2.6 libSDP_BuildImagesPreferenceRequest . . . 1151
9.9.2.7 libSDP_CalculateImageMask . . . 1151
9.9.2.8 LIBSDP_CARRIER_PACKAGE_SKU . . . 1151
9.9.2.9 libSDP_CheckValidFirmwareInfo . . . 1152
9.9.2.10 libSDP_DownloadFW . . . 1152
9.9.2.11 libSDP_ExtractFirmwareParametersByPath . . . 1152
9.9.2.12 libSDP_getFileType . . . 1152
9.9.2.13 libSDP_GetModelFamily . . . 1152
9.9.2.14 libSDP_GetVersion . . . 1152
9.9.2.15 libsdp_set_log_func . . . 1152
9.9.2.16 libsdp_SetReadBlockSize . . . 1152
9.9.2.17 LIBSDP_SKU_STRING_LENGTH . . . 1152
9.9.2.18 LITEFW_CARRIER_PACKAGE_SKU . . . 1152
9.9.2.19 LITEFW_SKU_STRING_LENGTH . . . 1152
9.9.2.20 NULL_TERMINATOR_CHAR_SIZE . . . 1152

9.9.3 Typedef Documentation . . . 1152
9.9.3.1 libSDP_FirmwareInfo . . . 1152
9.9.3.2 libsdplogger . . . 1152
9.9.3.3 litefw_FirmwareInfo . . . 1152
9.9.3.4 litefw_FirmwarePartNo . . . 1153
9.9.3.5 litefwlogger . . . 1153
9.9.4 Enumeration Type Documentation . . . 1153
9.9.4.1 litefw_Fw_Type . . . 1153
9.9.4.2 litefw_fwdwl_error_codes . . . 1153
9.9.4.3 litefw_Models . . . 1154
9.9.4.4 litefw_QDL_FLOW_CONTROLs . . . 1154
9.9.4.5 litefw_QDL_MODEs . . . 1154
9.9.5 Function Documentation . . . 1154
9.9.5.1 litefw_BuildImagesPreferenceRequest . . . 1154
9.9.5.2 litefw_CalculateImageMask . . . 1155
9.9.5.3 litefw_CheckValidFirmwareInfo . . . 1155
9.9.5.4 litefw_DownloadFW . . . 1155
9.9.5.5 litefw_ExtractFirmwareParametersByPath . . . 1156
9.9.5.6 litefw_ExtractFirmwarePartNoByPath . . . 1156
9.9.5.7 litefw_getFileType . . . 1157
9.9.5.8 litefw_GetModelFamily . . . 1157
9.9.5.9 litefw_GetQTLDownloadMode . . . 1157
9.9.5.10 litefw_GetQTLHWFlowControl . . . 1158
9.9.5.11 litefw_GetVersion . . . 1158
9.9.5.12 litefw_logsenable . . . 1158
9.9.5.13 litefw_set_log_func . . . 1158
9.9.5.14 litefw_SetQTLDownloadMode . . . 1158
9.9.5.15 litefw_SetQTLHWFlowControl . . . 1159
9.9.5.16 litefw_SetReadBlockSize . . . 1159
9.9.5.17 litefw_switch_9x07_to_downloadmode . . . 1159
9.9.5.18 litefw_switch_to_BootHoldMode . . . 1159
9.10 loc.h File Reference . . . 1160
9.10.1 Macro Definition Documentation . . . 1163
9.10.1.1 LOC_UINT8_MAX_STRING_SZ . . . 1163
9.10.1.2 LOCEVENTMASKBATCHFULLNOTIFICATION . . . 1163
9.10.1.3 LOCEVENTMASKENGINESTATE . . . 1163
9.10.1.4 LOCEVENTMASKFIXSESSIONSTATE . . . 1164
9.10.1.5 LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION . . . 1164
9.10.1.6 LOCEVENTMASKGEOFENCEBREACHNOTIFICATION . . . 1164
9.10.1.7 LOCEVENTMASKGEOFENCEGENALERT . . . 1164

9.10.1.8 LOCEVENTMASKGNSSMEASUREMENTREPORT . . . 1164
9.10.1.9 LOCEVENTMASKGNSSSVINFO . . . 1164
9.10.1.10 LOCEVENTMASKINJECTPOSITIONREQ . . . 1164
9.10.1.11 LOCEVENTMASKINJECTPREDICTEDORBITSREQ . . . 1164
9.10.1.12 LOCEVENTMASKINJECTTIMEREQ . . . 1164
9.10.1.13 LOCEVENTMASKINJECTWIFIAPDATAREQ . . . 1164
9.10.1.14 LOCEVENTMASKINVALIDVALUE . . . 1164
9.10.1.15 LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT . . . 1165
9.10.1.16 LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ . . . 1165
9.10.1.17 LOCEVENTMASKMOTIONDATACONTROL . . . 1165
9.10.1.18 LOCEVENTMASKNIGEOFENCENOTIFICATION . . . 1165
9.10.1.19 LOCEVENTMASKNINOTIFYVERIFYREQ . . . 1165
9.10.1.20 LOCEVENTMASKNMEA . . . 1165
9.10.1.21 LOCEVENTMASKPEDOMETERCONTROL . . . 1165
9.10.1.22 LOCEVENTMASKPOSITIONREPORT . . . 1165
9.10.1.23 LOCEVENTMASKSENSORSTREAMINGREADYSTATUS . . . 1165
9.10.1.24 LOCEVENTMASKSETSPISTREAMINGREPORT . . . 1165
9.10.1.25 LOCEVENTMASKTIMESYNCREQ . . . 1166
9.10.1.26 LOCEVENTMASKVEHICLEDATAREADYSTATUS . . . 1166
9.10.1.27 LOCEVENTMASKWIFIREQ . . . 1166
9.10.1.28 MAX_LOC_NMEA_STR_LEN . . . 1166
9.10.1.29 MAX_SENSOR_DATA_LEN . . . 1166
9.10.1.30 MAX_TEMP_DATA_LEN . . . 1166
9.10.2 Typedef Documentation . . . 1166
9.10.2.1 unpack_loc_DeleteAssistData_t . . . 1166
9.10.2.2 unpack_loc_SLQSLOCGetServer_t . . . 1166
9.10.2.3 unpack_loc_SLQSLOCInjectPosition_t . . . 1166
9.10.2.4 unpack_loc_SLQSLOCInjectSensorData_t . . . 1166
9.10.2.5 unpack_loc_SLQSLOCInjectUTCTime_t . . . 1166
9.10.2.6 unpack_loc_SLQSLOCSetCradleMountConfig_t . . . 1166
9.10.3 Enumeration Type Documentation . . . 1166
9.10.3.1 anonymous enum . . . 1166
9.10.4 Function Documentation . . . 1166
9.10.4.1 pack_loc_DeleteAssistData . . . 1166
9.10.4.2 pack_loc_EventRegister . . . 1167
9.10.4.3 pack_loc_SetExtPowerState . . . 1167
9.10.4.4 pack_loc_SetOperationMode . . . 1168
9.10.4.5 pack_loc_SLQSLOCGetBestAvailPos . . . 1168
9.10.4.6 pack_loc_SLQSLOCGetServer . . . 1168
9.10.4.7 pack_loc_SLQSLOCInjectPosition . . . 1169

9.10.4.8 pack_loc_SLQSLOCInjectSensorData . . . 1169
9.10.4.9 pack_loc_SLQSLOCInjectUTCTime . . . 1170
9.10.4.10 pack_loc_SLQSLOCSetCradleMountConfig . . . 1170
9.10.4.11 pack_loc_SLQSLOCSetServer . . . 1170
9.10.4.12 pack_loc_Start . . . 1171
9.10.4.13 pack_loc_Stop . . . 1171
9.10.4.14 unpack_loc_BestAvailPos_Ind . . . 1172
9.10.4.15 unpack_loc_CradleMountCallback_Ind . . . 1172
9.10.4.16 unpack_loc_DeleteAssistData . . . 1172
9.10.4.17 unpack_loc_DeleteAssistData_Ind . . . 1173
9.10.4.18 unpack_loc_EngineState_Ind . . . 1173
9.10.4.19 unpack_loc_EventNMEA_Ind . . . 1173
9.10.4.20 unpack_loc_EventRegister . . . 1174
9.10.4.21 unpack_loc_EventTimeSyncCallback_Ind . . . 1174
9.10.4.22 unpack_loc_GetServer_Ind . . . 1174
9.10.4.23 unpack_loc_GnssSvInfo_Ind . . . 1175
9.10.4.24 unpack_loc_InjectPositionCallback_Ind . . . 1175
9.10.4.25 unpack_loc_InjectSensorDataCallback_Ind . . . 1175
9.10.4.26 unpack_loc_InjectTimeSyncDataCallback_Ind . . . 1176
9.10.4.27 unpack_loc_InjectUTCTimeCallback_Ind . . . 1176
9.10.4.28 unpack_loc_PositionRpt_Ind . . . 1176
9.10.4.29 unpack_loc_SensorStreamingCallback_Ind . . . 1177
9.10.4.30 unpack_loc_SetExtPowerConfig_Ind . . . 1177
9.10.4.31 unpack_loc_SetExtPowerState . . . 1177
9.10.4.32 unpack_loc_SetOperationMode . . . 1178
9.10.4.33 unpack_loc_SetOperationMode_Ind . . . 1178
9.10.4.34 unpack_loc_SetServer_Ind . . . 1178
9.10.4.35 unpack_loc_SLQSLOCGetBestAvailPos . . . 1179
9.10.4.36 unpack_loc_SLQSLOCGetServer . . . 1179
9.10.4.37 unpack_loc_SLQSLOCInjectPosition . . . 1179
9.10.4.38 unpack_loc_SLQSLOCInjectSensorData . . . 1180
9.10.4.39 unpack_loc_SLQSLOCInjectUTCTime . . . 1180
9.10.4.40 unpack_loc_SLQSLOCSetCradleMountConfig . . . 1180
9.10.4.41 unpack_loc_SLQSLOCSetServer . . . 1181
9.10.4.42 unpack_loc_Start . . . 1181
9.10.4.43 unpack_loc_Stop . . . 1181
9.11 nas.h File Reference . . . 1182
9.11.1 Macro Definition Documentation . . . 1194
9.11.1.1 LITE_NW_SCAN_LTE_OP_MODE_MAX_LENGTH . . . 1194
9.11.1.2 NAS_IMSI_M_S1_LENGTH . . . 1194

9.11.1.3 NAS_IMSI_M_S2_LENGTH . . . 1194
9.11.1.4 NAS_MAX_DESCRIPTION_LENGTH . . . 1194
9.11.1.5 NAS_MAX_NUM_NETWORKS . . . 1194
9.11.1.6 NAS_MAX_PHY_CA_AGG_SCELL_ARRRY_SIZE . . . 1194
9.11.1.7 NAS_MAX_SCC_RX_INFO_INSTANCES . . . 1194
9.11.1.8 NAS_MAX_SLQS_NAS_PCI_INFO_LENGTH . . . 1194
9.11.1.9 NAS_MAX_SLQS_NAS_PCI_INFO_PLMN_LENGTH . . . 1194
9.11.1.10 NAS_MCC_MNC_INST_LENGTH . . . 1194
9.11.1.11 NAS_NAM_NAME_LENGTH . . . 1194
9.11.1.12 NAS_OTA_MESSAGE_MAX_BUF_SIZE . . . 1194
9.11.1.13 NAS_PLMN_LENGTH . . . 1194
9.11.1.14 NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST . . . 1194
9.11.2 Typedef Documentation . . . 1194
9.11.2.1 nas_LTEBandPrefExtTlv . . . 1194
9.11.2.2 nas_NR5gBandPrefTlv . . . 1194
9.11.2.3 unpack_nas_InitiateDomainAttach_t . . . 1195
9.11.2.4 unpack_nas_SetACCOLC_t . . . 1195
9.11.2.5 unpack_nas_SetCDMANetworkParameters_t . . . 1195
9.11.2.6 unpack_nas_SetLURejectCallback_t . . . 1195
9.11.2.7 unpack_nas_SetRFInfoCallback_t . . . 1195
9.11.2.8 unpack_nas_SLQSConfigSigInfo_t . . . 1195
9.11.2.9 unpack_nas_SLQSInitiateNetworkRegistration_t . . . 1195
9.11.2.10 unpack_nas_SLQSNasConfigSigInfo2_t . . . 1195
9.11.2.11 unpack_nas_SLQSNasIndicationRegisterExt_t . . . 1195
9.11.2.12 unpack_nas_SLQSNASSeteDRXParams_t . . . 1195
9.11.2.13 unpack_nas_SLQSNasSwiIndicationRegister_t . . . 1195
9.11.2.14 unpack_nas_SLQSNASSwiSetChannelLock_t . . . 1195
9.11.2.15 unpack_nas_SLQSSetBandPreference_t . . . 1195
9.11.2.16 unpack_nas_SLQSSetSignalStrengthsCallback_t . . . 1195
9.11.2.17 unpack_nas_SLQSSetSysSelectionPref_t . . . 1195
9.11.2.18 unpack_nas_SLQSSetSysSelectionPrefExt_t . . . 1195
9.11.2.19 unpack_nas_SLQSSwiPSDetach_t . . . 1195
9.11.3 Enumeration Type Documentation . . . 1195
9.11.3.1 LITE_TYPE_OF_SERVICE_DOMAIN . . . 1195
9.11.3.2 LITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND . . . 1195
9.11.3.3 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB . . . 1197
9.11.3.4 LITEQMI_NAS_LTE_CPHY_SCELL_STATE . . . 1197
9.11.3.5 NAS_LTE_CPHY_CA_BW_NRB_LITE . . . 1197
9.11.3.6 NAS_LTE_CPHY_SCELL_STATE_LITE . . . 1197
9.11.4 Function Documentation . . . 1197

9.11.4.1 pack_nas_GetACCOLC . . . 1197
9.11.4.2 pack_nas_GetANAAAAuthenticationStatus . . . 1198
9.11.4.3 pack_nas_GetCDMANetworkParameters . . . 1198
9.11.4.4 pack_nas_GetHomeNetwork . . . 1198
9.11.4.5 pack_nas_GetHomeNetwork3GPP2 . . . 1199
9.11.4.6 pack_nas_GetNetworkPreference . . . 1199
9.11.4.7 pack_nas_GetRFInfo . . . 1199
9.11.4.8 pack_nas_GetServingNetwork . . . 1200
9.11.4.9 pack_nas_GetServingNetworkCapabilities . . . 1200
9.11.4.10 pack_nas_GetSignalStrengths . . . 1200
9.11.4.11 pack_nas_InitiateDomainAttach . . . 1201
9.11.4.12 pack_nas_PerformNetworkScan . . . 1201
9.11.4.13 pack_nas_PerformNetworkScanPCI . . . 1201
9.11.4.14 pack_nas_SetACCOLC . . . 1202
9.11.4.15 pack_nas_SetCDMANetworkParameters . . . 1202
9.11.4.16 pack_nas_SetLURejectCallback . . . 1202
9.11.4.17 pack_nas_SetNetworkPreference . . . 1203
9.11.4.18 pack_nas_SetRFInfoCallback . . . 1203
9.11.4.19 pack_nas_SLQSConfigSigInfo . . . 1203
9.11.4.20 pack_nas_SLQSGetErrorRate . . . 1204
9.11.4.21 pack_nas_SlqsGetLTECphyCAInfo . . . 1204
9.11.4.22 pack_nas_SLQSGetNetworkTime . . . 1204
9.11.4.23 pack_nas_SLQSGetOperatorNameData . . . 1205
9.11.4.24 pack_nas_SLQSGetPLMNName . . . 1205
9.11.4.25 pack_nas_SLQSGetServingSystem . . . 1205
9.11.4.26 pack_nas_SLQSGetSignalStrength . . . 1206
9.11.4.27 pack_nas_SLQSGetSysInfo . . . 1206
9.11.4.28 pack_nas_SLQSGetSysSelectionPref . . . 1206
9.11.4.29 pack_nas_SLQSGetSysSelectionPrefExt . . . 1207
9.11.4.30 pack_nas_SLQSInitiateNetworkRegistration . . . 1207
9.11.4.31 pack_nas_SLQSNasConfigSigInfo2 . . . 1208
9.11.4.32 pack_nas_SLQSNasGet3GPP2Subscription . . . 1208
9.11.4.33 pack_nas_SLQSNasGetCellLocationInfo . . . 1208
9.11.4.34 pack_nas_SLQSNASGeteDRXParams . . . 1209
9.11.4.35 pack_nas_SLQSNASGeteDRXParamsExt . . . 1209
9.11.4.36 pack_nas_SLQSNASGetForbiddenNetworks . . . 1209
9.11.4.37 pack_nas_SLQSNasGetHDRColorCode . . . 1210
9.11.4.38 pack_nas_SLQSNasGetRFInfo . . . 1210
9.11.4.39 pack_nas_SLQSNasGetSigInfo . . . 1210
9.11.4.40 pack_nas_SLQSNasGetTxRxInfo . . . 1211

9.11.4.41 pack_nas_SLQSNasIndicationRegisterExt . . . 1211
9.11.4.42 pack_nas_SLQSNASSeteDRXParams . . . 1212
9.11.4.43 pack_nas_SLQSNASSwiGetChannelLock . . . 1212
9.11.4.44 pack_nas_SLQSNasSwiIndicationRegister . . . 1212
9.11.4.45 pack_nas_SLQSNasSwiModemStatus . . . 1213
9.11.4.46 pack_nas_SLQSNASSwiSetChannelLock . . . 1213
9.11.4.47 pack_nas_SLQSSetBandPreference . . . 1213
9.11.4.48 pack_nas_SLQSSetSignalStrengthsCallback . . . 1214
9.11.4.49 pack_nas_SLQSSetSysSelectionPref . . . 1214
9.11.4.50 pack_nas_SLQSSetSysSelectionPrefExt . . . 1214
9.11.4.51 pack_nas_SLQSSwiGetHDRPersonality . . . 1215
9.11.4.52 pack_nas_SLQSSwiGetHDRProtSubtype . . . 1215
9.11.4.53 pack_nas_SLQSSwiGetHRPDStats . . . 1215
9.11.4.54 pack_nas_SLQSSwiGetLteCQI . . . 1216
9.11.4.55 pack_nas_SLQSSwiGetLteSccRxInfo . . . 1216
9.11.4.56 pack_nas_SLQSSwiNetworkDebug . . . 1216
9.11.4.57 pack_nas_SLQSSwiPSDetach . . . 1217
9.11.4.58 unpack_nas_GetACCOLC . . . 1217
9.11.4.59 unpack_nas_GetANAAAAuthenticationStatus . . . 1218
9.11.4.60 unpack_nas_GetCDMANetworkParameters . . . 1218
9.11.4.61 unpack_nas_GetHomeNetwork . . . 1218
9.11.4.62 unpack_nas_GetHomeNetwork3GPP2 . . . 1219
9.11.4.63 unpack_nas_GetNetworkPreference . . . 1219
9.11.4.64 unpack_nas_GetRFInfo . . . 1219
9.11.4.65 unpack_nas_GetServingNetwork . . . 1220
9.11.4.66 unpack_nas_GetServingNetworkCapabilities . . . 1220
9.11.4.67 unpack_nas_GetSignalStrengths . . . 1220
9.11.4.68 unpack_nas_InitiateDomainAttach . . . 1221
9.11.4.69 unpack_nas_PerformNetworkScan . . . 1221
9.11.4.70 unpack_nas_SetACCOLC . . . 1222
9.11.4.71 unpack_nas_SetCDMANetworkParameters . . . 1222
9.11.4.72 unpack_nas_SetDataCapabilitiesCallback_ind . . . 1222
9.11.4.73 unpack_nas_SetEventReportInd . . . 1223
9.11.4.74 unpack_nas_SetLURejectCallback . . . 1223
9.11.4.75 unpack_nas_SetNasLTECphyCaIndCallback_ind . . . 1223
9.11.4.76 unpack_nas_SetNetworkPreference . . . 1224
9.11.4.77 unpack_nas_SetRFInfoCallback . . . 1224
9.11.4.78 unpack_nas_SetRoamingIndicatorCallback_ind . . . 1224
9.11.4.79 unpack_nas_SetServingSystemCallback_ind . . . 1225
9.11.4.80 unpack_nas_SLQSConfigSigInfo . . . 1225

9.11.4.81 unpack_nas_SLQSGetErrorRate . . . 1225
9.11.4.82 unpack_nas_SlqsGetLTECphyCAInfo . . . 1226
9.11.4.83 unpack_nas_SLQSGetNetworkTime . . . 1226
9.11.4.84 unpack_nas_SLQSGetOperatorNameData . . . 1226
9.11.4.85 unpack_nas_SLQSGetPLMNName . . . 1227
9.11.4.86 unpack_nas_SLQSGetServingSystem . . . 1227
9.11.4.87 unpack_nas_SLQSGetSignalStrength . . . 1227
9.11.4.88 unpack_nas_SLQSGetSysInfo . . . 1228
9.11.4.89 unpack_nas_SLQSGetSysSelectionPref . . . 1228
9.11.4.90 unpack_nas_SLQSGetSysSelectionPrefExt . . . 1229
9.11.4.91 unpack_nas_SLQSInitiateNetworkRegistration . . . 1229
9.11.4.92 unpack_nas_SLQSNasConfigSigInfo2 . . . 1229
9.11.4.93 unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind . . . 1230
9.11.4.94 unpack_nas_SLQSNasGet3GPP2Subscription . . . 1230
9.11.4.95 unpack_nas_SLQSNasGetCellLocationInfo . . . 1231
9.11.4.96 unpack_nas_SLQSNASGeteDRXParams . . . 1231
9.11.4.97 unpack_nas_SLQSNASGeteDRXParamsExt . . . 1231
9.11.4.98 unpack_nas_SLQSNASGetForbiddenNetworks . . . 1232
9.11.4.99 unpack_nas_SLQSNasGetHDRColorCode . . . 1232
9.11.4.100 unpack_nas_SLQSNasGetRFInfo . . . 1232
9.11.4.101 unpack_nas_SLQSNasGetSigInfo . . . 1233
9.11.4.102 unpack_nas_SLQSNasGetTxRxInfo . . . 1233
9.11.4.103 unpack_nas_SLQSNasIndicationRegisterExt . . . 1234
9.11.4.104 unpack_nas_SLQSNasNetworkRejectCallback_Ind . . . 1234
9.11.4.105 unpack_nas_SLQSNasNetworkTimeCallBack_ind . . . 1234
9.11.4.106 unpack_nas_SLQSNASSeteDRXParams . . . 1235
9.11.4.107 unpack_nas_SLQSNasSigInfoCallback_ind . . . 1235
9.11.4.108 unpack_nas_SLQSNASSwiGetChannelLock . . . 1235
9.11.4.109 unpack_nas_SLQSNasSwiIndicationRegister . . . 1236
9.11.4.110 unpack_nas_SLQSNasSwiModemStatus . . . 1236
9.11.4.111 unpack_nas_SLQSNasSwiOTAMessageCallback_ind . . . 1236
9.11.4.112 unpack_nas_SLQSNASSwiSetChannelLock . . . 1237
9.11.4.113 unpack_nas_SLQSNasSysInfoCallback_ind . . . 1237
9.11.4.114 unpack_nas_SLQSNasTimerCallback_ind . . . 1237
9.11.4.115 unpack_nas_SLQSSetBandPreference . . . 1238
9.11.4.116 unpack_nas_SLQSSetSignalStrengthsCallback . . . 1238
9.11.4.117 unpack_nas_SLQSSetSysSelectionPref . . . 1238
9.11.4.118 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind . . . 1239
9.11.4.119 unpack_nas_SLQSSetSysSelectionPrefExt . . . 1239
9.11.4.120 unpack_nas_SLQSSwiGetHDRPersonality . . . 1239

9.11.4.121 unpack_nas_SLQSSwiGetHDRProtSubtype . . . 1240
9.11.4.122 unpack_nas_SLQSSwiGetHRPDStats . . . 1240
9.11.4.123 unpack_nas_SLQSSwiGetLteCQI . . . 1240
9.11.4.124 unpack_nas_SLQSSwiGetLteSccRxInfo . . . 1241
9.11.4.125 unpack_nas_SLQSSwiHDRPersonalityCallback_Ind . . . 1241
9.11.4.126 unpack_nas_SLQSSwiNetworkDebug . . . 1241
9.11.4.127 unpack_nas_SLQSSwiPSDetach . . . 1242
9.11.4.128 unpack_nas_SLQSSwiRandIndicatorCallback_Ind . . . 1242
9.11.4.129 unpack_valid_nas_GetCDMANetworkParameters . . . 1242
9.11.4.130 unpack_valid_nas_SLQSGetServingSystem . . . 1243
9.11.4.131 unpack_valid_nas_SLQSGetSignalStrength . . . 1244
9.11.4.132 unpack_valid_nas_SLQSNasGetSigInfo . . . 1244
9.12 pds.h File Reference . . . 1245
9.12.1 Typedef Documentation . . . 1247
9.12.1.1 unpack_pds_ForceXTRADownload_t . . . 1247
9.12.1.2 unpack_pds_PDSInjectTimeReference_t . . . 1247
9.12.1.3 unpack_pds_ResetPDSData_t . . . 1247
9.12.1.4 unpack_pds_SetEventReportCallback_t . . . 1247
9.12.1.5 unpack_pds_SetPDSDefaults_t . . . 1247
9.12.1.6 unpack_pds_SetPDSState_t . . . 1247
9.12.1.7 unpack_pds_SetPortAutomaticTracking_t . . . 1248
9.12.1.8 unpack_pds_SetServiceAutomaticTracking_t . . . 1248
9.12.1.9 unpack_pds_SetXTRAAutomaticDownload_t . . . 1248
9.12.1.10 unpack_pds_SetXTRANetwork_t . . . 1248
9.12.1.11 unpack_pds_SLQSPDSDeterminePosition_t . . . 1248
9.12.1.12 unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t . . . 1248
9.12.1.13 unpack_pds_SLQSPDSInjectPositionData_t . . . 1248
9.12.1.14 unpack_pds_SLQSSetAGPSConfig_t . . . 1248
9.12.1.15 unpack_pds_SLQSSetPositionMethodState_t . . . 1248
9.12.1.16 unpack_pds_StartPDSTrackingSessionExt_t . . . 1248
9.12.1.17 unpack_pds_StopPDSTrackingSession_t . . . 1248
9.12.2 Function Documentation . . . 1248
9.12.2.1 pack_pds_ForceXTRADownload . . . 1248
9.12.2.2 pack_pds_GetPDSDefaults . . . 1248
9.12.2.3 pack_pds_GetPDSState . . . 1249
9.12.2.4 pack_pds_GetPortAutomaticTracking . . . 1249
9.12.2.5 pack_pds_GetServiceAutomaticTracking . . . 1249
9.12.2.6 pack_pds_GetXTRAAutomaticDownload . . . 1250
9.12.2.7 pack_pds_GetXTRANetwork . . . 1250
9.12.2.8 pack_pds_GetXTRAValidity . . . 1250

9.12.2.9 pack_pds_PDSInjectTimeReference . . . 1251
9.12.2.10 pack_pds_ResetPDSData . . . 1251
9.12.2.11 pack_pds_SetEventReportCallback . . . 1251
9.12.2.12 pack_pds_SetPDSDefaults . . . 1252
9.12.2.13 pack_pds_SetPDSState . . . 1252
9.12.2.14 pack_pds_SetPortAutomaticTracking . . . 1252
9.12.2.15 pack_pds_SetServiceAutomaticTracking . . . 1253
9.12.2.16 pack_pds_SetXTRAAutomaticDownload . . . 1253
9.12.2.17 pack_pds_SetXTRANetwork . . . 1254
9.12.2.18 pack_pds_SLQSGetAGPSConfig . . . 1254
9.12.2.19 pack_pds_SLQSGetGPSStateInfo . . . 1254
9.12.2.20 pack_pds_SLQSPDSDeterminePosition . . . 1255
9.12.2.21 pack_pds_SLQSPDSInjectAbsoluteTimeReference . . . 1255
9.12.2.22 pack_pds_SLQSPDSInjectPositionData . . . 1255
9.12.2.23 pack_pds_SLQSSetAGPSConfig . . . 1256
9.12.2.24 pack_pds_SLQSSetPositionMethodState . . . 1256
9.12.2.25 pack_pds_StartPDSTrackingSessionExt . . . 1256
9.12.2.26 pack_pds_StopPDSTrackingSession . . . 1257
9.12.2.27 unpack_pds_ForceXTRADownload . . . 1257
9.12.2.28 unpack_pds_GetPDSDefaults . . . 1258
9.12.2.29 unpack_pds_GetPDSState . . . 1258
9.12.2.30 unpack_pds_GetPortAutomaticTracking . . . 1258
9.12.2.31 unpack_pds_GetServiceAutomaticTracking . . . 1259
9.12.2.32 unpack_pds_GetXTRAAutomaticDownload . . . 1259
9.12.2.33 unpack_pds_GetXTRANetwork . . . 1259
9.12.2.34 unpack_pds_GetXTRAValidity . . . 1260
9.12.2.35 unpack_pds_PDSInjectTimeReference . . . 1260
9.12.2.36 unpack_pds_ResetPDSData . . . 1260
9.12.2.37 unpack_pds_SetEventReport_Ind . . . 1261
9.12.2.38 unpack_pds_SetEventReportCallback . . . 1261
9.12.2.39 unpack_pds_SetPDSDefaults . . . 1261
9.12.2.40 unpack_pds_SetPDSState . . . 1262
9.12.2.41 unpack_pds_SetPdsState_Ind . . . 1262
9.12.2.42 unpack_pds_SetPortAutomaticTracking . . . 1262
9.12.2.43 unpack_pds_SetServiceAutomaticTracking . . . 1263
9.12.2.44 unpack_pds_SetXTRAAutomaticDownload . . . 1263
9.12.2.45 unpack_pds_SetXTRANetwork . . . 1263
9.12.2.46 unpack_pds_SLQSGetAGPSConfig . . . 1264
9.12.2.47 unpack_pds_SLQSGetGPSStateInfo . . . 1264
9.12.2.48 unpack_pds_SLQSPDSDeterminePosition . . . 1264

9.12.2.49 unpack_pds_SLQSPDSInjectAbsoluteTimeReference . . . 1265
9.12.2.50 unpack_pds_SLQSPDSInjectPositionData . . . 1265
9.12.2.51 unpack_pds_SLQSSetAGPSConfig . . . 1265
9.12.2.52 unpack_pds_SLQSSetPositionMethodState . . . 1266
9.12.2.53 unpack_pds_StartPDSTrackingSessionExt . . . 1266
9.12.2.54 unpack_pds_StopPDSTrackingSession . . . 1266
9.13 qaGobiApiTableBandClasses.h File Reference . . . 1267
9.13.1 Detailed Description . . . 1267
9.13.2 Band Classes (Value - Description) . . . 1267
9.13.2.1 LTE Bands . . . 1268
9.14 qaGobiApiTableCallControlReturnReasons.h File Reference . . . 1270
9.14.1 Detailed Description . . . 1270
9.14.2 S1 . . . 1270
9.15 qaGobiApiTableCallEndReasons.h File Reference . . . 1271
9.15.1 Detailed Description . . . 1271
9.15.2 Call end reason codes (Code - Reason) . . . 1271
9.15.2.1 Technology-agnostic call end reasons . . . 1271
9.15.2.2 EVDO CDMA 1xEV-DO . . . 1272
9.15.2.3 WCDMA/GSM call end reasons . . . 1272
9.15.2.4 EVDO CDMA 1xEV-DO . . . 1274
9.15.2.5 call end reason type . . . 1275
9.15.2.6 Mobile IP call end reasons (Type=1) . . . 1275
9.15.2.7 Internal call end reasons (Type=2) . . . 1277
9.15.2.8 Call Manager defined call end reasons (Type=3) . . . 1279
9.15.2.9 3GPP specification defined call end reasons (Type=6) . . . 1284
9.15.2.10 PPP call end reasons (Type=7) . . . 1286
9.15.2.11 EHRPD call end reasons (Type=8) . . . 1286
9.15.2.12 IPV6 call end reasons (Type=9) . . . 1287
9.16 qaGobiApiTableCarrierCodes.h File Reference . . . 1287
9.16.1 Detailed Description . . . 1287
9.16.2 Carrier Codes (Number - Carrier) . . . 1288
9.17 qaGobiApiTableCodingScheme.h File Reference . . . 1289
9.17.1 Detailed Description . . . 1289
9.17.2 S1 . . . 1290
9.17.2.1 Use of bits 3..0\n\n . . . 1290
9.17.3 Coding Group Bits 7..4(0001) . . . 1290
9.17.3.1 use of bits 3..0 . . . 1290
9.17.4 Coding Group Bits 7..4(0010) . . . 1290
9.17.4.1 use of bits 3..0 . . . 1290
9.17.5 Coding Group Bits 7..4(0011) . . . 1291

9.17.5.1 use of bits 3..0 . . . 1291
9.17.6 Coding Group Bits 7..4(01xx) . . . 1291
9.17.6.1 use of bits 3..0 . . . 1291
9.17.7 Coding Group Bits 7..4(1001) . . . 1291
9.17.7.1 Reserved coding groups . . . 1291
9.17.8 Coding Group Bits 7..4(1010..1101) . . . 1292
9.17.8.1 Reserved coding groups . . . 1292
9.17.9 Coding Group Bits 7..4(1110) . . . 1292
9.17.9.1 Defined by the WAP Forum . . . 1292
9.17.10 Coding Group Bits 7..4 (1111) . . . 1292
9.17.10.1 Data coding / message handling . . . 1292
9.17.11 Macro Definition Documentation . . . 1292
9.17.11.1 __GOBI_API_CODING_SCHEME_H__ . . . 1292
9.18 qaGobiApiTableGpsCapabilityCodes.h File Reference . . . 1292
9.18.1 Detailed Description . . . 1292
9.18.2 GPS capability (Value - Capability) . . . 1292
9.19 qaGobiApiTablePowerModes.h File Reference . . . 1293
9.19.1 Detailed Description . . . 1293
9.19.2 Power Modes (Value - Description) . . . 1293
9.20 qaGobiApiTableRadioInterfaces.h File Reference . . . 1293
9.20.1 Detailed Description . . . 1293
9.20.2 Radio interface . . . 1293
9.20.2.1 Technology (Value - Radio Interface Technology) . . . 1293
9.21 qaGobiApiTableRegionCodes.h File Reference . . . 1294
9.21.1 Detailed Description . . . 1294
9.21.2 Region Codes (Code - Region) . . . 1294
9.22 qaGobiApiTableServiceOptions.h File Reference . . . 1294
9.22.1 Detailed Description . . . 1294
9.22.2 Service Option codes (Code - Reason) . . . 1295
9.22.2.1 Description . . . 1295
9.23 qaGobiApiTableSupServiceInfoClasses.h File Reference . . . 1297
9.23.1 Detailed Description . . . 1297
9.23.2 Supplementary Service Information Classes (Value - Service Class) . . . 1297
9.24 qaGobiApiTableSwiAudio.h File Reference . . . 1297
9.24.1 Detailed Description . . . 1297
9.24.2 ACDB Device (Device ID - description) . . . 1297
9.24.3 Physical Interface (Device ID - description - Interface parameters) . . . 1298
9.25 qaGobiApiTableSwiOMADMSessionStatus.h File Reference . . . 1298
9.25.1 Detailed Description . . . 1298
9.25.2 OMA DM Session Status (Session Status - Meaning - Usage) . . . 1298

9.26 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference . . . 1299
9.26.1 Detailed Description . . . 1299
9.26.2 OMA DM Update Complete Status (Update Complete Status - Meaning - Usage) . . . 1299
9.27 qaGobiApiTableVoiceCallEndReasons.h File Reference . . . 1300
9.27.1 Detailed Description . . . 1300
9.27.2 Voice Call and supplementary services end reason codes (Code - Reason) . . . 1300
9.27.2.1 General . . . 1300
9.27.2.2 service Errors . . . 1302
9.27.2.3 control cause values . . . 1303
9.27.2.4 reject causes . . . 1305
9.27.2.5 reject causes . . . 1305
9.27.2.6 reject causes . . . 1305
9.27.2.7 stratum reject causes . . . 1306
9.27.2.8 reject causes . . . 1306
9.27.2.9 IP end reasons . . . 1306
9.28 qmerrno.h File Reference . . . 1307
9.28.1 Enumeration Type Documentation . . . 1309
9.28.1.1 eQCWWANError . . . 1309
9.28.1.2 qm_wds_ds_profile_extended_err_codes . . . 1314
9.29 qos.h File Reference . . . 1314
9.29.1 Macro Definition Documentation . . . 1316
9.29.1.1 LITEQMI_MAX_QOS_FILTERS . . . 1316
9.29.1.2 LITEQMI_MAX_QOS_FLOW_PER_APN_STATS . . . 1316
9.29.1.3 LITEQMI_MAX_QOS_FLOWS . . . 1316
9.29.2 Function Documentation . . . 1316
9.29.2.1 pack_qos_BindDataPort . . . 1316
9.29.2.2 pack_qos_SLQSQosGetNetworkStatus . . . 1316
9.29.2.3 pack_qos_SLQSQosSwiReadApnExtraParams . . . 1317
9.29.2.4 pack_qos_SLQSQosSwiReadDataStats . . . 1318
9.29.2.5 pack_qos_SLQSSetQosEventCallback . . . 1318
9.29.2.6 unpack_qos_BindDataPort . . . 1319
9.29.2.7 unpack_qos_SLQSQosGetNetworkStatus . . . 1319
9.29.2.8 unpack_qos_SLQSQosSwiReadApnExtraParams . . . 1320
9.29.2.9 unpack_qos_SLQSQosSwiReadDataStats . . . 1320
9.29.2.10 unpack_qos_SLQSSetQosEventCallback . . . 1321
9.29.2.11 unpack_qos_SLQSSetQosEventCallback_ind . . . 1321
9.29.2.12 unpack_qos_SLQSSetQosNWStatusCallback_ind . . . 1322
9.29.2.13 unpack_qos_SLQSSetQosPriEventCallback_ind . . . 1323
9.29.2.14 unpack_qos_SLQSSetQosStatusCallback_ind . . . 1323
9.30 rms.h File Reference . . . 1324

9.30.1 Macro Definition Documentation . . . 1324
9.30.1.1 __LITEQMI_RMS_H__ . . . 1324
9.30.2 Function Documentation . . . 1324
9.30.2.1 pack_rms_GetSMSWake . . . 1324
9.30.2.2 pack_rms_SetSMSWake . . . 1325
9.30.2.3 unpack_rms_GetSMSWake . . . 1325
9.30.2.4 unpack_rms_SetSMSWake . . . 1325
9.31 sar.h File Reference . . . 1326
9.31.1 Typedef Documentation . . . 1326
9.31.1.1 unpack_sar_SLQSSetRfSarState_t . . . 1326
9.31.2 Function Documentation . . . 1326
9.31.2.1 pack_sar_SLQSGetRfSarState . . . 1326
9.31.2.2 pack_sar_SLQSSetRfSarState . . . 1327
9.31.2.3 unpack_sar_SLQSGetRfSarState . . . 1327
9.31.2.4 unpack_sar_SLQSSetRfSarState . . . 1327
9.32 sms.h File Reference . . . 1328
9.32.1 Macro Definition Documentation . . . 1331
9.32.1.1 MAX_CDMA_ENC_MO_TXT_MSG_SIZE . . . 1331
9.32.1.2 MAX_MS_TRANSFER_ROUTE_MSG . . . 1331
9.32.1.3 MAX_MSC_ADDRESS_SIZE . . . 1331
9.32.1.4 MAX_MSE_TWS_MSG . . . 1331
9.32.1.5 MAX_SMS_LIST_SIZE . . . 1331
9.32.1.6 MAX_SMS_MESSAGE_SIZE . . . 1331
9.32.1.7 SMS_CONFIG_LEN . . . 1332
9.32.1.8 SMS_MAX_SMS_ROUTES . . . 1332
9.32.1.9 SMS_NUM_OF_SET . . . 1332
9.32.1.10 SMSC_TYPE_LEN . . . 1332
9.32.2 Enumeration Type Documentation . . . 1332
9.32.2.1 eqmiCbkSetStatus . . . 1332
9.32.3 Function Documentation . . . 1332
9.32.3.1 pack_sms_GetSMSCAddress . . . 1332
9.32.3.2 pack_sms_SaveSMS . . . 1332
9.32.3.3 pack_sms_SendSMS . . . 1333
9.32.3.4 pack_sms_SetNewSMSCallback . . . 1333
9.32.3.5 pack_sms_SetSMSCAddress . . . 1333
9.32.3.6 pack_sms_SLQSDeleteSMS . . . 1334
9.32.3.7 pack_sms_SLQSGetIndicationRegister . . . 1334
9.32.3.8 pack_sms_SLQSGetMessageWaiting . . . 1334
9.32.3.9 pack_sms_SLQSGetSMS . . . 1335
9.32.3.10 pack_sms_SLQSGetSmsBroadcastConfig . . . 1335

9.32.3.11 pack_sms_SLQSGetSMSList . . . 1335
9.32.3.12 pack_sms_SLQSGetTransLayerInfo . . . 1336
9.32.3.13 pack_sms_SLQSGetTransNWRegInfo . . . 1336
9.32.3.14 pack_sms_SLQSModifySMSStatus . . . 1336
9.32.3.15 pack_sms_SLQSSendAsyncSMS . . . 1337
9.32.3.16 pack_sms_SLQSSetIndicationRegister . . . 1337
9.32.3.17 pack_sms_SLQSSetSmsBroadcastActivation . . . 1338
9.32.3.18 pack_sms_SLQSSetSmsBroadcastConfig . . . 1338
9.32.3.19 pack_sms_SLQSSetSmsStorage . . . 1338
9.32.3.20 pack_sms_SLQSSmsGetMaxStorageSize . . . 1339
9.32.3.21 pack_sms_SLQSSmsGetMessageProtocol . . . 1339
9.32.3.22 pack_sms_SLQSSmsSetRoutes . . . 1339
9.32.3.23 pack_sms_SLQSSwiGetSMSStorage . . . 1340
9.32.3.24 unpack_sms_GetSMSCAddress . . . 1340
9.32.3.25 unpack_sms_SaveSMS . . . 1340
9.32.3.26 unpack_sms_SendSMS . . . 1341
9.32.3.27 unpack_sms_SetNewSMSCallback . . . 1341
9.32.3.28 unpack_sms_SetNewSMSCallback_ind . . . 1341
9.32.3.29 unpack_sms_SetSMSCAddress . . . 1342
9.32.3.30 unpack_sms_SLQSDeleteSMS . . . 1342
9.32.3.31 unpack_sms_SLQSGetIndicationRegister . . . 1343
9.32.3.32 unpack_sms_SLQSGetMessageWaiting . . . 1343
9.32.3.33 unpack_sms_SLQSGetSMS . . . 1343
9.32.3.34 unpack_sms_SLQSGetSmsBroadcastConfig . . . 1344
9.32.3.35 unpack_sms_SLQSGetSMSList . . . 1344
9.32.3.36 unpack_sms_SLQSGetTransLayerInfo . . . 1344
9.32.3.37 unpack_sms_SLQSGetTransNWRegInfo . . . 1345
9.32.3.38 unpack_sms_SLQSModifySMSStatus . . . 1345
9.32.3.39 unpack_sms_SLQSNWRegInfoCallback_ind . . . 1345
9.32.3.40 unpack_sms_SLQSSendAsyncSMS . . . 1346
9.32.3.41 unpack_sms_SLQSSetIndicationRegister . . . 1346
9.32.3.42 unpack_sms_SLQSSetSmsBroadcastActivation . . . 1346
9.32.3.43 unpack_sms_SLQSSetSmsBroadcastConfig . . . 1347
9.32.3.44 unpack_sms_SLQSSetSmsStorage . . . 1347
9.32.3.45 unpack_sms_SLQSSmsGetMaxStorageSize . . . 1347
9.32.3.46 unpack_sms_SLQSSmsGetMessageProtocol . . . 1348
9.32.3.47 unpack_sms_SLQSSmsSetRoutes . . . 1348
9.32.3.48 unpack_sms_SLQSSwiGetSMSStorage . . . 1349
9.32.3.49 unpack_sms_SLQSTransLayerInfoCallback_ind . . . 1349
9.32.3.50 unpack_sms_SLQSWmsAsyncRawSendCallBack_ind . . . 1349

9.32.3.51 unpack_sms_SLQSWmsMemoryFullCallBack_ind . . . 1350
9.32.3.52 unpack_sms_SLQSWmsMessageWaitingCallBack_ind . . . 1350
9.33 swiaudio.h File Reference . . . 1350
9.33.1 Macro Definition Documentation . . . 1352
9.33.1.1 SWIAUDIO_MAX_LEN_IFACE_TABLE . . . 1352
9.33.2 Typedef Documentation . . . 1352
9.33.2.1 unpack_swiaudio_SLQSSetM2MAudioAVCFG_t . . . 1352
9.33.2.2 unpack_swiaudio_SLQSSetM2MAudioLPBK_t . . . 1352
9.33.2.3 unpack_swiaudio_SLQSSetM2MAudioNVDef_t . . . 1352
9.33.2.4 unpack_swiaudio_SLQSSetM2MAudioProfile_t . . . 1352
9.33.2.5 unpack_swiaudio_SLQSSetM2MAudioVolume_t . . . 1352
9.33.2.6 unpack_swiaudio_SLQSSetM2MAVMute_t . . . 1352
9.33.2.7 unpack_swiaudio_SLQSSetM2MSpkrGain_t . . . 1352
9.33.3 Function Documentation . . . 1352
9.33.3.1 pack_swiaudio_SLQSGetM2MAudioProfile . . . 1352
9.33.3.2 pack_swiaudio_SLQSGetM2MAudioVolume . . . 1353
9.33.3.3 pack_swiaudio_SLQSGetM2MAVMute . . . 1353
9.33.3.4 pack_swiaudio_SLQSGetM2MSpkrGain . . . 1353
9.33.3.5 pack_swiaudio_SLQSSetM2MAudioAVCFG . . . 1354
9.33.3.6 pack_swiaudio_SLQSSetM2MAudioLPBK . . . 1354
9.33.3.7 pack_swiaudio_SLQSSetM2MAudioNVDef . . . 1354
9.33.3.8 pack_swiaudio_SLQSSetM2MAudioProfile . . . 1355
9.33.3.9 pack_swiaudio_SLQSSetM2MAudioVolume . . . 1355
9.33.3.10 pack_swiaudio_SLQSSetM2MAVMute . . . 1356
9.33.3.11 pack_swiaudio_SLQSSetM2MSpkrGain . . . 1356
9.33.3.12 unpack_swiaudio_SLQSGetM2MAudioProfile . . . 1356
9.33.3.13 unpack_swiaudio_SLQSGetM2MAudioVolume . . . 1357
9.33.3.14 unpack_swiaudio_SLQSGetM2MAVMute . . . 1357
9.33.3.15 unpack_swiaudio_SLQSGetM2MSpkrGain . . . 1357
9.33.3.16 unpack_swiaudio_SLQSSetM2MAudioAVCFG . . . 1358
9.33.3.17 unpack_swiaudio_SLQSSetM2MAudioLPBK . . . 1358
9.33.3.18 unpack_swiaudio_SLQSSetM2MAudioNVDef . . . 1358
9.33.3.19 unpack_swiaudio_SLQSSetM2MAudioProfile . . . 1359
9.33.3.20 unpack_swiaudio_SLQSSetM2MAudioVolume . . . 1359
9.33.3.21 unpack_swiaudio_SLQSSetM2MAVMute . . . 1359
9.33.3.22 unpack_swiaudio_SLQSSetM2MSpkrGain . . . 1360
9.34 swiavms.h File Reference . . . 1360
9.34.1 Macro Definition Documentation . . . 1362
9.34.1.1 LITEQMI_MAX_GET_SETTINGS_AVMS_APN_STRING_LENGTH . . . 1362
9.34.1.2 LITEQMI_MAX_GET_SETTINGS_AVMS_PWD_STRING_LENGTH . . . 1362

9.34.1.3 LITEQMI_MAX_GET_SETTINGS_AVMS_UNAME_STRING_LENGTH 1362
9.34.1.4 LITEQMI_MAX_SWIOMA_STR_LEN 1362
9.34.1.5 MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER 1362
9.34.1.6 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH 1362
9.34.1.7 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH 1362
9.34.1.8 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH 1362
9.34.1.9 MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH 1362
9.34.1.10 MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH 1362
9.34.1.11 MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS . 1362
9.34.1.12 MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 1362
9.34.1.13 MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 1362
9.34.2 Function Documentation 1362
9.34.2.1 pack_swiavms_SLQSAVMSGetSettings 1362
9.34.2.2 pack_swiavms_SLQSAVMSGetSettings_v2 1363
9.34.2.3 pack_swiavms_SLQSAVMSSendSelection 1364
9.34.2.4 pack_swiavms_SLQSAVMSSessionGetInfo 1364
9.34.2.5 pack_swiavms_SLQSAvmsSetEventReport 1365
9.34.2.6 pack_swiavms_SLQSAVMSSetSettings 1365
9.34.2.7 pack_swiavms_SLQSAVMSSetSettings_v2 1366
9.34.2.8 pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField 1367
9.34.2.9 pack_swiavms_SLQSAVMSStartSession 1367
9.34.2.10 pack_swiavms_SLQSAVMSStopSession 1368
9.34.2.11 unpack_swiavms_SLQSAVMSEventReportInd 1368
9.34.2.12 unpack_swiavms_SLQSAVMSGetSettings 1369
9.34.2.13 unpack_swiavms_SLQSAVMSGetSettings_v2 1369
9.34.2.14 unpack_swiavms_SLQSAVMSSendSelection 1370
9.34.2.15 unpack_swiavms_SLQSAVMSSessionGetInfo 1370
9.34.2.16 unpack_swiavms_SLQSAvmsSetEventReport 1371
9.34.2.17 unpack_swiavms_SLQSAVMSSetSettings 1371
9.34.2.18 unpack_swiavms_SLQSAVMSSetSettings_v2 1372
9.34.2.19 unpack_swiavms_SLQSAVMSStartSession 1372
9.34.2.20 unpack_swiavms_SLQSAVMSStopSession 1373
9.34.2.21 unpack_swiavms_SLQSAVMSStopSession_avc2 1373
9.35 SwiDataTypes.h File Reference 1374
9.35.1 Detailed Description 1374
9.35.2 Macro Definition Documentation 1374
9.35.2.1 MEMSET_STATIC_OUTPUT_STRUCT 1374

9.35.2.2 QMI_NO_LTE_FW_SUPPORT . . . 1375
9.35.2.3 QMI_TLV_PLACEHOLDER . . . 1375
9.35.2.4 SWI_API . . . 1375
9.35.2.5 UNUSEDPARAM . . . 1375
9.35.3 Typedef Documentation . . . 1375
9.35.3.1 BOOL . . . 1375
9.35.3.2 BYTE . . . 1375
9.35.3.3 CHAR . . . 1375
9.35.3.4 FLOAT . . . 1375
9.35.3.5 INT32 . . . 1375
9.35.3.6 INT8 . . . 1375
9.35.3.7 LPCSTR . . . 1375
9.35.3.8 qmuint16 . . . 1375
9.35.3.9 qmulong . . . 1375
9.35.3.10 SHORT . . . 1375
9.35.3.11 ULONG . . . 1375
9.35.3.12 ULONGLONG . . . 1375
9.35.3.13 USHORT . . . 1375
9.35.3.14 WORD . . . 1375
9.36 swidms.h File Reference . . . 1375
9.36.1 Typedef Documentation . . . 1376
9.36.1.1 unpack_swidms_SLQSSwiDmsSetUsbNetNum_t . . . 1376
9.36.2 Function Documentation . . . 1376
9.36.2.1 pack_swidms_SLQSSwiDmsGetMTU . . . 1376
9.36.2.2 pack_swidms_SLQSSwiDmsGetUsbComp . . . 1377
9.36.2.3 pack_swidms_SLQSSwiDmsGetUsbNetNum . . . 1377
9.36.2.4 pack_swidms_SLQSSwiDmsSetMTU . . . 1377
9.36.2.5 pack_swidms_SLQSSwiDmsSetUsbComp . . . 1378
9.36.2.6 pack_swidms_SLQSSwiDmsSetUsbNetNum . . . 1378
9.36.2.7 unpack_swidms_SLQSSwiDmsGetMTU . . . 1378
9.36.2.8 unpack_swidms_SLQSSwiDmsGetUsbComp . . . 1379
9.36.2.9 unpack_swidms_SLQSSwiDmsGetUsbNetNum . . . 1379
9.36.2.10 unpack_swidms_SLQSSwiDmsSetMTU . . . 1380
9.36.2.11 unpack_swidms_SLQSSwiDmsSetUsbComp . . . 1380
9.36.2.12 unpack_swidms_SLQSSwiDmsSetUsbNetNum . . . 1380
9.37 swiloc.h File Reference . . . 1381
9.37.1 Typedef Documentation . . . 1381
9.37.1.1 unpack_swiloc_SwiLocSetAutoStart_t . . . 1381
9.37.2 Function Documentation . . . 1381
9.37.2.1 pack_swiloc_SwiLocGetAutoStart . . . 1381

9.37.2.2 pack_swiloc_SwiLocSetAutoStart . . . 1381
9.37.2.3 unpack_swiloc_SwiLocGetAutoStart . . . 1382
9.37.2.4 unpack_swiloc_SwiLocSetAutoStart . . . 1382
9.38 swioma.h File Reference . . . 1382
9.38.1 Macro Definition Documentation . . . 1384
9.38.1.1 LITEQMI_MAX_SWIOMA_STR_LEN . . . 1384
9.38.2 Typedef Documentation . . . 1384
9.38.2.1 unpack_swioma_SLQSOMADMAlertCallback_t . . . 1384
9.38.2.2 unpack_swioma_SLQSOMADMCancelSession_t . . . 1384
9.38.2.3 unpack_swioma_SLQSOMADMSendSelection_t . . . 1384
9.38.2.4 unpack_swioma_SLQSOMADMSetSettings_t . . . 1384
9.38.3 Function Documentation . . . 1384
9.38.3.1 pack_swioma_SLQSOMADMAlertCallback . . . 1384
9.38.3.2 pack_swioma_SLQSOMADMCancelSession . . . 1384
9.38.3.3 pack_swioma_SLQSOMADMGetSessionInfo . . . 1385
9.38.3.4 pack_swioma_SLQSOMADMGetSettings . . . 1386
9.38.3.5 pack_swioma_SLQSOMADMSendSelection . . . 1386
9.38.3.6 pack_swioma_SLQSOMADMSetSettings . . . 1387
9.38.3.7 pack_swioma_SLQSOMADMStartSession . . . 1388
9.38.3.8 unpack_swioma_SLQSOMADMAlertCallback . . . 1388
9.38.3.9 unpack_swioma_SLQSOMADMAlertCallback_ind . . . 1389
9.38.3.10 unpack_swioma_SLQSOMADMCancelSession . . . 1389
9.38.3.11 unpack_swioma_SLQSOMADMGetSessionInfo . . . 1390
9.38.3.12 unpack_swioma_SLQSOMADMGetSettings . . . 1390
9.38.3.13 unpack_swioma_SLQSOMADMSendSelection . . . 1391
9.38.3.14 unpack_swioma_SLQSOMADMSetSettings . . . 1391
9.38.3.15 unpack_swioma_SLQSOMADMStartSession . . . 1392
9.39 swiomaext.h File Reference . . . 1392
9.39.1 Macro Definition Documentation . . . 1393
9.39.1.1 LITEQMI_MAX_SWIOMA_STR_LEN . . . 1393
9.39.2 Typedef Documentation . . . 1393
9.39.2.1 unpack_swioma_SLQSOMADMCancelSessionExt_t . . . 1393
9.39.2.2 unpack_swioma_SLQSOMADMSendSelectionExt_t . . . 1393
9.39.2.3 unpack_swioma_SLQSOMADMSetSettingsExt_t . . . 1393
9.39.2.4 unpack_swioma_SLQSOMADMStartSessionExt_t . . . 1393
9.39.3 Function Documentation . . . 1393
9.39.3.1 pack_swioma_SLQSOMADMCancelSessionExt . . . 1393
9.39.3.2 pack_swioma_SLQSOMADMSendSelectionExt . . . 1394
9.39.3.3 pack_swioma_SLQSOMADMSetSettingsExt . . . 1394
9.39.3.4 pack_swioma_SLQSOMADMStartSessionExt . . . 1395

9.39.3.5 unpack_swioma_SLQSOMADMCancelSessionExt . . . 1396
9.39.3.6 unpack_swioma_SLQSOMADMSendSelectionExt . . . 1396
9.39.3.7 unpack_swioma_SLQSOMADMSetSettingsExt . . . 1396
9.39.3.8 unpack_swioma_SLQSOMADMStartSessionExt . . . 1397
9.40 switype_256bit.h File Reference . . . 1397
9.40.1 Macro Definition Documentation . . . 1398
9.40.1.1 SWI_UINT256_BIT_OFFSET . . . 1398
9.40.1.2 SWI_UINT256_BITS_PER_WORD . . . 1398
9.40.1.3 SWI_UINT256_INT_VALUE . . . 1398
9.40.1.4 SWI_UINT256_WORD_COUNT . . . 1398
9.40.1.5 SWI_UINT256_WORD_OFFSET . . . 1398
9.40.2 Function Documentation . . . 1398
9.40.2.1 swi_uint256_clear_bit . . . 1398
9.40.2.2 swi_uint256_get_bit . . . 1398
9.40.2.3 swi_uint256_print_mask . . . 1398
9.40.2.4 swi_uint256_set_bit . . . 1399
9.41 SWIWWANCMAPI.h File Reference . . . 1399
9.42 tmd.h File Reference . . . 1399
9.42.1 Macro Definition Documentation . . . 1400
9.42.1.1 MAX_MITIGATION_DEV_ID_LEN . . . 1400
9.42.1.2 MAX_MITIGATION_DEV_LIST_LEN . . . 1400
9.42.1.3 TMD_MAX_DEV_LIST . . . 1400
9.42.2 Function Documentation . . . 1400
9.42.2.1 pack_tmd_SLQSTmdDeRegNotMitigationLvl . . . 1400
9.42.2.2 pack_tmd_SLQSTmdGetMitigationDevList . . . 1400
9.42.2.3 pack_tmd_SLQSTmdGetMitigationLvl . . . 1400
9.42.2.4 pack_tmd_SLQSTmdRegNotMitigationLvl . . . 1401
9.42.2.5 unpack_tmd_SLQSTmdDeRegNotMitigationLvl . . . 1401
9.42.2.6 unpack_tmd_SLQSTmdGetMitigationDevList . . . 1402
9.42.2.7 unpack_tmd_SLQSTmdGetMitigationLvl . . . 1402
9.42.2.8 unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind . . . 1402
9.42.2.9 unpack_tmd_SLQSTmdRegNotMitigationLvl . . . 1403
9.43 uim.h File Reference . . . 1403
9.43.1 Macro Definition Documentation . . . 1406
9.43.1.1 MAX_DESCRIPTION_LENGTH . . . 1406
9.43.1.2 MAX_ICCID_LENGTH . . . 1406
9.43.1.3 MAX_NO_OF_APPLICATIONS . . . 1406
9.43.1.4 MAX_NO_OF_SLOTS . . . 1406
9.43.1.5 MAX_SLOTS_STATUS . . . 1406
9.43.1.6 UIM_MAX_ACTIVE_PERS_FEATURES . . . 1406

9.43.1.7 UIM_MAX_CONTENT_LENGTH . . . 1406
9.43.1.8 UIM_MAX_DESCRIPTION_LENGTH . . . 1406
9.43.1.9 UIM_MAX_NO_OF_APPLICATIONS . . . 1406
9.43.1.10 UIM_MAX_NO_OF_SLOTS . . . 1406
9.43.1.11 UIM_UINT8_MAX_STRING_SZ . . . 1406
9.43.2 Typedef Documentation . . . 1406
9.43.2.1 unpack_uim_SLQSUIMPowerDown_t . . . 1406
9.43.2.2 unpack_uim_SLQSUIMPowerUp_t . . . 1406
9.43.2.3 unpack_uim_SLQSUIMRefreshComplete_t . . . 1407
9.43.2.4 unpack_uim_SLQSUIMRefreshOK_t . . . 1407
9.43.2.5 unpack_uim_SLQSUIMRefreshRegister_t . . . 1407
9.43.2.6 unpack_uim_SLQSUIMReset_t . . . 1407
9.43.2.7 unpack_uim_SLQSUIMSwitchSlot_t . . . 1407
9.43.3 Function Documentation . . . 1407
9.43.3.1 pack_uim_ChangePin . . . 1407
9.43.3.2 pack_uim_GetCardStatus . . . 1407
9.43.3.3 pack_uim_ReadTransparent . . . 1407
9.43.3.4 pack_uim_SetPinProtection . . . 1408
9.43.3.5 pack_uim_SLQSUIMAuthenticate . . . 1408
9.43.3.6 pack_uim_SLQSUIMDepersonalization . . . 1409
9.43.3.7 pack_uim_SLQSUIMEventRegister . . . 1409
9.43.3.8 pack_uim_SLQSUIMGetConfiguration . . . 1409
9.43.3.9 pack_uim_SLQSUIMGetFileAttributes . . . 1410
9.43.3.10 pack_uim_SLQSUIMGetSlotsStatus . . . 1410
9.43.3.11 pack_uim_SLQSUIMPowerDown . . . 1410
9.43.3.12 pack_uim_SLQSUIMPowerUp . . . 1411
9.43.3.13 pack_uim_SLQSUIMRefreshComplete . . . 1411
9.43.3.14 pack_uim_SLQSUIMRefreshGetLastEvent . . . 1412
9.43.3.15 pack_uim_SLQSUIMRefreshOK . . . 1412
9.43.3.16 pack_uim_SLQSUIMRefreshRegister . . . 1412
9.43.3.17 pack_uim_SLQSUIMReset . . . 1413
9.43.3.18 pack_uim_SLQSUIMSwitchSlot . . . 1413
9.43.3.19 pack_uim_UnblockPin . . . 1413
9.43.3.20 pack_uim_VerifyPin . . . 1414
9.43.3.21 unpack_uim_ChangePin . . . 1414
9.43.3.22 unpack_uim_GetCardStatus . . . 1415
9.43.3.23 unpack_uim_ReadTransparent . . . 1415
9.43.3.24 unpack_uim_SetPinProtection . . . 1415
9.43.3.25 unpack_uim_SetUimSlotStatusChangeCallback_ind . . . 1416
9.43.3.26 unpack_uim_SLQSUIMAuthenticate . . . 1416

9.43.3.27 unpack_uim_SLQSUIMDepersonalization . . . 1416
9.43.3.28 unpack_uim_SLQSUIMEventRegister . . . 1417
9.43.3.29 unpack_uim_SLQSUIMGetConfiguration . . . 1417
9.43.3.30 unpack_uim_SLQSUIMGetFileAttributes . . . 1417
9.43.3.31 unpack_uim_SLQSUIMGetSlotsStatus . . . 1418
9.43.3.32 unpack_uim_SLQSUIMPowerDown . . . 1418
9.43.3.33 unpack_uim_SLQSUIMPowerUp . . . 1419
9.43.3.34 unpack_uim_SLQSUIMRefreshCallback_Ind . . . 1419
9.43.3.35 unpack_uim_SLQSUIMRefreshComplete . . . 1419
9.43.3.36 unpack_uim_SLQSUIMRefreshGetLastEvent . . . 1420
9.43.3.37 unpack_uim_SLQSUIMRefreshOK . . . 1420
9.43.3.38 unpack_uim_SLQSUIMRefreshRegister . . . 1420
9.43.3.39 unpack_uim_SLQSUIMReset . . . 1421
9.43.3.40 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind . . . 1421
9.43.3.41 unpack_uim_SLQSUIMSwitchSlot . . . 1421
9.43.3.42 unpack_uim_UnblockPin . . . 1422
9.43.3.43 unpack_uim_VerifyPin . . . 1422
9.44 voice.h File Reference . . . 1422
9.44.1 Detailed Description . . . 1428
9.44.2 S1 . . . 1428
9.44.3 Macro Definition Documentation . . . 1428
9.44.3.1 BARRING_PASSWORD_LENGTH . . . 1428
9.44.3.2 MAX_VOICE_CALL_NO_LEN . . . 1428
9.44.3.3 MAX_VOICE_DESCRIPTION_LENGTH . . . 1428
9.44.3.4 MAXVOICEUSSDLENGTH . . . 1428
9.44.3.5 VOICE_MAX_NO_OF_CALLS . . . 1428
9.44.4 Typedef Documentation . . . 1428
9.44.4.1 unpack_voice_AnswerUSSD_t . . . 1428
9.44.4.2 unpack_voice_CancelUSSD_t . . . 1428
9.44.4.3 unpack_voice_OriginateUSSD_t . . . 1428
9.44.4.4 unpack_voice_SLQSVoiceALSSelectLine_t . . . 1428
9.44.4.5 unpack_voice_SLQSVoiceALSSetLineSwitching_t . . . 1428
9.44.4.6 unpack_voice_SLQSVoiceBindSubscription_t . . . 1428
9.44.4.7 unpack_voice_SLQSVoiceIndicationRegister_t . . . 1428
9.44.4.8 unpack_voice_SLQSVoiceOrigUSSDNoWait_t . . . 1428
9.44.4.9 unpack_voice_SLQSVoiceSetPreferredPrivacy_t . . . 1428
9.44.5 Enumeration Type Documentation . . . 1428
9.44.5.1 liteServiceClassInformation . . . 1429
9.44.6 Function Documentation . . . 1429
9.44.6.1 pack_voice_AnswerUSSD . . . 1429

9.44.6.2 pack_voice_CancelUSSD . . . 1429
9.44.6.3 pack_voice_OriginateUSSD . . . 1430
9.44.6.4 pack_voice_SLQSOriginateUSSD . . . 1430
9.44.6.5 pack_voice_SLQSVoiceALSSelectLine . . . 1430
9.44.6.6 pack_voice_SLQSVoiceALSSetLineSwitching . . . 1431
9.44.6.7 pack_voice_SLQSVoiceAnswerCall . . . 1431
9.44.6.8 pack_voice_SLQSVoiceBindSubscription . . . 1431
9.44.6.9 pack_voice_SLQSVoiceBurstDTMF . . . 1432
9.44.6.10 pack_voice_SLQSVoiceDialCall . . . 1432
9.44.6.11 pack_voice_SLQSVoiceEndCall . . . 1433
9.44.6.12 pack_voice_SLQSVoiceGetAllCallInfo . . . 1433
9.44.6.13 pack_voice_SLQSVoiceGetCallBarring . . . 1433
9.44.6.14 pack_voice_SLQSVoiceGetCallForwardingStatus . . . 1434
9.44.6.15 pack_voice_SLQSVoiceGetCallInfo . . . 1434
9.44.6.16 pack_voice_SLQSVoiceGetCallWaiting . . . 1434
9.44.6.17 pack_voice_SLQSVoiceGetCLIP . . . 1435
9.44.6.18 pack_voice_SLQSVoiceGetCLIR . . . 1435
9.44.6.19 pack_voice_SLQSVoiceGetCNAP . . . 1435
9.44.6.20 pack_voice_SLQSVoiceGetCOLP . . . 1436
9.44.6.21 pack_voice_SLQSVoiceGetCOLR . . . 1436
9.44.6.22 pack_voice_SLQSVoiceGetConfig . . . 1436
9.44.6.23 pack_voice_SLQSVoiceIndicationRegister . . . 1437
9.44.6.24 pack_voice_SLQSVoiceManageCalls . . . 1437
9.44.6.25 pack_voice_SLQSVoiceOrigUSSDNoWait . . . 1437
9.44.6.26 pack_voice_SLQSVoiceSendFlash . . . 1438
9.44.6.27 pack_voice_SLQSVoiceSetCallBarringPassword . . . 1438
9.44.6.28 pack_voice_SLQSVoiceSetConfig . . . 1439
9.44.6.29 pack_voice_SLQSVoiceSetPreferredPrivacy . . . 1439
9.44.6.30 pack_voice_SLQSVoiceSetSUPSService . . . 1439
9.44.6.31 pack_voice_SLQSVoiceStartContDTMF . . . 1440
9.44.6.32 pack_voice_SLQSVoiceStopContDTMF . . . 1440
9.44.6.33 unpack_voice_allCallStatusCallback_ind . . . 1440
9.44.6.34 unpack_voice_AnswerUSSD . . . 1441
9.44.6.35 unpack_voice_CancelUSSD . . . 1441
9.44.6.36 unpack_voice_DTMFEventCallback_ind . . . 1441
9.44.6.37 unpack_voice_OriginateUSSD . . . 1442
9.44.6.38 unpack_voice_OTASPStatusCallback_ind . . . 1442
9.44.6.39 unpack_voice_SLQSOriginateUSSD . . . 1443
9.44.6.40 unpack_voice_SLQSVoiceALSSelectLine . . . 1443
9.44.6.41 unpack_voice_SLQSVoiceALSSetLineSwitching . . . 1443

9.44.6.42 unpack_voice_SLQSVoiceAnswerCall . . . 1444
9.44.6.43 unpack_voice_SLQSVoiceBindSubscription . . . 1444
9.44.6.44 unpack_voice_SLQSVoiceBurstDTMF . . . 1444
9.44.6.45 unpack_voice_SLQSVoiceDialCall . . . 1445
9.44.6.46 unpack_voice_SLQSVoiceEndCall . . . 1445
9.44.6.47 unpack_voice_SLQSVoiceGetAllCallInfo . . . 1445
9.44.6.48 unpack_voice_SLQSVoiceGetCallBarring . . . 1446
9.44.6.49 unpack_voice_SLQSVoiceGetCallForwardingStatus . . . 1446
9.44.6.50 unpack_voice_SLQSVoiceGetCallInfo . . . 1446
9.44.6.51 unpack_voice_SLQSVoiceGetCallWaiting . . . 1447
9.44.6.52 unpack_voice_SLQSVoiceGetCLIP . . . 1447
9.44.6.53 unpack_voice_SLQSVoiceGetCLIR . . . 1447
9.44.6.54 unpack_voice_SLQSVoiceGetCNAP . . . 1448
9.44.6.55 unpack_voice_SLQSVoiceGetCOLP . . . 1448
9.44.6.56 unpack_voice_SLQSVoiceGetCOLR . . . 1448
9.44.6.57 unpack_voice_SLQSVoiceGetConfig . . . 1449
9.44.6.58 unpack_voice_SLQSVoiceIndicationRegister . . . 1449
9.44.6.59 unpack_voice_SLQSVoiceManageCalls . . . 1449
9.44.6.60 unpack_voice_SLQSVoiceOrigUSSDNoWait . . . 1450
9.44.6.61 unpack_voice_SLQSVoiceSendFlash . . . 1450
9.44.6.62 unpack_voice_SLQSVoiceSetCallBarringPassword . . . 1450
9.44.6.63 unpack_voice_SLQSVoiceSetConfig . . . 1451
9.44.6.64 unpack_voice_SLQSVoiceSetPreferredPrivacy . . . 1451
9.44.6.65 unpack_voice_SLQSVoiceSetSUPSService . . . 1451
9.44.6.66 unpack_voice_SLQSVoiceStartContDTMF . . . 1452
9.44.6.67 unpack_voice_SLQSVoiceStopContDTMF . . . 1452
9.44.6.68 unpack_voice_SLQSVoiceSUPSCallback_ind . . . 1452
9.44.6.69 unpack_voice_SUPSNotificationCallback_ind . . . 1453
9.44.6.70 unpack_voice_USSDNotificationCallback_ind . . . 1453
9.44.6.71 unpack_voice_VoiceInfoRecCallback_ind . . . 1454
9.44.6.72 unpack_voice_voicePrivacyChangeCallback_ind . . . 1454
9.45 wds.h File Reference . . . 1454
9.45.1 Detailed Description . . . 1461
9.45.2 SO Mask . . . 1461
9.45.3 RAT Mask . . . 1461
9.45.4 Macro Definition Documentation . . . 1462
9.45.4.1 BYT_STAT_STAT_MASK . . . 1462
9.45.4.2 IPV6_ADDRESS_ARRAY_SIZE . . . 1462
9.45.4.3 MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE . . . 1462
9.45.4.4 PACK_WDS_IPV4 . . . 1462

9.45.4.5 PACK_WDS_IPV6 . . . 1462
9.45.4.6 WDS_DHCP_MAX_NUM_OPTIONS . . . 1462
9.45.4.7 WDS_DHCP_OPTION_DATA_BUF_SIZE . . . 1462
9.45.4.8 WDS_PROFILE_3GPP . . . 1462
9.45.4.9 WDS_PROFILE_3GPP2 . . . 1462
9.45.4.10 WDS_TFTID_SOURCE_IP_SIZE . . . 1462
9.45.5 Typedef Documentation . . . 1462
9.45.5.1 unpack_wds_DHCPv4ClientLeaseChange_t . . . 1462
9.45.5.2 unpack_wds_SetAutoconnect_t . . . 1462
9.45.5.3 unpack_wds_SetDefaultProfile_t . . . 1462
9.45.5.4 unpack_wds_SetDefaultProfileNum_t . . . 1462
9.45.5.5 unpack_wds_SetMobileIP_t . . . 1462
9.45.5.6 unpack_wds_SetMobileIPParameters_t . . . 1462
9.45.5.7 unpack_wds_SetMuxID_t . . . 1463
9.45.5.8 unpack_wds_SLQSResetPacketStatics_t . . . 1463
9.45.5.9 unpack_wds_SLQSSet3GPPConfigItem_t . . . 1463
9.45.5.10 unpack_wds_SLQSSetWdsEventCallback_t . . . 1463
9.45.5.11 unpack_wds_SLQSSSetDHCPv4ClientConfig_t . . . 1463
9.45.5.12 unpack_wds_SLQSSSetLoopback_t . . . 1463
9.45.5.13 unpack_wds_SLQSStopDataSession_t . . . 1463
9.45.5.14 unpack_wds_SLQSWdsGoActive_t . . . 1463
9.45.5.15 unpack_wds_SLQSWdsGoDormant_t . . . 1463
9.45.5.16 unpack_wds_SLQSWdsSetEventReport_t . . . 1463
9.45.5.17 UnpackQmiProfileInfo . . . 1463
9.45.6 Enumeration Type Documentation . . . 1463
9.45.6.1 liteQmiDataBearerMasks . . . 1463
9.45.7 Function Documentation . . . 1463
9.45.7.1 pack_wds_DHCPv4ClientLeaseChange . . . 1463
9.45.7.2 pack_wds_GetAutoconnect . . . 1464
9.45.7.3 pack_wds_GetByteTotals . . . 1464
9.45.7.4 pack_wds_GetConnectionRate . . . 1465
9.45.7.5 pack_wds_GetDataBearerTechnology . . . 1465
9.45.7.6 pack_wds_GetDefaultProfile . . . 1466
9.45.7.7 pack_wds_GetDefaultProfileNum . . . 1466
9.45.7.8 pack_wds_GetDormancyState . . . 1466
9.45.7.9 pack_wds_GetLastMobileIPError . . . 1467
9.45.7.10 pack_wds_GetMobileIP . . . 1467
9.45.7.11 pack_wds_GetMobileIPProfile . . . 1468
9.45.7.12 pack_wds_GetPacketStatistics . . . 1468
9.45.7.13 pack_wds_GetPacketStatus . . . 1469

9.45.7.14 pack_wds_GetSessionDuration . . . 1469
9.45.7.15 pack_wds_GetSessionState . . . 1469
9.45.7.16 pack_wds_RMSetTransferStatistics . . . 1470
9.45.7.17 pack_wds_SetAutoconnect . . . 1470
9.45.7.18 pack_wds_SetDefaultProfile . . . 1471
9.45.7.19 pack_wds_SetDefaultProfileNum . . . 1471
9.45.7.20 pack_wds_SetMobileIP . . . 1471
9.45.7.21 pack_wds_SetMobileIPParameters . . . 1472
9.45.7.22 pack_wds_SetMobileIPProfile . . . 1472
9.45.7.23 pack_wds_SetMuxID . . . 1473
9.45.7.24 pack_wds_SLQSCreateProfile . . . 1473
9.45.7.25 pack_wds_SLQSDeleteProfile . . . 1474
9.45.7.26 pack_wds_SLQSGet3GPPConfigItem . . . 1474
9.45.7.27 pack_wds_SLQSGetCurrDataSystemStat . . . 1474
9.45.7.28 pack_wds_SLQSGetCurrentChannelRate . . . 1475
9.45.7.29 pack_wds_SLQSGetDataBearerTechnology . . . 1475
9.45.7.30 pack_wds_SLQSGetDUNCallInfo . . . 1476
9.45.7.31 pack_wds_SLQSGetProfileSettings . . . 1476
9.45.7.32 pack_wds_SLQSGetRuntimeSettings . . . 1477
9.45.7.33 pack_wds_SLQSModifyProfile . . . 1477
9.45.7.34 pack_wds_SLQSResetPacketStatics . . . 1477
9.45.7.35 pack_wds_SLQSSet3GPPConfigItem . . . 1478
9.45.7.36 pack_wds_SLQSSetIPFamilyPreference . . . 1478
9.45.7.37 pack_wds_SLQSSetWdsEventCallback . . . 1479
9.45.7.38 pack_wds_SLQSSGetDHCPv4ClientConfig . . . 1479
9.45.7.39 pack_wds_SLQSSGetLoopback . . . 1479
9.45.7.40 pack_wds_SLQSSSetDHCPv4ClientConfig . . . 1480
9.45.7.41 pack_wds_SLQSSSetLoopback . . . 1480
9.45.7.42 pack_wds_SLQSStartDataSession . . . 1481
9.45.7.43 pack_wds_SLQSStopDataSession . . . 1481
9.45.7.44 pack_wds_SLQSWdsGoActive . . . 1481
9.45.7.45 pack_wds_SLQSWdsGoDormant . . . 1482
9.45.7.46 pack_wds_SLQSWdsSetEventReport . . . 1482
9.45.7.47 pack_wds_SLQSWdsSwiPDPRuntimeSettings . . . 1483
9.45.7.48 unpack_wds_DHCPv4ClientLease_ind . . . 1483
9.45.7.49 unpack_wds_DHCPv4ClientLeaseChange . . . 1483
9.45.7.50 unpack_wds_GetAutoconnect . . . 1484
9.45.7.51 unpack_wds_GetByteTotals . . . 1484
9.45.7.52 unpack_wds_GetConnectionRate . . . 1484
9.45.7.53 unpack_wds_GetDataBearerTechnology . . . 1485

9.45.7.54 unpack_wds_GetDefaultProfile . . . 1485
9.45.7.55 unpack_wds_GetDefaultProfileNum . . . 1486
9.45.7.56 unpack_wds_GetDormancyState . . . 1486
9.45.7.57 unpack_wds_GetLastMobileIPError . . . 1486
9.45.7.58 unpack_wds_GetMobileIP . . . 1487
9.45.7.59 unpack_wds_GetMobileIPProfile . . . 1487
9.45.7.60 unpack_wds_GetPacketStatistics . . . 1487
9.45.7.61 unpack_wds_GetPacketStatus . . . 1488
9.45.7.62 unpack_wds_GetSessionDuration . . . 1488
9.45.7.63 unpack_wds_GetSessionState . . . 1488
9.45.7.64 unpack_wds_RMSetTransferStatistics . . . 1489
9.45.7.65 unpack_wds_RMTransferStatistics_ind . . . 1489
9.45.7.66 unpack_wds_SetAutoconnect . . . 1489
9.45.7.67 unpack_wds_SetDefaultProfile . . . 1490
9.45.7.68 unpack_wds_SetDefaultProfileNum . . . 1490
9.45.7.69 unpack_wds_SetMobileIP . . . 1490
9.45.7.70 unpack_wds_SetMobileIPParameters . . . 1491
9.45.7.71 unpack_wds_SetMobileIPProfile . . . 1491
9.45.7.72 unpack_wds_SetMuxID . . . 1491
9.45.7.73 unpack_wds_SLQSCreateProfile . . . 1492
9.45.7.74 unpack_wds_SLQSDeleteProfile . . . 1492
9.45.7.75 unpack_wds_SLQSDUNCallInfoCallBack_ind . . . 1492
9.45.7.76 unpack_wds_SLQSGet3GPPConfigItem . . . 1493
9.45.7.77 unpack_wds_SLQSGetCurrDataSystemStat . . . 1493
9.45.7.78 unpack_wds_SLQSGetCurrentChannelRate . . . 1493
9.45.7.79 unpack_wds_SLQSGetDataBearerTechnology . . . 1494
9.45.7.80 unpack_wds_SLQSGetDUNCallInfo . . . 1494
9.45.7.81 unpack_wds_SLQSGetProfileSettings . . . 1494
9.45.7.82 unpack_wds_SLQSGetRuntimeSettings . . . 1495
9.45.7.83 unpack_wds_SLQSModifyProfile . . . 1495
9.45.7.84 unpack_wds_SLQSResetPacketStatics . . . 1495
9.45.7.85 unpack_wds_SLQSSet3GPPConfigItem . . . 1496
9.45.7.86 unpack_wds_SLQSSetIPFamilyPreference . . . 1496
9.45.7.87 unpack_wds_SLQSSetPacketSrvStatusCallback . . . 1496
9.45.7.88 unpack_wds_SLQSSetWdsEventCallback . . . 1497
9.45.7.89 unpack_wds_SLQSSetWdsEventCallback_ind . . . 1497
9.45.7.90 unpack_wds_SLQSSGetDHCPv4ClientConfig . . . 1497
9.45.7.91 unpack_wds_SLQSSGetLoopback . . . 1498
9.45.7.92 unpack_wds_SLQSSSetDHCPv4ClientConfig . . . 1498
9.45.7.93 unpack_wds_SLQSSSetLoopback . . . 1498

9.45.7.94 unpack_wds_SLQSStartDataSession 1499
9.45.7.95 unpack_wds_SLQSStopDataSession 1499
9.45.7.96 unpack_wds_SLQSWdsGoActive 1499
9.45.7.97 unpack_wds_SLQSWdsGoDormant 1500
9.45.7.98 unpack_wds_SLQSWdsSetEventReport 1500
9.45.7.99 unpack_wds_SLQSWdsSwiPDPRuntimeSettings 1500

Index **1502**

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) 29
Streaming Download Protocol (lite-fw) 30

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Tables 31

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

_litefw_FirmwareInfo_ . . . 33
_litefw_FirmwarePartNo_ . . . 34
altSrcInfo_t . . . 34
appStats . . . 35
audio_RXAGCList . . . 38
audio_RXAVCList . . . 39
audio_RXPCMIIRFltr . . . 40
audio_TXAGCList . . . 41
audio_TXPCMIIRFltr . . . 42
CarrierImage_t . . . 44
cat_AlPhaIdentifierTlv . . . 45
cat_commonEventTlv . . . 46
cat_currentCatEvent . . . 46
cat_EndProactiveSessionTlv . . . 47
cat_EventIDDataTlv . . . 48
cat_EventListTlv . . . 48
cat_RefreshTlv . . . 49
cdmaSSInfo . . . 49
connectionStatus . . . 50
crashInfoParams . . . 50
crashInformation . . . 51
currNetworkInfo . . . 52
dms_ActivationStatusTlv . . . 52
dms_OperatingModeTlv . . . 53
dms_PSMActiveTimerIndTlv . . . 54
dms_PSMActiveTimerTlv . . . 54
dms_PSMDurationDueToOOSTlv . . . 55
dms_PSMDurationThresholdTlv . . . 55
dms_PSMEarlyWakeupTimeTlv . . . 56
dms_PSMEnableStateIndTlv . . . 56
dms_PSMEnableStateTlv . . . 57
dms_PSMPeriodicUpdateTimerIndTlv . . . 57
dms_PSMPeriodicUpdateTimerTlv . . . 58
dms_PSMRandomizationWindowTlv . . . 58
dms_TemperatureTlv . . . 59
dms_UimAutoSwitchActSlotTlv . . . 60
dms_UimStatusTlv . . . 60
dms_VoltageTlv . . . 61

DMScustSettingInfo . . . 62
DMScustSettingList . . . 63
DMSgetCustomFeatureV2 . . . 63
DMSgetCustomInput . . . 64
dunchannelRate . . . 64
eriDataparams . . . 65
eTWSPLMNInfoTlv . . . 66
FMSImageElement . . . 66
FMSImageIdElement . . . 67
FMSImageIDEntries . . . 68
FMSImageList . . . 69
FMSPrefImageList . . . 69
hdrSSInfo . . . 70
image_info_t . . . 71
ims_AMRModeInfo . . . 71
ims_AMROctAlgnInfo . . . 72
ims_AMRWBModeInfo . . . 73
ims_AMRWBOctAlgnInfo . . . 73
ims_CSCFPortNameInfo . . . 74
ims_EnabAMRWBInfo . . . 74
ims_EnabSCRAMRInfo . . . 75
ims_EnabSCRAMRWBInfo . . . 75
ims_IMSDomainInfo . . . 76
ims_IMSTestModeInfo . . . 76
ims_MinSessExpInfo . . . 76
ims_PCSCFPortInfo . . . 77
ims_PhCtxtURIInfo . . . 77
ims_RngBkTmrInfo . . . 78
ims_RngTmrInfo . . . 78
ims_RTPRTCPInactTmrDurInfo . . . 79
ims_SessDurInfo . . . 79
ims_SigCompEnInfo . . . 80
ims_SIPPortInfo . . . 80
ims_SIPRegnTmrInfo . . . 81
ims_SMSFmtInfo . . . 81
ims_SMSoIPNwInfo . . . 82
ims_SubscrTmrInfo . . . 82
ims_TmrT1Info . . . 83
ims_TmrT2Info . . . 83
ims_TmrTfInfo . . . 84
imsa_IMSFailErrCodeTlv . . . 84
imsa_IMSRegStatusErrorCodeInfo . . . 85
imsa_IMSRegStatusInfo . . . 85
imsa_NewIMSRegStatusInfo . . . 86
imsa_RatHandoverStatusInfo . . . 86
imsa_SmsRatInfo . . . 87
imsa_SmsSvcStatusInfo . . . 88
imsa_UtRatInfo . . . 88
imsa_UtSvcStatusInfo . . . 89
imsa_VoipRatInfo . . . 90
imsa_VoipSvcStatusInfo . . . 90
imsa_VtRatInfo . . . 91
imsa_VtSvcStatusInfo . . . 91
ipv6AddressInfo . . . 92
LibPackGPRSRequestedQoS . . . 92
LibpackProfile3GPP . . . 93
LibpackProfile3GPP2 . . . 100
LibPackprofile_3GPP . . . 106

LibPackprofile_3GPP2 . . . 112
LibPackQosClassID . . . 117
LibPackTFTIDParams . . . 118
LibPackUMTSQoS . . . 120
LibPackUMTSReqQoSSigInd . . . 123
loc_accelAcceptReady . . . 123
loc_accelTempAcceptReady . . . 124
loc_BdsSV . . . 125
loc_BdsSVInfo . . . 125
loc_CellDb . . . 126
loc_ClkInfo . . . 126
loc_GnssData . . . 128
loc_gpsTime . . . 129
loc_gyroAcceptReady . . . 130
loc_gyroTempAcceptReady . . . 131
loc_IPv4Config . . . 132
loc_IPv4Info . . . 132
loc_IPv6Config . . . 133
loc_IPv6Info . . . 133
loc_LocApplicationInfo . . . 134
loc_precisionDilution . . . 135
loc_satelliteInfo . . . 136
loc_sensorDataUsage . . . 138
loc_SV . . . 139
loc_SVInfo . . . 140
loc_svUsedforFix . . . 140
loc_urlAddr . . . 141
loc_URLAddrInfo . . . 141
lteSSInfo . . . 142
messageModeTlv . . . 143
nas_acqOrderPref . . . 143
nas_AcqOrderPrefTlv . . . 144
nas_ActPilotPNElement . . . 145
nas_AddCDMASysInfo . . . 145
nas_AddSysInfo . . . 146
nas_BandPrefInfoTlv . . . 146
nas_BandPrefTlv . . . 147
nas_CallBarringSysInfo . . . 149
nas_callBarStatus . . . 149
nas_CDMAChannel . . . 150
nas_CDMAECIOThresh . . . 151
nas_CDMAInfo . . . 152
nas_CDMARSSIThresh . . . 153
nas_CDMASysInfo . . . 153
nas_CDMASysInfoExt . . . 157
nas_cellParams . . . 157
nas_ciotAcqOrderPref . . . 158
nas_CiotAcqOrderPrefTlv . . . 159
nas_CiotLteOpModePrefTlv . . . 160
nas_CommInfo . . . 160
nas_CSGID . . . 162
nas_CsgId . . . 163
nas_currentPLMN . . . 163
nas_dataSrvCapabilities . . . 164
nas_DataStatusDetail . . . 165
nas_detailSvcInfo . . . 167
nas_DeviceConfigDetail . . . 168
nas_dirNum . . . 169

nas_DRCParams . . . 169
nas_ecioListElement . . . 170
nas_ECIOThresh . . . 171
nas_EdrxCiotLteMode . . . 171
nas_EdrxCycleLength . . . 172
nas_EdrxEnableType . . . 172
nas_EdrxPagingTimeWindow . . . 173
nas_EdrxRatType . . . 173
nas_EmerModeTlv . . . 174
nas_errorRateListElement . . . 175
nas_ForbiddenNetworks3GPP . . . 176
nas_GERANInfo . . . 177
nas_geranInstInfo . . . 178
nas_gsmCellInfo . . . 179
nas_GSMRSSIThresh . . . 180
nas_GSMSrvStatusInfo . . . 181
nas_GSMSysInfo . . . 182
nas_GWAcqOrderPrefTlv . . . 185
nas_HDRECIOThresh . . . 185
nas_HDRIOThresh . . . 186
nas_HDRPersonality_Ind_Data . . . 186
nas_HDRRSSIThresh . . . 187
nas_HDRSINRThresh . . . 188
nas_HDRSINRThreshold . . . 188
nas_HDRSysInfo . . . 189
nas_homeSIDNID . . . 191
nas_infoInterFreq . . . 191
nas_IOThresh . . . 193
nas_lteBandPrefExt . . . 193
nas_LTEBandPrefTlv . . . 194
nas_LteCiotOpModeTlv . . . 196
nas_lteEARFCN . . . 196
nas_lteGsmCellInfo . . . 197
nas_LTEInfo . . . 198
nas_LTEInfoInterfreq . . . 201
nas_LTEInfoIntrafreq . . . 201
nas_LTEInfoNeighboringGSM . . . 203
nas_LTEInfoNeighboringWCDMA . . . 204
nas_LteM1BandPrefTlv . . . 205
nas_LteNb1BandPrefTlv . . . 207
nas_LTEOperationMode . . . 209
nas_LteOpMode . . . 209
nas_lteOpModeTlv . . . 210
nas_ltePCI . . . 211
nas_lteRsrpinformation . . . 211
nas_LTERSRPThresh . . . 212
nas_LTERSRQThresh . . . 212
nas_LTERSSIThresh . . . 213
nas_LTESigRptCfg . . . 214
nas_LTESigRptConfig . . . 214
nas_lteSnrinformation . . . 215
nas_LTESNRThresh . . . 216
nas_LTESNRThreshold . . . 216
nas_LTESysInfo . . . 217
nas_lteWcdmaCellInfo . . . 219
nas_minBasedIMSI . . . 220
nas_MNRInfo . . . 221
nas_ModePrefTlv . . . 222

nas_namName . . . 222
nas_netSelectionPref . . . 223
nas_NetSelPrefTlv . . . 224
nas_NetworkStat1x . . . 224
nas_NetworkStatEVDO . . . 226
nas_nmrCellInfo . . . 228
nas_nr5gBandPref . . . 229
nas_NR5GCellStatusTlv . . . 230
nas_NR5GSerStatTlv . . . 231
nas_NR5GSystemInfoTlv . . . 232
nas_operatorNameString . . . 235
nas_OperatorPLMNData . . . 235
nas_operatorPLMNList . . . 236
nas_PhyCaAggPcellInfo . . . 237
nas_PhyCaAggScellDlBw . . . 238
nas_PhyCaAggScellIndex . . . 238
nas_PhyCaAggScellIndType . . . 239
nas_PhyCaAggScellInfo . . . 240
nas_PilotSetData . . . 242
nas_PilotSetParams . . . 243
nas_PlmnID . . . 243
nas_PLMNNetworkName . . . 244
nas_PLMNNetworkNameData . . . 244
nas_PRLPrefTlv . . . 246
nas_protocolSubtypeElement . . . 247
nas_qaQmi3Gpp2TimeZone . . . 248
nas_QmiNas3GppNetworkInfo . . . 249
nas_QmiNas3GppNetworkRAT . . . 250
nas_QmisNasPcsDigit . . . 251
nas_QmisNasSlqsNasPCICellInfo . . . 252
nas_QmisNasSlqsNasPCIInfo . . . 253
nas_RankIndicatorTlv . . . 254
nas_RatDisabledMaskTlv . . . 254
nas_RejectReasonTlv . . . 255
nas_RFBandInfoElements . . . 255
nas_RfBandInfoExtFormat . . . 256
nas_RfBandInfoExtFormatElements . . . 257
nas_RfBandInfoList . . . 258
nas_RfBandwidthInfo . . . 258
nas_RfBandwidthInfoElements . . . 259
nas_RfDedicatedBandInfo . . . 260
nas_RfDedicatedBandInfoElements . . . 260
nas_RFInfoTlv . . . 261
nas_roamIndList . . . 262
nas_RoamPrefTlv . . . 263
nas_RSRPThresh . . . 263
nas_rsrqInformation . . . 264
nas_RSRQThresh . . . 264
nas_RSSIThresh . . . 265
nas_rxInfo . . . 266
nas_RxSigInfo . . . 267
nas_rxSignalStrengthListElement . . . 268
nas_SccRxInfo . . . 269
nas_serviceProviderName . . . 270
nas_servSystem . . . 271
nas_sidNid . . . 272
nas_SignalStrengthTlv . . . 273
nas_SLQSSignalStrengthsIndReq . . . 273

nas_SLQSSignalStrengthsInformation . . . 275
nas_SLQSSignalStrengthsTlv . . . 276
nas_SrvDomainPrefTlv . . . 277
nas_SrvStatusInfo . . . 277
nas_sysInfoCommon . . . 278
nas_TDSCDMAECIOThresh . . . 281
nas_TDSCDMARSCPThresh . . . 281
nas_TDSCDMARSSIThresh . . . 282
nas_TDSCDMASINRCONFThresh . . . 282
nas_TDSCDMASINRThresh . . . 283
nas_timeInfo . . . 283
nas_trueIMSI . . . 285
nas_txInfo . . . 286
nas_UMTSInfo . . . 286
nas_UMTSinstInfo . . . 288
nas_umtsLTENbrCell . . . 289
nas_UniversalTime . . . 290
nas_wcdmaCellInfo . . . 291
nas_WCDMAECIOThresh . . . 292
nas_WCDMAInfoLTENeighborCell . . . 292
nas_WCDMARSSIThresh . . . 293
nas_WCDMASysInfo . . . 294
nas_wcdmaUARFCN . . . 297
NASAcqOrderPrefTlv . . . 298
NASBandPreferenceTlv . . . 298
NASCiotAcqOrderPrefTlv . . . 299
NASCiotLteOpModePrefTlv . . . 299
NASEmergencyModeTlv . . . 299
NasGetLTECphyCaInfo . . . 300
NASGWAcqOrderPrefTlv . . . 301
NASLTEBandPreferenceTlv . . . 301
NASLteM1BandPrefTlv . . . 302
NASLteNasReleaseInfoTlv . . . 302
NASLteNB1BandPrefTlv . . . 303
NASModePreferenceTlv . . . 303
NASNetSelPreferenceTlv . . . 304
NASNr5gBandPrefTlv . . . 304
NASOTAMessageTlv . . . 305
NASPhyCaAggPcellInfo . . . 306
NASPhyCaAggScellArray . . . 307
NASPhyCaAggScellDlBw . . . 308
NASPhyCaAggScellIndex . . . 308
NASPhyCaAggScellIndType . . . 309
NASPhyCaAggScellInfo . . . 310
NASPRLPreferenceTlv . . . 311
NASQmiCbkNasSwiOTAMessageInd . . . 311
NASQmiCbkNasSystemSelPrefInd . . . 312
NASRatDisabledMaskTlv . . . 314
NASRoamPreferenceTlv . . . 315
NASServDomainPrefTlv . . . 315
NASServingSystemInfo . . . 315
NASTimeInfoTlv . . . 317
newMTMessageTlv . . . 318
pack_audio_SLQSGetAudioPathConfig_t . . . 318
pack_audio_SLQSGetAudioProfile_t . . . 319
pack_audio_SLQSGetAudioVolTLBConfig_t . . . 320
pack_audio_SLQSSetAudioPathConfig_t . . . 320
pack_audio_SLQSSetAudioProfile_t . . . 322

pack_audio_SLQSSetAudioVolTLBConfig_t . . . 324
pack_cat_CATSendEnvelopeCommand_t . . . 324
pack_cat_CATSendTerminalResponse_t . . . 325
pack_cat_SetCATEventCallback_t . . . 326
pack_dms_ActivateAutomatic_t . . . 327
pack_dms_GetCustFeaturesV2_t . . . 327
pack_dms_ResetToFactoryDefaults_t . . . 328
pack_dms_SetActivationStatusCallback_t . . . 328
pack_dms_SetCrashAction_t . . . 328
pack_dms_SetCustFeature_t . . . 329
pack_dms_SetCustFeaturesV2_t . . . 331
pack_dms_SetEventReport_t . . . 332
pack_dms_SetIndicationRegister_t . . . 332
pack_dms_SetPower_t . . . 333
pack_dms_SetUSBComp_t . . . 334
pack_dms_SLQSDmsSwiIndicationRegister_t . . . 335
pack_dms_SLQSSetPowerSaveModeConfig_t . . . 335
pack_dms_SLQSSwiGetCrashInfo_t . . . 336
pack_dms_SLQSSwiSetDyingGaspCfg_t . . . 336
pack_dms_SLQSSwiSetHostDevInfo_t . . . 337
pack_dms_SLQSSwiSetOSInfo_t . . . 338
pack_dms_SwiSetEventReport_t . . . 338
pack_dms_SwiUimSelect_t . . . 339
pack_dms_UIMChangePIN_t . . . 340
pack_dms_UIMGetControlKeyStatus_t . . . 340
pack_dms_UIMGetICCID_t . . . 341
pack_dms_UIMSetControlKeyProtection_t . . . 341
pack_dms_UIMSetPINProtection_t . . . 342
pack_dms_UIMUnblockControlKey_t . . . 343
pack_dms_UIMUnblockPIN_t . . . 343
pack_dms_UIMVerifyPIN_t . . . 344
pack_fms_GetImagesPreference_t . . . 344
pack_fms_GetStoredImages_t . . . 345
pack_fms_SetImagesPreference_t . . . 345
pack_ims_SLQSImsConfigIndicationRegister_t . . . 346
pack_ims_SLQSSetIMSSMSConfig_t . . . 347
pack_ims_SLQSSetIMSUserConfig_t . . . 348
pack_ims_SLQSSetIMSVoIPConfig_t . . . 349
pack_ims_SLQSSetRegMgrConfig_t . . . 351
pack_ims_SLQSSetSIPConfig_t . . . 352
pack_imsa_SLQSRegisterIMSAIndication_t . . . 353
pack_loc_Delete_Assist_Data_t . . . 354
pack_loc_EventRegister_t . . . 355
pack_loc_SetExtPowerState_t . . . 357
pack_loc_SetOperationMode_t . . . 357
pack_loc_SLQSLOCGetBestAvailPos_t . . . 358
pack_loc_SLQSLOCGetServer_t . . . 359
pack_loc_SLQSLOCInjectPosition_t . . . 359
pack_loc_SLQSLOCInjectSensorData_t . . . 364
pack_loc_SLQSLOCInjectUTCTime_t . . . 366
pack_loc_SLQSLOCSetCradleMountConfig_t . . . 366
pack_loc_SLQSLOCSetServer_t . . . 367
pack_loc_Start_t . . . 368
pack_loc_Stop_t . . . 370
pack_nas_InitiateDomainAttach_t . . . 370
pack_nas_PerformNetworkScanPCI_t . . . 371
pack_nas_SetACCOLC_t . . . 371
pack_nas_SetCDMANetworkParameters_t . . . 371

pack_nas_SetNetworkPreference_t . . . 373
pack_nas_SLQSConfigSigInfo_t . . . 374
pack_nas_SLQSGetPLMNName_t . . . 375
pack_nas_SLQSInitiateNetworkRegistration_t . . . 376
pack_nas_SLQSNasConfigSigInfo2_t . . . 377
pack_nas_SLQSNasGet3GPP2Subscription_t . . . 382
pack_nas_SLQSNASGeteDRXParamsExt_t . . . 382
pack_nas_SLQSNasGetTxRxInfo_t . . . 383
pack_nas_SLQSNasIndicationRegisterExt_t . . . 384
pack_nas_SLQSNASSeteDRXParams_t . . . 387
pack_nas_SLQSNasSwiIndicationRegister_t . . . 388
pack_nas_SLQSNASSwiSetChannelLock_t . . . 390
pack_nas_SLQSSetBandPreference_t . . . 390
pack_nas_SLQSSetSignalStrengthsCallback_t . . . 392
pack_nas_SLQSSetSysSelectionPref_t . . . 392
pack_nas_SLQSSetSysSelectionPrefExt_t . . . 398
pack_nas_SLQSSwiPSDetach_t . . . 408
pack_pds_PDSInjectTimeReference_t . . . 408
pack_pds_ResetPDSData_t . . . 409
pack_pds_SetEventReportCallback_t . . . 410
pack_pds_SetPDSDefaults_t . . . 410
pack_pds_SetPDSState_t . . . 411
pack_pds_SetPortAutomaticTracking_t . . . 411
pack_pds_SetServiceAutomaticTracking_t . . . 412
pack_pds_SetXTRAAutomaticDownload_t . . . 412
pack_pds_SetXTRANetwork_t . . . 413
pack_pds_SLQSGetAGPSConfig_t . . . 413
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t . . . 414
pack_pds_SLQSPDSInjectPositionData_t . . . 415
pack_pds_SLQSSetAGPSConfig_t . . . 417
pack_pds_SLQSSetPositionMethodState_t . . . 417
pack_pds_StartPDSTrackingSessionExt_t . . . 418
pack_qmi_t . . . 419
pack_qos_BindDataPort_t . . . 420
pack_qos_SLQSQosSwiReadApnExtraParams_t . . . 421
pack_qos_SLQSQosSwiReadDataStats_t . . . 421
pack_qos_SLQSSetQosEventCallback_t . . . 421
pack_rms_SetSMSWake_t . . . 422
pack_sar_SLQSSetRfSarState_t . . . 422
pack_sms_SaveSMS_t . . . 423
pack_sms_SendSMS_t . . . 424
pack_sms_SetNewSMSCallback_t . . . 425
pack_sms_SetSMSCAddress_t . . . 425
pack_sms_SLQSDeleteSMS_t . . . 425
pack_sms_SLQSGetSMS_t . . . 426
pack_sms_SLQSGetSmsBroadcastConfig_t . . . 427
pack_sms_SLQSGetSMSList_t . . . 427
pack_sms_SLQSModifySMSStatus_t . . . 428
pack_sms_SLQSSendAsyncSMS_t . . . 429
pack_sms_SLQSSetIndicationRegister_t . . . 429
pack_sms_SLQSSetSmsBroadcastActivation_t . . . 430
pack_sms_SLQSSetSmsBroadcastConfig_t . . . 430
pack_sms_SLQSSetSmsStorage_t . . . 431
pack_sms_SLQSSmsGetMaxStorageSize_t . . . 432
pack_sms_SLQSSmsSetRoutes_t . . . 432
pack_swiaudio_SLQSGetM2MAudioProfile_t . . . 432
pack_swiaudio_SLQSGetM2MAudioVolume_t . . . 433
pack_swiaudio_SLQSGetM2MAVMute_t . . . 433

pack_swiaudio_SLQSGetM2MSpkrGain_t . . . 434
pack_swiaudio_SLQSSetM2MAudioAVCFG_t . . . 434
pack_swiaudio_SLQSSetM2MAudioLPBK_t . . . 435
pack_swiaudio_SLQSSetM2MAudioProfile_t . . . 435
pack_swiaudio_SLQSSetM2MAudioVolume_t . . . 437
pack_swiaudio_SLQSSetM2MAVMute_t . . . 437
pack_swiaudio_SLQSSetM2MSpkrGain_t . . . 438
pack_swiavms_SLQSAVMSSendSelection_t . . . 439
pack_swiavms_SLQSAVMSSetSettings_t . . . 439
pack_swiavms_SLQSAVMSSetSettings_v2_t . . . 441
pack_swiavms_SLQSAVMSStartSession_t . . . 443
pack_swiavms_SLQSAVMSStopSession_t . . . 443
pack_swidms_SLQSSwiDmsSetMTU_t . . . 444
pack_swidms_SLQSSwiDmsSetUsbComp_t . . . 444
pack_swidms_SLQSSwiDmsSetUsbNetNum_t . . . 445
pack_swiloc_SwiLocSetAutoStart_t . . . 446
pack_swioma_SLQSOMADMCancelSession_t . . . 447
pack_swioma_SLQSOMADMCancelSessionExt_t . . . 448
pack_swioma_SLQSOMADMGetSessionInfo_t . . . 448
pack_swioma_SLQSOMADMSendSelection_t . . . 449
pack_swioma_SLQSOMADMSendSelectionExt_t . . . 449
pack_swioma_SLQSOMADMSetSettings_t . . . 450
pack_swioma_SLQSOMADMSetSettingsExt_t . . . 451
pack_swioma_SLQSOMADMStartSession_t . . . 452
pack_swioma_SLQSOMADMStartSessionExt_t . . . 452
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t . . . 453
pack_tmd_SLQSTmdGetMitigationLvl_t . . . 453
pack_tmd_SLQSTmdRegNotMitigationLvl_t . . . 454
pack_uim_ChangePin_t . . . 454
pack_uim_ReadTransparent_t . . . 455
pack_uim_SetPinProtection_t . . . 457
pack_uim_SLQSUIMAuthenticate_t . . . 458
pack_uim_SLQSUIMDepersonalization_t . . . 458
pack_uim_SLQSUIMEventRegister_t . . . 459
pack_uim_SLQSUIMGetConfiguration_t . . . 459
pack_uim_SLQSUIMGetFileAttributes_t . . . 460
pack_uim_SLQSUIMPowerDown_t . . . 460
pack_uim_SLQSUIMPowerUp_t . . . 461
pack_uim_SLQSUIMRefreshComplete_t . . . 461
pack_uim_SLQSUIMRefreshGetLastEvent_t . . . 462
pack_uim_SLQSUIMRefreshOK_t . . . 462
pack_uim_SLQSUIMRefreshRegister_t . . . 463
pack_uim_SLQSUIMSwitchSlot_t . . . 463
pack_uim_UnblockPin_t . . . 464
pack_uim_VerifyPin_t . . . 465
pack_voice_AnswerUSSD_t . . . 466
pack_voice_OriginateUSSD_t . . . 467
pack_voice_SLQSOriginateUSSD_t . . . 467
pack_voice_SLQSVoiceALSSelectLine_t . . . 468
pack_voice_SLQSVoiceALSSetLineSwitching_t . . . 468
pack_voice_SLQSVoiceAnswerCall_t . . . 469
pack_voice_SLQSVoiceBindSubscription_t . . . 469
pack_voice_SLQSVoiceBurstDTMF_t . . . 469
pack_voice_SLQSVoiceDialCall_t . . . 470
pack_voice_SLQSVoiceEndCall_t . . . 472
pack_voice_SLQSVoiceGetCallBarring_t . . . 472
pack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 473
pack_voice_SLQSVoiceGetCallInfo_t . . . 474

pack_voice_SLQSVoiceGetCallWaiting_t . . . 474
pack_voice_SLQSVoiceGetConfig_t . . . 475
pack_voice_SLQSVoiceIndicationRegister_t . . . 477
pack_voice_SLQSVoiceManageCalls_t . . . 477
pack_voice_SLQSVoiceOrigUSSDNoWait_t . . . 478
pack_voice_SLQSVoiceSendFlash_t . . . 479
pack_voice_SLQSVoiceSetCallBarringPassword_t . . . 479
pack_voice_SLQSVoiceSetConfig_t . . . 480
pack_voice_SLQSVoiceSetPreferredPrivacy_t . . . 482
pack_voice_SLQSVoiceSetSUPSService_t . . . 482
pack_voice_SLQSVoiceStartContDTMF_t . . . 484
pack_voice_SLQSVoiceStopContDTMF_t . . . 485
pack_wds_DHCPv4ClientLeaseChange_t . . . 485
pack_wds_GetDefaultProfile_t . . . 486
pack_wds_GetDefaultProfileNum_t . . . 486
pack_wds_GetDormancyState_t . . . 487
pack_wds_GetLastMobileIPError_t . . . 487
pack_wds_GetMobileIP_t . . . 487
pack_wds_GetMobileIPProfile_t . . . 487
pack_wds_GetPacketStatistics_t . . . 488
pack_wds_GetPacketStatus_t . . . 488
pack_wds_GetSessionDuration_t . . . 489
pack_wds_RMSetTransferStatistics_t . . . 489
pack_wds_SetAutoconnect_t . . . 490
pack_wds_SetDefaultProfile_t . . . 490
pack_wds_SetDefaultProfileNum_t . . . 492
pack_wds_SetMobileIP_t . . . 493
pack_wds_SetMobileIPParameters_t . . . 493
pack_wds_SetMobileIPProfile_t . . . 494
pack_wds_SLQSCreateProfile_t . . . 496
pack_wds_SLQSDeleteProfile_t . . . 497
pack_wds_SLQSGetCurrDataSystemStat_t . . . 497
pack_wds_SLQSGetDataBearerTechnology_t . . . 498
pack_wds_SLQSGetDUNCallInfo_t . . . 498
pack_wds_SLQSGetProfileSettings_t . . . 499
pack_wds_SLQSGetRuntimeSettings_t . . . 500
pack_wds_SLQSModifyProfile_t . . . 501
pack_wds_SLQSSet3GPPConfigItem_t . . . 502
pack_wds_SLQSSetIPFamilyPreference_t . . . 504
pack_wds_SLQSSetWdsEventCallback_t . . . 504
pack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 505
pack_wds_SLQSSSetDHCPv4ClientConfig_t . . . 505
pack_wds_SLQSSSetLoopback_t . . . 506
pack_wds_SLQSStartDataSession_t . . . 506
pack_wds_SLQSStopDataSession_t . . . 508
pack_wds_SLQSWdsSetEventReport_t . . . 508
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 510
PackCreateProfileOut . . . 510
packgetDyingGaspCfg . . . 511
packgetDyingGaspStatistics . . . 511
PackSwiAvmsSetSettingsAPNInfo . . . 512
PackSwiAvmsSetSettingsConnectionRetryTimers . . . 513
PackSwiAvmsSetSettingsPeriodInfo . . . 514
PackSwiAVMSSettingsAPNInfo . . . 514
PackSwiAVMSSettingsConnectionRetryTimers . . . 515
PackSwiAVMSSettingsPeriodsInfo . . . 516
qmiSmsMessageList . . . 516
qmiWDSDataBearerTechnology . . . 516

qmTlvResult . . . 517
qos_BindDataPortMuxID_t . . . 517
qos_BindDataPortPeripheralEndPointID_t . . . 518
qos_BindDataPortSIODataPort_t . . . 518
RFBandInfoElements . . . 519
rmTrasnferStaticsReq . . . 519
sensorData_t . . . 520
slot_t . . . 522
slotInf . . . 523
slots_t . . . 524
sms_BroadcastConfig . . . 525
sms_CDMABroadcastConfig . . . 525
sms_getIndicationReg . . . 526
sms_getMsgWaitingInfo . . . 527
sms_getTransLayerInfo . . . 527
sms_getTransNWRegInfo . . . 528
sms_maxStorageSizeReq . . . 529
sms_maxStorageSizeResp . . . 529
sms_messageWaitingInfoContent . . . 530
sms_msgProtocolResp . . . 531
sms_qaQmi3GPP2BroadcastCfgInfo . . . 531
sms_qaQmi3GPPBroadcastCfgInfo . . . 532
sms_routeEntry . . . 532
sms_sendAsyncsmsParams . . . 533
sms_setIndicationReg . . . 536
sms_setRoutesReq . . . 536
sms_transLayerInfo . . . 537
sMSCAddressInfo . . . 538
sMSCAddressTlv . . . 538
sMSEtwsMessageInfo . . . 539
sMSEtwsMessageTlv . . . 539
sMSEtwsPlmnInfo . . . 540
sMSMessageModeInfo . . . 540
sMSMTMessageInfo . . . 541
sMSOnIMSInfo . . . 541
sMSOnIMSTlv . . . 542
sMSTransferRouteMTMessageInfo . . . 542
swi_uint256_t . . . 543
swiaudio_PCMparams . . . 543
swidms_ehrpdMTUSizeTlv . . . 544
swidms_hrpdMTUSizeTlv . . . 544
swidms_intfaceCfgTlv . . . 545
swidms_mtuSize3gppTlv . . . 546
swidms_supportedIntBitmaskTlv . . . 547
swidms_usbMTUSizeTlv . . . 547
tdscdmaSigInfoExt . . . 548
tempData_t . . . 548
tmd_mitigationDevList . . . 550
transferRouteMessageTlv . . . 550
uim_appStatus . . . 551
uim_authenticateResult . . . 554
uim_authenticationData . . . 554
uim_cardResult . . . 555
uim_cardStatus . . . 556
uim_changeUIMPIN . . . 557
uim_depersonalizationInformation . . . 558
uim_encryptedPIN1 . . . 559
uim_fileAttributes . . . 560

uim_fileInfo . . . 563
uim_hotSwapStatus . . . 564
uim_personalizationStatus . . . 564
uim_readResult . . . 565
uim_readTransparentInfo . . . 566
uim_refreshevent . . . 566
uim_registerRefresh . . . 568
uim_remainingRetries . . . 569
uim_sessionInformation . . . 570
uim_setPINProtection . . . 570
uim_slotInfo . . . 571
uim_UIMSessionInformation . . . 573
uim_unblockUIMPIN . . . 574
uim_verifyUIMPIN . . . 575
unpack_audio_SLQSGetAudioPathConfig_t . . . 575
unpack_audio_SLQSGetAudioProfile_t . . . 578
unpack_audio_SLQSGetAudioVolTLBConfig_t . . . 579
unpack_audio_SLQSSetAudioVolTLBConfig_t . . . 579
unpack_cat_SetCatEventCallback_ind_t . . . 580
unpack_cat_SetCATEventCallback_t . . . 580
unpack_dms_GetActivationState_t . . . 581
unpack_dms_GetBandCapability_t . . . 582
unpack_dms_GetCrashAction_t . . . 584
unpack_dms_GetCustFeature_t . . . 585
unpack_dms_GetCustFeaturesV2_t . . . 587
unpack_dms_GetDeviceCap_t . . . 588
unpack_dms_GetDeviceCapabilities_t . . . 589
unpack_dms_GetDeviceHardwareRev_t . . . 591
unpack_dms_GetDeviceMfr_t . . . 592
unpack_dms_GetDeviceSerialNumbers_t . . . 592
unpack_dms_GetFirmwareInfo_t . . . 594
unpack_dms_GetFirmwareRevision_t . . . 596
unpack_dms_GetFirmwareRevisions_t . . . 597
unpack_dms_GetFSN_t . . . 598
unpack_dms_GetHardwareRevision_t . . . 599
unpack_dms_GetIMSI_t . . . 599
unpack_dms_GetManufacturer_t . . . 600
unpack_dms_GetModelID_t . . . 601
unpack_dms_GetNetworkTime_t . . . 601
unpack_dms_GetOfflineReason_t . . . 602
unpack_dms_GetPower_t . . . 603
unpack_dms_GetPRLVersion_t . . . 604
unpack_dms_GetSerialNumbers_t . . . 605
unpack_dms_GetUSBComp_t . . . 606
unpack_dms_GetVoiceNumber_t . . . 608
unpack_dms_PSMCfgChange_ind_t . . . 610
unpack_dms_ResetToFactoryDefaults_t . . . 610
unpack_dms_SetActivationStatusCallback_t . . . 611
unpack_dms_SetCrashAction_t . . . 611
unpack_dms_SetCustFeature_t . . . 612
unpack_dms_SetCustFeaturesV2_t . . . 612
unpack_dms_SetEventReport_ind_t . . . 613
unpack_dms_SetEventReport_t . . . 614
unpack_dms_SetFirmwarePreference_t . . . 614
unpack_dms_SetIndicationRegister_t . . . 615
unpack_dms_SetPower_t . . . 615
unpack_dms_SetUSBComp_t . . . 616
unpack_dms_SLQSDmsSwiGetPCInfo_t . . . 616

unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t . . . 619
unpack_dms_SLQSDmsSwiGetResetInfo_t . . . 620
unpack_dms_SLQSDmsSwiGetUimSelection_t . . . 621
unpack_dms_SLQSDmsSwiIndicationRegister_t . . . 622
unpack_dms_SLQSGetBandCapability_t . . . 623
unpack_dms_SLQSGetERIFile_t . . . 627
unpack_dms_SLQSGetPowerSaveModeConfig_t . . . 628
unpack_dms_SLQSSetPowerSaveModeConfig_t . . . 629
unpack_dms_SLQSSwiClearDyingGaspStatistics_t . . . 630
unpack_dms_SLQSSwiGetCrashInfo_t . . . 630
unpack_dms_SLQSSwiGetDyingGaspCfg_t . . . 631
unpack_dms_SLQSSwiGetDyingGaspStatistics_t . . . 632
unpack_dms_SLQSSwiGetFirmwareCurr_t . . . 632
unpack_dms_SLQSSwiGetFwUpdateStatus_t . . . 633
unpack_dms_SLQSSwiGetHostDevInfo_t . . . 635
unpack_dms_SLQSSwiGetOSInfo_t . . . 636
unpack_dms_SLQSSwiGetSerialNoExt_t . . . 637
unpack_dms_SLQSSwiSetDyingGaspCfg_t . . . 638
unpack_dms_SLQSSwiSetHostDevInfo_t . . . 638
unpack_dms_SLQSSwiSetOSInfo_t . . . 639
unpack_dms_SLQSUIMGetState_t . . . 639
unpack_dms_SwiEventReportCallBack_ind_t . . . 640
unpack_dms_SwiSetEventReport_t . . . 641
unpack_dms_SwiUimSelect_t . . . 641
unpack_dms_UIMGetControlKeyStatus_t . . . 642
unpack_dms_UIMGetICCID_t . . . 643
unpack_dms_UIMGetPINStatus_t . . . 644
unpack_dms_UIMSetControlKeyProtection_t . . . 645
unpack_dms_UIMSetPINProtection_t . . . 646
unpack_dms_UIMUnblockControlKey_t . . . 647
unpack_fms_GetImagesPreference_t . . . 647
unpack_fms_GetStoredImages_t . . . 648
unpack_fms_SetImagesPreference_t . . . 649
unpack_ims_SLQSGetIMSSMSConfig_t . . . 649
unpack_ims_SLQSGetIMSUserConfig_t . . . 650
unpack_ims_SLQSGetIMSVoIPConfig_t . . . 651
unpack_ims_SLQSGetRegMgrConfig_t . . . 654
unpack_ims_SLQSGetSIPConfig_t . . . 655
unpack_ims_SLQSRegMgrCfgCallBack_ind_t . . . 656
unpack_ims_SLQSSetIMSSMSConfig_t . . . 657
unpack_ims_SLQSSetIMSUserConfig_t . . . 658
unpack_ims_SLQSSetIMSVoIPConfig_t . . . 658
unpack_ims_SLQSSetRegMgrConfig_t . . . 659
unpack_ims_SLQSSetSIPConfig_t . . . 659
unpack_ims_SLQSSIPCfgCallBack_ind_t . . . 660
unpack_ims_SLQSSMSCfgCallBack_ind_t . . . 661
unpack_ims_SLQSUserCfgCallBack_ind_t . . . 662
unpack_ims_SLQSVoIPCfgCallBack_ind_t . . . 663
unpack_imsa_SLQSGetIMSARegStatus_t . . . 665
unpack_imsa_SLQSGetIMSAServiceStatus_t . . . 666
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t . . . 668
unpack_imsa_SLQSImsaRatStatusCallBack_ind_t . . . 669
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t . . . 670
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t . . . 671
unpack_loc_BestAvailPos_Ind_t . . . 672
unpack_loc_CradleMountCallback_Ind_t . . . 679
unpack_loc_Delete_Assist_Data_t . . . 679
unpack_loc_DeleteAssistData_Ind_t . . . 680

unpack_loc_EngineState_Ind_t . . . 681
unpack_loc_EventNMEA_Ind_t . . . 682
unpack_loc_EventRegister_t . . . 683
unpack_loc_EventTimeSyncCallback_Ind_t . . . 683
unpack_loc_GetServer_Ind_t . . . 683
unpack_loc_GnssSvInfo_Ind_t . . . 685
unpack_loc_InjectPositionCallback_Ind_t . . . 686
unpack_loc_InjectSensorDataCallback_Ind_t . . . 686
unpack_loc_InjectTimeSyncDataCallback_Ind_t . . . 688
unpack_loc_InjectUTCTimeCallback_Ind_t . . . 689
unpack_loc_PositionRpt_Ind_t . . . 689
unpack_loc_SensorStreamingCallback_Ind_t . . . 696
unpack_loc_SetExtPowerConfig_Ind_t . . . 697
unpack_loc_SetExtPowerState_t . . . 698
unpack_loc_SetOperationMode_Ind_t . . . 699
unpack_loc_SetOperationMode_t . . . 699
unpack_loc_SetServer_Ind_t . . . 700
unpack_loc_SLQSLOCGetBestAvailPos_t . . . 701
unpack_loc_Start_t . . . 701
unpack_loc_Stop_t . . . 702
unpack_nas_GetACCOLC_t . . . 702
unpack_nas_GetANAAAAuthenticationStatus_t . . . 703
unpack_nas_GetCDMANetworkParameters_t . . . 703
unpack_nas_GetHomeNetwork3GPP2_t . . . 706
unpack_nas_GetHomeNetwork_t . . . 708
unpack_nas_GetNetworkPreference_t . . . 709
unpack_nas_GetRFInfo_t . . . 710
unpack_nas_GetServingNetwork_t . . . 711
unpack_nas_GetServingNetworkCapabilities_t . . . 714
unpack_nas_GetSignalStrengths_t . . . 715
unpack_nas_PerformNetworkScan_t . . . 716
unpack_nas_SetDataCapabilitiesCallback_ind_t . . . 717
unpack_nas_SetEventReportInd_t . . . 718
unpack_nas_SetNasLTECphyCaIndCallback_ind_t . . . 719
unpack_nas_SetNetworkPreference_t . . . 720
unpack_nas_SetRoamingIndicatorCallback_ind_t . . . 721
unpack_nas_SetServingSystemCallback_ind_t . . . 721
unpack_nas_SLQSGetErrorRate_t . . . 722
unpack_nas_SlqsGetLTECphyCAInfo_t . . . 723
unpack_nas_SLQSGetNetworkTime_t . . . 724
unpack_nas_SLQSGetOperatorNameData_t . . . 724
unpack_nas_SLQSGetPLMNName_t . . . 726
unpack_nas_SLQSGetServingSystem_t . . . 729
unpack_nas_SLQSGetSignalStrength_t . . . 733
unpack_nas_SLQSGetSysInfo_t . . . 736
unpack_nas_SLQSGetSysSelectionPref_t . . . 740
unpack_nas_SLQSGetSysSelectionPrefExt_t . . . 744
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t . . . 747
unpack_nas_SLQSNasGet3GPP2Subscription_t . . . 748
unpack_nas_SLQSNasGetCellLocationInfo_t . . . 749
unpack_nas_SLQSNASGeteDRXParams_t . . . 751
unpack_nas_SLQSNASGeteDRXParamsExt_t . . . 752
unpack_nas_SLQSNASGetForbiddenNetworks_t . . . 753
unpack_nas_SLQSNasGetHDRColorCode_t . . . 754
unpack_nas_SLQSNasGetRFInfo_t . . . 754
unpack_nas_SLQSNasGetSigInfo_t . . . 755
unpack_nas_SLQSNasGetTxRxInfo_t . . . 756
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t . . . 757

unpack_nas_SLQSNasNetworkTimeCallBack_ind_t . . . 760
unpack_nas_SLQSNasSigInfoCallback_ind_t . . . 761
unpack_nas_SLQSNASSwiGetChannelLock_t . . . 763
unpack_nas_SLQSNasSwiModemStatus_t . . . 763
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t . . . 764
unpack_nas_SLQSNasTimerCallback_ind_t . . . 765
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t . . . 765
unpack_nas_SLQSSwiGetHDRPersonality_t . . . 766
unpack_nas_SLQSSwiGetHDRProtSubtype_t . . . 767
unpack_nas_SLQSSwiGetHRPDStats_t . . . 768
unpack_nas_SLQSSwiGetLteCQI_t . . . 769
unpack_nas_SLQSSwiGetLteSccRxInfo_t . . . 770
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t . . . 770
unpack_nas_SLQSSwiNetworkDebug_t . . . 771
unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t . . . 772
unpack_nas_SLQSSysInfoCallback_ind_t . . . 772
unpack_omaDmConfigTlv_t . . . 776
unpack_omaDmFotaTlv_t . . . 777
unpack_omaDmNotificationsTlv_t . . . 779
unpack_pds_GetPDSDefaults_t . . . 780
unpack_pds_GetPDSState_t . . . 781
unpack_pds_GetPortAutomaticTracking_t . . . 782
unpack_pds_GetServiceAutomaticTracking_t . . . 783
unpack_pds_GetXTRAAutomaticDownload_t . . . 783
unpack_pds_GetXTRANetwork_t . . . 784
unpack_pds_GetXTRAValidity_t . . . 784
unpack_pds_SetEventReport_Ind_t . . . 785
unpack_pds_SetPdsState_Ind_t . . . 787
unpack_pds_SLQSGetAGPSConfig_t . . . 788
unpack_pds_SLQSGetGPSStateInfo_t . . . 789
unpack_qmi_t . . . 794
unpack_qos_BindDataPort_t . . . 794
unpack_qos_dataRate_t . . . 795
unpack_qos_IPv4Addr_t . . . 795
unpack_qos_IPv6Addr_t . . . 795
unpack_qos_IPv6TrafCls_t . . . 796
unpack_qos_pktErrRate_t . . . 796
unpack_qos_Port_t . . . 797
unpack_qos_QosFlowInfo_t . . . 797
unpack_qos_QosFlowInfoState_t . . . 799
unpack_qos_SLQSQosGetNetworkStatus_t . . . 799
unpack_qos_SLQSQosSwiReadApnExtraParams_t . . . 800
unpack_qos_SLQSQosSwiReadDataStats_t . . . 802
unpack_qos_SLQSSetQosEventCallback_ind_t . . . 803
unpack_qos_SLQSSetQosEventCallback_t . . . 804
unpack_qos_SLQSSetQosNWStatusCallback_ind_t . . . 804
unpack_qos_SLQSSetQosPriEventCallback_ind_t . . . 805
unpack_qos_SLQSSetQosStatusCallback_ind_t . . . 805
unpack_qos_swiQosFilter_t . . . 807
unpack_qos_swiQosFlow_t . . . 810
unpack_qos_tokenBucket_t . . . 814
unpack_qos_Tos_t . . . 815
unpack_QosFlowStat_t . . . 815
unpack_result_t . . . 816
unpack_rms_GetSMSWake_t . . . 817
unpack_rms_SetSMSWake_t . . . 817
unpack_RMTransferStatistics_ind_t . . . 818
unpack_sar_SLQSGetRfSarState_t . . . 819

unpack_sms_GetSMSCAddress_t . . . 819
unpack_sms_SaveSMS_t . . . 820
unpack_sms_SendSMS_t . . . 821
unpack_sms_SetNewSMSCallback_ind_t . . . 822
unpack_sms_SetNewSMSCallback_t . . . 823
unpack_sms_SetSMSCAddress_t . . . 823
unpack_sms_SLQSDeleteSMS_t . . . 824
unpack_sms_SLQSGetIndicationRegister_t . . . 824
unpack_sms_SLQSGetMessageWaiting_t . . . 825
unpack_sms_SLQSGetSMS_t . . . 825
unpack_sms_SLQSGetSmsBroadcastConfig_t . . . 826
unpack_sms_SLQSGetSMSList_t . . . 827
unpack_sms_SLQSGetTransLayerInfo_t . . . 828
unpack_sms_SLQSGetTransNWRegInfo_t . . . 828
unpack_sms_SLQSModifySMSStatus_t . . . 829
unpack_sms_SLQSNWRegInfoCallback_ind_t . . . 829
unpack_sms_SLQSSendAsyncSMS_t . . . 830
unpack_sms_SLQSSetIndicationRegister_t . . . 830
unpack_sms_SLQSSetSmsBroadcastActivation_t . . . 831
unpack_sms_SLQSSetSmsBroadcastConfig_t . . . 831
unpack_sms_SLQSSetSmsStorage_t . . . 832
unpack_sms_SLQSSmsGetMaxStorageSize_t . . . 832
unpack_sms_SLQSSmsGetMessageProtocol_t . . . 833
unpack_sms_SLQSSmsSetRoutes_t . . . 833
unpack_sms_SLQSSwiGetSMSStorage_t . . . 834
unpack_sms_SLQSTransLayerInfoCallback_ind_t . . . 834
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t . . . 835
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t . . . 837
unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t . . . 838
unpack_swiaudio_SLQSGetM2MAudioProfile_t . . . 838
unpack_swiaudio_SLQSGetM2MAudioVolume_t . . . 840
unpack_swiaudio_SLQSGetM2MAVMute_t . . . 840
unpack_swiaudio_SLQSGetM2MSpkrGain_t . . . 841
unpack_swiavms_SLQSAVMSEventReportInd_t . . . 842
unpack_swiavms_SLQSAVMSGetSettings_t . . . 843
unpack_swiavms_SLQSAVMSGetSettings_v2_t . . . 846
unpack_swiavms_SLQSAVMSSendSelection_t . . . 848
unpack_swiavms_SLQSAVMSSessionGetInfo_t . . . 849
unpack_swiavms_SLQSAvmsSetEventReport_t . . . 850
unpack_swiavms_SLQSAVMSSetSettings_t . . . 850
unpack_swiavms_SLQSAVMSSetSettings_v2_t . . . 851
unpack_swiavms_SLQSAVMSStartSession_t . . . 851
unpack_swiavms_SLQSAVMSStopSession_t . . . 852
unpack_swidms_SLQSSwiDmsGetMTU_t . . . 852
unpack_swidms_SLQSSwiDmsGetUsbComp_t . . . 853
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t . . . 854
unpack_swidms_SLQSSwiDmsSetMTU_t . . . 854
unpack_swidms_SLQSSwiDmsSetUsbComp_t . . . 855
unpack_swiloc_SwiLocGetAutoStart_t . . . 855
unpack_swioma_SLQSOMADMAlertCallback_ind_t . . . 857
unpack_swioma_SLQSOMADMGetSessionInfo_t . . . 858
unpack_swioma_SLQSOMADMGetSettings_t . . . 861
unpack_swioma_SLQSOMADMStartSession_t . . . 863
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t . . . 863
unpack_tmd_SLQSTmdGetMitigationDevList_t . . . 864
unpack_tmd_SLQSTmdGetMitigationLvl_t . . . 865
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t . . . 865
unpack_tmd_SLQSTmdRegNotMitigationLvl_t . . . 866

unpack_uim_ChangePin_t . . . 867
unpack_uim_GetCardStatus_t . . . 868
unpack_uim_ReadTransparent_t . . . 868
unpack_uim_SetPinProtection_t . . . 869
unpack_uim_SetUimSlotStatusChangeCallback_ind_t . . . 870
unpack_uim_SLQSUIMAuthenticate_t . . . 871
unpack_uim_SLQSUIMDepersonalization_t . . . 872
unpack_uim_SLQSUIMEventRegister_t . . . 872
unpack_uim_SLQSUIMGetConfiguration_t . . . 873
unpack_uim_SLQSUIMGetFileAttributes_t . . . 874
unpack_uim_SLQSUIMGetSlotsStatus_t . . . 874
unpack_uim_SLQSUIMRefreshCallback_Ind_t . . . 875
unpack_uim_SLQSUIMRefreshGetLastEvent_t . . . 876
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t . . . 876
unpack_uim_UnblockPin_t . . . 877
unpack_uim_VerifyPin_t . . . 878
unpack_voice_allCallStatusCallback_ind_t . . . 879
unpack_voice_DTMFEventCallback_ind_t . . . 881
unpack_voice_OTASPStatusCallback_ind_t . . . 882
unpack_voice_SLQSOriginateUSSD_t . . . 883
unpack_voice_SLQSVoiceAnswerCall_t . . . 885
unpack_voice_SLQSVoiceBurstDTMF_t . . . 886
unpack_voice_SLQSVoiceDialCall_t . . . 886
unpack_voice_SLQSVoiceEndCall_t . . . 887
unpack_voice_SLQSVoiceGetAllCallInfo_t . . . 888
unpack_voice_SLQSVoiceGetCallBarring_t . . . 892
unpack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 893
unpack_voice_SLQSVoiceGetCallInfo_t . . . 895
unpack_voice_SLQSVoiceGetCallWaiting_t . . . 898
unpack_voice_SLQSVoiceGetCLIP_t . . . 899
unpack_voice_SLQSVoiceGetCLIR_t . . . 901
unpack_voice_SLQSVoiceGetCNAP_t . . . 902
unpack_voice_SLQSVoiceGetCOLP_t . . . 904
unpack_voice_SLQSVoiceGetCOLR_t . . . 905
unpack_voice_SLQSVoiceGetConfig_t . . . 906
unpack_voice_SLQSVoiceManageCalls_t . . . 908
unpack_voice_SLQSVoiceSendFlash_t . . . 909
unpack_voice_SLQSVoiceSetCallBarringPassword_t . . . 910
unpack_voice_SLQSVoiceSetConfig_t . . . 911
unpack_voice_SLQSVoiceSetSUPSService_t . . . 913
unpack_voice_SLQSVoiceStartContDTMF_t . . . 914
unpack_voice_SLQSVoiceStopContDTMF_t . . . 914
unpack_voice_SLQSVoiceSUPSCallback_ind_t . . . 915
unpack_voice_SUPSNotificationCallback_ind_t . . . 918
unpack_voice_USSDNotificationCallback_ind_t . . . 920
unpack_voice_VoiceInfoRecCallback_ind_t . . . 920
unpack_voice_voicePrivacyChangeCallback_ind_t . . . 923
unpack_wds_DHCPv4ClientLease_ind_t . . . 924
unpack_wds_GetAutoconnect_t . . . 925
unpack_wds_GetByteTotals_t . . . 925
unpack_wds_GetConnectionRate_t . . . 926
unpack_wds_GetDataBearerTechnology_t . . . 927
unpack_wds_GetDefaultProfile_t . . . 928
unpack_wds_GetDefaultProfileNum_t . . . 931
unpack_wds_GetDormancyState_t . . . 931
unpack_wds_GetLastMobileIPError_t . . . 932
unpack_wds_GetMobileIP_t . . . 933
unpack_wds_GetMobileIPProfile_t . . . 933

unpack_wds_GetPacketStatistics_t . . . 935
unpack_wds_GetPacketStatus_t . . . 937
unpack_wds_GetSessionDuration_t . . . 939
unpack_wds_GetSessionState_t . . . 940
unpack_wds_RMSetTransferStatistics_t . . . 940
unpack_wds_SetMobileIPProfile_t . . . 941
unpack_wds_SLQSCreateProfile_t . . . 941
unpack_wds_SLQSDeleteProfile_t . . . 942
unpack_wds_SLQSDUNCallInfoCallBack_ind_t . . . 943
unpack_wds_SLQSGet3GPPConfigItem_t . . . 944
unpack_wds_SLQSGetCurrDataSystemStat_t . . . 946
unpack_wds_SLQSGetCurrentChannelRate_t . . . 947
unpack_wds_SLQSGetDataBearerTechnology_t . . . 948
unpack_wds_SLQSGetDUNCallInfo_t . . . 949
unpack_wds_SLQSGetProfileSettings_t . . . 952
unpack_wds_SLQSGetRuntimeSettings_t . . . 952
unpack_wds_SLQSModifyProfile_t . . . 956
unpack_wds_SLQSSetIPFamilyPreference_t . . . 956
unpack_wds_SLQSSetPacketSrvStatusCallback_t . . . 957
unpack_wds_SLQSSetWdsEventCallback_ind_t . . . 959
unpack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 964
unpack_wds_SLQSSGetLoopback_t . . . 965
unpack_wds_SLQSStartDataSession_t . . . 966
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 967
UnPackGetProfileSettingOut . . . 969
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo . . . 970
UnpackSwiAvmsEventReportConfig . . . 972
UnpackSwiAvmsEventReportConnectionRequest . . . 973
UnpackSwiAvmsEventReportDataSessionStatus . . . 973
UnpackSwiAvmsEventReportHTTPStatus . . . 974
UnpackSwiAvmsEventReportNotification . . . 976
UnpackSwiAvmsEventReportPackageID . . . 977
UnpackSwiAvmsEventReportRegStatus . . . 977
UnpackSwiAvmsEventReportSessionType . . . 978
UnpackSwiAvmsEventReportWAMSParamChange . . . 978
unpackWdsProfileParam . . . 979
voice_airTimer . . . 979
voice_allCallsAlphaIDInfo . . . 980
voice_allCallsDiagInfo . . . 980
voice_allCallsUUSInfo . . . 981
voice_alphaIDInfo . . . 981
voice_arrAlertingPattern . . . 982
voice_arrAlertingType . . . 983
voice_arrAlphaID . . . 984
voice_arrCalledPartyNum . . . 984
voice_arrCallEndReason . . . 985
voice_arrCallInfo . . . 986
voice_arrConnectPartyNum . . . 986
voice_arrDiagInfo . . . 987
voice_arrRedirPartyNum . . . 987
voice_arrRemotePartyName . . . 988
voice_arrRemotePartyNum . . . 988
voice_arrSvcOption . . . 989
voice_arrUUSInfo . . . 989
voice_burstDTMFInfo . . . 990
voice_calledPartyInfo . . . 991
voice_calledPartySubAdd . . . 992
voice_callerIDInfo . . . 993

voice_callFwdTypeAndPlan . . . 994
voice_callFWExtInfo . . . 995
voice_callFWInfo . . . 997
voice_callInfo . . . 998
voice_callingPartyInfo . . . 999
voice_ccSUPSType . . . 1001
voice_CLIPResp . . . 1002
voice_CLIRResp . . . 1002
voice_CNAPResp . . . 1003
voice_COLPResp . . . 1004
voice_COLRResp . . . 1004
voice_connectNumInfo . . . 1005
voice_CUGInfo . . . 1007
voice_curAMRConfig . . . 1007
voice_diagInfo . . . 1008
voice_DTMFInfo . . . 1009
voice_DTMFLengths . . . 1009
voice_ECTNum . . . 1010
voice_extDispRecInfo . . . 1011
voice_getAllCallInformation . . . 1012
voice_getAllCallRmtPtyName . . . 1012
voice_getAllCallRmtPtyNum . . . 1013
voice_getCallFWExtInfo . . . 1013
voice_getCallFWInfo . . . 1014
voice_lineCtrlInfo . . . 1014
voice_newPwdData . . . 1015
voice_NSSAudioCtrl . . . 1016
voice_peerNumberInfo . . . 1016
voice_prefVoiceSO . . . 1018
voice_redirNumInfo . . . 1020
voice_remotePartyName . . . 1022
voice_remotePartyNum . . . 1022
voice_roamTimer . . . 1023
voice_signalInfo . . . 1024
voice_SUPSInfo . . . 1025
voice_USSDNotificationNetworkInfo . . . 1025
voice_USSInfo . . . 1026
voice_UUSInfo . . . 1026
wds_channelRate . . . 1028
wds_ChannelRateTlv . . . 1028
wds_ConnStatusTlv . . . 1029
wds_currNetworkInfo . . . 1029
wds_DataBearTechTlv . . . 1030
wds_DataULongLongTlv . . . 1031
wds_DataULongTlv . . . 1032
wds_DHCPLeaseOptTlv . . . 1032
wds_DHCPLeaseStateTlv . . . 1033
wds_DHCPOpt . . . 1033
wds_DHCPProfileIdTlv . . . 1034
wds_DHCPv4HWConfig . . . 1034
wds_DHCPv4Option . . . 1035
wds_DHCPv4OptionList . . . 1036
wds_DHCPv4ProfileId . . . 1036
wds_Domain . . . 1037
wds_DomainNameList . . . 1037
wds_DormStatTlv . . . 1038
wds_GPRSQoS . . . 1038
wds_IPv4AdTlv . . . 1039

wds_IPV6AddressInfo . . . 1040
wds_IPV6GWAddressInfo . . . 1040
wds_LastMdmCallEndRsnTlv . . . 1041
wds_PCSCFFQDNAddress . . . 1041
wds_PCSCFFQDNAddressList . . . 1042
wds_PCSCFIPv4ServerAddressList . . . 1042
wds_ProfileIdentifier . . . 1043
wds_profileInfo . . . 1044
wds_RXBytesOKTlv . . . 1044
wds_transferStatInd . . . 1045
wds_TrStatInd . . . 1045
wds_TXBytesOKTlv . . . 1046
wds_UMTSMinQoS . . . 1047
wdsDhcpv4HwConfig . . . 1049
wdsDhcpv4Option . . . 1050
wdsDhcpv4OptionList . . . 1050
wdsDhcpv4ProfileId . . . 1051

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 . . . 1053
audio.h . . . 1053
cat.h . . . 1059
common.h . . . 1062
dms.h . . . 1068
fms.h . . . 1128
ims.h . . . 1131
imsa.h . . . 1143
lite-fw.h . . . 1149
loc.h . . . 1160
nas.h . . . 1182
pds.h . . . 1245
qaGobiApiTableBandClasses.h
Network Access Service API Band Classes table . . . 1267
qaGobiApiTableCallControlReturnReasons.h
Call Control Return Reasons table . . . 1270
qaGobiApiTableCallEndReasons.h
Wireless Data Service Call End Reasons . . . 1271
qaGobiApiTableCarrierCodes.h
Carrier Codes table . . . 1287
qaGobiApiTableCodingScheme.h
Data Coding Scheme . . . 1289
qaGobiApiTableGpsCapabilityCodes.h
Position Determination Service API GPS Capability Codes . . . 1292
qaGobiApiTablePowerModes.h
Device Management Service API Power Modes table . . . 1293
qaGobiApiTableRadioInterfaces.h
Network Access Service API Radio Interfaces table . . . 1293
qaGobiApiTableRegionCodes.h
Region Codes table . . . 1294
qaGobiApiTableServiceOptions.h
Voice Service Options . . . 1294
qaGobiApiTableSupServiceInfoClasses.h
Voice Supplementary Service Information Classes . . . 1297
qaGobiApiTableSwiAudio.h
Swi Audio related tables . . . 1297

qaGobiApiTableSwiOMADMSessionStatus.h
Session Status table . . . 1298
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h
Update Complete Status table . . . 1299
qaGobiApiTableVoiceCallEndReasons.h
Voice Service Call and supplementary services end reasons . . . 1300
qmerrno.h . . . 1307
qos.h . . . 1314
rms.h . . . 1324
sar.h . . . 1326
sms.h . . . 1328
swiaudio.h . . . 1350
swiavms.h . . . 1360
SwiDataTypes.h
SWI data types . . . 1374
swidms.h . . . 1375
swiloc.h . . . 1381
swioma.h . . . 1382
swiomaext.h . . . 1392
switype_256bit.h . . . 1397
SWIWWANCMAPI.h . . . 1399
tmd.h . . . 1399
uim.h . . . 1403
voice.h . . . 1422
wds.h . . . 1454

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

6.1.1 Detailed Description

6.2 Streaming Download Protocol (lite-fw)

Files

- file lite-fw.h

6.2.1 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_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]

8.1.1 Detailed Description

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

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.

8.1.2 Field Documentation

8.1.2.1 char _litefw_FirmwareInfo_::szCarrier_str[20+1]

8.1.2.2 char _litefw_FirmwareInfo_::szCarrierPriversion_str[20+1]

8.1.2.3 char _litefw_FirmwareInfo_::szFwversion_str[20+1]

8.1.2.4 char _litefw_FirmwareInfo_::szModelid_str[20+1]

8.1.2.5 char _litefw_FirmwareInfo_::szPackageid_str[20+1]

8.1.2.6 char _litefw_FirmwareInfo_::szSku_str[20+1]

8.2 _litefw_FirmwarePartNo_ Struct Reference

Data Fields

- char szPartno_str [20+1]

8.2.1 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

8.2.2 Field Documentation

8.2.2.1 char _litefw_FirmwarePartNo_::szPartno_str[20+1]

8.3 altSrcInfo_t Struct Reference

Data Fields

- uint32_t source
- uint32_t linkage
- uint32_t coverage

8.3.1 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

8.3.2 Field Documentation

8.3.2.1 uint32_t altSrcInfo_t::coverage

8.3.2.2 uint32_t altSrcInfo_t::linkage

8.3.2.3 uint32_t altSrcInfo_t::source

8.4 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

8.4.1 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.

8.4.2 Field Documentation

8.4.2.1 uint8_t appStats::aidLength

8.4.2.2 uint8_t appStats::aidVal[255]

8.4.2.3 uint8_t appStats::appState

8.4.2.4 uint8_t appStats::appType

8.4.2.5 uint8_t appStats::persoFeature

8.4.2.6 uint8_t appStats::persoRetries

8.4.2.7 uint8_t appStats::persoState

8.4.2.8 uint8_t appStats::persoUnblockRetries

8.4.2.9 uint8_t appStats::pin1Retries

8.4.2.10 uint8_t appStats::pin1State

8.4.2.11 uint8_t appStats::pin2Retries

8.4.2.12 uint8_t appStats::pin2State

8.4.2.13 uint8_t appStats::puk1Retries

8.4.2.14 uint8_t appStats::puk2Retries

8.4.2.15 uint8_t appStats::univPin

8.5 audio_RXAGCList Struct Reference

Data Fields

- uint16_t * pRXStaticGain

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

8.5.1 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

8.5.2 Field Documentation

8.5.2.1 uint16_t* audio_RXAGCList::pRXAIG

8.5.2.2 uint16_t* audio_RXAGCList::pRXComprSlope

8.5.2.3 uint16_t* audio_RXAGCList::pRXComprThres

8.5.2.4 uint16_t* audio_RXAGCList::pRXExpSlope

8.5.2.5 uint16_t* audio_RXAGCList::pRXExpThres

8.5.2.6 uint16_t* audio_RXAGCList::pRXStaticGain

8.6 audio_RXAVCList Struct Reference

Data Fields

- uint16_t * pAVRXAVCSens
- uint16_t * pAVRXAVCHeadroom

8.6.1 Detailed Description

This structure contains the SLQSGetAudioPathConfig parameters related to AV_RXAVCLIST.

Parameters

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

8.6.2 Field Documentation

8.6.2.1 uint16_t* audio_RXAVCList::pAVRXAVCHeadroom

8.6.2.2 uint16_t* audio_RXAVCList::pAVRXAVCSens

8.7 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

8.7.1 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

8.7.2 Field Documentation

8.7.2.1 uint16_t* audio_RXPCMIIRFltr::pFlag

8.7.2.2 uint8_t* audio_RXPCMIIRFltr::pStage0Val

8.7.2.3 uint8_t* audio_RXPCMIIRFltr::pStage1Val

8.7.2.4 uint8_t* audio_RXPCMIIRFltr::pStage2Val

8.7.2.5 uint8_t* audio_RXPCMIIRFltr::pStage3Val

8.7.2.6 uint8_t* audio_RXPCMIIRFltr::pStage4Val

8.7.2.7 uint16_t* audio_RXPCMIIRFltr::pStageCnt

8.8 audio_TXAGCList Struct Reference

Data Fields

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

8.8.1 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

8.8.2 Field Documentation

8.8.2.1 uint16_t* audio_TXAGCList::pTXAIG

8.8.2.2 uint16_t* audio_TXAGCList::pTXComprSlope

8.8.2.3 uint16_t* audio_TXAGCList::pTXComprThres

8.8.2.4 uint16_t* audio_TXAGCList::pTXExpSlope

8.8.2.5 uint16_t* audio_TXAGCList::pTXExpThres

8.8.2.6 uint16_t* audio_TXAGCList::pTXStaticGain

8.9 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

8.9.1 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

8.9.2 Field Documentation

8.9.2.1 uint16_t* audio_TXPCMIIRFltr::pFlag

8.9.2.2 uint8_t* audio_TXPCMIIRFltr::pStage0Val

8.9.2.3 uint8_t* audio_TXPCMIIRFltr::pStage1Val

8.9.2.4 uint8_t* audio_TXPCMIIRFltr::pStage2Val

8.9.2.5 uint8_t* audio_TXPCMIIRFltr::pStage3Val

8.9.2.6 uint8_t* audio_TXPCMIIRFltr::pStage4Val

8.9.2.7 uint16_t* audio_TXPCMIIRFltr::pStageCnt

8.10 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]

8.10.1 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

8.10.2 Field Documentation

8.10.2.1 uint8_t CarrierImage_t::m_FwBuildId[100]

8.10.2.2 uint8_t CarrierImage_t::m_FwImageId[100]

8.10.2.3 uint32_t CarrierImage_t::m_nCarrierId

8.10.2.4 uint32_t CarrierImage_t::m_nFolderId

8.10.2.5 uint32_t CarrierImage_t::m_nStorage

8.10.2.6 uint8_t CarrierImage_t::m_PriBuildId[100]

8.10.2.7 uint8_t CarrierImage_t::m_PriImageId[100]

8.11 cat_AlPhaIdentifierTlv Struct Reference

Data Fields

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

8.11.1 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]

8.11.2 Field Documentation

8.11.2.1 uint8_t cat_AlPhaIdentifierTlv::AlphaID[255]

8.11.2.2 uint16_t cat_AlPhaIdentifierTlv::AlphaIDLength

8.11.2.3 uint8_t cat_AlPhaIdentifierTlv::ReferenceID

8.12 cat_commonEventTlv Struct Reference

Data Fields

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

8.12.1 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

8.12.2 Field Documentation

8.12.2.1 union cat_currentCatEvent cat_commonEventTlv::CatEvent

8.12.2.2 uint8_t cat_commonEventTlv::EventID

8.12.2.3 uint16_t cat_commonEventTlv::EventLength

8.12.2.4 uint8_t cat_commonEventTlv::TlvPresent

8.13 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

8.13.1 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

8.13.2 Field Documentation

8.13.2.1 struct cat_AlPhaIdentifierTlv cat_currentCatEvent::CatAlphaIdtfr

8.13.2.2 struct cat_EndProactiveSessionTlv cat_currentCatEvent::CatEndPS

8.13.2.3 struct cat_EventListTlv cat_currentCatEvent::CatEventLst

8.13.2.4 struct cat_EventIDDataTlv cat_currentCatEvent::CatEvIDData

8.13.2.5 struct cat_RefreshTlv cat_currentCatEvent::CatRefresh

8.14 cat_EndProactiveSessionTlv Struct Reference

Data Fields

- uint8_t EndProactiveSession

8.14.1 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

8.14.2 Field Documentation

8.14.2.1 uint8_t cat_EndProactiveSessionTlv::EndProactiveSession

8.15 cat_EventIDDataTlv Struct Reference

Data Fields

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

8.15.1 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]

8.15.2 Field Documentation

8.15.2.1 uint8_t cat_EventIDDataTlv::Data[255]

8.15.2.2 uint16_t cat_EventIDDataTlv::DataLength

8.15.2.3 uint32_t cat_EventIDDataTlv::ReferenceID

8.16 cat_EventListTlv Struct Reference

Data Fields

- uint32_t SetupEventList

8.16.1 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

8.16.2 Field Documentation

8.16.2.1 uint32_t cat_EventListTlv::SetupEventList

8.17 cat_RefreshTlv Struct Reference

Data Fields

- uint16_t RefreshMode
- uint8_t RefreshStage

8.17.1 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

8.17.2 Field Documentation

8.17.2.1 uint16_t cat_RefreshTlv::RefreshMode

8.17.2.2 uint8_t cat_RefreshTlv::RefreshStage

8.18 cdmaSSInfo Struct Reference

Data Fields

- int8_t rssi
- int16_t ecio

8.18.1 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).

8.18.2 Field Documentation

8.18.2.1 int16_t cdmaSSInfo::ecio

8.18.2.2 int8_t cdmaSSInfo::rssi

8.19 connectionStatus Struct Reference

Data Fields

- uint8_t MDMConnStatus
- uint64_t MDMCallDuration

8.19.1 Detailed Description

This structure contains modem connection status

Parameters

MDMConnStatus	• Current link status – 0x01 - DISCONNECTED – 0x02 - CONNECTED – 0xff - Invalid data.
MDMCallDuration	• 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.

8.19.2 Field Documentation

8.19.2.1 uint64_t connectionStatus::MDMCallDuration

8.19.2.2 uint8_t connectionStatus::MDMConnStatus

8.20 crashInfoParams Struct Reference

Data Fields

- uint8_t crashStatus
- crashInformation crashInfo

8.20.1 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

8.20.2 Field Documentation

8.20.2.1 crashInformation crashInfoParams::crashInfo

8.20.2.2 uint8_t crashInfoParams::crashStatus

8.21 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]

8.21.1 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[OUT]	• length of the gcdumpString field as an input, length of the gcdumpString field returned by API as an OUTPUT

gcdumpString[OUT]	• gcdump string for the crash

8.21.2 Field Documentation

8.21.2.1 uint32_t crashInformation::crashData

8.21.2.2 uint32_t crashInformation::crashId

8.21.2.3 char crashInformation::crashString[255]

8.21.2.4 uint16_t crashInformation::crashStrlen

8.21.2.5 char crashInformation::gcdumpString[1024]

8.21.2.6 uint16_t crashInformation::gcdumpStrlen

8.21.2.7 uint16_t crashInformation::numCrashes

8.22 currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.22.1 Detailed Description

Network information structure

Parameters

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

8.22.2 Field Documentation

8.22.2.1 uint8_t currNetworkInfo::NetworkType

8.22.2.2 uint32_t currNetworkInfo::RATMask

8.22.2.3 uint32_t currNetworkInfo::SOMask

8.23 dms_ActivationStatusTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t activationStatus

8.23.1 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

8.23.2 Field Documentation

8.23.2.1 uint32_t dms_ActivationStatusTlv::activationStatus

8.23.2.2 uint16_t dms_ActivationStatusTlv::TlvPresent

8.24 dms_OperatingModeTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t operatingMode

8.24.1 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

8.24.2 Field Documentation

8.24.2.1 uint32_t dms_OperatingModeTlv::operatingMode

8.24.2.2 uint16_t dms_OperatingModeTlv::TlvPresent

8.25 dms_PSMActiveTimerIndTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ActiveTimerInd

8.25.1 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).

8.25.2 Field Documentation

8.25.2.1 uint32_t dms_PSMActiveTimerIndTlv::ActiveTimerInd

8.25.2.2 uint8_t dms_PSMActiveTimerIndTlv::TlvPresent

8.26 dms_PSMActiveTimerTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t activeTimer

8.26.1 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).

8.26.2 Field Documentation

8.26.2.1 uint32_t dms_PSMActiveTimerTlv::activeTimer

8.26.2.2 uint8_t dms_PSMActiveTimerTlv::TlvPresent

8.27 dms_PSMDurationDueToOOSTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t durationDueToOOS

8.27.1 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).

8.27.2 Field Documentation

8.27.2.1 uint32_t dms_PSMDurationDueToOOSTlv::durationDueToOOS

8.27.2.2 uint8_t dms_PSMDurationDueToOOSTlv::TlvPresent

8.28 dms_PSMDurationThresholdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t durationThreshold

8.28.1 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).

8.28.2 Field Documentation

8.28.2.1 uint32_t dms_PSMDurationThresholdTlv::durationThreshold

8.28.2.2 uint8_t dms_PSMDurationThresholdTlv::TlvPresent

8.29 dms_PSMEarlyWakeupTimeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t earlyWakeupTime

8.29.1 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.

8.29.2 Field Documentation

8.29.2.1 uint32_t dms_PSMEarlyWakeupTimeTlv::earlyWakeupTime

8.29.2.2 uint8_t dms_PSMEarlyWakeupTimeTlv::TlvPresent

8.30 dms_PSMEnableStateIndTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EnableStateInd

8.30.1 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

8.30.2 Field Documentation

8.30.2.1 uint8_t dms_PSMEnableStateIndTlv::EnableStateInd

8.30.2.2 uint8_t dms_PSMEnableStateIndTlv::TlvPresent

8.31 dms_PSMEnableStateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t enableState

8.31.1 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

8.31.2 Field Documentation

8.31.2.1 uint8_t dms_PSMEnableStateTlv::enableState

8.31.2.2 uint8_t dms_PSMEnableStateTlv::TlvPresent

8.32 dms_PSMPeriodicUpdateTimerIndTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t PeriodicUpdateTimerInd

8.32.1 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).

8.32.2 Field Documentation

8.32.2.1 uint32_t dms_PSMPeriodicUpdateTimerIndTlv::PeriodicUpdateTimerInd

8.32.2.2 uint8_t dms_PSMPeriodicUpdateTimerIndTlv::TlvPresent

8.33 dms_PSMPeriodicUpdateTimerTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t periodicUpdateTimer

8.33.1 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).

8.33.2 Field Documentation

8.33.2.1 uint32_t dms_PSMPeriodicUpdateTimerTlv::periodicUpdateTimer

8.33.2.2 uint8_t dms_PSMPeriodicUpdateTimerTlv::TlvPresent

8.34 dms_PSMRandomizationWindowTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t randomizationWindow

8.34.1 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)

8.34.2 Field Documentation

8.34.2.1 uint32_t dms_PSMRandomizationWindowTlv::randomizationWindow

8.34.2.2 uint8_t dms_PSMRandomizationWindowTlv::TlvPresent

8.35 dms_TemperatureTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t TempStat
- uint16_t Temperature

8.35.1 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

8.35.2 Field Documentation

8.35.2.1 uint16_t dms_TemperatureTlv::Temperature

8.35.2.2 uint8_t dms_TemperatureTlv::TempStat

8.35.2.3 uint8_t dms_TemperatureTlv::TlvPresent

8.36 dms_UimAutoSwitchActSlotTlv Struct Reference

Data Fields

- uint8_t uimAutoSwitchActSlot
- uint8_t TlvPresent

8.36.1 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.

8.36.2 Field Documentation

8.36.2.1 uint8_t dms_UimAutoSwitchActSlotTlv::TlvPresent

8.36.2.2 uint8_t dms_UimAutoSwitchActSlotTlv::uimAutoSwitchActSlot

8.37 dms_UimStatusTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t intf
- uint8_t event

8.37.1 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.

8.37.2 Field Documentation

8.37.2.1 uint8_t dms_UimStatusTlv::event

8.37.2.2 uint8_t dms_UimStatusTlv::intf

8.37.2.3 uint8_t dms_UimStatusTlv::TlvPresent

8.38 dms_VoltageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t VoltStat
- uint16_t Voltage

8.38.1 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

8.38.2 Field Documentation

8.38.2.1 uint8_t dms_VoltageTlv::TlvPresent

8.38.2.2 uint16_t dms_VoltageTlv::Voltage

8.38.2.3 uint8_t dms_VoltageTlv::VoltStat

8.39 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

8.39.1 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

8.39.2 Field Documentation

8.39.2.1 uint16_t DMScustSettingInfo::cust_attr

8.39.2.2 uint8_t DMScustSettingInfo::cust_id[64+1]

8.39.2.3 uint8_t DMScustSettingInfo::cust_value[8+1]

8.39.2.4 uint16_t DMScustSettingInfo::id_length

8.39.2.5 uint16_t DMScustSettingInfo::value_length

8.40 DMScustSettingList Struct Reference

Data Fields

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

8.40.1 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

8.40.2 Field Documentation

8.40.2.1 DMScustSettingInfo DMScustSettingList::custSetting[255+1]

8.40.2.2 uint8_t DMScustSettingList::list_type

8.40.2.3 uint16_t DMScustSettingList::num_instances

8.41 DMSgetCustomFeatureV2 Struct Reference

Data Fields

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

8.41.1 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

8.41.2 Field Documentation

8.41.2.1 DMScustSettingInfo* DMSgetCustomFeatureV2::pCustSettingInfo

8.41.2.2 DMScustSettingList* DMSgetCustomFeatureV2::pCustSettingList

8.41.2.3 DMSgetCustomInput* DMSgetCustomFeatureV2::pGetCustomInput

8.42 DMSgetCustomInput Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint8_t list_type

8.42.1 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

8.42.2 Field Documentation

8.42.2.1 uint8_t DMSgetCustomInput::cust_id[64+1]

8.42.2.2 uint8_t DMSgetCustomInput::list_type

8.43 dunchannelRate Struct Reference

Data Fields

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

8.43.1 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.

8.43.2 Field Documentation

8.43.2.1 uint32_t dunchannelRate::CurrChanRxRate

8.43.2.2 uint32_t dunchannelRate::CurrChanTxRate

8.43.2.3 uint32_t dunchannelRate::MaxChanRxRate

8.43.2.4 uint32_t dunchannelRate::MaxChanTxRate

8.44 eriDataparams Struct Reference

Data Fields

- uint16_t eriDataLen
- uint8_t eriData [1024]

8.44.1 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)

8.44.2 Field Documentation

8.44.2.1 uint8_t eriDataparams::eriData[1024]

8.44.2.2 uint16_t eriDataparams::eriDataLen

8.45 eTWSPLMNInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsPlmnInfo ETWSPLMNInfo

8.45.1 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

8.45.2 Field Documentation

8.45.2.1 sMSEtwsPlmnInfo eTWSPLMNInfoTlv::ETWSPLMNInfo

8.45.2.2 uint8_t eTWSPLMNInfoTlv::TlvPresent

8.46 FMSImageElement Struct Reference

Data Fields

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

8.46.1 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

8.46.2 Field Documentation

8.46.2.1 uint8_t FMSImageElement::buildId[100]

8.46.2.2 uint8_t FMSImageElement::buildIdLength

8.46.2.3 uint8_t FMSImageElement::imageId[100]

8.46.2.4 uint8_t FMSImageElement::imageType

8.47 FMSImageIdElement Struct Reference

Data Fields

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

8.47.1 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)

8.47.2 Field Documentation

8.47.2.1 uint8_t FMSImageIdElement::buildID[100]

8.47.2.2 uint8_t FMSImageIdElement::buildIDLength

8.47.2.3 uint8_t FMSImageIdElement::failureCount

8.47.2.4 uint8_t FMSImageIdElement::imageID[100]

8.47.2.5 uint8_t FMSImageIdElement::storageIndex

8.48 FMSImageIDEntries Struct Reference

Data Fields

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

8.48.1 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

8.48.2 Field Documentation

8.48.2.1 uint8_t FMSImageIDEntries::executingImage

8.48.2.2 FMSImageIdElement FMSImageIDEntries::imageIDElement[50]

8.48.2.3 uint8_t FMSImageIDEntries::imageIDSize

8.48.2.4 uint8_t FMSImageIDEntries::imageType

8.48.2.5 uint8_t FMSImageIDEntries::maxImages

8.49 FMSImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageIDEntries imageIDEntries [2]

8.49.1 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)

8.49.2 Field Documentation

8.49.2.1 FMSImageIDEntries FMSImageList::imageIDEntries[2]

8.49.2.2 uint8_t FMSImageList::listSize

8.50 FMSPrefImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageElement listEntries [2]

8.50.1 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

8.50.2 Field Documentation

8.50.2.1 FMSImageElement FMSPrefImageList::listEntries[2]

8.50.2.2 uint8_t FMSPrefImageList::listSize

8.51 hdrSSInfo Struct Reference

Data Fields

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

8.51.1 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.

8.51.2 Field Documentation

8.51.2.1 int16_t hdrSSInfo::ecio

8.51.2.2 int32_t hdrSSInfo::io

8.51.2.3 int8_t hdrSSInfo::rssi

8.51.2.4 uint8_t hdrSSInfo::sinr

8.52 image_info_t Struct Reference

Data Fields

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

8.52.1 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

8.52.2 Field Documentation

8.52.2.1 uint8_t image_info_t::buildID[255]

8.52.2.2 uint8_t image_info_t::buildIDLen

8.52.2.3 uint8_t image_info_t::imageType

8.52.2.4 uint8_t image_info_t::uniqueID[16]

8.53 ims_AMRModeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrMode

8.53.1 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

8.53.2 Field Documentation

8.53.2.1 uint8_t ims_AMRModeInfo::amrMode

8.53.2.2 uint8_t ims_AMRModeInfo::TlvPresent

8.54 ims_AMROctAlgnInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrOctAlgn

8.54.1 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

8.54.2 Field Documentation

8.54.2.1 uint8_t ims_AMROctAlgnInfo::amrOctAlgn

8.54.2.2 uint8_t ims_AMROctAlgnInfo::TlvPresent

8.55 ims_AMRWBModeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t amrWBMode

8.55.1 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

8.55.2 Field Documentation

8.55.2.1 uint16_t ims_AMRWBModeInfo::amrWBMode

8.55.2.2 uint8_t ims_AMRWBModeInfo::TlvPresent

8.56 ims_AMRWBOctAlgnInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrWBOctAlgn

8.56.1 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

8.56.2 Field Documentation

8.56.2.1 uint8_t ims_AMRWBOctAlgnInfo::amrWBOctAlgn

8.56.2.2 uint8_t ims_AMRWBOctAlgnInfo::TlvPresent

8.57 ims_CSCFPortNameInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t cscfPortName [255]

8.57.1 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

8.57.2 Field Documentation

8.57.2.1 uint8_t ims_CSCFPortNameInfo::cscfPortName[255]

8.57.2.2 uint8_t ims_CSCFPortNameInfo::TlvPresent

8.58 ims_EnabAMRWBInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t amrWBEnable

8.58.1 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

8.58.2 Field Documentation

8.58.2.1 uint8_t ims_EnabAMRWBInfo::amrWBEnable

8.58.2.2 uint8_t ims_EnabAMRWBInfo::TlvPresent

8.59 ims_EnabSCRAMRInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t scrAmrEnable

8.59.1 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

8.59.2 Field Documentation

8.59.2.1 uint8_t ims_EnabSCRAMRInfo::scrAmrEnable

8.59.2.2 uint8_t ims_EnabSCRAMRInfo::TlvPresent

8.60 ims_EnabSCRAMRWBInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t scrAmrWBEnable

8.60.1 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

8.60.2 Field Documentation

8.60.2.1 uint8_t ims_EnabSCRAMRWBInfo::scrAmrWBEnable

8.60.2.2 uint8_t ims_EnabSCRAMRWBInfo::TlvPresent

8.61 ims_IMSDomainInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t imsDomainName [255]

8.61.1 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

8.61.2 Field Documentation

8.61.2.1 uint8_t ims_IMSDomainInfo::imsDomainName[255]

8.61.2.2 uint8_t ims_IMSDomainInfo::TlvPresent

8.62 ims_IMSTestModeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t imsTestMode

8.62.1 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

8.62.2 Field Documentation

8.62.2.1 uint8_t ims_IMSTestModeInfo::imsTestMode

8.62.2.2 uint8_t ims_IMSTestModeInfo::TlvPresent

8.63 ims_MinSessExpInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t minSessExp

8.63.1 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

8.63.2 Field Documentation

8.63.2.1 uint16_t ims_MinSessExpInfo::minSessExp

8.63.2.2 uint8_t ims_MinSessExpInfo::TlvPresent

8.64 ims_PCSCFPortInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t priCSCFPort

8.64.1 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

8.64.2 Field Documentation

8.64.2.1 uint16_t ims_PCSCFPortInfo::priCSCFPort

8.64.2.2 uint8_t ims_PCSCFPortInfo::TlvPresent

8.65 ims_PhCtxtURIInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t PhCtxtURI [255]

8.65.1 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

8.65.2 Field Documentation

8.65.2.1 uint8_t ims_PhCtxtURIInfo::PhCtxtURI[255]

8.65.2.2 uint8_t ims_PhCtxtURIInfo::TlvPresent

8.66 ims_RngBkTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RingBkTmr

8.66.1 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

8.66.2 Field Documentation

8.66.2.1 uint16_t ims_RngBkTmrInfo::RingBkTmr

8.66.2.2 uint8_t ims_RngBkTmrInfo::TlvPresent

8.67 ims_RngTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RingTmr

8.67.1 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

8.67.2 Field Documentation

8.67.2.1 uint16_t ims_RngTmrInfo::RingTmr

8.67.2.2 uint8_t ims_RngTmrInfo::TlvPresent

8.68 ims_RTPRTCPInactTmrDurInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t InactTmr

8.68.1 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

8.68.2 Field Documentation

8.68.2.1 uint16_t ims_RTPRTCPInactTmrDurInfo::InactTmr

8.68.2.2 uint8_t ims_RTPRTCPInactTmrDurInfo::TlvPresent

8.69 ims_SessDurInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t sessExp

8.69.1 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

8.69.2 Field Documentation

8.69.2.1 uint16_t ims_SessDurInfo::sessExp

8.69.2.2 uint8_t ims_SessDurInfo::TlvPresent

8.70 ims_SigCompEnInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t SigCompEn

8.70.1 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

8.70.2 Field Documentation

8.70.2.1 uint8_t ims_SigCompEnInfo::SigCompEn

8.70.2.2 uint8_t ims_SigCompEnInfo::TlvPresent

8.71 ims_SIPPortInfo Struct Reference

Data Fields

- uint8_t TlvPresent

- uint16_t SIPLocalPort

8.71.1 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

8.71.2 Field Documentation

8.71.2.1 uint16_t ims_SIPPortInfo::SIPLocalPort

8.71.2.2 uint8_t ims_SIPPortInfo::TlvPresent

8.72 ims_SIPRegnTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrSIPRegn

8.72.1 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

8.72.2 Field Documentation

8.72.2.1 uint8_t ims_SIPRegnTmrInfo::TlvPresent

8.72.2.2 uint32_t ims_SIPRegnTmrInfo::tmrSIPRegn

8.73 ims_SMSFmtInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t smsFormat

8.73.1 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

8.73.2 Field Documentation

8.73.2.1 uint8_t ims_SMSFmtInfo::smsFormat

8.73.2.2 uint8_t ims_SMSFmtInfo::TlvPresent

8.74 ims_SMSoIPNwInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t smsoIPNW

8.74.1 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

8.74.2 Field Documentation

8.74.2.1 uint8_t ims_SMSoIPNwInfo::smsoIPNW

8.74.2.2 uint8_t ims_SMSoIPNwInfo::TlvPresent

8.75 ims_SubscrTmrInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t subscrTmr

8.75.1 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.

8.75.2 Field Documentation

8.75.2.1 uint32_t ims_SubscrTmrInfo::subscrTmr

8.75.2.2 uint8_t ims_SubscrTmrInfo::TlvPresent

8.76 ims_TmrT1Info Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrT1

8.76.1 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.

8.76.2 Field Documentation

8.76.2.1 uint8_t ims_TmrT1Info::TlvPresent

8.76.2.2 uint32_t ims_TmrT1Info::tmrT1

8.77 ims_TmrT2Info Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrT2

8.77.1 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.

8.77.2 Field Documentation

8.77.2.1 uint8_t ims_TmrT2Info::TlvPresent

8.77.2.2 uint32_t ims_TmrT2Info::tmrT2

8.78 ims_TmrTfInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t tmrTf

8.78.1 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

8.78.2 Field Documentation

8.78.2.1 uint8_t ims_TmrTfInfo::TlvPresent

8.78.2.2 uint32_t ims_TmrTfInfo::tmrTf

8.79 imsa_IMSFailErrCodeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ImsFailErrCode

8.79.1 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

8.79.2 Field Documentation

8.79.2.1 uint32_t imsa_IMSFailErrCodeTlv::ImsFailErrCode

8.79.2.2 uint8_t imsa_IMSFailErrCodeTlv::TlvPresent

8.80 imsa_IMSRegStatusErrorCodeInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ErrorCode

8.80.1 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

8.80.2 Field Documentation

8.80.2.1 uint16_t imsa_IMSRegStatusErrorCodeInfo::ErrorCode

8.80.2.2 uint8_t imsa_IMSRegStatusErrorCodeInfo::TlvPresent

8.81 imsa_IMSRegStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t ImsRegistered

8.81.1 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

8.81.2 Field Documentation

8.81.2.1 uint8_t imsa_IMSRegStatusInfo::ImsRegistered

8.81.2.2 uint8_t imsa_IMSRegStatusInfo::TlvPresent

8.82 imsa_NewIMSRegStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ImsRegStatus

8.82.1 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

8.82.2 Field Documentation

8.82.2.1 uint32_t imsa_NewIMSRegStatusInfo::ImsRegStatus

8.82.2.2 uint8_t imsa_NewIMSRegStatusInfo::TlvPresent

8.83 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]

8.83.1 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

8.83.2 Field Documentation

8.83.2.1 uint8_t imsa_RatHandoverStatusInfo::ErrorCodeData[256]

8.83.2.2 uint8_t imsa_RatHandoverStatusInfo::ErrorCodeLen

8.83.2.3 uint32_t imsa_RatHandoverStatusInfo::RatHandoverStatus

8.83.2.4 uint32_t imsa_RatHandoverStatusInfo::SourceRAT

8.83.2.5 uint32_t imsa_RatHandoverStatusInfo::TargetRAT

8.83.2.6 uint8_t imsa_RatHandoverStatusInfo::TlvPresent

8.84 imsa_SmsRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SmsRatVal

8.84.1 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

8.84.2 Field Documentation

8.84.2.1 uint32_t imsa_SmsRatInfo::SmsRatVal

8.84.2.2 uint8_t imsa_SmsRatInfo::TlvPresent

8.85 imsa_SmsSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SmsSvcStatus

8.85.1 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

8.85.2 Field Documentation

8.85.2.1 uint32_t imsa_SmsSvcStatusInfo::SmsSvcStatus

8.85.2.2 uint8_t imsa_SmsSvcStatusInfo::TlvPresent

8.86 imsa_UtRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t UtRatVal

8.86.1 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

8.86.2 Field Documentation

8.86.2.1 uint8_t imsa_UtRatInfo::TlvPresent

8.86.2.2 uint32_t imsa_UtRatInfo::UtRatVal

8.87 imsa_UtSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t UtSvcStatus

8.87.1 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

8.87.2 Field Documentation

8.87.2.1 uint8_t imsa_UtSvcStatusInfo::TlvPresent

8.87.2.2 uint32_t imsa_UtSvcStatusInfo::UtSvcStatus

8.88 imsa_VoipRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t VoipRatVal

8.88.1 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

8.88.2 Field Documentation

8.88.2.1 uint8_t imsa_VoipRatInfo::TlvPresent

8.88.2.2 uint32_t imsa_VoipRatInfo::VoipRatVal

8.89 imsa_VoipSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t VoipSvcStatus

8.89.1 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

8.89.2 Field Documentation

8.89.2.1 uint8_t imsa_VoipSvcStatusInfo::TlvPresent

8.89.2.2 uint32_t imsa_VoipSvcStatusInfo::VoipSvcStatus

8.90 imsa_VtRatInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t VtRatVal

8.90.1 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

8.90.2 Field Documentation

8.90.2.1 uint8_t imsa_VtRatInfo::TlvPresent

8.90.2.2 uint32_t imsa_VtRatInfo::VtRatVal

8.91 imsa_VtSvcStatusInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t VtSvcStatus

8.91.1 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

8.91.2 Field Documentation

8.91.2.1 uint8_t imsa_VtSvcStatusInfo::TlvPresent

8.91.2.2 uint32_t imsa_VtSvcStatusInfo::VtSvcStatus

8.92 ipv6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.92.1 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)

8.92.2 Field Documentation

8.92.2.1 uint16_t ipv6AddressInfo::IPAddressV6[8]

8.92.2.2 uint8_t ipv6AddressInfo::IPV6PrefixLen

8.93 LibPackGPRSRequestedQoS Struct Reference

Data Fields

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

8.93.1 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

8.93.2 Field Documentation

8.93.2.1 uint32_t LibPackGPRSRequestedQoS::delayClass

8.93.2.2 uint32_t LibPackGPRSRequestedQoS::meanThroughputClass

8.93.2.3 uint32_t LibPackGPRSRequestedQoS::peakThroughputClass

8.93.2.4 uint32_t LibPackGPRSRequestedQoS::precedenceClass

8.93.2.5 uint32_t LibPackGPRSRequestedQoS::reliabilityClass

8.94 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

8.94.1 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 • 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**

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 • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings(). • 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 • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings(). • 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 SLQSGetProfileSettings(). • Bit to check in ParamPresenceMask - **49**
pSupportEmergencyCalls	• 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**

8.94.2 Field Documentation

8.94.2.1 uint8_t* LibpackProfile3GPP::pAddrAllocPref

8.94.2.2 uint8_t* LibpackProfile3GPP::pAPNClass

8.94.2.3 uint8_t* LibpackProfile3GPP::pAPNDisabledFlag

8.94.2.4 uint8_t* LibpackProfile3GPP::pAPNName

8.94.2.5 uint16_t* LibpackProfile3GPP::pAPNnameSize

8.94.2.6 uint8_t* LibpackProfile3GPP::pAuthenticationPref

8.94.2.7 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSMinimumQoS

8.94.2.8 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSRequestedQos

8.94.2.9 uint8_t* LibpackProfile3GPP::pImCnFlag

8.94.2.10 uint32_t* LibpackProfile3GPP::pIPv4AddrPref

8.94.2.11 uint16_t* LibpackProfile3GPP::pIPv6AddPref

8.94.2.12 uint8_t* LibpackProfile3GPP::pPassword

8.94.2.13 uint16_t* LibpackProfile3GPP::pPasswordSize

8.94.2.14 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingDhcp

8.94.2.15 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingPCO

8.94.2.16 uint32_t* LibpackProfile3GPP::pPDNInactivTimeout

8.94.2.17 uint8_t* LibpackProfile3GPP::pPdpAccessConFlag

8.94.2.18 uint8_t* LibpackProfile3GPP::pPdpContext

8.94.2.19 uint8_t* LibpackProfile3GPP::pPdpDataCompType

8.94.2.20 uint8_t* LibpackProfile3GPP::pPdpHdrCompType

8.94.2.21 uint8_t* LibpackProfile3GPP::pPDPtype

8.94.2.22 uint32_t* LibpackProfile3GPP::pPriDNSIPv4AddPref

8.94.2.23 uint16_t* LibpackProfile3GPP::pPriDNSIPv6addpref

8.94.2.24 uint8_t* LibpackProfile3GPP::pPrimaryID

8.94.2.25 uint8_t* LibpackProfile3GPP::pProfilename

8.94.2.26 uint16_t* LibpackProfile3GPP::pProfilenameSize

8.94.2.27 LibPackQosClassID* LibpackProfile3GPP::pQosClassID

8.94.2.28 uint32_t* LibpackProfile3GPP::pSecDNSIPv4AddPref

8.94.2.29 uint16_t* LibpackProfile3GPP::pSecDNSIPv6addpref

8.94.2.30 uint8_t* LibpackProfile3GPP::pSecondaryFlag

8.94.2.31 uint8_t* LibpackProfile3GPP::pSupportEmergencyCalls

8.94.2.32 **LibPackTFTIDParams*** LibpackProfile3GPP::pTFTID1Params

8.94.2.33 **LibPackTFTIDParams*** LibpackProfile3GPP::pTFTID2Params

8.94.2.34 **LibPackUMTSQoS*** LibpackProfile3GPP::pUMTSMinQoS

8.94.2.35 **LibPackUMTSReqQoSSigInd*** LibpackProfile3GPP::pUMTSMinQosSigInd

8.94.2.36 **LibPackUMTSQoS*** LibpackProfile3GPP::pUMTSReqQoS

8.94.2.37 **LibPackUMTSReqQoSSigInd*** LibpackProfile3GPP::pUMTSReqQoSSigInd

8.94.2.38 uint8_t* LibpackProfile3GPP::pUsername

8.94.2.39 uint16_t* LibpackProfile3GPP::pUsernameSize

8.95 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

8.95.1 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**

8.95.2 Field Documentation

8.95.2.1 uint8_t* LibpackProfile3GPP2::pAllowLinger

8.95.2.2 uint8_t* LibpackProfile3GPP2::pAPNClass3GPP2

8.95.2.3 uint8_t* LibpackProfile3GPP2::pAPNEnabled3GPP2

8.95.2.4 uint8_t* LibpackProfile3GPP2::pApnString

8.95.2.5 uint16_t* LibpackProfile3GPP2::pApnStringSize

8.95.2.6 uint8_t* LibpackProfile3GPP2::pAppPriority

8.95.2.7 uint32_t* LibpackProfile3GPP2::pAppType

8.95.2.8 uint8_t* LibpackProfile3GPP2::pAuthPassword

8.95.2.9 uint16_t* LibpackProfile3GPP2::pAuthPasswordSize

8.95.2.10 uint8_t* LibpackProfile3GPP2::pAuthProtocol

8.95.2.11 uint8_t* LibpackProfile3GPP2::pAuthRetryCount

8.95.2.12 uint16_t* LibpackProfile3GPP2::pAuthTimeout

8.95.2.13 uint8_t* LibpackProfile3GPP2::pDataMode

8.95.2.14 uint8_t* LibpackProfile3GPP2::pDataRate

8.95.2.15 uint16_t* LibpackProfile3GPP2::pIpcpAckTimeout

8.95.2.16 uint8_t* LibpackProfile3GPP2::pIpcpCreqRetryCount

8.95.2.17 uint8_t* LibpackProfile3GPP2::pIsPcscfAddressNedded

8.95.2.18 uint16_t* LibpackProfile3GPP2::pLcpAckTimeout

8.95.2.19 uint8_t* LibpackProfile3GPP2::pLcpCreqRetryCount

8.95.2.20 uint8_t* LibpackProfile3GPP2::pNegoDnsSrvrPref

8.95.2.21 uint32_t* LibpackProfile3GPP2::pPDNInactivTimeout3GPP2

8.95.2.22 uint8_t* LibpackProfile3GPP2::pPdnType

8.95.2.23 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimer1x

8.95.2.24 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimerDO

8.95.2.25 uint32_t* LibpackProfile3GPP2::pPrimaryV4DnsAddress

8.95.2.26 uint16_t* LibpackProfile3GPP2::pPriV6DnsAddress

8.95.2.27 uint8_t* LibpackProfile3GPP2::pRATType

8.95.2.28 uint32_t* LibpackProfile3GPP2::pSecondaryV4DnsAddress

8.95.2.29 uint16_t* LibpackProfile3GPP2::pSecV6DnsAddress

8.95.2.30 uint8_t* LibpackProfile3GPP2::pUserId

8.95.2.31 uint16_t* LibpackProfile3GPP2::pUserIdSize

8.96 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

8.96.1 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

pSecondaryDNSIPv6addpref	• Secondary DNS IPv6 address preference
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
pQosClassID	• 3GPP LTE QoS parameters
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 • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
pPDNInactivTimeout	• 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 SLQSGetProfileSettings().
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 SLQSGetProfileSettings().
pSupportEmergencyCalls	• 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

8.96.2 Field Documentation

8.96.2.1 uint8_t* LibPackprofile_3GPP::pAddrAllocPref

8.96.2.2 uint8_t* LibPackprofile_3GPP::pAPNClass

8.96.2.3 uint8_t* LibPackprofile_3GPP::pAPNDisabledFlag

8.96.2.4 uint8_t* LibPackprofile_3GPP::pAPNName

8.96.2.5 uint16_t* LibPackprofile_3GPP::pAPNnameSize

8.96.2.6 uint8_t* LibPackprofile_3GPP::pAuthenticationPref

8.96.2.7 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSMinimumQoS

8.96.2.8 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSRequestedQos

8.96.2.9 uint8_t* LibPackprofile_3GPP::pImCnFlag

8.96.2.10 uint32_t* LibPackprofile_3GPP::pIPv4AddrPref

8.96.2.11 uint16_t* LibPackprofile_3GPP::pIPv6AddPref

8.96.2.12 uint8_t* LibPackprofile_3GPP::pPassword

8.96.2.13 uint16_t* LibPackprofile_3GPP::pPasswordSize

8.96.2.14 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingDhcp

8.96.2.15 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingPCO

8.96.2.16 uint32_t* LibPackprofile_3GPP::pPDNInactivTimeout

8.96.2.17 uint8_t* LibPackprofile_3GPP::pPdpAccessConFlag

8.96.2.18 uint8_t* LibPackprofile_3GPP::pPdpContext

8.96.2.19 uint8_t* LibPackprofile_3GPP::pPdpDataCompType

8.96.2.20 uint8_t* LibPackprofile_3GPP::pPdpHdrCompType

8.96.2.21 uint8_t* LibPackprofile_3GPP::pPDPtype

8.96.2.22 uint32_t* LibPackprofile_3GPP::pPriDNSIPv4AddPref

8.96.2.23 uint16_t* LibPackprofile_3GPP::pPriDNSIPv6addpref

8.96.2.24 uint8_t* LibPackprofile_3GPP::pPrimaryID

8.96.2.25 uint8_t* LibPackprofile_3GPP::pProfilename

8.96.2.26 uint16_t* LibPackprofile_3GPP::pProfilenameSize

8.96.2.27 LibPackQosClassID* LibPackprofile_3GPP::pQosClassID

8.96.2.28 uint32_t* LibPackprofile_3GPP::pSecDNSIPv4AddPref

8.96.2.29 uint16_t* LibPackprofile_3GPP::pSecDNSIPv6addpref

8.96.2.30 uint8_t* LibPackprofile_3GPP::pSecondaryFlag

8.96.2.31 uint8_t* LibPackprofile_3GPP::pSupportEmergencyCalls

8.96.2.32 **LibPackTFTIDParams*** LibPackprofile_3GPP::pTFTID1Params

8.96.2.33 **LibPackTFTIDParams*** LibPackprofile_3GPP::pTFTID2Params

8.96.2.34 **LibPackUMTSQoS*** LibPackprofile_3GPP::pUMTSMinQoS

8.96.2.35 **LibPackUMTSReqQoSSigInd*** LibPackprofile_3GPP::pUMTSMinQosSigInd

8.96.2.36 **LibPackUMTSQoS*** LibPackprofile_3GPP::pUMTSReqQoS

8.96.2.37 **LibPackUMTSReqQoSSigInd*** LibPackprofile_3GPP::pUMTSReqQoSSigInd

8.96.2.38 **uint8_t*** LibPackprofile_3GPP::pUsername

8.96.2.39 **uint16_t*** LibPackprofile_3GPP::pUsernameSize

8.97 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

8.97.1 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().

8.97.2 Field Documentation

8.97.2.1 uint8_t* LibPackprofile_3GPP2::pAllowLinger

8.97.2.2 uint8_t* LibPackprofile_3GPP2::pAPNClass3GPP2

8.97.2.3 uint8_t* LibPackprofile_3GPP2::pAPNEnabled3GPP2

8.97.2.4 uint8_t* LibPackprofile_3GPP2::pApnString

8.97.2.5 uint16_t* LibPackprofile_3GPP2::pApnStringSize

8.97.2.6 uint8_t* LibPackprofile_3GPP2::pAppPriority

8.97.2.7 uint32_t* LibPackprofile_3GPP2::pAppType

8.97.2.8 uint8_t* LibPackprofile_3GPP2::pAuthPassword

8.97.2.9 uint16_t* LibPackprofile_3GPP2::pAuthPassword_tSize

8.97.2.10 uint8_t* LibPackprofile_3GPP2::pAuthProtocol

8.97.2.11 uint8_t* LibPackprofile_3GPP2::pAuthRetryCount

8.97.2.12 uint16_t* LibPackprofile_3GPP2::pAuthTimeout

8.97.2.13 uint8_t* LibPackprofile_3GPP2::pDataMode

8.97.2.14 uint8_t* LibPackprofile_3GPP2::pDataRate

8.97.2.15 uint16_t* LibPackprofile_3GPP2::plpcpAckTimeout

8.97.2.16 uint8_t* LibPackprofile_3GPP2::plpcpCreqRetryCount

8.97.2.17 uint8_t* LibPackprofile_3GPP2::plsPcscfAddressNedded

8.97.2.18 uint16_t* LibPackprofile_3GPP2::pLcpAckTimeout

8.97.2.19 uint8_t* LibPackprofile_3GPP2::pLcpCreqRetryCount

8.97.2.20 uint8_t* LibPackprofile_3GPP2::pNegoDnsSrvrPref

8.97.2.21 uint32_t* LibPackprofile_3GPP2::pPDNInactivTimeout3GPP2

8.97.2.22 uint8_t* LibPackprofile_3GPP2::pPdnType

8.97.2.23 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimer1x

8.97.2.24 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimerDO

8.97.2.25 uint32_t* LibPackprofile_3GPP2::pPrimaryV4DnsAddress

8.97.2.26 uint16_t* LibPackprofile_3GPP2::pPriV6DnsAddress

8.97.2.27 uint8_t* LibPackprofile_3GPP2::pRATType

8.97.2.28 uint32_t* LibPackprofile_3GPP2::pSecondaryV4DnsAddress

8.97.2.29 uint16_t* LibPackprofile_3GPP2::pSecV6DnsAddress

8.97.2.30 uint8_t* LibPackprofile_3GPP2::pUserId

8.97.2.31 uint16_t* LibPackprofile_3GPP2::pUserIdSize

8.98 LibPackQosClassID Struct Reference

Data Fields

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

8.98.1 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

8.98.2 Field Documentation

8.98.2.1 uint32_t LibPackQosClassID::gDlBitRate

8.98.2.2 uint32_t LibPackQosClassID::gUlBitRate

8.98.2.3 uint32_t LibPackQosClassID::maxDlBitRate

8.98.2.4 uint32_t LibPackQosClassID::maxUlBitRate

8.98.2.5 uint8_t LibPackQosClassID::QCI

8.99 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

8.99.1 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

8.99.2 Field Documentation

8.99.2.1 uint16_t LibPackTFTIDParams::destPortRangeEnd

8.99.2.2 uint16_t LibPackTFTIDParams::destPortRangeStart

8.99.2.3 uint8_t LibPackTFTIDParams::eValid

8.99.2.4 uint8_t LibPackTFTIDParams::filterId

8.99.2.5 uint32_t LibPackTFTIDParams::flowLabel

8.99.2.6 uint32_t LibPackTFTIDParams::IPSECSPI

8.99.2.7 uint8_t LibPackTFTIDParams::ipVersion

8.99.2.8 uint8_t LibPackTFTIDParams::nextHeader

8.99.2.9 uint16_t* LibPackTFTIDParams::pSourceIP

8.99.2.10 uint8_t LibPackTFTIDParams::sourceIPMask

8.99.2.11 uint16_t LibPackTFTIDParams::srcPortRangeEnd

8.99.2.12 uint16_t LibPackTFTIDParams::srcPortRangeStart

8.99.2.13 uint16_t LibPackTFTIDParams::tosMask

8.100 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

8.100.1 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.

8.100.2 Field Documentation

8.100.2.1 uint8_t LibPackUMTSQoS::deliveryErrSDU

8.100.2.2 uint32_t LibPackUMTSQoS::grntDownlinkBitrate

8.100.2.3 uint32_t LibPackUMTSQoS::grntUplinkBitrate

8.100.2.4 uint32_t LibPackUMTSQoS::maxDownlinkBitrate

8.100.2.5 uint32_t LibPackUMTSQoS::maxSDUSize

8.100.2.6 uint32_t LibPackUMTSQoS::maxUplinkBitrate

8.100.2.7 uint8_t LibPackUMTSQoS::qosDeliveryOrder

8.100.2.8 uint8_t LibPackUMTSQoS::resBerRatio

8.100.2.9 uint8_t LibPackUMTSQoS::sduErrorRatio

8.100.2.10 uint8_t LibPackUMTSQoS::trafficClass

8.100.2.11 uint32_t LibPackUMTSQoS::trafficPriority

8.100.2.12 uint32_t LibPackUMTSQoS::transferDelay

8.101 LibPackUMTSReqQoSSigInd Struct Reference

Data Fields

- LibPackUMTSQoS UMTSReqQoS
- uint8_t SigInd

8.101.1 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

8.101.2 Field Documentation

8.101.2.1 uint8_t LibPackUMTSReqQoSSigInd::SigInd

8.101.2.2 LibPackUMTSQoS LibPackUMTSReqQoSSigInd::UMTSReqQoS

8.102 loc_accelAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.102.1 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.

8.102.2 Field Documentation

8.102.2.1 uint16_t loc_accelAcceptReady::batchPerSec

8.102.2.2 uint8_t loc_accelAcceptReady::injectEnable

8.102.2.3 uint16_t loc_accelAcceptReady::samplesPerBatch

8.103 loc_accelTempAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.103.1 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.

8.103.2 Field Documentation

8.103.2.1 uint16_t loc_accelTempAcceptReady::batchPerSec

8.103.2.2 uint8_t loc_accelTempAcceptReady::injectEnable

8.103.2.3 uint16_t loc_accelTempAcceptReady::samplesPerBatch

8.104 loc_BdsSV Struct Reference

Data Fields

- uint16_t id
- uint8_t mask

8.104.1 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

8.104.2 Field Documentation

8.104.2.1 uint16_t loc_BdsSV::id

8.104.2.2 uint8_t loc_BdsSV::mask

8.105 loc_BdsSVInfo Struct Reference

Data Fields

- uint8_t len
- loc_BdsSV * pSV

8.105.1 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

8.105.2 Field Documentation

8.105.2.1 uint8_t loc_BdsSVInfo::len

8.105.2.2 loc_BdsSV* loc_BdsSVInfo::pSV

8.106 loc_CellDb Struct Reference

Data Fields

- uint32_t mask

8.106.1 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

8.106.2 Field Documentation

8.106.2.1 uint32_t loc_CellDb::mask

8.107 loc_ClkInfo Struct Reference

Data Fields

- uint32_t mask

8.107.1 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

8.107.2 Field Documentation

8.107.2.1 uint32_t loc_ClkInfo::mask

8.108 loc_GnssData Struct Reference

Data Fields

- uint64_t mask

8.108.1 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

8.108.2 Field Documentation

8.108.2.1 uint64_t loc_GnssData::mask

8.109 loc_gpsTime Struct Reference

Data Fields

- uint16_t gpsWeek
- uint32_t gpsTimeOfWeekMs

8.109.1 Detailed Description

This structure contains GPS Time info.

Parameters

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

8.109.2 Field Documentation

8.109.2.1 uint32_t loc_gpsTime::gpsTimeOfWeekMs

8.109.2.2 uint16_t loc_gpsTime::gpsWeek

8.110 loc_gyroAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.110.1 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
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.

8.110.2 Field Documentation

8.110.2.1 uint16_t loc_gyroAcceptReady::batchPerSec

8.110.2.2 uint8_t loc_gyroAcceptReady::injectEnable

8.110.2.3 uint16_t loc_gyroAcceptReady::samplesPerBatch

8.111 loc_gyroTempAcceptReady Struct Reference

Data Fields

- uint8_t injectEnable
- uint16_t samplesPerBatch
- uint16_t batchPerSec

8.111.1 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.

8.111.2 Field Documentation

8.111.2.1 uint16_t loc_gyroTempAcceptReady::batchPerSec

8.111.2.2 uint8_t loc_gyroTempAcceptReady::injectEnable

8.111.2.3 uint16_t loc_gyroTempAcceptReady::samplesPerBatch

8.112 loc_IPv4Config Struct Reference

Data Fields

- uint32_t IPv4Addr
- uint16_t IPv4Port

8.112.1 Detailed Description

This structure specifies information regarding the IPv4 address and port.

Parameters

IPv4Addr	• IPv4 address.
IPv4Port	• IPv4 port.

8.112.2 Field Documentation

8.112.2.1 uint32_t loc_IPv4Config::IPv4Addr

8.112.2.2 uint16_t loc_IPv4Config::IPv4Port

8.113 loc_IPv4Info Struct Reference

Data Fields

- uint32_t address
- uint16_t port
- uint8_t TlvPresent

8.113.1 Detailed Description

This structure contains LOC Server IPV4 info field.

Parameters

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

8.113.2 Field Documentation

8.113.2.1 uint32_t loc_IPv4Info::address

8.113.2.2 uint16_t loc_IPv4Info::port

8.113.2.3 uint8_t loc_IPv4Info::TlvPresent

8.114 loc_IPv6Config Struct Reference

Data Fields

- uint16_t IPv6Addr [8]
- uint32_t IPv6Port

8.114.1 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.

8.114.2 Field Documentation

8.114.2.1 uint16_t loc_IPv6Config::IPv6Addr[8]

8.114.2.2 uint32_t loc_IPv6Config::IPv6Port

8.115 loc_IPv6Info Struct Reference

Data Fields

- uint16_t address [8]
- uint32_t port
- uint8_t TlvPresent

8.115.1 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

8.115.2 Field Documentation

8.115.2.1 uint16_t loc_IPv6Info::address[8]

8.115.2.2 uint32_t loc_IPv6Info::port

8.115.2.3 uint8_t loc_IPv6Info::TlvPresent

8.116 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

8.116.1 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

8.116.2 Field Documentation

8.116.2.1 uint8_t loc_LocApplicationInfo::appNameLength

8.116.2.2 uint8_t loc_LocApplicationInfo::appProviderLength

8.116.2.3 uint8_t loc_LocApplicationInfo::appVersionLength

8.116.2.4 uint8_t loc_LocApplicationInfo::appVersionValid

8.116.2.5 uint8_t* loc_LocApplicationInfo::pAppName

8.116.2.6 uint8_t* loc_LocApplicationInfo::pAppProvider

8.116.2.7 uint8_t* loc_LocApplicationInfo::pAppVersion

8.117 loc_precisionDilution Struct Reference

Data Fields

- uint32_t PDOP
- uint32_t HDOP
- uint32_t VDOP

8.117.1 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)

8.117.2 Field Documentation

8.117.2.1 uint32_t loc_precisionDilution::HDOP

8.117.2.2 uint32_t loc_precisionDilution::PDOP

8.117.2.3 uint32_t loc_precisionDilution::VDOP

8.118 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

8.118.1 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

8.118.2 Field Documentation

8.118.2.1 float loc_satelliteInfo::azimuth

8.118.2.2 float loc_satelliteInfo::elevation

8.118.2.3 uint16_t loc_satelliteInfo::gnssSvId

8.118.2.4 uint8_t loc_satelliteInfo::healthStatus

8.118.2.5 float loc_satelliteInfo::snr

8.118.2.6 uint8_t loc_satelliteInfo::svInfoMask

8.118.2.7 uint8_t loc_satelliteInfo::svListLen

8.118.2.8 uint32_t loc_satelliteInfo::svStatus

8.118.2.9 uint32_t loc_satelliteInfo::system

8.118.2.10 uint32_t loc_satelliteInfo::validMask

8.119 loc_sensorDataUsage Struct Reference

Data Fields

- uint32_t usageMask
- uint32_t aidingIndicatorMask

8.119.1 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

8.119.2 Field Documentation

8.119.2.1 uint32_t loc_sensorDataUsage::aidingIndicatorMask

8.119.2.2 uint32_t loc_sensorDataUsage::usageMask

8.120 loc_SV Struct Reference

Data Fields

- uint16_t id
- uint32_t system
- uint8_t mask

8.120.1 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

8.120.2 Field Documentation

8.120.2.1 uint16_t loc_SV::id

8.120.2.2 uint8_t loc_SV::mask

8.120.2.3 uint32_t loc_SV::system

8.121 loc_SVInfo Struct Reference

Data Fields

- uint8_t len
- loc_SV * pSV

8.121.1 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

8.121.2 Field Documentation

8.121.2.1 uint8_t loc_SVInfo::len

8.121.2.2 loc_SV* loc_SVInfo::pSV

8.122 loc_svUsedforFix Struct Reference

Data Fields

- uint8_t gnssSvUsedList_len
- uint16_t gnssSvUsedList [255]

8.122.1 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

8.122.2 Field Documentation

8.122.2.1 uint16_t loc_svUsedforFix::gnssSvUsedList[255]

8.122.2.2 uint8_t loc_svUsedforFix::gnssSvUsedList_len

8.123 loc_urlAddr Struct Reference

Data Fields

- char address [256]
- uint8_t TlvPresent

8.123.1 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

8.123.2 Field Documentation

8.123.2.1 char loc_urlAddr::address[256]

8.123.2.2 uint8_t loc_urlAddr::TlvPresent

8.124 loc_URLAddrInfo Struct Reference

Data Fields

- char urlAddr [256]

8.124.1 Detailed Description

This structure specifies information regarding the URL.

Parameters

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

8.124.2 Field Documentation

8.124.2.1 char loc_URLAddrInfo::urlAddr[256]

8.125 lteSSInfo Struct Reference

Data Fields

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

8.125.1 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,

8.125.2 Field Documentation

8.125.2.1 int16_t lteSSInfo::rsrp

8.125.2.2 int8_t lteSSInfo::rsrq

8.125.2.3 int8_t lteSSInfo::rssi

8.125.2.4 int16_t lteSSInfo::snr

8.126 messageModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMessageModeInfo MessageModeInfo

8.126.1 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

8.126.2 Field Documentation

8.126.2.1 sMSMessageModeInfo messageModeTlv::MessageModeInfo

8.126.2.2 uint8_t messageModeTlv::TlvPresent

8.127 nas_acqOrderPref Struct Reference

Data Fields

- uint8_t acqOrdeLen
- uint8_t * pAcqOrder

8.127.1 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

8.127.2 Field Documentation

8.127.2.1 uint8_t nas_acqOrderPref::acqOrdeLen

8.127.2.2 uint8_t* nas_acqOrderPref::pAcqOrder

8.128 nas_AcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t acqOrdeLen
- uint8_t * pAcqOrder
- uint8_t TlvPresent

8.128.1 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

8.128.2 Field Documentation

8.128.2.1 uint8_t nas_AcqOrderPrefTlv::acqOrdeLen

8.128.2.2 uint8_t* nas_AcqOrderPrefTlv::pAcqOrder

8.128.2.3 uint8_t nas_AcqOrderPrefTlv::TlvPresent

8.129 nas_ActPilotPNElement Struct Reference

Data Fields

- uint16_t ActSetPilotPN
- uint8_t ActSetPilotPNStrength

8.129.1 Detailed Description

This structure contains pilot PN Element.

Parameters

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

8.129.2 Field Documentation

8.129.2.1 uint16_t nas_ActPilotPNElement::ActSetPilotPN

8.129.2.2 uint8_t nas_ActPilotPNElement::ActSetPilotPNStrength

8.130 nas_AddCDMASysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx
- uint16_t regPrd

8.130.1 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.

8.130.2 Field Documentation

8.130.2.1 uint16_t nas_AddCDMASysInfo::geoSysIdx

8.130.2.2 uint16_t nas_AddCDMASysInfo::regPrd

8.131 nas_AddSysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx
- uint32_t cellBroadcastCap

8.131.1 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

8.131.2 Field Documentation

8.131.2.1 uint32_t nas_AddSysInfo::cellBroadcastCap

8.131.2.2 uint16_t nas_AddSysInfo::geoSysIdx

8.132 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

8.132.1 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

8.132.2 Field Documentation

8.132.2.1 uint64_t nas_BandPrefInfoTlv::bits_129_192

8.132.2.2 uint64_t nas_BandPrefInfoTlv::bits_193_256

8.132.2.3 uint64_t nas_BandPrefInfoTlv::bits_1_64

8.132.2.4 uint64_t nas_BandPrefInfoTlv::bits_65_128

8.132.2.5 uint8_t nas_BandPrefInfoTlv::TlvPresent

8.133 nas_BandPrefTlv Struct Reference

Data Fields

- uint64_t BandPref
- uint8_t TlvPresent

8.133.1 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

8.133.2 Field Documentation

8.133.2.1 uint64_t nas_BandPrefTlv::BandPref

8.133.2.2 uint8_t nas_BandPrefTlv::TlvPresent

8.134 nas_CallBarringSysInfo Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.134.1 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

8.134.2 Field Documentation

8.134.2.1 uint32_t nas_CallBarringSysInfo::csBarStatus

8.134.2.2 uint32_t nas_CallBarringSysInfo::psBarStatus

8.135 nas_callBarStatus Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.135.1 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

8.135.2 Field Documentation

8.135.2.1 uint32_t nas_callBarStatus::csBarStatus

8.135.2.2 uint32_t nas_callBarStatus::psBarStatus

8.136 nas_CDMAChannel Struct Reference

Data Fields

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

8.136.1 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

8.136.2 Field Documentation

8.136.2.1 uint16_t nas_CDMAChannel::priChA

8.136.2.2 uint16_t nas_CDMAChannel::priChB

8.136.2.3 uint16_t nas_CDMAChannel::secChA

8.136.2.4 uint16_t nas_CDMAChannel::secChB

8.137 nas_CDMAECIOThresh Struct Reference

Data Fields

- uint8_t CDMAECIOThreshListLen
- int16_t * pCDMAECIOThreshList

8.137.1 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)

8.137.2 Field Documentation

8.137.2.1 uint8_t nas_CDMAECIOThresh::CDMAECIOThreshListLen

8.137.2.2 int16_t* nas_CDMAECIOThresh::pCDMAECIOThreshList

8.138 nas_CDMAInfo Struct Reference

Data Fields

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

8.138.1 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

8.138.2 Field Documentation

8.138.2.1 uint16_t nas_CDMAInfo::baseId

8.138.2.2 uint32_t nas_CDMAInfo::baseLat

8.138.2.3 uint32_t nas_CDMAInfo::baseLong

8.138.2.4 uint16_t nas_CDMAInfo::nid

8.138.2.5 uint16_t nas_CDMAInfo::refpn

8.138.2.6 uint16_t nas_CDMAInfo::sid

8.139 nas_CDMARSSIThresh Struct Reference

Data Fields

- uint8_t CDMARSSIThreshListLen
- int16_t * pCDMARSSIThreshList

8.139.1 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).

8.139.2 Field Documentation

8.139.2.1 uint8_t nas_CDMARSSIThresh::CDMARSSIThreshListLen

8.139.2.2 int16_t* nas_CDMARSSIThresh::pCDMARSSIThreshList

8.140 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]

8.140.1 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}.

8.140.2 Field Documentation

8.140.2.1 uint16_t nas_CDMASysInfo::baseId

8.140.2.2 uint32_t nas_CDMASysInfo::baseLat

8.140.2.3 uint32_t nas_CDMASysInfo::baseLong

8.140.2.4 uint8_t nas_CDMASysInfo::bsInfoValid

8.140.2.5 uint8_t nas_CDMASysInfo::bsPRev

8.140.2.6 uint8_t nas_CDMASysInfo::bsPRevValid

8.140.2.7 uint8_t nas_CDMASysInfo::ccsSupported

8.140.2.8 uint8_t nas_CDMASysInfo::ccsSupportedValid

8.140.2.9 uint8_t nas_CDMASysInfo::cdmaSysIdValid

8.140.2.10 uint8_t nas_CDMASysInfo::isSysPrlMatch

8.140.2.11 uint8_t nas_CDMASysInfo::isSysPrlMatchValid

8.140.2.12 uint8_t nas_CDMASysInfo::MCC[3]

8.140.2.13 uint8_t nas_CDMASysInfo::MNC[3]

8.140.2.14 uint16_t nas_CDMASysInfo::networkID

8.140.2.15 uint8_t nas_CDMASysInfo::networkIdValid

8.140.2.16 uint16_t nas_CDMASysInfo::packetZone

8.140.2.17 uint8_t nas_CDMASysInfo::packetZoneValid

8.140.2.18 uint8_t nas_CDMASysInfo::pRevInUse

8.140.2.19 uint8_t nas_CDMASysInfo::pRevInUseValid

8.140.2.20 nas_sysInfoCommon nas_CDMASysInfo::sysInfoCDMA

8.140.2.21 uint16_t nas_CDMASysInfo::systemID

8.141 nas_CDMASysInfoExt Struct Reference

Data Fields

- uint16_t MCC
- uint8_t imsi_11_12

8.141.1 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

8.141.2 Field Documentation

8.141.2.1 uint8_t nas_CDMASysInfoExt::imsi_11_12

8.141.2.2 uint16_t nas_CDMASysInfoExt::MCC

8.142 nas_cellParams Struct Reference

Data Fields

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

8.142.1 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.

8.142.2 Field Documentation

8.142.2.1 uint16_t nas_cellParams::pci

8.142.2.2 int16_t nas_cellParams::rsrp

8.142.2.3 int16_t nas_cellParams::rsrq

8.142.2.4 int16_t nas_cellParams::rssi

8.142.2.5 int16_t nas_cellParams::srxlev

8.143 nas_ciotAcqOrderPref Struct Reference

Data Fields

- uint8_t ciotAcqOrderLen
- uint32_t * pCiotAcqOrder

8.143.1 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

8.143.2 Field Documentation

8.143.2.1 uint8_t nas_ciotAcqOrderPref::ciotAcqOrderLen

8.143.2.2 uint32_t* nas_ciotAcqOrderPref::pCiotAcqOrder

8.144 nas_CiotAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t ciotAcqOrderLen
- uint32_t * pCiotAcqOrder
- uint8_t TlvPresent

8.144.1 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

8.144.2 Field Documentation

8.144.2.1 uint8_t nas_CiotAcqOrderPrefTlv::ciotAcqOrderLen

8.144.2.2 uint32_t* nas_CiotAcqOrderPrefTlv::pCiotAcqOrder

8.144.2.3 uint8_t nas_CiotAcqOrderPrefTlv::TlvPresent

8.145 nas_CiotLteOpModePrefTlv Struct Reference

Data Fields

- uint64_t ciotLteOpModePref
- uint8_t TlvPresent

8.145.1 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

8.145.2 Field Documentation

8.145.2.1 uint64_t nas_CiotLteOpModePrefTlv::ciotLteOpModePref

8.145.2.2 uint8_t nas_CiotLteOpModePrefTlv::TlvPresent

8.146 nas_CommInfo Struct Reference

Data Fields

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

- uint8_t psState

8.146.1 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

8.146.2 Field Documentation

8.146.2.1 uint8_t nas_CommInfo::imsRegState

8.146.2.2 uint8_t nas_CommInfo::modemMode

8.146.2.3 uint8_t nas_CommInfo::psState

8.146.2.4 uint8_t nas_CommInfo::systemMode

8.146.2.5 int8_t nas_CommInfo::temperature

8.147 nas_CSGID Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t mncPcsDigits
- uint32_t id
- uint8_t rat

8.147.1 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

8.147.2 Field Documentation

8.147.2.1 uint32_t nas_CSGID::id

8.147.2.2 uint16_t nas_CSGID::mcc

8.147.2.3 uint16_t nas_CSGID::mnc

8.147.2.4 uint8_t nas_CSGID::mncPcsDigits

8.147.2.5 uint8_t nas_CSGID::rat

8.148 nas_CsgId Struct Reference

Data Fields

- uint32_t csgId
- uint8_t TlvPresent

8.148.1 Detailed Description

This structure contains csg ID.

Parameters

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

8.148.2 Field Documentation

8.148.2.1 uint32_t nas_CsgId::csgId

8.148.2.2 uint8_t nas_CsgId::TlvPresent

8.149 nas_currentPLMN Struct Reference

Data Fields

- uint16_t MCC

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

8.149.1 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

8.149.2 Field Documentation

8.149.2.1 uint16_t nas_currentPLMN::MCC

8.149.2.2 uint16_t nas_currentPLMN::MNC

8.149.2.3 uint8_t nas_currentPLMN::netDescr[255]

8.149.2.4 uint8_t nas_currentPLMN::netDescrLength

8.150 nas_dataSrvCapabilities Struct Reference

Data Fields

- uint8_t dataCapabilitiesLen
- uint8_t dataCapabilities [32]

8.150.1 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+

8.150.2 Field Documentation

8.150.2.1 uint8_t nas_dataSrvCapabilities::dataCapabilities[32]

8.150.2.2 uint8_t nas_dataSrvCapabilities::dataCapabilitiesLen

8.151 nas_DataStatusDetail Struct Reference

Data Fields

- uint32_t IPAddress
- uint8_t LastErrCode

8.151.1 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

8.151.2 Field Documentation

8.151.2.1 uint32_t nas_DataStatusDetail::IPAddress

8.151.2.2 uint8_t nas_DataStatusDetail::LastErrCode

8.152 nas_detailSvcInfo Struct Reference

Data Fields

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

8.152.1 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

8.152.2 Field Documentation

8.152.2.1 uint8_t nas_detailSvcInfo::hdrHybrid

8.152.2.2 uint8_t nas_detailSvcInfo::hdrSrvStatus

8.152.2.3 uint8_t nas_detailSvcInfo::isSysForbidden

8.152.2.4 uint8_t nas_detailSvcInfo::srvCapability

8.152.2.5 uint8_t nas_detailSvcInfo::srvStatus

8.153 nas_DeviceConfigDetail Struct Reference

Data Fields

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

8.153.1 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

8.153.2 Field Documentation

8.153.2.1 uint8_t nas_DeviceConfigDetail::Chipset

8.153.2.2 uint8_t nas_DeviceConfigDetail::HWVersion

8.153.2.3 uint8_t nas_DeviceConfigDetail::QLIC

8.153.2.4 uint8_t nas_DeviceConfigDetail::Technology

8.154 nas_dirNum Struct Reference

Data Fields

- uint8_t dirNumLen
- uint8_t dirNum [255]

8.154.1 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.

8.154.2 Field Documentation

8.154.2.1 uint8_t nas_dirNum::dirNum[255]

8.154.2.2 uint8_t nas_dirNum::dirNumLen

8.155 nas_DRCParams Struct Reference

Data Fields

- uint8_t DRCValue
- uint8_t DRCCover

8.155.1 Detailed Description

This structure contains DRC information.

Parameters

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

8.155.2 Field Documentation

8.155.2.1 uint8_t nas_DRCParams::DRCCover

8.155.2.2 uint8_t nas_DRCParams::DRCValue

8.156 nas_ecioListElement Struct Reference

Data Fields

- int16_t ecio
- uint8_t radioIf

8.156.1 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

8.156.2 Field Documentation

8.156.2.1 int16_t nas_ecioListElement::ecio

8.156.2.2 uint8_t nas_ecioListElement::radioIf

8.157 nas_ECIOThresh Struct Reference

Data Fields

- uint8_t ECIOThresListLen
- int16_t * pECIOThresList

8.157.1 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

8.157.2 Field Documentation

8.157.2.1 uint8_t nas_ECIOThresh::ECIOThresListLen

8.157.2.2 int16_t* nas_ECIOThresh::pECIOThresList

8.158 nas_EdrxCiotLteMode Struct Reference

Data Fields

- uint32_t lteOpMode
- uint8_t TlvPresent

8.158.1 Detailed Description

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

Parameters

lteOpMode	• 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.

8.158.2 Field Documentation

8.158.2.1 uint32_t nas_EdrxCiotLteMode::lteOpMode

8.158.2.2 uint8_t nas_EdrxCiotLteMode::TlvPresent

8.159 nas_EdrxCycleLength Struct Reference

Data Fields

- uint8_t cycleLength
- uint8_t TlvPresent

8.159.1 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.

8.159.2 Field Documentation

8.159.2.1 uint8_t nas_EdrxCycleLength::cycleLength

8.159.2.2 uint8_t nas_EdrxCycleLength::TlvPresent

8.160 nas_EdrxEnableType Struct Reference

Data Fields

- uint8_t edrxEnabled
- uint8_t TlvPresent

8.160.1 Detailed Description

This structure contains the parameters for eDRX Enabled Indicator Type.

Parameters

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

8.160.2 Field Documentation

8.160.2.1 uint8_t nas_EdrxEnableType::edrxEnabled

8.160.2.2 uint8_t nas_EdrxEnableType::TlvPresent

8.161 nas_EdrxPagingTimeWindow Struct Reference

Data Fields

- uint8_t edrxPtw
- uint8_t TlvPresent

8.161.1 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.

8.161.2 Field Documentation

8.161.2.1 uint8_t nas_EdrxPagingTimeWindow::edrxPtw

8.161.2.2 uint8_t nas_EdrxPagingTimeWindow::TlvPresent

8.162 nas_EdrxRatType Struct Reference

Data Fields

- uint8_t edrxRatType
- uint8_t TlvPresent

8.162.1 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.

8.162.2 Field Documentation

8.162.2.1 uint8_t nas_EdrxRatType::edrxRatType

8.162.2.2 uint8_t nas_EdrxRatType::TlvPresent

8.163 nas_EmerModeTlv Struct Reference

Data Fields

- uint8_t EmerMode
- uint8_t TlvPresent

8.163.1 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

8.163.2 Field Documentation

8.163.2.1 uint8_t nas_EmerModeTlv::EmerMode

8.163.2.2 uint8_t nas_EmerModeTlv::TlvPresent

8.164 nas_errorRateListElement Struct Reference

Data Fields

- uint16_t errorRate
- uint8_t radioIf

8.164.1 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

8.164.2 Field Documentation

8.164.2.1 uint16_t nas_errorRateListElement::errorRate

8.164.2.2 uint8_t nas_errorRateListElement::radioIf

8.165 nas_ForbiddenNetworks3GPP Struct Reference

Data Fields

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

8.165.1 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

8.165.2 Field Documentation

8.165.2.1 uint16_t nas_ForbiddenNetworks3GPP::forbiddenNwInstLen

8.165.2.2 uint16_t nas_ForbiddenNetworks3GPP::MCC[255]

8.165.2.3 uint16_t nas_ForbiddenNetworks3GPP::MNC[255]

8.165.2.4 uint8_t nas_ForbiddenNetworks3GPP::TlvPresent

8.166 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]

8.166.1 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.

8.166.2 Field Documentation

8.166.2.1 uint16_t nas_GERANInfo::arfcn

8.166.2.2 uint8_t nas_GERANInfo::bsic

8.166.2.3 uint32_t nas_GERANInfo::cellID

8.166.2.4 nas_nmrCellInfo nas_GERANInfo::insNmrCellInfo[255]

8.166.2.5 uint16_t nas_GERANInfo::lac

8.166.2.6 uint8_t nas_GERANInfo::nmrInst

8.166.2.7 uint8_t nas_GERANInfo::plmn[3]

8.166.2.8 uint16_t nas_GERANInfo::rxLev

8.166.2.9 uint32_t nas_GERANInfo::timingAdvance

8.167 nas_geranInstInfo Struct Reference

Data Fields

- uint16_t geranArfcn

- uint8_t geranBsicNcc
- uint8_t geranBsicBcc
- int16_t geranRssi

8.167.1 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.

8.167.2 Field Documentation

8.167.2.1 uint16_t nas_geranInstInfo::geranArfcn

8.167.2.2 uint8_t nas_geranInstInfo::geranBsicBcc

8.167.2.3 uint8_t nas_geranInstInfo::geranBsicNcc

8.167.2.4 int16_t nas_geranInstInfo::geranRssi

8.168 nas_gsmCellInfo Struct Reference

Data Fields

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

8.168.1 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.

8.168.2 Field Documentation

8.168.2.1 uint16_t nas_gsmCellInfo::arfcn

8.168.2.2 uint8_t nas_gsmCellInfo::band1900

8.168.2.3 uint8_t nas_gsmCellInfo::bsicId

8.168.2.4 uint8_t nas_gsmCellInfo::cellIdValid

8.168.2.5 int16_t nas_gsmCellInfo::rssi

8.168.2.6 int16_t nas_gsmCellInfo::srxlev

8.169 nas_GSMRSSIThresh Struct Reference

Data Fields

- uint8_t GSMRSSIThreshListLen
- int16_t * pGSMRSSIThreshList

8.169.1 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)

8.169.2 Field Documentation

8.169.2.1 uint8_t nas_GSMRSSIThresh::GSMRSSIThreshListLen

8.169.2.2 int16_t* nas_GSMRSSIThresh::pGSMRSSIThreshList

8.170 nas_GSMSrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t trueSrvStatus
- uint8_t isPrefDataPath

8.170.1 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

8.170.2 Field Documentation

8.170.2.1 uint8_t nas_GSMSrvStatusInfo::isPrefDataPath

8.170.2.2 uint8_t nas_GSMSrvStatusInfo::srvStatus

8.170.2.3 uint8_t nas_GSMSrvStatusInfo::trueSrvStatus

8.171 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

8.171.1 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_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.

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

8.171.2 Field Documentation

8.171.2.1 uint32_t nas_GSMSysInfo::cellId

8.171.2.2 uint8_t nas_GSMSysInfo::cellIdValid

8.171.2.3 uint8_t nas_GSMSysInfo::dtmSupp

8.171.2.4 uint8_t nas_GSMSysInfo::dtmSuppValid

8.171.2.5 uint8_t nas_GSMSysInfo::egprsSupp

8.171.2.6 uint8_t nas_GSMSysInfo::egprsSuppValid

8.171.2.7 uint16_t nas_GSMSysInfo::lac

8.171.2.8 uint8_t nas_GSMSysInfo::lacValid

8.171.2.9 uint8_t nas_GSMSysInfo::MCC[3]

8.171.2.10 uint8_t nas_GSMSysInfo::MNC[3]

8.171.2.11 uint8_t nas_GSMSysInfo::networkIdValid

8.171.2.12 uint8_t nas_GSMSysInfo::regRejectInfoValid

8.171.2.13 uint8_t nas_GSMSysInfo::rejCause

8.171.2.14 uint8_t nas_GSMSysInfo::rejectSrvDomain

8.171.2.15 nas_sysInfoCommon nas_GSMSysInfo::sysInfoGSM

8.172 nas_GWAcqOrderPrefTlv Struct Reference

Data Fields

- uint32_t GWAcqOrderPref
- uint8_t TlvPresent

8.172.1 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

8.172.2 Field Documentation

8.172.2.1 uint32_t nas_GWAcqOrderPrefTlv::GWAcqOrderPref

8.172.2.2 uint8_t nas_GWAcqOrderPrefTlv::TlvPresent

8.173 nas_HDRECIOThresh Struct Reference

Data Fields

- uint8_t HDRECIOThreshListLen
- int16_t * pHDRECIOThreshList

8.173.1 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).

8.173.2 Field Documentation

8.173.2.1 uint8_t nas_HDRECIOThresh::HDRECIOThreshListLen

8.173.2.2 int16_t* nas_HDRECIOThresh::pHDRECIOThreshList

8.174 nas_HDRIOThresh Struct Reference

Data Fields

- uint8_t HDRIOThreshListLen
- int16_t * pHDRIOThreshList

8.174.1 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).

8.174.2 Field Documentation

8.174.2.1 uint8_t nas_HDRIOThresh::HDRIOThreshListLen

8.174.2.2 int16_t* nas_HDRIOThresh::pHDRIOThreshList

8.175 nas_HDRPersonality_Ind_Data Struct Reference

Data Fields

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

8.175.1 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**

8.175.2 Field Documentation

8.175.2.1 uint16_t* nas_HDRPersonality_Ind_Data::pCurrentPersonality

8.175.2.2 uint8_t* nas_HDRPersonality_Ind_Data::pPersonalityListLength

8.175.2.3 nas_protocolSubtypeElement* nas_HDRPersonality_Ind_Data::pProtocolSubtypeElement

8.176 nas_HDRRSSIThresh Struct Reference

Data Fields

- uint8_t HDRRSSIThreshListLen
- int16_t * pHDRRSSIThreshList

8.176.1 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).

8.176.2 Field Documentation

8.176.2.1 uint8_t nas_HDRRSSIThresh::HDRRSSIThreshListLen

8.176.2.2 int16_t* nas_HDRRSSIThresh::pHDRRSSIThreshList

8.177 nas_HDRSINRThresh Struct Reference

Data Fields

- uint8_t HDRSINRThresListLen
- uint8_t * pHDRSINRThresList

8.177.1 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

8.177.2 Field Documentation

8.177.2.1 uint8_t nas_HDRSINRThresh::HDRSINRThresListLen

8.177.2.2 uint8_t* nas_HDRSINRThresh::pHDRSINRThresList

8.178 nas_HDRSINRThreshold Struct Reference

Data Fields

- uint8_t HDRSINRThreshListLen
- uint16_t * pHDRSINRThreshList

8.178.1 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

8.178.2 Field Documentation

8.178.2.1 uint8_t nas_HDRSINRThreshold::HDRSINRThreshListLen

8.178.2.2 uint16_t* nas_HDRSINRThreshold::pHDRSINRThreshList

8.179 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]

8.179.1 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[SLQ-S_SYSTEM_ID-_SIZE]	• IS-856 system ID. • Only applicable for HDR.

8.179.2 Field Documentation

8.179.2.1 uint8_t nas_HDRSysInfo::hdrActiveProt

8.179.2.2 uint8_t nas_HDRSysInfo::hdrActiveProtValid

8.179.2.3 uint8_t nas_HDRSysInfo::hdrPersonality

8.179.2.4 uint8_t nas_HDRSysInfo::hdrPersonalityValid

8.179.2.5 uint8_t nas_HDRSysInfo::is856SysId[16]

8.179.2.6 uint8_t nas_HDRSysInfo::is856SysIdValid

8.179.2.7 uint8_t nas_HDRSysInfo::isSysPrlMatch

8.179.2.8 uint8_t nas_HDRSysInfo::isSysPrlMatchValid

8.179.2.9 nas_sysInfoCommon nas_HDRSysInfo::sysInfoHDR

8.180 nas_homeSIDNID Struct Reference

Data Fields

- uint8_t numInstances
- nas_sidNid SidNid [255]

8.180.1 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

8.180.2 Field Documentation

8.180.2.1 uint8_t nas_homeSIDNID::numInstances

8.180.2.2 nas_sidNid nas_homeSIDNID::SidNid[255]

8.181 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]

8.181.1 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.

8.181.2 Field Documentation

8.181.2.1 uint8_t nas_infoInterFreq::cell_resel_priority

8.181.2.2 nas_cellParams nas_infoInterFreq::cellInterFreqParams[255]

8.181.2.3 uint8_t nas_infoInterFreq::cells_len

8.181.2.4 uint16_t nas_infoInterFreq::earfcn

8.181.2.5 uint8_t nas_infoInterFreq::threshXHigh

8.181.2.6 uint8_t nas_infoInterFreq::threshXLow

8.182 nas_IOThresh Struct Reference

Data Fields

- uint8_t IOThresListLen
- int32_t * pIOThresList

8.182.1 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

8.182.2 Field Documentation

8.182.2.1 uint8_t nas_IOThresh::IOThresListLen

8.182.2.2 int32_t* nas_IOThresh::pIOThresList

8.183 nas_lteBandPrefExt Struct Reference

Data Fields

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

8.183.1 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.

8.183.2 Field Documentation

8.183.2.1 uint64_t nas_lteBandPrefExt::bits_129_192

8.183.2.2 uint64_t nas_lteBandPrefExt::bits_193_256

8.183.2.3 uint64_t nas_lteBandPrefExt::bits_1_64

8.183.2.4 uint64_t nas_lteBandPrefExt::bits_65_128

8.184 nas_LTEBandPrefTlv Struct Reference

Data Fields

- uint64_t LTEBandPref
- uint8_t TlvPresent

8.184.1 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

8.184.2 Field Documentation

8.184.2.1 uint64_t nas_LTEBandPrefTlv::LTEBandPref

8.184.2.2 uint8_t nas_LTEBandPrefTlv::TlvPresent

8.185 nas_LteCiotOpModeTlv Struct Reference

Data Fields

- uint32_t campedCiotLteOpMode
- uint8_t TlvPresent

8.185.1 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

8.185.2 Field Documentation

8.185.2.1 uint32_t nas_LteCiotOpModeTlv::campedCiotLteOpMode

8.185.2.2 uint8_t nas_LteCiotOpModeTlv::TlvPresent

8.186 nas_lteEARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn0
- uint32_t earfcn1

8.186.1 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.

8.186.2 Field Documentation

8.186.2.1 uint32_t nas_lteEARFCN::earfcn0

8.186.2.2 uint32_t nas_lteEARFCN::earfcn1

8.186.2.3 uint8_t nas_lteEARFCN::status

8.187 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]

8.187.1 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.

8.187.2 Field Documentation

8.187.2.1 uint8_t nas_lteGsmCellInfo::cellReselPriority

8.187.2.2 uint8_t nas_lteGsmCellInfo::cells_len

8.187.2.3 nas_gsmCellInfo nas_lteGsmCellInfo::GsmCellInfo[255]

8.187.2.4 uint8_t nas_lteGsmCellInfo::nccPermitted

8.187.2.5 uint8_t nas_lteGsmCellInfo::threshGsmHigh

8.187.2.6 uint8_t nas_lteGsmCellInfo::threshGsmLow

8.188 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

8.188.1 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

8.188.2 Field Documentation

8.188.2.1 uint8_t nas_LTEInfo::band

8.188.2.2 uint8_t nas_LTEInfo::bandwidth

8.188.2.3 uint8_t nas_LTEInfo::emmConnState

8.188.2.4 uint8_t nas_LTEInfo::emmState

8.188.2.5 uint8_t nas_LTEInfo::emmSubState

8.188.2.6 uint16_t nas_LTEInfo::RXChan

8.188.2.7 uint16_t nas_LTEInfo::TXChan

8.189 nas_LTEInfoInterfreq Struct Reference

Data Fields

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

8.189.1 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.

8.189.2 Field Documentation

8.189.2.1 uint8_t nas_LTEInfoInterfreq::freqsLen

8.189.2.2 nas_infoInterFreq nas_LTEInfoInterfreq::InfoInterfreq[255]

8.189.2.3 uint8_t nas_LTEInfoInterfreq::ueInIdle

8.190 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]

8.190.1 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
threshServingLow	• 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.

8.190.2 Field Documentation

8.190.2.1 nas_cellParams nas_LTEInfoIntrafreq::CellParams[255]

8.190.2.2 uint8_t nas_LTEInfoIntrafreq::cellReselPriority

8.190.2.3 uint8_t nas_LTEInfoIntrafreq::cellsLen

8.190.2.4 uint16_t nas_LTEInfoIntrafreq::earfcn

8.190.2.5 uint32_t nas_LTEInfoIntrafreq::globalCellId

8.190.2.6 uint8_t nas_LTEInfoIntrafreq::plmn[3]

8.190.2.7 uint16_t nas_LTEInfoIntrafreq::servingCellId

8.190.2.8 uint8_t nas_LTEInfoIntrafreq::sIntraSearch

8.190.2.9 uint8_t nas_LTEInfoIntrafreq::sNonIntraSearch

8.190.2.10 uint16_t nas_LTEInfoIntrafreq::tac

8.190.2.11 uint8_t nas_LTEInfoIntrafreq::threshServingLow

8.190.2.12 uint8_t nas_LTEInfoIntrafreq::ueInIdle

8.191 nas_LTEInfoNeighboringGSM Struct Reference

Data Fields

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

8.191.1 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.

8.191.2 Field Documentation

8.191.2.1 uint8_t nas_LTEInfoNeighboringGSM::freqsLen

8.191.2.2 nas_lteGsmCellInfo nas_LTEInfoNeighboringGSM::LteGsmCellInfo[255]

8.191.2.3 uint8_t nas_LTEInfoNeighboringGSM::ueInIdle

8.192 nas_LTEInfoNeighboringWCDMA Struct Reference

Data Fields

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

8.192.1 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.
LTEWCDMACellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_lteWcdmaCellInfo for more information.

8.192.2 Field Documentation

8.192.2.1 uint8_t nas_LTEInfoNeighboringWCDMA::freqsLen

8.192.2.2 nas_lteWcdmaCellInfo nas_LTEInfoNeighboringWCDMA::LTEWCDMACellInfo[255]

8.192.2.3 uint8_t nas_LTEInfoNeighboringWCDMA::ueInIdle

8.193 nas_LteM1BandPrefTlv Struct Reference

Data Fields

- uint64_t lteM1BandPref
- uint8_t TlvPresent

8.193.1 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

8.193.2 Field Documentation

8.193.2.1 uint64_t nas_LteM1BandPrefTlv::lteM1BandPref

8.193.2.2 uint8_t nas_LteM1BandPrefTlv::TlvPresent

8.194 nas_LteNb1BandPrefTlv Struct Reference

Data Fields

- uint64_t lteNb1BandPref
- uint8_t TlvPresent

8.194.1 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

8.194.2 Field Documentation

8.194.2.1 uint64_t nas_LteNb1BandPrefTlv::lteNb1BandPref

8.194.2.2 uint8_t nas_LteNb1BandPrefTlv::TlvPresent

8.195 nas_LTEOperationMode Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t * pLTEOperationMode

8.195.1 Detailed Description

This structure contains the nas_LTEOperationMode response parameters.

Parameters

TlvPresent	• Tlv Present
pLTEOperation-Mode	• LTE Operational Mode

8.195.2 Field Documentation

8.195.2.1 uint32_t* nas_LTEOperationMode::pLTEOperationMode

8.195.2.2 uint8_t nas_LTEOperationMode::TlvPresent

8.196 nas_LteOpMode Struct Reference

Data Fields

- uint32_t lteOpMode
- uint8_t TlvPresent

8.196.1 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

8.196.2 Field Documentation

8.196.2.1 uint32_t nas_LteOpMode::lteOpMode

8.196.2.2 uint8_t nas_LteOpMode::TlvPresent

8.197 nas_lteOpModeTlv Struct Reference

Data Fields

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

8.197.1 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

8.197.2 Field Documentation

8.197.2.1 uint32_t nas_lteOpModeTlv::lteOpMode[255]

8.197.2.2 uint8_t nas_lteOpModeTlv::lteOpModeLen

8.197.2.3 uint16_t nas_lteOpModeTlv::MCC[255]

8.197.2.4 uint16_t nas_lteOpModeTlv::MNC[255]

8.197.2.5 uint8_t nas_lteOpModeTlv::TlvPresent

8.198 nas_ltePCI Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn
- uint32_t pci

8.198.1 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

8.198.2 Field Documentation

8.198.2.1 uint32_t nas_ltePCI::earfcn

8.198.2.2 uint32_t nas_ltePCI::pci

8.198.2.3 uint8_t nas_ltePCI::status

8.199 nas_lteRsrpinformation Struct Reference

Data Fields

- int16_t rsrplevel

8.199.1 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).

8.199.2 Field Documentation

8.199.2.1 int16_t nas_lteRsrpinformation::rsrplevel

8.200 nas_LTERSRPThresh Struct Reference

Data Fields

- uint8_t LTERSRPThreshListLen
- int16_t * pLTERSRPThreshList

8.200.1 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).

8.200.2 Field Documentation

8.200.2.1 uint8_t nas_LTERSRPThresh::LTERSRPThreshListLen

8.200.2.2 int16_t* nas_LTERSRPThresh::pLTERSRPThreshList

8.201 nas_LTERSRQThresh Struct Reference

Data Fields

- uint8_t LTERSRQThreshListLen

- int16_t * pLTERSRQThreshList

8.201.1 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)

8.201.2 Field Documentation

8.201.2.1 uint8_t nas_LTERSRQThresh::LTERSRQThreshListLen

8.201.2.2 int16_t* nas_LTERSRQThresh::pLTERSRQThreshList

8.202 nas_LTERSSIThresh Struct Reference

Data Fields

- uint8_t LTERSSIThreshListLen
- int16_t * pLTERSSIThreshList

8.202.1 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)

8.202.2 Field Documentation

8.202.2.1 uint8_t nas_LTERSSIThresh::LTERSSIThreshListLen

8.202.2.2 int16_t* nas_LTERSSIThresh::pLTERSSIThreshList

8.203 nas_LTESigRptCfg Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.203.1 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

8.203.2 Field Documentation

8.203.2.1 uint8_t nas_LTESigRptCfg::avgPeriod

8.203.2.2 uint8_t nas_LTESigRptCfg::rptRate

8.204 nas_LTESigRptConfig Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.204.1 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

8.204.2 Field Documentation

8.204.2.1 uint8_t nas_LTESigRptConfig::avgPeriod

8.204.2.2 uint8_t nas_LTESigRptConfig::rptRate

8.205 nas_lteSnrinformation Struct Reference

Data Fields

- int16_t snrlevel

8.205.1 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.

8.205.2 Field Documentation

8.205.2.1 int16_t nas_lteSnrinformation::snrlevel

8.206 nas_LTESNRThresh Struct Reference

Data Fields

- uint8_t LTESNRThresListLen
- int16_t * pLTESNRThresList

8.206.1 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

8.206.2 Field Documentation

8.206.2.1 uint8_t nas_LTESNRThresh::LTESNRThresListLen

8.206.2.2 int16_t* nas_LTESNRThresh::pLTESNRThresList

8.207 nas_LTESNRThreshold Struct Reference

Data Fields

- uint8_t LTESNRThreshListLen
- int16_t * pLTESNRThreshList

8.207.1 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).

8.207.2 Field Documentation

8.207.2.1 uint8_t nas_LTESNRThreshold::LTESNRThreshListLen

8.207.2.2 int16_t* nas_LTESNRThreshold::pLTESNRThreshList

8.208 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

8.208.1 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

8.208.2 Field Documentation

8.208.2.1 uint32_t nas_LTESysInfo::cellId

8.208.2.2 uint8_t nas_LTESysInfo::cellIdValid

8.208.2.3 uint16_t nas_LTESysInfo::lac

8.208.2.4 uint8_t nas_LTESysInfo::lacValid

8.208.2.5 uint8_t nas_LTESysInfo::MCC[3]

8.208.2.6 uint8_t nas_LTESysInfo::MNC[3]

8.208.2.7 uint8_t nas_LTESysInfo::networkIdValid

8.208.2.8 uint8_t nas_LTESysInfo::regRejectInfoValid

8.208.2.9 uint8_t nas_LTESysInfo::rejCause

8.208.2.10 uint8_t nas_LTESysInfo::rejectSrvDomain

8.208.2.11 nas_sysInfoCommon nas_LTESysInfo::sysInfoLTE

8.208.2.12 uint16_t nas_LTESysInfo::tac

8.208.2.13 uint8_t nas_LTESysInfo::tacValid

8.209 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]

8.209.1 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.

8.209.2 Field Documentation

8.209.2.1 uint8_t nas_lteWcdmaCellInfo::cellReselPriority

8.209.2.2 uint8_t nas_lteWcdmaCellInfo::cellsLen

8.209.2.3 uint16_t nas_lteWcdmaCellInfo::threshXhigh

8.209.2.4 uint16_t nas_lteWcdmaCellInfo::threshXlow

8.209.2.5 uint16_t nas_lteWcdmaCellInfo::uarfcn

8.209.2.6 nas_wcdmaCellInfo nas_lteWcdmaCellInfo::WCDMACellInfo[255]

8.210 nas_minBasedIMSI Struct Reference

Data Fields

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

- uint8_t imsiMS2 [3]

8.210.1 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

8.210.2 Field Documentation

8.210.2.1 uint16_t nas_minBasedIMSI::imsiM1112

8.210.2.2 uint8_t nas_minBasedIMSI::imsiMS1[7]

8.210.2.3 uint8_t nas_minBasedIMSI::imsiMS2[3]

8.210.2.4 uint8_t nas_minBasedIMSI::mccM[3]

8.211 nas_MNRInfo Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint32_t rat

8.211.1 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

8.211.2 Field Documentation

8.211.2.1 uint16_t nas_MNRInfo::mcc

8.211.2.2 uint16_t nas_MNRInfo::mnc

8.211.2.3 uint32_t nas_MNRInfo::rat

8.212 nas_ModePrefTlv Struct Reference

Data Fields

- uint16_t ModePref
- uint8_t TlvPresent

8.212.1 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

8.212.2 Field Documentation

8.212.2.1 uint16_t nas_ModePrefTlv::ModePref

8.212.2.2 uint8_t nas_ModePrefTlv::TlvPresent

8.213 nas_namName Struct Reference

Data Fields

- uint8_t namNameLen
- uint8_t namName [12]

8.213.1 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.

8.213.2 Field Documentation

8.213.2.1 uint8_t nas_namName::namName[12]

8.213.2.2 uint8_t nas_namName::namNameLen

8.214 nas_netSelectionPref Struct Reference

Data Fields

- uint8_t netReg
- uint16_t mcc
- uint16_t mnc

8.214.1 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

8.214.2 Field Documentation

8.214.2.1 uint16_t nas_netSelectionPref::mcc

8.214.2.2 uint16_t nas_netSelectionPref::mnc

8.214.2.3 uint8_t nas_netSelectionPref::netReg

8.215 nas_NetSelPrefTlv Struct Reference

Data Fields

- uint8_t NetSelPref
- uint8_t TlvPresent

8.215.1 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

8.215.2 Field Documentation

8.215.2.1 uint8_t nas_NetSelPrefTlv::NetSelPref

8.215.2.2 uint8_t nas_NetSelPrefTlv::TlvPresent

8.216 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

8.216.1 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

8.216.2 Field Documentation

8.216.2.1 uint8_t nas_NetworkStat1x::ActSetCnt

8.216.2.2 uint8_t nas_NetworkStat1x::NeighborSetCnt

8.216.2.3 nas_ActPilotPNElement* nas_NetworkStat1x::pActPilotPNElements

8.216.2.4 uint16_t* nas_NetworkStat1x::pNeighborSetPilotPN

8.216.2.5 uint16_t nas_NetworkStat1x::RX_EC_IO

8.216.2.6 uint32_t nas_NetworkStat1x::RX_PWR

8.216.2.7 uint16_t nas_NetworkStat1x::SO

8.216.2.8 uint8_t nas_NetworkStat1x::State

8.216.2.9 uint32_t nas_NetworkStat1x::TX_PWR

8.217 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

8.217.1 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)

8.217.2 Field Documentation

8.217.2.1 uint8_t nas_NetworkStatEVDO::MACIndex

8.217.2.2 uint16_t nas_NetworkStatEVDO::PER

8.217.2.3 uint16_t nas_NetworkStatEVDO::PilotEnergy

8.217.2.4 uint16_t* nas_NetworkStatEVDO::pSectorID

8.217.2.5 uint16_t nas_NetworkStatEVDO::RX_PWR

8.217.2.6 uint8_t nas_NetworkStatEVDO::SectorIDLen

8.217.2.7 uint8_t nas_NetworkStatEVDO::SNR

8.217.2.8 uint8_t nas_NetworkStatEVDO::State

8.218 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

8.218.1 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

8.218.2 Field Documentation

8.218.2.1 uint16_t nas_nmrCellInfo::nmrArfcn

8.218.2.2 uint8_t nas_nmrCellInfo::nmrBsic

8.218.2.3 uint32_t nas_nmrCellInfo::nmrCellID

8.218.2.4 uint16_t nas_nmrCellInfo::nmrLac

8.218.2.5 uint8_t nas_nmrCellInfo::nmrPlmn[3]

8.218.2.6 uint16_t nas_nmrCellInfo::nmrRxLev

8.219 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

8.219.1 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.

8.219.2 Field Documentation

8.219.2.1 uint64_t nas_nr5gBandPref::bits_129_192

8.219.2.2 uint64_t nas_nr5gBandPref::bits_193_256

8.219.2.3 uint64_t nas_nr5gBandPref::bits_1_64

8.219.2.4 uint64_t nas_nr5gBandPref::bits_65_128

8.220 nas_NR5GCellStatusTlv Struct Reference

Data Fields

- uint32_t nr5gCellStatus
- uint8_t TlvPresent

8.220.1 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

8.220.2 Field Documentation

8.220.2.1 uint32_t nas_NR5GCellStatusTlv::nr5gCellStatus

8.220.2.2 uint8_t nas_NR5GCellStatusTlv::TlvPresent

8.221 nas_NR5GSerStatTlv Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t trueSrvStatus
- uint8_t isPrefDataPath
- uint8_t TlvPresent

8.221.1 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

8.221.2 Field Documentation

8.221.2.1 uint8_t nas_NR5GSerStatTlv::isPrefDataPath

8.221.2.2 uint8_t nas_NR5GSerStatTlv::srvStatus

8.221.2.3 uint8_t nas_NR5GSerStatTlv::TlvPresent

8.221.2.4 uint8_t nas_NR5GSerStatTlv::trueSrvStatus

8.222 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

8.222.1 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

8.222.2 Field Documentation

8.222.2.1 uint32_t nas_NR5GSystemInfoTlv::cellId

8.222.2.2 uint8_t nas_NR5GSystemInfoTlv::cellIdValid

8.222.2.3 uint16_t nas_NR5GSystemInfoTlv::lac

8.222.2.4 uint8_t nas_NR5GSystemInfoTlv::lacValid

8.222.2.5 int8_t nas_NR5GSystemInfoTlv::MCC[3]

8.222.2.6 int8_t nas_NR5GSystemInfoTlv::MNC[3]

8.222.2.7 uint8_t nas_NR5GSystemInfoTlv::nwIdValid

8.222.2.8 uint8_t nas_NR5GSystemInfoTlv::regRejectInfoValid

8.222.2.9 uint8_t nas_NR5GSystemInfoTlv::rejCause

8.222.2.10 uint8_t nas_NR5GSystemInfoTlv::rejectSrvDomain

8.222.2.11 uint8_t nas_NR5GSystemInfoTlv::roamStatus

8.222.2.12 uint8_t nas_NR5GSystemInfoTlv::roamStatusValid

8.222.2.13 uint8_t nas_NR5GSystemInfoTlv::srvcapability

8.222.2.14 uint8_t nas_NR5GSystemInfoTlv::srvCapValid

8.222.2.15 uint8_t nas_NR5GSystemInfoTlv::srvDomain

8.222.2.16 uint8_t nas_NR5GSystemInfoTlv::srvDomainValid

8.222.2.17 uint8_t nas_NR5GSystemInfoTlv::sysForbidden

8.222.2.18 uint8_t nas_NR5GSystemInfoTlv::sysForbiddenValid

8.222.2.19 uint16_t nas_NR5GSystemInfoTlv::tac

8.222.2.20 uint8_t nas_NR5GSystemInfoTlv::tacValid

8.222.2.21 uint8_t nas_NR5GSystemInfoTlv::TlvPresent

8.223 nas_operatorNameString Struct Reference

Data Fields

- uint8_t PLMNName [255]

8.223.1 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.

8.223.2 Field Documentation

8.223.2.1 uint8_t nas_operatorNameString::PLMNName[255]

8.224 nas_OperatorPLMNData Struct Reference

Data Fields

- uint8_t mcc [3]
- uint8_t mnc [3]
- uint16_t lac1
- uint16_t lac2
- uint8_t PLMNRecID

8.224.1 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.

8.224.2 Field Documentation

8.224.2.1 uint16_t nas_OperatorPLMNData::lac1

8.224.2.2 uint16_t nas_OperatorPLMNData::lac2

8.224.2.3 uint8_t nas_OperatorPLMNData::mcc[3]

8.224.2.4 uint8_t nas_OperatorPLMNData::mnc[3]

8.224.2.5 uint8_t nas_OperatorPLMNData::PLMNRecID

8.225 nas_operatorPLMNList Struct Reference

Data Fields

- uint16_t numInstance
- nas_OperatorPLMNData PLMNData [255]

8.225.1 Detailed Description

This structure contains Operator PLMN List.

Parameters

numInstance	• Number of sets of the elements.
PLMNData	• Refer OperatorPLMNLData for details (Optional).

8.225.2 Field Documentation

8.225.2.1 uint16_t nas_operatorPLMNList::numInstance

8.225.2.2 nas_OperatorPLMNData nas_operatorPLMNList::PLMNData[255]

8.226 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

8.226.1 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.

8.226.2 Field Documentation

8.226.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggPcellInfo::dl_bw_value

8.226.2.2 int nas_PhyCaAggPcellInfo::freq

8.226.2.3 int nas_PhyCaAggPcellInfo::iLTEbandValue

8.226.2.4 int nas_PhyCaAggPcellInfo::pci

8.226.2.5 uint8_t nas_PhyCaAggPcellInfo::TlvPresent

8.227 nas_PhyCaAggScellDlBw Struct Reference

Data Fields

- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- uint8_t TlvPresent

8.227.1 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.

8.227.2 Field Documentation

8.227.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellDlBw::dl_bw_value

8.227.2.2 uint8_t nas_PhyCaAggScellDlBw::TlvPresent

8.228 nas_PhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.228.1 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.

8.228.2 Field Documentation

8.228.2.1 uint8_t nas_PhyCaAggScellIndex::scell_idx

8.228.2.2 uint8_t nas_PhyCaAggScellIndex::TlvPresent

8.229 nas_PhyCaAggScellIndType Struct Reference

Data Fields

- int pci
- int freq
- NAS_LTE_CPHY_SCELL_STATE_LITE scell_state
- uint8_t TlvPresent

8.229.1 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.

8.229.2 Field Documentation

8.229.2.1 int nas_PhyCaAggScellIndType::freq

8.229.2.2 int nas_PhyCaAggScellIndType::pci

8.229.2.3 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellIndType::scell_state

8.229.2.4 uint8_t nas_PhyCaAggScellIndType::TlvPresent

8.230 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

8.230.1 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.

8.230.2 Field Documentation

8.230.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellInfo::dl_bw_value

8.230.2.2 int nas_PhyCaAggScellInfo::freq

8.230.2.3 uint16_t nas_PhyCaAggScellInfo::iLTEbandValue

8.230.2.4 int nas_PhyCaAggScellInfo::pci

8.230.2.5 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellInfo::scell_state

8.230.2.6 uint8_t nas_PhyCaAggScellInfo::TlvPresent

8.231 nas_PilotSetData Struct Reference

Data Fields

- uint8_t NumPilots
- nas_PilotSetParams * pPilotSetInfo

8.231.1 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.

8.231.2 Field Documentation

8.231.2.1 uint8_t nas_PilotSetData::NumPilots

8.231.2.2 nas_PilotSetParams* nas_PilotSetData::pPilotSetInfo

8.232 nas_PilotSetParams Struct Reference

Data Fields

- uint32_t PilotType
- uint16_t PilotPN
- uint16_t PilotStrength

8.232.1 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)

8.232.2 Field Documentation

8.232.2.1 uint16_t nas_PilotSetParams::PilotPN

8.232.2.2 uint16_t nas_PilotSetParams::PilotStrength

8.232.2.3 uint32_t nas_PilotSetParams::PilotType

8.233 nas_PlmnID Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t pcsDigit
- uint8_t TlvPresent

8.233.1 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

8.233.2 Field Documentation

8.233.2.1 uint16_t nas_PlmnID::mcc

8.233.2.2 uint16_t nas_PlmnID::mnc

8.233.2.3 uint8_t nas_PlmnID::pcsDigit

8.233.2.4 uint8_t nas_PlmnID::TlvPresent

8.234 nas_PLMNNetworkName Struct Reference

Data Fields

- uint8_t numInstance
- nas_PLMNNetworkNameData PLMNNetName [255]

8.234.1 Detailed Description

This structure contains NITZ Information list.

Parameters

numInstance	• Number of sets of the elements.
PLMNNetName	• Refer nas_PLMNNetworkNameData for details (Optional).

8.234.2 Field Documentation

8.234.2.1 uint8_t nas_PLMNNetworkName::numInstance

8.234.2.2 nas_PLMNNetworkNameData nas_PLMNNetworkName::PLMNNetName[255]

8.235 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]

8.235.1 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.

8.235.2 Field Documentation

8.235.2.1 uint8_t nas_PLMNNetworkNameData::codingScheme

8.235.2.2 uint8_t nas_PLMNNetworkNameData::countryInitials

8.235.2.3 uint8_t nas_PLMNNetworkNameData::longName[255]

8.235.2.4 uint8_t nas_PLMNNetworkNameData::longNameLen

8.235.2.5 uint8_t nas_PLMNNetworkNameData::longNameSpareBits

8.235.2.6 uint8_t nas_PLMNNetworkNameData::shortName[255]

8.235.2.7 uint8_t nas_PLMNNetworkNameData::shortNameLen

8.235.2.8 uint8_t nas_PLMNNetworkNameData::shortNameSpareBits

8.236 nas_PRLPrefTlv Struct Reference

Data Fields

- uint16_t PRLPref
- uint8_t TlvPresent

8.236.1 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

8.236.2 Field Documentation

8.236.2.1 uint16_t nas_PRLPrefTlv::PRLPref

8.236.2.2 uint8_t nas_PRLPrefTlv::TlvPresent

8.237 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

8.237.1 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

8.237.2 Field Documentation

8.237.2.1 uint16_t nas_protocolSubtypeElement::AccessMac

8.237.2.2 uint16_t nas_protocolSubtypeElement::AuthProt

8.237.2.3 uint16_t nas_protocolSubtypeElement::ControlMac

8.237.2.4 uint16_t nas_protocolSubtypeElement::EncryptProt

8.237.2.5 uint16_t nas_protocolSubtypeElement::ForwardMac

8.237.2.6 uint16_t nas_protocolSubtypeElement::IdleState

8.237.2.7 uint16_t nas_protocolSubtypeElement::KeyExchange

8.237.2.8 uint16_t nas_protocolSubtypeElement::MultDisc

8.237.2.9 uint16_t nas_protocolSubtypeElement::PhysicalLayer

8.237.2.10 uint16_t nas_protocolSubtypeElement::ReverseMac

8.237.2.11 uint16_t nas_protocolSubtypeElement::SecProt

8.237.2.12 uint16_t nas_protocolSubtypeElement::VirtStream

8.238 nas_qaQmi3Gpp2TimeZone Struct Reference

Data Fields

- uint8_t leapSeconds
- uint8_t localTimeOffset
- uint8_t daylightSavings

8.238.1 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)

8.238.2 Field Documentation

8.238.2.1 uint8_t nas_qaQmi3Gpp2TimeZone::daylightSavings

8.238.2.2 uint8_t nas_qaQmi3Gpp2TimeZone::leapSeconds

8.238.2.3 uint8_t nas_qaQmi3Gpp2TimeZone::localTimeOffset

8.239 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]

8.239.1 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.

8.239.2 Field Documentation

8.239.2.1 char nas_QmiNas3GppNetworkInfo::Desription[255]

8.239.2.2 uint32_t nas_QmiNas3GppNetworkInfo::Forbidden

8.239.2.3 uint32_t nas_QmiNas3GppNetworkInfo::InUse

8.239.2.4 uint16_t nas_QmiNas3GppNetworkInfo::MCC

8.239.2.5 uint16_t nas_QmiNas3GppNetworkInfo::MNC

8.239.2.6 uint32_t nas_QmiNas3GppNetworkInfo::Preferred

8.239.2.7 uint32_t nas_QmiNas3GppNetworkInfo::Roaming

8.240 nas_QmiNas3GppNetworkRAT Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t RAT

8.240.1 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

8.240.2 Field Documentation

8.240.2.1 uint16_t nas_QmiNas3GppNetworkRAT::MCC

8.240.2.2 uint16_t nas_QmiNas3GppNetworkRAT::MNC

8.240.2.3 uint8_t nas_QmiNas3GppNetworkRAT::RAT

8.241 nas_QmisNasPcsDigit Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t includes_pcs_digit

8.241.1 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

8.241.2 Field Documentation

8.241.2.1 uint8_t nas_QmisNasPcsDigit::includes_pcs_digit

8.241.2.2 uint16_t nas_QmisNasPcsDigit::MCC

8.241.2.3 uint16_t nas_QmisNasPcsDigit::MNC

8.242 nas_QmisNasSlqsNasPCICellInfo Struct Reference

Data Fields

- uint32_t freq
- uint16_t cellID
- uint32_t GlobalCellID
- uint8_t PlmnLen
- nas_QmisNasPcsDigit nasQmisNasPcsDigit [255]

8.242.1 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.

8.242.2 Field Documentation

8.242.2.1 uint16_t nas_QmisNasSlqsNasPCICellInfo::cellID

8.242.2.2 uint32_t nas_QmisNasSlqsNasPCICellInfo::freq

8.242.2.3 uint32_t nas_QmisNasSlqsNasPCICellInfo::GlobalCellID

8.242.2.4 nas_QmisNasPcsDigit nas_QmisNasSlqsNasPCICellInfo::nasQmisNasPcsDigit[255]

8.242.2.5 uint8_t nas_QmisNasSlqsNasPCICellInfo::PlmnLen

8.243 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

8.243.1 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.

8.243.2 Field Documentation

8.243.2.1 nas_QmisNasSlqsNasPCICellInfo nas_QmisNasSlqsNasPCIInfo::nasQmisNasSlqsNasPCICellInfo[255]

8.243.2.2 uint8_t nas_QmisNasSlqsNasPCIInfo::PCICellInfoLen

8.243.2.3 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrp

8.243.2.4 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrpRx0

8.243.2.5 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrpRx1

8.243.2.6 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrq

8.243.2.7 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrqRx0

8.243.2.8 uint16_t nas_QmisNasSlqsNasPCIInfo::rsrqRx1

8.244 nas_RankIndicatorTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t count0
- uint32_t count1

8.244.1 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

8.244.2 Field Documentation

8.244.2.1 uint32_t nas_RankIndicatorTlv::count0

8.244.2.2 uint32_t nas_RankIndicatorTlv::count1

8.244.2.3 uint8_t nas_RankIndicatorTlv::TlvPresent

8.245 nas_RatDisabledMaskTlv Struct Reference

Data Fields

- uint16_t ratDisabledMask
- uint8_t TlvPresent

8.245.1 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

8.245.2 Field Documentation

8.245.2.1 uint16_t nas_RatDisabledMaskTlv::ratDisabledMask

8.245.2.2 uint8_t nas_RatDisabledMaskTlv::TlvPresent

8.246 nas_RejectReasonTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t serviceDomain
- uint32_t rejectCause

8.246.1 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

8.246.2 Field Documentation

8.246.2.1 uint32_t nas_RejectReasonTlv::rejectCause

8.246.2.2 uint32_t nas_RejectReasonTlv::serviceDomain

8.246.2.3 uint8_t nas_RejectReasonTlv::TlvPresent

8.247 nas_RFBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBandClass
- uint16_t activeChannel

8.247.1 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)

8.247.2 Field Documentation

8.247.2.1 uint16_t nas_RFBandInfoElements::activeBandClass

8.247.2.2 uint16_t nas_RFBandInfoElements::activeChannel

8.247.2.3 uint8_t nas_RFBandInfoElements::radioInterface

8.248 nas_RfBandInfoExtFormat Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t * pInstancesSize
- struct nas_RfBandInfoExtFormatElements * pRfBandInfoExtFormatParam

8.248.1 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

8.248.2 Field Documentation

8.248.2.1 uint8_t* nas_RfBandInfoExtFormat::pInstancesSize

8.248.2.2 struct nas_RfBandInfoExtFormatElements* nas_RfBandInfoExtFormat::pRfBandInfoExtFormatParam

8.248.2.3 uint8_t nas_RfBandInfoExtFormat::TlvPresent

8.249 nas_RfBandInfoExtFormatElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBand
- uint32_t activeChannel

8.249.1 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)

8.249.2 Field Documentation

8.249.2.1 uint16_t nas_RfBandInfoExtFormatElements::activeBand

8.249.2.2 uint32_t nas_RfBandInfoExtFormatElements::activeChannel

8.249.2.3 uint8_t nas_RfBandInfoExtFormatElements::radioInterface

8.250 nas_RfBandInfoList Struct Reference

Data Fields

- uint8_t * pInstanceSize
- struct nas_RFBandInfoElements * pRFBandInfoParam

8.250.1 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

8.250.2 Field Documentation

8.250.2.1 uint8_t* nas_RfBandInfoList::pInstanceSize

8.250.2.2 struct nas_RFBandInfoElements* nas_RfBandInfoList::pRFBandInfoParam

8.251 nas_RfBandwidthInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t * pInstancesSize
- struct nas_RfBandwidthInfoElements * pRfBandwidthInfoParam

8.251.1 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

8.251.2 Field Documentation

8.251.2.1 uint8_t* nas_RfBandwidthInfo::pInstancesSize

8.251.2.2 struct nas_RfBandwidthInfoElements* nas_RfBandwidthInfo::pRfBandwidthInfoParam

8.251.2.3 uint8_t nas_RfBandwidthInfo::TlvPresent

8.252 nas_RfBandwidthInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint32_t bandwidth

8.252.1 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

8.252.2 Field Documentation

8.252.2.1 uint32_t nas_RfBandwidthInfoElements::bandwidth

8.252.2.2 uint8_t nas_RfBandwidthInfoElements::radioInterface

8.253 nas_RfDedicatedBandInfo Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t * pInstancesSize
- struct nas_RfDedicatedBandInfoElements * pRfDedicatedBandInfoParam

8.253.1 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

8.253.2 Field Documentation

8.253.2.1 uint8_t* nas_RfDedicatedBandInfo::pInstancesSize

8.253.2.2 struct nas_RfDedicatedBandInfoElements* nas_RfDedicatedBandInfo::pRfDedicatedBandInfoParam

8.253.2.3 uint8_t nas_RfDedicatedBandInfo::TlvPresent

8.254 nas_RfDedicatedBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t dedicatedBand

8.254.1 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

8.254.2 Field Documentation

8.254.2.1 uint16_t nas_RfDedicatedBandInfoElements::dedicatedBand

8.254.2.2 uint8_t nas_RfDedicatedBandInfoElements::radioInterface

8.255 nas_RFInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t radioInterfaceSize
- uint32_t radioInterface [255]
- uint32_t activeBandClass [255]
- uint32_t activeChannel [255]

8.255.1 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.

8.255.2 Field Documentation

8.255.2.1 uint32_t nas_RFInfoTlv::activeBandClass[255]

8.255.2.2 uint32_t nas_RFInfoTlv::activeChannel[255]

8.255.2.3 uint32_t nas_RFInfoTlv::radioInterface[255]

8.255.2.4 uint8_t nas_RFInfoTlv::radioInterfaceSize

8.255.2.5 uint8_t nas_RFInfoTlv::TlvPresent

8.256 nas_roamIndList Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t radioInterface [32]
- uint8_t roamIndicator [32]

8.256.1 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

8.256.2 Field Documentation

8.256.2.1 uint8_t nas_roamIndList::numInstances

8.256.2.2 uint8_t nas_roamIndList::radioInterface[32]

8.256.2.3 uint8_t nas_roamIndList::roamIndicator[32]

8.257 nas_RoamPrefTlv Struct Reference

Data Fields

- uint16_t RoamPref
- uint8_t TlvPresent

8.257.1 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

8.257.2 Field Documentation

8.257.2.1 uint16_t nas_RoamPrefTlv::RoamPref

8.257.2.2 uint8_t nas_RoamPrefTlv::TlvPresent

8.258 nas_RSRPThresh Struct Reference

Data Fields

- uint8_t RSRPThresListLen
- int16_t * pRSRPThresList

8.258.1 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

8.258.2 Field Documentation

8.258.2.1 int16_t* nas_RSRPThresh::pRSRPThresList

8.258.2.2 uint8_t nas_RSRPThresh::RSRPThresListLen

8.259 nas_rsrqInformation Struct Reference

Data Fields

- int8_t rsrq
- uint8_t radioIf

8.259.1 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

8.259.2 Field Documentation

8.259.2.1 uint8_t nas_rsrqInformation::radioIf

8.259.2.2 int8_t nas_rsrqInformation::rsrq

8.260 nas_RSRQThresh Struct Reference

Data Fields

- uint8_t RSRQThresListLen
- int8_t * pRSRQThresList

8.260.1 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

8.260.2 Field Documentation

8.260.2.1 int8_t* nas_RSRQThresh::pRSRQThresList

8.260.2.2 uint8_t nas_RSRQThresh::RSRQThresListLen

8.261 nas_RSSIThresh Struct Reference

Data Fields

- uint8_t RSSIThresListLen
- int8_t * pRSSIThresList

8.261.1 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.

8.261.2 Field Documentation

8.261.2.1 int8_t* nas_RSSIThresh::pRSSIThresList

8.261.2.2 uint8_t nas_RSSIThresh::RSSIThresListLen

8.262 nas_rxInfo Struct Reference

Data Fields

- uint8_t isRadioTuned
- int32_t rxPower
- int32_t ecio
- int32_t rscp
- int32_t rsrp
- uint32_t phase

8.262.1 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.

8.262.2 Field Documentation

8.262.2.1 int32_t nas_rxInfo::ecio

8.262.2.2 uint8_t nas_rxInfo::isRadioTuned

8.262.2.3 uint32_t nas_rxInfo::phase

8.262.2.4 int32_t nas_rxInfo::rscp

8.262.2.5 int32_t nas_rxInfo::rsrp

8.262.2.6 int32_t nas_rxInfo::rxPower

8.263 nas_RxSigInfo Struct Reference

Data Fields

- uint8_t rxChainIndex
- uint8_t isRadioTuned
- int32_t rxPower
- int32_t rsrp

8.263.1 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

8.263.2 Field Documentation

8.263.2.1 uint8_t nas_RxSigInfo::isRadioTuned

8.263.2.2 int32_t nas_RxSigInfo::rsrp

8.263.2.3 uint8_t nas_RxSigInfo::rxChainIndex

8.263.2.4 int32_t nas_RxSigInfo::rxPower

8.264 nas_rxSignalStrengthListElement Struct Reference

Data Fields

- int16_t rxSignalStrength
- uint8_t radioIf

8.264.1 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.

8.264.2 Field Documentation

8.264.2.1 uint8_t nas_rxSignalStrengthListElement::radioIf

8.264.2.2 int16_t nas_rxSignalStrengthListElement::rxSignalStrength

8.265 nas_SccRxInfo Struct Reference

Data Fields

- int32_t rsrq
- int16_t snr
- uint8_t numInstances
- nas_RxSigInfo sigInfo [255]
- uint8_t TlvPresent

8.265.1 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.

8.265.2 Field Documentation

8.265.2.1 uint8_t nas_SccRxInfo::numInstances

8.265.2.2 int32_t nas_SccRxInfo::rsrq

8.265.2.3 nas_RxSigInfo nas_SccRxInfo::sigInfo[255]

8.265.2.4 int16_t nas_SccRxInfo::snr

8.265.2.5 uint8_t nas_SccRxInfo::TlvPresent

8.266 nas_serviceProviderName Struct Reference

Data Fields

- uint8_t displayCondition
- uint8_t spnLength
- uint8_t spn [255]

8.266.1 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.

8.266.2 Field Documentation

8.266.2.1 uint8_t nas_serviceProviderName::displayCondition

8.266.2.2 uint8_t nas_serviceProviderName::spn[255]

8.266.2.3 uint8_t nas_serviceProviderName::spnLength

8.267 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]

8.267.1 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

8.267.2 Field Documentation

8.267.2.1 uint8_t nas_servSystem::csAttachState

8.267.2.2 uint8_t nas_servSystem::numRadioInterfaces

8.267.2.3 uint8_t nas_servSystem::psAttachState

8.267.2.4 uint8_t nas_servSystem::radioInterface[32]

8.267.2.5 uint8_t nas_servSystem::regState

8.267.2.6 uint8_t nas_servSystem::selNetwork

8.268 nas_sidNid Struct Reference

Data Fields

- uint16_t nid
- uint16_t sid

8.268.1 Detailed Description

This structure contains the parameters for SidNid Information

Parameters

nid	• Network ID
sid	• System ID

8.268.2 Field Documentation

8.268.2.1 uint16_t nas_sidNid::nid

8.268.2.2 uint16_t nas_sidNid::sid

8.269 nas_SignalStrengthTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- int8_t signalStrength
- uint32_t radioInterface

8.269.1 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

8.269.2 Field Documentation

8.269.2.1 uint32_t nas_SignalStrengthTlv::radioInterface

8.269.2.2 int8_t nas_SignalStrengthTlv::signalStrength

8.269.2.3 uint8_t nas_SignalStrengthTlv::TlvPresent

8.270 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

8.270.1 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.

8.270.2 Field Documentation

8.270.2.1 uint8_t nas_SLQSSignalStrengthsIndReq::ecioDelta

8.270.2.2 int16_t nas_SLQSSignalStrengthsIndReq::ecioThresholdList[10]

8.270.2.3 uint8_t nas_SLQSSignalStrengthsIndReq::ecioThresholdListLen

8.270.2.4 uint8_t nas_SLQSSignalStrengthsIndReq::ioDelta

8.270.2.5 uint8_t nas_SLQSSignalStrengthsIndReq::lteRsrpDelta

8.270.2.6 uint16_t nas_SLQSSignalStrengthsIndReq::lteSnrDelta

8.270.2.7 uint8_t nas_SLQSSignalStrengthsIndReq::rsrqDelta

8.270.2.8 uint8_t nas_SLQSSignalStrengthsIndReq::rxSignalStrengthDelta

8.270.2.9 uint8_t nas_SLQSSignalStrengthsIndReq::sinrDelta

8.270.2.10 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdList[5]

8.270.2.11 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdListLen

8.271 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

8.271.1 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**

8.271.2 Field Documentation

8.271.2.1 nas_ecioListElement nas_SLQSSignalStrengthsInformation::ecioInfo

8.271.2.2 nas_errorRateListElement nas_SLQSSignalStrengthsInformation::errorRateInfo

8.271.2.3 uint32_t nas_SLQSSignalStrengthsInformation::io

8.271.2.4 nas_lteRsrpinformation nas_SLQSSignalStrengthsInformation::lteRsrpinfo

8.271.2.5 nas_lteSnrinformation nas_SLQSSignalStrengthsInformation::lteSnrinfo

8.271.2.6 nas_rsrqInformation nas_SLQSSignalStrengthsInformation::rsrqInfo

8.271.2.7 nas_rxSignalStrengthListElement nas_SLQSSignalStrengthsInformation::rxSignalStrengthInfo

8.271.2.8 uint8_t nas_SLQSSignalStrengthsInformation::sinr

8.272 nas_SLQSSignalStrengthsTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- nas_SLQSSignalStrengthsInformation sSLQSSignalStrengthsInfo

8.272.1 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

8.272.2 Field Documentation

8.272.2.1 nas_SLQSSignalStrengthsInformation nas_SLQSSignalStrengthsTlv::sSLQSSignalStrengthsInfo

8.272.2.2 uint8_t nas_SLQSSignalStrengthsTlv::TlvPresent

8.273 nas_SrvDomainPrefTlv Struct Reference

Data Fields

- uint32_t SrvDomainPref
- uint8_t TlvPresent

8.273.1 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

8.273.2 Field Documentation

8.273.2.1 uint32_t nas_SrvDomainPrefTlv::SrvDomainPref

8.273.2.2 uint8_t nas_SrvDomainPrefTlv::TlvPresent

8.274 nas_SrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t isPrefDataPath

8.274.1 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

8.274.2 Field Documentation

8.274.2.1 uint8_t nas_SrvStatusInfo::isPrefDataPath

8.274.2.2 uint8_t nas_SrvStatusInfo::srvStatus

8.275 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

8.275.1 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

8.275.2 Field Documentation

8.275.2.1 uint8_t nas_sysInfoCommon::isSysForbidden

8.275.2.2 uint8_t nas_sysInfoCommon::isSysForbiddenValid

8.275.2.3 uint8_t nas_sysInfoCommon::roamStatus

8.275.2.4 uint8_t nas_sysInfoCommon::roamStatusValid

8.275.2.5 uint8_t nas_sysInfoCommon::srvCapability

8.275.2.6 uint8_t nas_sysInfoCommon::srvCapabilityValid

8.275.2.7 uint8_t nas_sysInfoCommon::srvDomain

8.275.2.8 uint8_t nas_sysInfoCommon::srvDomainValid

8.276 nas_TDSCDMAECIOThresh Struct Reference

Data Fields

- uint8_t TDSCDMAECIOThreshListLen
- float * pTDSCDMAECIOThreshList

8.276.1 Detailed Description

This structure contains TDSCDMA ECIO threshold related parameters.

Parameters

TDSCDMAECIOThreshListLen	• 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.

8.276.2 Field Documentation

8.276.2.1 float* nas_TDSCDMAECIOThresh::pTDSCDMAECIOThreshList

8.276.2.2 uint8_t nas_TDSCDMAECIOThresh::TDSCDMAECIOThreshListLen

8.277 nas_TDSCDMARSCPThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSCPThreshListLen
- int16_t * pTDSCDMARSCPThreshList

8.277.1 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).

8.277.2 Field Documentation

8.277.2.1 int16_t* nas_TDSCDMARSCPThresh::pTDSCDMARSCPThreshList

8.277.2.2 uint8_t nas_TDSCDMARSCPThresh::TDSCDMARSCPThreshListLen

8.278 nas_TDSCDMARSSIThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSSIThreshListLen
- float * pTDSCDMARSSIThreshList

8.278.1 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.

8.278.2 Field Documentation

8.278.2.1 float* nas_TDSCDMARSSIThresh::pTDSCDMARSSIThreshList

8.278.2.2 uint8_t nas_TDSCDMARSSIThresh::TDSCDMARSSIThreshListLen

8.279 nas_TDSCDMASINRCONFThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRCONFThreshListLen
- float * pTDSCDMASINRCONFThreshList

8.279.1 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

8.279.2 Field Documentation

8.279.2.1 float* nas_TDSCDMASINRCONFThresh::pTDSCDMASINRCONFThreshList

8.279.2.2 uint8_t nas_TDSCDMASINRCONFThresh::TDSCDMASINRCONFThreshListLen

8.280 nas_TDSCDMASINRThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRThreshListLen
- float * pTDSCDMASINRThreshList

8.280.1 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

8.280.2 Field Documentation

8.280.2.1 float* nas_TDSCDMASINRThresh::pTDSCDMASINRThreshList

8.280.2.2 uint8_t nas_TDSCDMASINRThresh::TDSCDMASINRThreshListLen

8.281 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

8.281.1 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.

8.281.2 Field Documentation

8.281.2.1 uint8_t nas_timeInfo::day

8.281.2.2 uint8_t nas_timeInfo::dayLtSavingAdj

8.281.2.3 uint8_t nas_timeInfo::dayOfWeek

8.281.2.4 uint8_t nas_timeInfo::hour

8.281.2.5 uint8_t nas_timeInfo::minute

8.281.2.6 uint8_t nas_timeInfo::month

8.281.2.7 uint8_t nas_timeInfo::radioInterface

8.281.2.8 uint8_t nas_timeInfo::second

8.281.2.9 int8_t nas_timeInfo::timeZone

8.281.2.10 uint8_t nas_timeInfo::TlvPresent

8.281.2.11 uint16_t nas_timeInfo::year

8.282 nas_trueIMSI Struct Reference

Data Fields

- uint8_t mccT [3]
- uint16_t imsiT1112
- uint8_t imsiTS1 [7]
- uint8_t imsiTS2 [3]
- uint8_t imsiTaddrNum

8.282.1 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

8.282.2 Field Documentation

8.282.2.1 uint16_t nas_truelMSI::imsiT1112

8.282.2.2 uint8_t nas_truelMSI::imsiTaddrNum

8.282.2.3 uint8_t nas_truelMSI::imsiTS1[7]

8.282.2.4 uint8_t nas_truelMSI::imsiTS2[3]

8.282.2.5 uint8_t nas_truelMSI::mccT[3]

8.283 nas_txInfo Struct Reference

Data Fields

- uint8_t isInTraffic
- int32_t txPower

8.283.1 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.

8.283.2 Field Documentation

8.283.2.1 uint8_t nas_txInfo::isInTraffic

8.283.2.2 int32_t nas_txInfo::txPower

8.284 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]

8.284.1 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.

8.284.2 Field Documentation

8.284.2.1 uint16_t nas_UMTSInfo::cellID

8.284.2.2 int16_t nas_UMTSInfo::ecio

8.284.2.3 uint8_t nas_UMTSInfo::geranInst

8.284.2.4 nas_geranInstInfo nas_UMTSInfo::GeranInstInfo[255]

8.284.2.5 uint16_t nas_UMTSInfo::lac

8.284.2.6 uint8_t nas_UMTSInfo::plmn[3]

8.284.2.7 uint16_t nas_UMTSInfo::psc

8.284.2.8 int16_t nas_UMTSInfo::rscp

8.284.2.9 uint16_t nas_UMTSInfo::uarfcn

8.284.2.10 uint8_t nas_UMTSInfo::umtsInst

8.284.2.11 nas_UMTSinstInfo nas_UMTSInfo::UMTSInstInfo[255]

8.285 nas_UMTSinstInfo Struct Reference

Data Fields

- uint16_t umtsUarfcn
- uint16_t umtsPsc
- int16_t umtsRscp
- int16_t umtsEcio

8.285.1 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).

8.285.2 Field Documentation

8.285.2.1 int16_t nas_UMTSinstInfo::umtsEcio

8.285.2.2 uint16_t nas_UMTSinstInfo::umtsPsc

8.285.2.3 int16_t nas_UMTSinstInfo::umtsRscp

8.285.2.4 uint16_t nas_UMTSinstInfo::umtsUarfcn

8.286 nas_umtsLTENbrCell Struct Reference

Data Fields

- uint16_t earfcn
- uint16_t pci
- uint32_t rsrp
- uint32_t rsrq
- int16_t srxlev
- uint8_t cellIsTDD

8.286.1 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.

8.286.2 Field Documentation

8.286.2.1 uint8_t nas_umtsLTENbrCell::cellIsTDD

8.286.2.2 uint16_t nas_umtsLTENbrCell::earfcn

8.286.2.3 uint16_t nas_umtsLTENbrCell::pci

8.286.2.4 uint32_t nas_umtsLTENbrCell::rsrp

8.286.2.5 uint32_t nas_umtsLTENbrCell::rsrq

8.286.2.6 int16_t nas_umtsLTENbrCell::srxlev

8.287 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

8.287.1 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.

8.287.2 Field Documentation

8.287.2.1 uint8_t nas_UniversalTime::day

8.287.2.2 uint8_t nas_UniversalTime::dayOfWeek

8.287.2.3 uint8_t nas_UniversalTime::hour

8.287.2.4 uint8_t nas_UniversalTime::minute

8.287.2.5 uint8_t nas_UniversalTime::month

8.287.2.6 uint8_t nas_UniversalTime::second

8.287.2.7 uint16_t nas_UniversalTime::year

8.288 nas_wcdmaCellInfo Struct Reference

Data Fields

- uint16_t psc
- int16_t cpich_rscp
- int16_t cpich_ecno
- int16_t srxlev

8.288.1 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.

8.288.2 Field Documentation

8.288.2.1 int16_t nas_wcdmaCellInfo::cpich_ecno

8.288.2.2 int16_t nas_wcdmaCellInfo::cpich_rscp

8.288.2.3 uint16_t nas_wcdmaCellInfo::psc

8.288.2.4 int16_t nas_wcdmaCellInfo::srxlev

8.289 nas_WCDMAECIOThresh Struct Reference

Data Fields

- uint8_t WCDMAECIOThreshListLen
- int16_t * pWCDMAECIOThreshList

8.289.1 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)

8.289.2 Field Documentation

8.289.2.1 int16_t* nas_WCDMAECIOThresh::pWCDMAECIOThreshList

8.289.2.2 uint8_t nas_WCDMAECIOThresh::WCDMAECIOThreshListLen

8.290 nas_WCDMAInfoLTENeighborCell Struct Reference

Data Fields

- uint32_t wcdmaRRCState
- uint8_t umtsLTENbrCellLen
- nas_umtsLTENbrCell UMTSLTENbrCell [255]

8.290.1 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.

8.290.2 Field Documentation

8.290.2.1 nas_umtsLTENbrCell nas_WCDMAInfoLTENeighborCell::UMTSLTENbrCell[255]

8.290.2.2 uint8_t nas_WCDMAInfoLTENeighborCell::umtsLTENbrCellLen

8.290.2.3 uint32_t nas_WCDMAInfoLTENeighborCell::wcdmaRRCState

8.291 nas_WCDMARSSIThresh Struct Reference

Data Fields

- uint8_t WCDMARSSIThreshListLen
- int16_t ∗ pWCDMARSSIThreshList

8.291.1 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)

8.291.2 Field Documentation

8.291.2.1 int16_t* nas_WCDMARSSIThresh::pWCDMARSSIThreshList

8.291.2.2 uint8_t nas_WCDMARSSIThresh::WCDMARSSIThreshListLen

8.292 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

8.292.1 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_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.

hsCallStatus-Valid	• 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

8.292.2 Field Documentation

8.292.2.1 uint32_t nas_WCDMASysInfo::cellId

8.292.2.2 uint8_t nas_WCDMASysInfo::cellIdValid

8.292.2.3 uint8_t nas_WCDMASysInfo::hsCallStatus

8.292.2.4 uint8_t nas_WCDMASysInfo::hsCallStatusValid

8.292.2.5 uint8_t nas_WCDMASysInfo::hsInd

8.292.2.6 uint8_t nas_WCDMASysInfo::hsIndValid

8.292.2.7 uint16_t nas_WCDMASysInfo::lac

8.292.2.8 uint8_t nas_WCDMASysInfo::lacValid

8.292.2.9 uint8_t nas_WCDMASysInfo::MCC[3]

8.292.2.10 uint8_t nas_WCDMASysInfo::MNC[3]

8.292.2.11 uint8_t nas_WCDMASysInfo::networkIdValid

8.292.2.12 uint16_t nas_WCDMASysInfo::psc

8.292.2.13 uint8_t nas_WCDMASysInfo::pscValid

8.292.2.14 uint8_t nas_WCDMASysInfo::regRejectInfoValid

8.292.2.15 uint8_t nas_WCDMASysInfo::rejCause

8.292.2.16 uint8_t nas_WCDMASysInfo::rejectSrvDomain

8.292.2.17 nas_sysInfoCommon nas_WCDMASysInfo::sysInfoWCDMA

8.293 nas_wcdmaUARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t uarfcn

8.293.1 Detailed Description

This structure contains WCDMA UARFCN infromation.

Parameters

status	• 0 - Disable • 1 - Enable
uarfcn	• UARFCN to which UE is locked

8.293.2 Field Documentation

8.293.2.1 uint8_t nas_wcdmaUARFCN::status

8.293.2.2 uint32_t nas_wcdmaUARFCN::uarfcn

8.294 NASAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t AcqOrderLen
- uint8_t AcqOrderPref [255]

8.294.1 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

8.294.2 Field Documentation

8.294.2.1 uint8_t NASAcqOrderPrefTlv::AcqOrderLen

8.294.2.2 uint8_t NASAcqOrderPrefTlv::AcqOrderPref[255]

8.294.2.3 uint8_t NASAcqOrderPrefTlv::TlvPresent

8.295 NASBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t band_pref

8.295.1 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

8.295.2 Field Documentation

8.295.2.1 uint64_t NASBandPreferenceTlv::band_pref

8.295.2.2 uint8_t NASBandPreferenceTlv::TlvPresent

8.296 NASCiotAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t CiotAcqOrderLen
- uint32_t CiotAcqOrderPref [255]

8.296.1 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

8.296.2 Field Documentation

8.296.2.1 uint8_t NASCiotAcqOrderPrefTlv::CiotAcqOrderLen

8.296.2.2 uint32_t NASCiotAcqOrderPrefTlv::CiotAcqOrderPref[255]

8.296.2.3 uint8_t NASCiotAcqOrderPrefTlv::TlvPresent

8.297 NASCiotLteOpModePrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t CiotLteOpModePref

8.297.1 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

8.297.2 Field Documentation

8.297.2.1 uint64_t NASCiotLteOpModePrefTlv::CiotLteOpModePref

8.297.2.2 uint8_t NASCiotLteOpModePrefTlv::TlvPresent

8.298 NASEmergencyModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EmerMode

8.298.1 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

8.298.2 Field Documentation

8.298.2.1 uint8_t NASEmergencyModeTlv::EmerMode

8.298.2.2 uint8_t NASEmergencyModeTlv::TlvPresent

8.299 NasGetLTECphyCaInfo Struct Reference

Data Fields

- NASPhyCaAggScellIndType PhyCaAggScellIndType
- NASPhyCaAggScellDlBw PhyCaAggScellDlBw
- NASPhyCaAggScellInfo PhyCaAggScellInfo
- NASPhyCaAggPcellInfo PhyCaAggPcellInfo
- NASPhyCaAggScellIndex PhyCaAggScellIndex
- NASPhyCaAggScellArray PhyCaAggScellArray

8.299.1 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**

8.299.2 Field Documentation

8.299.2.1 NASPhyCaAggPcellInfo NasGetLTECphyCaInfo::PhyCaAggPcellInfo

8.299.2.2 NASPhyCaAggScellArray NasGetLTECphyCaInfo::PhyCaAggScellArray

8.299.2.3 NASPhyCaAggScellDlBw NasGetLTECphyCaInfo::PhyCaAggScellDlBw

8.299.2.4 NASPhyCaAggScellIndex NasGetLTECphyCaInfo::PhyCaAggScellIndex

8.299.2.5 NASPhyCaAggScellIndType NasGetLTECphyCaInfo::PhyCaAggScellIndType

8.299.2.6 NASPhyCaAggScellInfo NasGetLTECphyCaInfo::PhyCaAggScellInfo

8.300 NASGWAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t GWAcqOrderPref

8.300.1 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

8.300.2 Field Documentation

8.300.2.1 uint32_t NASGWAcqOrderPrefTlv::GWAcqOrderPref

8.300.2.2 uint8_t NASGWAcqOrderPrefTlv::TlvPresent

8.301 NASLTEBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LTEBandPref

8.301.1 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

8.301.2 Field Documentation

8.301.2.1 uint64_t NASLTEBandPreferenceTlv::LTEBandPref

8.301.2.2 uint8_t NASLTEBandPreferenceTlv::TlvPresent

8.302 NASLteM1BandPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LteM1BandPref

8.302.1 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

8.302.2 Field Documentation

8.302.2.1 uint64_t NASLteM1BandPrefTlv::LteM1BandPref

8.302.2.2 uint8_t NASLteM1BandPrefTlv::TlvPresent

8.303 NASLteNasReleaseInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t nas_release
- uint8_t nas_major
- uint8_t nas_minor

8.303.1 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

8.303.2 Field Documentation

8.303.2.1 uint8_t NASLteNasReleaseInfoTlv::nas_major

8.303.2.2 uint8_t NASLteNasReleaseInfoTlv::nas_minor

8.303.2.3 uint8_t NASLteNasReleaseInfoTlv::nas_release

8.303.2.4 uint8_t NASLteNasReleaseInfoTlv::TlvPresent

8.304 NASLteNB1BandPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LteNB1BandPref

8.304.1 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

8.304.2 Field Documentation

8.304.2.1 uint64_t NASLteNB1BandPrefTlv::LteNB1BandPref

8.304.2.2 uint8_t NASLteNB1BandPrefTlv::TlvPresent

8.305 NASModePreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ModePref

8.305.1 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

8.305.2 Field Documentation

8.305.2.1 uint16_t NASModePreferenceTlv::ModePref

8.305.2.2 uint8_t NASModePreferenceTlv::TlvPresent

8.306 NASNetSelPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t NetSelPref

8.306.1 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

8.306.2 Field Documentation

8.306.2.1 uint8_t NASNetSelPreferenceTlv::NetSelPref

8.306.2.2 uint8_t NASNetSelPreferenceTlv::TlvPresent

8.307 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

8.307.1 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

8.307.2 Field Documentation

8.307.2.1 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits1_64

8.307.2.2 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits_129_192

8.307.2.3 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits_193_256

8.307.2.4 uint64_t NASNr5gBandPrefTlv::Nr5gBandPrefbits_65_128

8.307.2.5 uint8_t NASNr5gBandPrefTlv::TlvPresent

8.308 NASOTAMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t message_type
- uint16_t data_len
- uint8_t data_buf [2048]

8.308.1 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

8.308.2 Field Documentation

8.308.2.1 uint8_t NASOTAMessageTlv::data_buf[2048]

8.308.2.2 uint16_t NASOTAMessageTlv::data_len

8.308.2.3 uint32_t NASOTAMessageTlv::message_type

8.308.2.4 uint8_t NASOTAMessageTlv::TlvPresent

8.309 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

8.309.1 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.

8.309.2 Field Documentation

8.309.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggPcellInfo::dl_bw_value

8.309.2.2 uint32_t NASPhyCaAggPcellInfo::freq

8.309.2.3 uint32_t NASPhyCaAggPcellInfo::iLTEbandValue

8.309.2.4 uint32_t NASPhyCaAggPcellInfo::pci

8.309.2.5 uint8_t NASPhyCaAggPcellInfo::TlvPresent

8.310 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

8.310.1 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.

8.310.2 Field Documentation

8.310.2.1 uint16_t NASPhyCaAggScellArray::band[255]

8.310.2.2 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellArray::cphy_ca_dl_bandwidth[255]

8.310.2.3 uint8_t NASPhyCaAggScellArray::cphy_scell_info_list_len

8.310.2.4 uint16_t NASPhyCaAggScellArray::freq[255]

8.310.2.5 uint16_t NASPhyCaAggScellArray::pci[255]

8.310.2.6 uint16_t NASPhyCaAggScellArray::scell_idx[255]

8.310.2.7 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellArray::scell_state[255]

8.310.2.8 uint8_t NASPhyCaAggScellArray::TlvPresent

8.311 NASPhyCaAggScellDlBw Struct Reference

Data Fields

- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint8_t TlvPresent

8.311.1 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.

8.311.2 Field Documentation

8.311.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellDlBw::dl_bw_value

8.311.2.2 uint8_t NASPhyCaAggScellDlBw::TlvPresent

8.312 NASPhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.312.1 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.

8.312.2 Field Documentation

8.312.2.1 uint8_t NASPhyCaAggScellIndex::scell_idx

8.312.2.2 uint8_t NASPhyCaAggScellIndex::TlvPresent

8.313 NASPhyCaAggScellIndType Struct Reference

Data Fields

- uint32_t pci
- uint32_t freq
- LITEQMI_NAS_LTE_CPHY_SCELL_STATE scell_state
- uint8_t TlvPresent

8.313.1 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.

8.313.2 Field Documentation

8.313.2.1 uint32_t NASPhyCaAggScellIndType::freq

8.313.2.2 uint32_t NASPhyCaAggScellIndType::pci

8.313.2.3 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellIndType::scell_state

8.313.2.4 uint8_t NASPhyCaAggScellIndType::TlvPresent

8.314 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

8.314.1 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.

8.314.2 Field Documentation

8.314.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellInfo::dl_bw_value

8.314.2.2 uint32_t NASPhyCaAggScellInfo::freq

8.314.2.3 uint32_t NASPhyCaAggScellInfo::iLTEbandValue

8.314.2.4 uint32_t NASPhyCaAggScellInfo::pci

8.314.2.5 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellInfo::scell_state

8.314.2.6 uint8_t NASPhyCaAggScellInfo::TlvPresent

8.315 NASPRLPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t PRLPref

8.315.1 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

8.315.2 Field Documentation

8.315.2.1 uint16_t NASPRLPreferenceTlv::PRLPref

8.315.2.2 uint8_t NASPRLPreferenceTlv::TlvPresent

8.316 NASQmiCbkNasSwiOTAMessageInd Struct Reference

Data Fields

- NASOTAMessageTlv otaMsgTlv
- NASLteNasReleaseInfoTlv nasRelInfoTlv
- NASTimeInfoTlv timeTlv

8.316.1 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**

8.316.2 Field Documentation

8.316.2.1 NASLteNasReleaseInfoTlv NASQmiCbkNasSwiOTAMessageInd::nasRelInfoTlv

8.316.2.2 NASOTAMessageTlv NASQmiCbkNasSwiOTAMessageInd::otaMsgTlv

8.316.2.3 NASTimeInfoTlv NASQmiCbkNasSwiOTAMessageInd::timeTlv

8.317 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

8.317.1 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

8.317.2 Field Documentation

8.317.2.1 NASAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::AOPTlv

8.317.2.2 NASBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::BPTlv

8.317.2.3 NASCiotAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::CiotAOPTlv

8.317.2.4 NASCiotLteOpModePrefTlv NASQmiCbkNasSystemSelPrefInd::CiotOpMPTlv

8.317.2.5 NASEmergencyModeTlv NASQmiCbkNasSystemSelPrefInd::EMTlv

8.317.2.6 NASGWAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::GWAOPTlv

8.317.2.7 NASLTEBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::LBPTlv

8.317.2.8 NASLteM1BandPrefTlv NASQmiCbkNasSystemSelPrefInd::M1BandPTlv

8.317.2.9 NASModePreferenceTlv NASQmiCbkNasSystemSelPrefInd::MPTlv

8.317.2.10 NASLteNB1BandPrefTlv NASQmiCbkNasSystemSelPrefInd::NB1BandPTlv

8.317.2.11 NASNr5gBandPrefTlv NASQmiCbkNasSystemSelPrefInd::NR5gBandPTlv

8.317.2.12 NASNetSelPreferenceTlv NASQmiCbkNasSystemSelPrefInd::NSPTlv

8.317.2.13 NASPRLPreferenceTlv NASQmiCbkNasSystemSelPrefInd::PRLPTlv

8.317.2.14 NASRatDisabledMaskTlv NASQmiCbkNasSystemSelPrefInd::RatDMTlv

8.317.2.15 NASRoamPreferenceTlv NASQmiCbkNasSystemSelPrefInd::RPTlv

8.317.2.16 NASServDomainPrefTlv NASQmiCbkNasSystemSelPrefInd::SDPTlv

8.318 NASRatDisabledMaskTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RatDisabledMask

8.318.1 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

8.318.2 Field Documentation

8.318.2.1 uint16_t NASRatDisabledMaskTlv::RatDisabledMask

8.318.2.2 uint8_t NASRatDisabledMaskTlv::TlvPresent

8.319 NASRoamPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RoamPref

8.319.1 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

8.319.2 Field Documentation

8.319.2.1 uint16_t NASRoamPreferenceTlv::RoamPref

8.319.2.2 uint8_t NASRoamPreferenceTlv::TlvPresent

8.320 NASServDomainPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SrvDomainPref

8.320.1 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

8.320.2 Field Documentation

8.320.2.1 uint32_t NASServDomainPrefTlv::SrvDomainPref

8.320.2.2 uint8_t NASServDomainPrefTlv::TlvPresent

8.321 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

8.321.1 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

8.321.2 Field Documentation

8.321.2.1 uint8_t NASServingSystemInfo::csAttachState

8.321.2.2 uint8_t NASServingSystemInfo::hdrPersonality

8.321.2.3 uint8_t NASServingSystemInfo::psAttachState

8.321.2.4 uint8_t NASServingSystemInfo::radioInterfaceList[255]

8.321.2.5 uint8_t NASServingSystemInfo::radioInterfaceNo

8.321.2.6 uint8_t NASServingSystemInfo::registrationState

8.321.2.7 uint8_t NASServingSystemInfo::selectedNetwork

8.322 NASTimeInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t time

8.322.1 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

8.322.2 Field Documentation

8.322.2.1 uint64_t NASTimeInfoTlv::time

8.322.2.2 uint8_t NASTimeInfoTlv::TlvPresent

8.323 newMTMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMTMessageInfo MTMessageInfo

8.323.1 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

8.323.2 Field Documentation

8.323.2.1 sMSMTMessageInfo newMTMessageTlv::MTMessageInfo

8.323.2.2 uint8_t newMTMessageTlv::TlvPresent

8.324 pack_audio_SLQSGetAudioPathConfig_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Item

8.324.1 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

8.324.2 Field Documentation

8.324.2.1 uint8_t pack_audio_SLQSGetAudioPathConfig_t::Item

8.324.2.2 uint8_t pack_audio_SLQSGetAudioPathConfig_t::Profile

8.325 pack_audio_SLQSGetAudioProfile_t Struct Reference

Data Fields

- uint8_t Generator

8.325.1 Detailed Description

This structure contains the pack parameters to Get Audio Profile.

Parameters

Generator	• Audio Generator – 0 - Voice – 1 - Key Beep – 2 - MIDI

8.325.2 Field Documentation

8.325.2.1 uint8_t pack_audio_SLQSGetAudioProfile_t::Generator

8.326 pack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator
- uint8_t Volume
- uint8_t Item

8.326.1 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

8.326.2 Field Documentation

8.326.2.1 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Generator

8.326.2.2 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Item

8.326.2.3 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Profile

8.326.2.4 uint8_t pack_audio_SLQSGetAudioVolTLBConfig_t::Volume

8.327 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

8.327.1 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

8.327.2 Field Documentation

8.327.2.1 uint16_t* pack_audio_SLQSSetAudioPathConfig_t::pCodecSTGain

8.327.2.2 uint16_t* pack_audio_SLQSSetAudioPathConfig_t::pDTMFTXGain

8.327.2.3 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pECMode

8.327.2.4 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pNSEnable

8.327.2.5 uint8_t pack_audio_SLQSSetAudioPathConfig_t::Profile

8.327.2.6 audio_RXAGCList* pack_audio_SLQSSetAudioPathConfig_t::pRXAGCList

8.327.2.7 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pRXAVCAGCSwitch

8.327.2.8 audio_RXAVCList* pack_audio_SLQSSetAudioPathConfig_t::pRXAVCList

8.327.2.9 audio_RXPCMIIRFltr* pack_audio_SLQSSetAudioPathConfig_t::pRXPCMIIRFltr

8.327.2.10 audio_TXAGCList* pack_audio_SLQSSetAudioPathConfig_t::pTXAGCList

8.327.2.11 uint8_t* pack_audio_SLQSSetAudioPathConfig_t::pTXAVCSwitch

8.327.2.12 uint16_t* pack_audio_SLQSSetAudioPathConfig_t::pTXGain

8.327.2.13 audio_TXPCMIIRFltr* pack_audio_SLQSSetAudioPathConfig_t::pTXPCMIIRFltr

8.328 pack_audio_SLQSSetAudioProfile_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t EarMute
- uint8_t MicMute
- uint8_t Generator
- uint8_t Volume

8.328.1 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

8.328.2 Field Documentation

8.328.2.1 uint8_t pack_audio_SLQSSetAudioProfile_t::EarMute

8.328.2.2 uint8_t pack_audio_SLQSSetAudioProfile_t::Generator

8.328.2.3 uint8_t pack_audio_SLQSSetAudioProfile_t::MicMute

8.328.2.4 uint8_t pack_audio_SLQSSetAudioProfile_t::Profile

8.328.2.5 uint8_t pack_audio_SLQSSetAudioProfile_t::Volume

8.329 pack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator
- uint8_t Volume
- uint8_t Item
- uint16_t VolValue

8.329.1 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

8.329.2 Field Documentation

8.329.2.1 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Generator

8.329.2.2 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Item

8.329.2.3 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Profile

8.329.2.4 uint8_t pack_audio_SLQSSetAudioVolTLBConfig_t::Volume

8.329.2.5 uint16_t pack_audio_SLQSSetAudioVolTLBConfig_t::VolValue

8.330 pack_cat_CATSendEnvelopeCommand_t Struct Reference

Data Fields

- uint32_t cmdID
- uint32_t dataLen
- uint8_t * pData

8.330.1 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]

8.330.2 Field Documentation

8.330.2.1 uint32_t pack_cat_CATSendEnvelopeCommand_t::cmdID

8.330.2.2 uint32_t pack_cat_CATSendEnvelopeCommand_t::dataLen

8.330.2.3 uint8_t* pack_cat_CATSendEnvelopeCommand_t::pData

8.331 pack_cat_CATSendTerminalResponse_t Struct Reference

Data Fields

- uint32_t refID
- uint32_t dataLen
- uint8_t * pData

8.331.1 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]

8.331.2 Field Documentation

8.331.2.1 uint32_t pack_cat_CATSendTerminalResponse_t::dataLen

8.331.2.2 uint8_t* pack_cat_CATSendTerminalResponse_t::pData

8.331.2.3 uint32_t pack_cat_CATSendTerminalResponse_t::refID

8.332 pack_cat_SetCATEventCallback_t Struct Reference

Data Fields

- uint32_t eventMask

8.332.1 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

8.332.2 Field Documentation

8.332.2.1 uint32_t pack_cat_SetCATEventCallback_t::eventMask

8.333 pack_dms_ActivateAutomatic_t Struct Reference

Data Fields

- uint8_t actCode [81]

8.333.1 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"

8.333.2 Field Documentation

8.333.2.1 uint8_t pack_dms_ActivateAutomatic_t::actCode[81]

8.334 pack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint8_t list_type
- uint16_t Tlvresult

8.334.1 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

8.334.2 Field Documentation

8.334.2.1 uint8_t pack_dms_GetCustFeaturesV2_t::cust_id[64+1]

8.334.2.2 uint8_t pack_dms_GetCustFeaturesV2_t::list_type

8.334.2.3 uint16_t pack_dms_GetCustFeaturesV2_t::Tlvresult

8.335 pack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint8_t spc [6]

8.335.1 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)

8.335.2 Field Documentation

8.335.2.1 uint8_t pack_dms_ResetToFactoryDefaults_t::spc[6]

8.336 pack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint8_t activationState

8.336.1 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

8.336.2 Field Documentation

8.336.2.1 uint8_t pack_dms_SetActivationStatusCallback_t::activationState

8.337 pack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t crashAction

8.337.1 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

8.337.2 Field Documentation

8.337.2.1 uint8_t pack_dms_SetCrashAction_t::crashAction

8.338 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

8.338.1 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

8.338.2 Field Documentation

8.338.2.1 uint8_t pack_dms_SetCustFeature_t::DHCPRelayEnabled

8.338.2.2 uint8_t pack_dms_SetCustFeature_t::DisableIMSI

8.338.2.3 uint32_t pack_dms_SetCustFeature_t::GpsEnable

8.338.2.4 uint8_t pack_dms_SetCustFeature_t::GPSLPM

8.338.2.5 uint8_t pack_dms_SetCustFeature_t::GPSSel

8.338.2.6 uint16_t pack_dms_SetCustFeature_t::IPFamSupport

8.338.2.7 uint8_t pack_dms_SetCustFeature_t::IsVoiceEnabled

8.338.2.8 uint8_t pack_dms_SetCustFeature_t::RMAutoConnect

8.338.2.9 uint8_t pack_dms_SetCustFeature_t::SMSSupport

8.339 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

8.339.1 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

8.339.2 Field Documentation

8.339.2.1 uint8_t pack_dms_SetCustFeaturesV2_t::cust_id[64+1]

8.339.2.2 uint8_t pack_dms_SetCustFeaturesV2_t::cust_value[8+1]

8.339.2.3 uint16_t pack_dms_SetCustFeaturesV2_t::Tlvresult

8.339.2.4 uint16_t pack_dms_SetCustFeaturesV2_t::value_length

8.340 pack_dms_SetEventReport_t Struct Reference

Data Fields

- uint8_t mode

8.340.1 Detailed Description

This structure is used to store pack_dms_SetEventReport parameters.

Parameters

mode	• Operating Mode • 0 - reset • 1 - set • 2 - no change

8.340.2 Field Documentation

8.340.2.1 uint8_t pack_dms_SetEventReport_t::mode

8.341 pack_dms_SetIndicationRegister_t Struct Reference

Data Fields

- uint8_t PSMStatus
- uint8_t PSMCfgChangeInfo
- uint8_t RptIMSCapability

8.341.1 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

8.341.2 Field Documentation

8.341.2.1 uint8_t pack_dms_SetIndicationRegister_t::PSMCfgChangeInfo

8.341.2.2 uint8_t pack_dms_SetIndicationRegister_t::PSMStatus

8.341.2.3 uint8_t pack_dms_SetIndicationRegister_t::RptIMSCapability

8.342 pack_dms_SetPower_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t Tlvresult

8.342.1 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.

8.342.2 Field Documentation

8.342.2.1 uint32_t pack_dms_SetPower_t::mode

8.342.2.2 uint16_t pack_dms_SetPower_t::Tlvresult

8.343 pack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint8_t USBComp
- uint16_t Tlvresult

8.343.1 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.

8.343.2 Field Documentation

8.343.2.1 uint16_t pack_dms_SetUSBComp_t::Tlvresult

8.343.2.2 uint8_t pack_dms_SetUSBComp_t::USBComp

8.344 pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint8_t resetInfoInd

8.344.1 Detailed Description

This structure contains set registration state for different indication pack

Parameters

resetInfoInd[IN]	• Values – 0 - Disable – 1 - Enable

8.344.2 Field Documentation

8.344.2.1 uint8_t pack_dms_SLQSDmsSwiIndicationRegister_t::resetInfoInd

8.345 pack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference

Data Fields

- uint8_t * pPsmEnableState
- uint32_t * pActiveTimer
- uint32_t * pPeriodicUpdateTimer

8.345.1 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.

8.345.2 Field Documentation

8.345.2.1 uint32_t* pack_dms_SLQSSetPowerSaveModeConfig_t::pActiveTimer

8.345.2.2 uint32_t* pack_dms_SLQSSetPowerSaveModeConfig_t::pPeriodicUpdateTimer

8.345.2.3 uint8_t* pack_dms_SLQSSetPowerSaveModeConfig_t::pPsmEnableState

8.346 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint8_t clear

8.346.1 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

8.346.2 Field Documentation

8.346.2.1 uint8_t pack_dms_SLQSSwiGetCrashInfo_t::clear

8.347 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.347.1 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.
pDestSMS-Content[IN]	• SMS COntent as a string of 8 bit ASCII text characters Max 160 chars. • Optional parameter.

8.347.2 Field Documentation

8.347.2.1 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSContent

8.347.2.2 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSNum

8.348 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- char manString [255]
- char modelString [255]
- char swVerString [255]
- char plasmaIDString [255]
- char hostID [255]

8.348.1 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

8.348.2 Field Documentation

8.348.2.1 char pack_dms_SLQSSwiSetHostDevInfo_t::hostID[255]

8.348.2.2 char pack_dms_SLQSSwiSetHostDevInfo_t::manString[255]

8.348.2.3 char pack_dms_SLQSSwiSetHostDevInfo_t::modelString[255]

8.348.2.4 char pack_dms_SLQSSwiSetHostDevInfo_t::plasmaIDString[255]

8.348.2.5 char pack_dms_SLQSSwiSetHostDevInfo_t::swVerString[255]

8.349 pack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- char nameString [255]
- char versionString [255]

8.349.1 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.

8.349.2 Field Documentation

8.349.2.1 char pack_dms_SLQSSwiSetOSInfo_t::nameString[255]

8.349.2.2 char pack_dms_SLQSSwiSetOSInfo_t::versionString[255]

8.350 pack_dms_SwiSetEventReport_t Struct Reference

Data Fields

- uint8_t * pTempReport
- uint8_t * pVoltReport
- uint8_t * pUIMStatusReport

8.350.1 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

8.350.2 Field Documentation

8.350.2.1 uint8_t* pack_dms_SwiSetEventReport_t::pTempReport

8.350.2.2 uint8_t* pack_dms_SwiSetEventReport_t::pUIMStatusReport

8.350.2.3 uint8_t* pack_dms_SwiSetEventReport_t::pVoltReport

8.351 pack_dms_SwiUimSelect_t Struct Reference

Data Fields

- uint8_t uim_select

8.351.1 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.

8.351.2 Field Documentation

8.351.2.1 uint8_t pack_dms_SwiUimSelect_t::uim_select

8.352 pack_dms_UIMChangePIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t oldValue [255]
- uint8_t newValue [255]

8.352.1 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

8.352.2 Field Documentation

8.352.2.1 uint8_t pack_dms_UIMChangePIN_t::id

8.352.2.2 uint8_t pack_dms_UIMChangePIN_t::newValue[255]

8.352.2.3 uint8_t pack_dms_UIMChangePIN_t::oldValue[255]

8.353 pack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facility

8.353.1 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)

8.353.2 Field Documentation

8.353.2.1 uint8_t pack_dms_UIMGetControlKeyStatus_t::facility

8.354 pack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.354.1 Detailed Description

This structure contains UIM Get ICCID pack

Parameters

Tlvresult	• Pack result.

8.354.2 Field Documentation

8.354.2.1 swi_uint256_t pack_dms_UIMGetICCID_t::ParamPresenceMask

8.354.2.2 uint16_t pack_dms_UIMGetICCID_t::Tlvresult

8.355 pack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t facility
- uint8_t facilityState
- uint8_t facilityCk [8]

8.355.1 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)

8.355.2 Field Documentation

8.355.2.1 uint8_t pack_dms_UIMSetControlKeyProtection_t::facility

8.355.2.2 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityCk[8]

8.355.2.3 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityState

8.356 pack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t id
- uint8_t bEnable
- uint8_t value [255]

8.356.1 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

8.356.2 Field Documentation

8.356.2.1 uint8_t pack_dms_UIMSetPINProtection_t::bEnable

8.356.2.2 uint8_t pack_dms_UIMSetPINProtection_t::id

8.356.2.3 uint8_t pack_dms_UIMSetPINProtection_t::value[255]

8.357 pack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t facility
- uint8_t facilityCk [8]

8.357.1 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)

8.357.2 Field Documentation

8.357.2.1 uint8_t pack_dms_UIMUnblockControlKey_t::facility

8.357.2.2 uint8_t pack_dms_UIMUnblockControlKey_t::facilityCk[8]

8.358 pack_dms_UIMUnblockPIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t pukValue [255]
- uint8_t newPin [255]

8.358.1 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

8.358.2 Field Documentation

8.358.2.1 uint8_t pack_dms_UIMUnblockPIN_t::id

8.358.2.2 uint8_t pack_dms_UIMUnblockPIN_t::newPin[255]

8.358.2.3 uint8_t pack_dms_UIMUnblockPIN_t::pukValue[255]

8.359 pack_dms_UIMVerifyPIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t value [255]

8.359.1 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

8.359.2 Field Documentation

8.359.2.1 uint8_t pack_dms_UIMVerifyPIN_t::id

8.359.2.2 uint8_t pack_dms_UIMVerifyPIN_t::value[255]

8.360 pack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint16_t TlvResult

8.360.1 Detailed Description

This structure contains the Get Image Preference information pack

Parameters

TlvResult	• Pack result

8.360.2 Field Documentation

8.360.2.1 uint16_t pack_fms_GetImagesPreference_t::TlvResult

8.361 pack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint16_t TlvResult

8.361.1 Detailed Description

This structure contains the Get Stored Images pack

Parameters

TlvResult	• Pack result

8.361.2 Field Documentation

8.361.2.1 uint16_t pack_fms_GetStoredImages_t::TlvResult

8.362 pack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t imageListSize
- FMSPrefImageList * pImageList
- uint32_t bForceDownload
- uint8_t modemIndex
- uint16_t TlvResult

8.362.1 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

8.362.2 Field Documentation

8.362.2.1 uint32_t pack_fms_SetImagesPreference_t::bForceDownload

8.362.2.2 uint32_t pack_fms_SetImagesPreference_t::imageListSize

8.362.2.3 uint8_t pack_fms_SetImagesPreference_t::modemindex

8.362.2.4 FMSPrefImageList* pack_fms_SetImagesPreference_t::pImageList

8.362.2.5 uint16_t pack_fms_SetImagesPreference_t::TlvResult

8.363 pack_ims_SLQSImsConfigIndicationRegister_t Struct Reference

Data Fields

- uint8_t * pSIPConfigEvents
- uint8_t * pRegMgrConfigEvents
- uint8_t * pSMSConfigEvents
- uint8_t * pUserConfigEvents
- uint8_t * pVoIPConfigEvents

8.363.1 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_US-ER_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_V-OIP_CONFIG_IND indication. – 0x00 - Disable – 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.363.2 Field Documentation

8.363.2.1 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pRegMgrConfigEvents

8.363.2.2 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pSIPConfigEvents

8.363.2.3 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pSMSConfigEvents

8.363.2.4 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pUserConfigEvents

8.363.2.5 uint8_t* pack_ims_SLQSImsConfigIndicationRegister_t::pVoIPConfigEvents

8.364 pack_ims_SLQSSetIMSSMSConfig_t Struct Reference

Data Fields

- uint8_t * pSMSFormat
- uint8_t * pSMSOverIPNwInd
- uint8_t * pPhoneCtxtURILen
- uint8_t * pPhoneCtxtURI

8.364.1 Detailed Description

This structure contains the SLQSSetIMSSMSConfig pack parameters.

Parameters

pSMSFormat	• SMS format – 0 - 3GPP – 1 - 3GPP2
pSMSOverIPNw-Ind	• SMS over IP Network Indication Flag – TRUE - Turn on mobile-originated SMS – FALSE - Turn off mobile-originated SMS
pPhoneCtxtURI-Len	• 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

8.364.2 Field Documentation

8.364.2.1 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pPhoneCtxtURI

8.364.2.2 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pPhoneCtxtURILen

8.364.2.3 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pSMSFormat

8.364.2.4 uint8_t* pack_ims_SLQSSetIMSSMSConfig_t::pSMSOverIPNwInd

8.365 pack_ims_SLQSSetIMSUserConfig_t Struct Reference

Data Fields

- uint8_t * pIMSDomainLen
- uint8_t * pIMSDomain

8.365.1 Detailed Description

This structure contains the SLQSSetIMSUserConfig pack parameters.

Parameters

pIMSDomainLen	• Length of IMS Domain Name to follow
pIMSDomain	• IMS domain name

8.365.2 Field Documentation

8.365.2.1 uint8_t* pack_ims_SLQSSetIMSUserConfig_t::pIMSDomain

8.365.2.2 uint8_t* pack_ims_SLQSSetIMSUserConfig_t::pIMSDomainLen

8.366 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

8.366.1 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.

8.366.2 Field Documentation

8.366.2.1 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrMode

8.366.2.2 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrOctetAligned

8.366.2.3 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrWbEnable

8.366.2.4 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrWBMode

8.366.2.5 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pAmrWBOctetAligned

8.366.2.6 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pMinSessionExpiryTimer

8.366.2.7 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pRingBackTimer

8.366.2.8 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pRingingTimer

8.366.2.9 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pRTPRTCPInactTimer

8.366.2.10 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pScrAmrEnable

8.366.2.11 uint8_t* pack_ims_SLQSSetIMSVoIPConfig_t::pScrAmrWbEnable

8.366.2.12 uint16_t* pack_ims_SLQSSetIMSVoIPConfig_t::pSessionExpiryTimer

8.367 pack_ims_SLQSSetRegMgrConfig_t Struct Reference

Data Fields

- uint16_t * pPriCSCFPort
- uint8_t * pCSCFPortNameLen
- uint8_t * pCSCFPortName
- uint8_t * pIMSTestMode

8.367.1 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

8.367.2 Field Documentation

8.367.2.1 uint8_t* pack_ims_SLQSSetRegMgrConfig_t::pCSCFPortName

8.367.2.2 uint8_t* pack_ims_SLQSSetRegMgrConfig_t::pCSCFPortNameLen

8.367.2.3 uint8_t* pack_ims_SLQSSetRegMgrConfig_t::pIMSTestMode

8.367.2.4 uint16_t* pack_ims_SLQSSetRegMgrConfig_t::pPriCSCFPort

8.368 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

8.368.1 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

8.368.2 Field Documentation

8.368.2.1 uint8_t* pack_ims_SLQSSetSIPConfig_t::pSigCompEnabled

8.368.2.2 uint16_t* pack_ims_SLQSSetSIPConfig_t::pSIPLocalPort

8.368.2.3 uint32_t* pack_ims_SLQSSetSIPConfig_t::pSubscribeTimer

8.368.2.4 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerSIPReg

8.368.2.5 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerT1

8.368.2.6 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerT2

8.368.2.7 uint32_t* pack_ims_SLQSSetSIPConfig_t::pTimerTf

8.369 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

8.369.1 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.

8.369.2 Field Documentation

8.369.2.1 int pack_imsa_SLQSRegisterIMSAIndication_t::has_PdpStatusConfig

8.369.2.2 int pack_imsa_SLQSRegisterIMSAIndication_t::has_RatHandoverStatusConfig

8.369.2.3 int pack_imsa_SLQSRegisterIMSAIndication_t::has_RegStatusConfig

8.369.2.4 int pack_imsa_SLQSRegisterIMSAIndication_t::has_ServiceStatusConfig

8.369.2.5 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::PdpStatusConfig

8.369.2.6 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::RatHandoverStatusConfig

8.369.2.7 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::RegStatusConfig

8.369.2.8 uint8_t pack_imsa_SLQSRegisterIMSAIndication_t::ServiceStatusConfig

8.370 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

8.370.1 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.

8.370.2 Field Documentation

8.370.2.1 loc_BdsSVInfo* pack_loc_Delete_Assist_Data_t::pBdsSVInfo

8.370.2.2 loc_CellDb* pack_loc_Delete_Assist_Data_t::pCellDb

8.370.2.3 loc_ClkInfo* pack_loc_Delete_Assist_Data_t::pClkInfo

8.370.2.4 loc_GnssData* pack_loc_Delete_Assist_Data_t::pGnssData

8.370.2.5 loc_SVInfo* pack_loc_Delete_Assist_Data_t::pSVInfo

8.370.2.6 uint16_t pack_loc_Delete_Assist_Data_t::TlvResult

8.371 pack_loc_EventRegister_t Struct Reference

Data Fields

- uint64_t eventRegister
- uint16_t TlvResult

8.371.1 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.

8.371.2 Field Documentation

8.371.2.1 uint64_t pack_loc_EventRegister_t::eventRegister

8.371.2.2 uint16_t pack_loc_EventRegister_t::TlvResult

8.372 pack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint32_t extPowerState
- uint16_t TlvResult

8.372.1 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.

8.372.2 Field Documentation

8.372.2.1 uint32_t pack_loc_SetExtPowerState_t::extPowerState

8.372.2.2 uint16_t pack_loc_SetExtPowerState_t::TlvResult

8.373 pack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t TlvResult

8.373.1 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.

8.373.2 Field Documentation

8.373.2.1 uint32_t pack_loc_SetOperationMode_t::mode

8.373.2.2 uint16_t pack_loc_SetOperationMode_t::Tlvresult

8.374 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint32_t xid
- uint16_t Tlvresult

8.374.1 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.

8.374.2 Field Documentation

8.374.2.1 uint16_t pack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.374.2.2 uint32_t pack_loc_SLQSLOCGetBestAvailPos_t::xid

8.375 pack_loc_SLQSLOCGetServer_t Struct Reference

Data Fields

- uint32_t serverType
- int has_serverAddrTypeMask
- uint8_t serverAddrTypeMask

8.375.1 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

8.375.2 Field Documentation

8.375.2.1 int pack_loc_SLQSLOCGetServer_t::has_serverAddrTypeMask

8.375.2.2 uint8_t pack_loc_SLQSLOCGetServer_t::serverAddrTypeMask

8.375.2.3 uint32_t pack_loc_SLQSLOCGetServer_t::serverType

8.376 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

8.376.1 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.

8.376.2 Field Documentation

8.376.2.1 altSrcInfo_t pack_loc_SLQSLOCInjectPosition_t::altitudeSrcInfo

8.376.2.2 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtEllipsoid

8.376.2.3 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtMeanSeaLevel

8.376.2.4 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeSrcInfo

8.376.2.5 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtEllipsoid

8.376.2.6 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtMeanSeaLevel

8.376.2.7 int pack_loc_SLQSLOCInjectPosition_t::has_horConfidence

8.376.2.8 int pack_loc_SLQSLOCInjectPosition_t::has_horReliability

8.376.2.9 int pack_loc_SLQSLOCInjectPosition_t::has_horUncCircular

8.376.2.10 int pack_loc_SLQSLOCInjectPosition_t::has_latitude

8.376.2.11 int pack_loc_SLQSLOCInjectPosition_t::has_longitude

8.376.2.12 int pack_loc_SLQSLOCInjectPosition_t::has_positionSrc

8.376.2.13 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorConfidence

8.376.2.14 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorUncCircular

8.376.2.15 int pack_loc_SLQSLOCInjectPosition_t::has_timestampAge

8.376.2.16 int pack_loc_SLQSLOCInjectPosition_t::has_timestampUtc

8.376.2.17 int pack_loc_SLQSLOCInjectPosition_t::has_vertConfidence

8.376.2.18 int pack_loc_SLQSLOCInjectPosition_t::has_vertRelicability

8.376.2.19 int pack_loc_SLQSLOCInjectPosition_t::has_vertUnc

8.376.2.20 uint8_t pack_loc_SLQSLOCInjectPosition_t::horConfidence

8.376.2.21 uint32_t pack_loc_SLQSLOCInjectPosition_t::horReliability

8.376.2.22 float pack_loc_SLQSLOCInjectPosition_t::horUncCircular

8.376.2.23 double pack_loc_SLQSLOCInjectPosition_t::latitude

8.376.2.24 double pack_loc_SLQSLOCInjectPosition_t::longitude

8.376.2.25 uint32_t pack_loc_SLQSLOCInjectPosition_t::positionSrc

8.376.2.26 uint8_t pack_loc_SLQSLOCInjectPosition_t::rawHorConfidence

8.376.2.27 float pack_loc_SLQSLOCInjectPosition_t::rawHorUncCircular

8.376.2.28 uint32_t pack_loc_SLQSLOCInjectPosition_t::timestampAge

8.376.2.29 uint64_t pack_loc_SLQSLOCInjectPosition_t::timestampUtc

8.376.2.30 uint8_t pack_loc_SLQSLOCInjectPosition_t::vertConfidence

8.376.2.31 uint32_t pack_loc_SLQSLOCInjectPosition_t::vertReliability

8.376.2.32 float pack_loc_SLQSLOCInjectPosition_t::vertUnc

8.377 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

8.377.1 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

8.377.2 Field Documentation

8.377.2.1 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::acceleroData

8.377.2.2 uint32_t pack_loc_SLQSLOCInjectSensorData_t::acceleroTimeSrc

8.377.2.3 tempData_t pack_loc_SLQSLOCInjectSensorData_t::accelTemp

8.377.2.4 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::gyroData

8.377.2.5 tempData_t pack_loc_SLQSLOCInjectSensorData_t::gyroTemp

8.377.2.6 uint32_t pack_loc_SLQSLOCInjectSensorData_t::gyroTimeSrc

8.377.2.7 int pack_loc_SLQSLOCInjectSensorData_t::has_acceleroTimeSrc

8.377.2.8 int pack_loc_SLQSLOCInjectSensorData_t::has_accelTemp

8.377.2.9 int pack_loc_SLQSLOCInjectSensorData_t::has_accleroData

8.377.2.10 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroData

8.377.2.11 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTemp

8.377.2.12 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTimeSrc

8.377.2.13 int pack_loc_SLQSLOCInjectSensorData_t::has_opaqueId

8.377.2.14 uint32_t pack_loc_SLQSLOCInjectSensorData_t::opaqueId

8.378 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference

Data Fields

- uint64_t timeMsec
- uint32_t timeUncMsec

8.378.1 Detailed Description

This structure contains inject UTC time parameter.

Parameters

timeMsec	• The UTC time since Jan. 1, 1970
timeUncMsec	• The time Uncertainty

8.378.2 Field Documentation

8.378.2.1 uint64_t pack_loc_SLQSLOCInjectUTCTime_t::timeMsec

8.378.2.2 uint32_t pack_loc_SLQSLOCInjectUTCTime_t::timeUncMsec

8.379 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference

Data Fields

- uint32_t state
- int has_confidence
- uint8_t confidence

8.379.1 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

8.379.2 Field Documentation

8.379.2.1 uint8_t pack_loc_SLQSLOCSetCradleMountConfig_t::confidence

8.379.2.2 int pack_loc_SLQSLOCSetCradleMountConfig_t::has_confidence

8.379.2.3 uint32_t pack_loc_SLQSLOCSetCradleMountConfig_t::state

8.380 pack_loc_SLQSLOCSetServer_t Struct Reference

Data Fields

- uint32_t serverType
- loc_IPv4Config * pIPv4Config
- loc_IPv6Config * pIPv6Config
- loc_URLAddrInfo * pURLAddr

8.380.1 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

8.380.2 Field Documentation

8.380.2.1 loc_IPv4Config* pack_loc_SLQSLOCSetServer_t::pIPv4Config

8.380.2.2 loc_IPv6Config* pack_loc_SLQSLOCSetServer_t::pIPv6Config

8.380.2.3 loc_URLAddrInfo* pack_loc_SLQSLOCSetServer_t::pURLAddr

8.380.2.4 uint32_t pack_loc_SLQSLOCSetServer_t::serverType

8.381 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

8.381.1 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.

8.381.2 Field Documentation

8.381.2.1 loc_LocApplicationInfo* pack_loc_Start_t::pApplicationInfo

8.381.2.2 uint32_t* pack_loc_Start_t::pConfigAltitudeAssumed

8.381.2.3 uint32_t* pack_loc_Start_t::pHorizontalAccuracyLvl

8.381.2.4 uint32_t* pack_loc_Start_t::pIntermediateReportState

8.381.2.5 uint32_t* pack_loc_Start_t::pMinIntervalTime

8.381.2.6 uint32_t* pack_loc_Start_t::pRecurrenceType

8.381.2.7 uint8_t pack_loc_Start_t::SessionId

8.381.2.8 uint16_t pack_loc_Start_t::Tlvresult

8.382 pack_loc_Stop_t Struct Reference

Data Fields

- uint8_t SessionId
- uint16_t Tlvresult

8.382.1 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.

8.382.2 Field Documentation

8.382.2.1 uint8_t pack_loc_Stop_t::SessionId

8.382.2.2 uint16_t pack_loc_Stop_t::Tlvresult

8.383 pack_nas_InitiateDomainAttach_t Struct Reference

Data Fields

- uint32_t action

8.383.1 Detailed Description

Pack structure for Initiates a domain attach/detach of the device.

Parameters

action	• Domain action to attempt 1 - Attach 2 - Detach

8.383.2 Field Documentation

8.383.2.1 uint32_t pack_nas_InitiateDomainAttach_t::action

8.384 pack_nas_PerformNetworkScanPCI_t Struct Reference

Data Fields

- uint8_t * pNetworkType
- uint32_t * pScanType
- uint64_t * pCiotOpModePref
- uint64_t * pLteM1BandPref
- uint64_t * pLteNB1BandPref

8.384.1 Field Documentation

8.384.1.1 uint64_t* pack_nas_PerformNetworkScanPCI_t::pCiotOpModePref

8.384.1.2 uint64_t* pack_nas_PerformNetworkScanPCI_t::pLteM1BandPref

8.384.1.3 uint64_t* pack_nas_PerformNetworkScanPCI_t::pLteNB1BandPref

8.384.1.4 uint8_t* pack_nas_PerformNetworkScanPCI_t::pNetworkType

8.384.1.5 uint32_t* pack_nas_PerformNetworkScanPCI_t::pScanType

8.385 pack_nas_SetACCOLC_t Struct Reference

Data Fields

- int8_t spc [6]
- uint8_t accolc

8.385.1 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

8.385.2 Field Documentation

8.385.2.1 uint8_t pack_nas_SetACCOLC_t::accolc

8.385.2.2 int8_t pack_nas_SetACCOLC_t::spc[6]

8.386 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

8.386.1 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

8.386.2 Field Documentation

8.386.2.1 uint32_t* pack_nas_SetCDMANetworkParameters_t::pApplication

8.386.2.2 uint32_t* pack_nas_SetCDMANetworkParameters_t::pBroadcast

8.386.2.3 uint8_t* pack_nas_SetCDMANetworkParameters_t::pCustomSCP

8.386.2.4 uint8_t* pack_nas_SetCDMANetworkParameters_t::pForceRev0

8.386.2.5 uint32_t* pack_nas_SetCDMANetworkParameters_t::pProtocol

8.386.2.6 uint32_t* pack_nas_SetCDMANetworkParameters_t::pRoaming

8.386.2.7 char* pack_nas_SetCDMANetworkParameters_t::pSPC

8.387 pack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint32_t TechnologyPref
- uint32_t Duration
- uint16_t Tlvresult

8.387.1 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

8.387.2 Field Documentation

8.387.2.1 uint32_t pack_nas_SetNetworkPreference_t::Duration

8.387.2.2 uint32_t pack_nas_SetNetworkPreference_t::TechnologyPref

8.387.2.3 uint16_t pack_nas_SetNetworkPreference_t::TlvResult

8.388 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

8.388.1 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

8.388.2 Field Documentation

8.388.2.1 nas_ECIOThresh∗ pack_nas_SLQSConfigSigInfo_t::pECIOThresh

8.388.2.2 nas_HDRSINRThresh∗ pack_nas_SLQSConfigSigInfo_t::pHDRSINRThresh

8.388.2.3 nas_IOThresh∗ pack_nas_SLQSConfigSigInfo_t::pIOThresh

8.388.2.4 nas_LTESigRptCfg∗ pack_nas_SLQSConfigSigInfo_t::pLTESigRptCfg

8.388.2.5 nas_LTESNRThresh∗ pack_nas_SLQSConfigSigInfo_t::pLTESNRThresh

8.388.2.6 nas_RSRPThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRPThresh

8.388.2.7 nas_RSRQThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRQThresh

8.388.2.8 nas_RSSIThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSSIThresh

8.388.2.9 nas_TDSCDMASINRCONFThresh∗ pack_nas_SLQSConfigSigInfo_t::pTDSCDMASINRCONFThresh

8.389 pack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t ∗ pMncPcsStatus

8.389.1 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.

8.389.2 Field Documentation

8.389.2.1 uint16_t pack_nas_SLQSGetPLMNName_t::mcc

8.389.2.2 uint16_t pack_nas_SLQSGetPLMNName_t::mnc

8.389.2.3 uint8_t* pack_nas_SLQSGetPLMNName_t::pMncPcsStatus

8.390 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference

Data Fields

- uint32_t regAction
- nas_MNRInfo * pMNRInfo
- uint32_t * pChangeDuration
- uint8_t * pMncPcsDigitStatus

8.390.1 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.

8.390.2 Field Documentation

8.390.2.1 uint32_t* pack_nas_SLQSInitiateNetworkRegistration_t::pChangeDuration

8.390.2.2 uint8_t* pack_nas_SLQSInitiateNetworkRegistration_t::pMncPcsDigitStatus

8.390.2.3 nas_MNRInfo* pack_nas_SLQSInitiateNetworkRegistration_t::pMNRInfo

8.390.2.4 uint32_t pack_nas_SLQSInitiateNetworkRegistration_t::regAction

8.391 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

8.391.1 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.

8.391.2 Field Documentation

8.391.2.1 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIODelta

8.391.2.2 nas_CDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIOThresh

8.391.2.3 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIDelta

8.391.2.4 nas_CDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIThresh

8.391.2.5 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIDelta

8.391.2.6 nas_GSMRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIThresh

8.391.2.7 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIODelta

8.391.2.8 nas_HDRECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIOThresh

8.391.2.9 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIODelta

8.391.2.10 nas_HDRIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIOThresh

8.391.2.11 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIDelta

8.391.2.12 nas_HDRRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIThresh

8.391.2.13 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRDelta

8.391.2.14 nas_HDRSINRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRThresh

8.391.2.15 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPDelta

8.391.2.16 nas_LTERSRPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPThresh

8.391.2.17 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQDelta

8.391.2.18 nas_LTERSRQThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQThresh

8.391.2.19 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIDelta

8.391.2.20 nas_LTERSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIThresh

8.391.2.21 nas_LTESigRptConfig* pack_nas_SLQSNasConfigSigInfo2_t::pLTESigRptConfig

8.391.2.22 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRDelta

8.391.2.23 nas_LTESNRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRThresh

8.391.2.24 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIODelta

8.391.2.25 nas_TDSCDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIOThresh

8.391.2.26 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPDelta

8.391.2.27 nas_TDSCDMARSCPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPThresh

8.391.2.28 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIDelta

8.391.2.29 nas_TDSCDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIThresh

8.391.2.30 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRDelta

8.391.2.31 nas_TDSCDMASINRThresh∗ pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRThresh

8.391.2.32 uint16_t∗ pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIODelta

8.391.2.33 nas_WCDMAECIOThresh∗ pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIOThresh

8.391.2.34 uint16_t∗ pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIDelta

8.391.2.35 nas_WCDMARSSIThresh∗ pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIThresh

8.392 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference

Data Fields

- uint8_t namID

8.392.1 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.

8.392.2 Field Documentation

8.392.2.1 uint8_t pack_nas_SLQSNasGet3GPP2Subscription_t::namID

8.393 pack_nas_SLQSNASGeteDRXParamsExt_t Struct Reference

Data Fields

- uint8_t ∗ pEdrxRAT
- uint32_t ∗ pLteOpMode

8.393.1 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.
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.

8.393.2 Field Documentation

8.393.2.1 uint8_t* pack_nas_SLQSNASGeteDRXParamsExt_t::pEdrxRAT

8.393.2.2 uint32_t* pack_nas_SLQSNASGeteDRXParamsExt_t::pLteOpMode

8.394 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- uint8_t radio_if

8.394.1 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

8.394.2 Field Documentation

8.394.2.1 uint8_t pack_nas_SLQSNasGetTxRxInfo_t::radio_if

8.395 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

8.395.1 Detailed Description

This structure contains pack indication register parameters.

Parameters

pSystem-SelectionInd	[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

pServing-SystemInd	[Optional] • Serving System indication registration. The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pDualStandBy-PrefInd	[Optional] • Dual Standby Preference indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSubscription-InfoInd	[Optional] • Subscription Information indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pNetworkTime-Ind	[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
pSignalStrength-Ind	[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
pHDRNewUATI-AssInd	[Optional] • HDR New UATI Assigned indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable

pHDRSession-CloseInd	[Optional] • HDR Session Closed indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pManaged-RoamingInd	[Optional] • Managed Roaming indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable
pNetworkReject-Ind	[Optional] • Network reject information indication registration.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSuppressSys-InfoInd	[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

8.395.2 Field Documentation

8.395.2.1 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDDTMInd

8.395.2.2 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDualStandByPrefInd

8.395.2.3 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pEdrxChangeInfoInd

8.395.2.4 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pErrorRateInd

8.395.2.5 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRNewUATIAssInd

8.395.2.6 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRSessionCloseInd

8.395.2.7 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pLTECphyCa

8.395.2.8 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pManagedRoamingInd

8.395.2.9 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkRejectInd

8.395.2.10 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkTimeInd

8.395.2.11 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pServingSystemInd

8.395.2.12 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSignalStrengthInd

8.395.2.13 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSubscriptionInfoInd

8.395.2.14 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSuppressSysInfoInd

8.395.2.15 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSysInfoInd

8.395.2.16 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSystemSelectionInd

8.396 pack_nas_SLQSNASSeteDRXParams_t Struct Reference

Data Fields

- uint8_t * pEdrxEnable
- uint8_t * pCycleLen
- uint8_t * pPagingTimeWindow
- uint8_t * pEdrxRatType
- uint32_t * pEdrxCiotLteMode

8.396.1 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.
pEdrxCiotLteMode	[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.

8.396.2 Field Documentation

8.396.2.1 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pCycleLen

8.396.2.2 uint32_t* pack_nas_SLQSNASSeteDRXParams_t::pEdrxCiotLteMode

8.396.2.3 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pEdrxEnable

8.396.2.4 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pEdrxRatType

8.396.2.5 uint8_t* pack_nas_SLQSNASSeteDRXParams_t::pPagingTimeWindow

8.397 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

8.397.1 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

8.397.2 Field Documentation

8.397.2.1 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsDl

8.397.2.2 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsUl

8.397.2.3 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmDl

8.397.2.4 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmUl

8.397.2.5 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmDl

8.397.2.6 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmUI

8.397.2.7 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pRankIndicatorInd

8.397.2.8 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pTimer

8.398 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN * pWcdmaUARFCN
- nas_lteEARFCN * pLteEARFCN
- nas_ltePCI * pLtePCI

8.398.1 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

8.398.2 Field Documentation

8.398.2.1 nas_lteEARFCN* pack_nas_SLQSNASSwiSetChannelLock_t::pLteEARFCN

8.398.2.2 nas_ltePCI* pack_nas_SLQSNASSwiSetChannelLock_t::pLtePCI

8.398.2.3 nas_wcdmaUARFCN* pack_nas_SLQSNASSwiSetChannelLock_t::pWcdmaUARFCN

8.399 pack_nas_SLQSSetBandPreference_t Struct Reference

Data Fields

- uint64_t bandPref

8.399.1 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

8.399.2 Field Documentation

8.399.2.1 uint64_t pack_nas_SLQSSetBandPreference_t::bandPref

8.400 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference

Data Fields

- uint8_t bEnable
- nas_SLQSSignalStrengthsIndReq * pSigIndReq

8.400.1 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

8.400.2 Field Documentation

8.400.2.1 uint8_t pack_nas_SLQSSetSignalStrengthsCallback_t::bEnable

8.400.2.2 nas_SLQSSignalStrengthsIndReq* pack_nas_SLQSSetSignalStrengthsCallback_t::pSigIndReq

8.401 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

8.401.1 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

8.401.2 Field Documentation

8.401.2.1 struct nas_acqOrderPref* pack_nas_SLQSSetSysSelectionPref_t::pAcqOrderPref

8.401.2.2 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pBandPref

8.401.2.3 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pChgDuration

8.401.2.4 struct nas_CSGID* pack_nas_SLQSSetSysSelectionPref_t::pCSGID

8.401.2.5 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pEmerMode

8.401.2.6 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pGWAcqOrderPref

8.401.2.7 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pLTEBandPref

8.401.2.8 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pMNCIncPCSDigStat

8.401.2.9 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pModePref

8.401.2.10 struct nas_netSelectionPref* pack_nas_SLQSSetSysSelectionPref_t::pNetSelPref

8.401.2.11 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pPRLPref

8.401.2.12 unsigned char* pack_nas_SLQSSetSysSelectionPref_t::pRAT

8.401.2.13 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pRoamPref

8.401.2.14 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvDomainPref

8.401.2.15 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvRegRestriction

8.401.2.16 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pTdscdmaBandPref

8.402 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

8.402.1 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

8.402.2 Field Documentation

8.402.2.1 struct nas_acqOrderPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pAcqOrderPref

8.402.2.2 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pBandPref

8.402.2.3 uint8_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pChgDuration

8.402.2.4 struct nas_ciotAcqOrderPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pCiotAcqOrderPref

8.402.2.5 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pCiotLteOpMode

8.402.2.6 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pCiotLteOpModePref

8.402.2.7 struct nas_CSGID* pack_nas_SLQSSetSysSelectionPrefExt_t::pCSGID

8.402.2.8 uint8_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pEmerMode

8.402.2.9 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pGWAcqOrderPref

8.402.2.10 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pLTEBandPref

8.402.2.11 struct nas_lteBandPrefExt* pack_nas_SLQSSetSysSelectionPrefExt_t::pLTEBandPrefExt

8.402.2.12 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pLteM1BandPref

8.402.2.13 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pLteNb1BandPref

8.402.2.14 uint8_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pMNCIncPCSDigStat

8.402.2.15 uint16_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pModePref

8.402.2.16 struct nas_netSelectionPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pNetSelPref

8.402.2.17 struct nas_nr5gBandPref* pack_nas_SLQSSetSysSelectionPrefExt_t::pNr5gBandPref

8.402.2.18 uint16_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pPRLPref

8.402.2.19 unsigned char* pack_nas_SLQSSetSysSelectionPrefExt_t::pRAT

8.402.2.20 uint16_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pRoamPref

8.402.2.21 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pSrvDomainPref

8.402.2.22 uint32_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pSrvRegRestriction

8.402.2.23 uint64_t* pack_nas_SLQSSetSysSelectionPrefExt_t::pTdscdmaBandPref

8.403 pack_nas_SLQSSwiPSDetach_t Struct Reference

Data Fields

- uint8_t * pDetachAction

8.403.1 Detailed Description

Structure for PS connection detach pack.

Parameters

pDetachAction[IN]	• Values – 2- Initiates an immediate packet domain detach.

8.403.2 Field Documentation

8.403.2.1 uint8_t* pack_nas_SLQSSwiPSDetach_t::pDetachAction

8.404 pack_pds_PDSInjectTimeReference_t Struct Reference

Data Fields

- uint64_t systemTime
- uint16_t systemDiscontinuities

8.404.1 Detailed Description

Structure contain the parameter for PDS Inject time.

Parameters

systemTime	• System time(milliseconds)
system-Discontinuities	• Number of system time discontinuities

8.404.2 Field Documentation

8.404.2.1 uint16_t pack_pds_PDSInjectTimeReference_t::systemDiscontinuities

8.404.2.2 uint64_t pack_pds_PDSInjectTimeReference_t::systemTime

8.405 pack_pds_ResetPDSData_t Struct Reference

Data Fields

- uint32_t * pGPSDataMask
- uint32_t * pCellDataMask

8.405.1 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

8.405.2 Field Documentation

8.405.2.1 uint32_t* pack_pds_ResetPDSData_t::pCellDataMask

8.405.2.2 uint32_t* pack_pds_ResetPDSData_t::pGPSDataMask

8.406 pack_pds_SetEventReportCallback_t Struct Reference

Data Fields

- uint8_t posDataNmea
- uint8_t rptPosData

8.406.1 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

8.406.2 Field Documentation

8.406.2.1 uint8_t pack_pds_SetEventReportCallback_t::posDataNmea

8.406.2.2 uint8_t pack_pds_SetEventReportCallback_t::rptPosData

8.407 pack_pds_SetPDSDefaults_t Struct Reference

Data Fields

- uint32_t operation
- uint8_t timeout
- uint32_t interval
- uint32_t accuracy

8.407.1 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)

8.407.2 Field Documentation

8.407.2.1 uint32_t pack_pds_SetPDSDefaults_t::accuracy

8.407.2.2 uint32_t pack_pds_SetPDSDefaults_t::interval

8.407.2.3 uint32_t pack_pds_SetPDSDefaults_t::operation

8.407.2.4 uint8_t pack_pds_SetPDSDefaults_t::timeout

8.408 pack_pds_SetPDSState_t Struct Reference

Data Fields

- uint32_t enable

8.408.1 Detailed Description

Structure contain PDS state parameters.

Parameters

enable	• Desired PDS state – Zero - disable – Non-Zero - enable

8.408.2 Field Documentation

8.408.2.1 uint32_t pack_pds_SetPDSState_t::enable

8.409 pack_pds_SetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t bAuto

8.409.1 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

8.409.2 Field Documentation

8.409.2.1 uint32_t pack_pds_SetPortAutomaticTracking_t::bAuto

8.410 pack_pds_SetServiceAutomaticTracking_t Struct Reference

Data Fields

- uint32_t bAuto

8.410.1 Detailed Description

Structure contain the parameter to Sets the automatic tracking state.

Parameters

bAuto	• Automatic tracking session started for service – 0x00 - Disabled – 0x01 - Enabled

8.410.2 Field Documentation

8.410.2.1 uint32_t pack_pds_SetServiceAutomaticTracking_t::bAuto

8.411 pack_pds_SetXTRAAutomaticDownload_t Struct Reference

Data Fields

- uint32_t bEnabled
- uint16_t interval

8.411.1 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

8.411.2 Field Documentation

8.411.2.1 uint32_t pack_pds_SetXTRAAutomaticDownload_t::bEnabled

8.411.2.2 uint16_t pack_pds_SetXTRAAutomaticDownload_t::interval

8.412 pack_pds_SetXTRANetwork_t Struct Reference

Data Fields

- uint32_t preference

8.412.1 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

8.412.2 Field Documentation

8.412.2.1 uint32_t pack_pds_SetXTRANetwork_t::preference

8.413 pack_pds_SLQSGetAGPSConfig_t Struct Reference

Data Fields

- uint8_t * pNetworkMode

8.413.1 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

8.413.2 Field Documentation

8.413.2.1 uint8_t* pack_pds_SLQSGetAGPSConfig_t::pNetworkMode

8.414 pack_pds_SLQSPDSInjectAbsoluteTimeReference_t Struct Reference

Data Fields

- uint64_t timeMsec
- uint32_t timeUncMsec
- uint8_t timeBase
- uint8_t forceFlag

8.414.1 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

8.414.2 Field Documentation

8.414.2.1 uint8_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::forceFlag

8.414.2.2 uint8_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeBase

8.414.2.3 uint64_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeMsec

8.414.2.4 uint32_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeUncMsec

8.415 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

8.415.1 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

8.415.2 Field Documentation

8.415.2.1 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pAltitudeWrtEllipsoid

8.415.2.2 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pAltitudeWrtSealevel

8.415.2.3 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pHorizontalConfidence

8.415.2.4 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pHorizontalUncCircular

8.415.2.5 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pLatitude

8.415.2.6 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pLongitude

8.415.2.7 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pPositionSource

8.415.2.8 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pTimeStamp

8.415.2.9 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pTimeType

8.415.2.10 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pVerticalConfidence

8.415.2.11 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pVerticalUnc

8.416 pack_pds_SLQSSetAGPSConfig_t Struct Reference

Data Fields

- uint32_t * pServerAddress
- uint32_t * pServerPort
- uint8_t * pServerURL
- uint8_t * pServerURLLength
- uint8_t * pNetworkMode

8.416.1 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

8.416.2 Field Documentation

8.416.2.1 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pNetworkMode

8.416.2.2 uint32_t* pack_pds_SLQSSetAGPSConfig_t::pServerAddress

8.416.2.3 uint32_t* pack_pds_SLQSSetAGPSConfig_t::pServerPort

8.416.2.4 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pServerURL

8.416.2.5 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pServerURLLength

8.417 pack_pds_SLQSSetPositionMethodState_t Struct Reference

Data Fields

- uint8_t * pXtraTimeState

- uint8_t * pXtraDataState
- uint8_t * pWifiState

8.417.1 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

8.417.2 Field Documentation

8.417.2.1 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pWifiState

8.417.2.2 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pXtraDataState

8.417.2.3 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pXtraTimeState

8.418 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

8.418.1 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)

8.418.2 Field Documentation

8.418.2.1 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixAccuracy

8.418.2.2 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixCount

8.418.2.3 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixInterval

8.418.2.4 uint8_t pack_pds_StartPDSTrackingSessionExt_t::fixTimeout

8.418.2.5 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionControl

8.418.2.6 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionOperation

8.418.2.7 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionServerOption

8.418.2.8 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionType

8.419 pack_qmi_t Struct Reference

Data Fields

- uint16_t xid
- int timeout
- uint16_t msgid
- uint8_t svc

8.419.1 Detailed Description

qmi request context

Parameters

in	*xid*	transaction id
out	*timeout*	timeout recommended in seconds
out	*msgid*	message id
out	*svc*	qmi service

8.419.2 Field Documentation

8.419.2.1 uint16_t pack_qmi_t::msgid

8.419.2.2 uint8_t pack_qmi_t::svc

8.419.2.3 int pack_qmi_t::timeout

8.419.2.4 uint16_t pack_qmi_t::xid

8.420 pack_qos_BindDataPort_t Struct Reference

Data Fields

- qos_BindDataPortPeripheralEndPointID_t * pPeripheralEndPointID
- qos_BindDataPortMuxID_t * pMuxID
- qos_BindDataPortSIODataPort_t * pSIODataPort

8.420.1 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.

8.420.2 Field Documentation

8.420.2.1 qos_BindDataPortMuxID_t* pack_qos_BindDataPort_t::pMuxID

8.420.2.2 qos_BindDataPortPeripheralEndPointID_t* pack_qos_BindDataPort_t::pPeripheralEndPointID

8.420.2.3 qos_BindDataPortSIODataPort_t* pack_qos_BindDataPort_t::pSIODataPort

8.421 pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference

Data Fields

- uint32_t apnId

8.421.1 Detailed Description

Structure that contains the APN ID to obtain extra APN parameters

Parameters

apnId[IN]	• APN id

8.421.2 Field Documentation

8.421.2.1 uint32_t pack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.422 pack_qos_SLQSQosSwiReadDataStats_t Struct Reference

Data Fields

- uint32_t apnId

8.422.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

apnId[IN]	• APN id

8.422.2 Field Documentation

8.422.2.1 uint32_t pack_qos_SLQSQosSwiReadDataStats_t::apnId

8.423 pack_qos_SLQSSetQosEventCallback_t Struct Reference

Data Fields

- uint8_t enable

8.423.1 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

8.423.2 Field Documentation

8.423.2.1 uint8_t pack_qos_SLQSSetQosEventCallback_t::enable

8.424 pack_rms_SetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask

8.424.1 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)

8.424.2 Field Documentation

8.424.2.1 uint32_t pack_rms_SetSMSWake_t::enabled

8.424.2.2 uint32_t pack_rms_SetSMSWake_t::wake_mask

8.425 pack_sar_SLQSSetRfSarState_t Struct Reference

Data Fields

- uint32_t RfSarState

8.425.1 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

8.425.2 Field Documentation

8.425.2.1 uint32_t pack_sar_SLQSSetRfSarState_t::RfSarState

8.426 pack_sms_SaveSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage

8.426.1 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

8.426.2 Field Documentation

8.426.2.1 uint32_t pack_sms_SaveSMS_t::messageFormat

8.426.2.2 uint32_t pack_sms_SaveSMS_t::messageSize

8.426.2.3 uint8_t* pack_sms_SaveSMS_t::pMessage

8.426.2.4 uint32_t pack_sms_SaveSMS_t::storageType

8.427 pack_sms_SendSMS_t Struct Reference

Data Fields

- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage
- uint8_t * pLinktimer

8.427.1 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

8.427.2 Field Documentation

8.427.2.1 uint32_t pack_sms_SendSMS_t::messageFormat

8.427.2.2 uint32_t pack_sms_SendSMS_t::messageSize

8.427.2.3 uint8_t* pack_sms_SendSMS_t::pLinktimer

8.427.2.4 uint8_t* pack_sms_SendSMS_t::pMessage

8.428 pack_sms_SetNewSMSCallback_t Struct Reference

Data Fields

- enum eqmiCbkSetStatus status

8.428.1 Detailed Description

This strucure contains pack set new SMS callback parameters.

Parameters

status	callback parameter

8.428.2 Field Documentation

8.428.2.1 enum eqmiCbkSetStatus pack_sms_SetNewSMSCallback_t::status

8.429 pack_sms_SetSMSCAddress_t Struct Reference

Data Fields

- uint8_t * pSMSCAddress
- uint8_t * pSMSCType

8.429.1 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).

8.429.2 Field Documentation

8.429.2.1 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCAddress

8.429.2.2 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCType

8.430 pack_sms_SLQSDeleteSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t * pMessageIndex
- uint32_t * pMessageTag
- uint8_t * pMessageMode

8.430.1 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)

8.430.2 Field Documentation

8.430.2.1 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageIndex

8.430.2.2 uint8_t* pack_sms_SLQSDeleteSMS_t::pMessageMode

8.430.2.3 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageTag

8.430.2.4 uint32_t pack_sms_SLQSDeleteSMS_t::storageType

8.431 pack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex
- uint8_t * pMessageMode

8.431.1 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)

8.431.2 Field Documentation

8.431.2.1 uint32_t pack_sms_SLQSGetSMS_t::messageIndex

8.431.2.2 uint8_t* pack_sms_SLQSGetSMS_t::pMessageMode

8.431.2.3 uint32_t pack_sms_SLQSGetSMS_t::storageType

8.432 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode

8.432.1 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)

8.432.2 Field Documentation

8.432.2.1 uint8_t pack_sms_SLQSGetSmsBroadcastConfig_t::mode

8.433 pack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t * pRequestedTag
- uint8_t * pMessageMode

8.433.1 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)

8.433.2 Field Documentation

8.433.2.1 uint8_t* pack_sms_SLQSGetSMSList_t::pMessageMode

8.433.2.2 uint32_t* pack_sms_SLQSGetSMSList_t::pRequestedTag

8.433.2.3 uint32_t pack_sms_SLQSGetSMSList_t::storageType

8.434 pack_sms_SLQSModifySMSStatus_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex
- uint32_t messageTag
- uint8_t * pMessageMode

8.434.1 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)

8.434.2 Field Documentation

8.434.2.1 uint32_t pack_sms_SLQSModifySMSStatus_t::messageIndex

8.434.2.2 uint32_t pack_sms_SLQSModifySMSStatus_t::messageTag

8.434.2.3 uint8_t* pack_sms_SLQSModifySMSStatus_t::pMessageMode

8.434.2.4 uint32_t pack_sms_SLQSModifySMSStatus_t::storageType

8.435 pack_sms_SLQSSendAsyncSMS_t Struct Reference

Data Fields

- sms_sendAsyncsmsParams * pSendSmsParams

8.435.1 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

8.435.2 Field Documentation

8.435.2.1 sms_sendAsyncsmsParams* pack_sms_SLQSSendAsyncSMS_t::pSendSmsParams

8.436 pack_sms_SLQSSetIndicationRegister_t Struct Reference

Data Fields

- sms_setIndicationReg * pSetIndicationRegReq

8.436.1 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

8.436.2 Field Documentation

8.436.2.1 sms_setIndicationReg* pack_sms_SLQSSetIndicationRegister_t::pSetIndicationRegReq

8.437 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference

Data Fields

- uint8_t mode
- uint8_t broadcastActivate

8.437.1 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

8.437.2 Field Documentation

8.437.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::broadcastActivate

8.437.2.2 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::mode

8.438 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode
- sms_qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig
- sms_qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig

8.438.1 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).

8.438.2 Field Documentation

8.438.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastConfig_t::mode

8.438.2.2 sms_qaQmi3GPPBroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pBroadcastConfig

8.438.2.3 sms_qaQmi3GPP2BroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pCDMABroadcastConfig

8.439 pack_sms_SLQSSetSmsStorage_t Struct Reference

Data Fields

- uint8_t smsStorage

8.439.1 Detailed Description

Structure contain Parameter to set the SMS Storage on the device

Parameters

smsStorage	• SMS Storage – 0x01 - device's permanent memory – 0x02 - UICC

8.439.2 Field Documentation

8.439.2.1 uint8_t pack_sms_SLQSSetSmsStorage_t::smsStorage

8.440 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeReq * pMaxStorageSizeReq

8.440.1 Detailed Description

This structure contains get maximum storage size.

Parameters

pMaxStorage-SizeReq	• Request parameters for SmsSLQSGetMaxStorageSize – See sms_maxStorageSizeReq for more information

8.440.2 Field Documentation

8.440.2.1 sms_maxStorageSizeReq* pack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeReq

8.441 pack_sms_SLQSSmsSetRoutes_t Struct Reference

Data Fields

- sms_setRoutesReq * pSetRoutesReq

8.441.1 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

8.441.2 Field Documentation

8.441.2.1 sms_setRoutesReq* pack_sms_SLQSSmsSetRoutes_t::pSetRoutesReq

8.442 pack_swiaudio_SLQSGetM2MAudioProfile_t Struct Reference

Data Fields

- uint8_t * pGenerator

8.442.1 Detailed Description

This structure contains the SLQSGetM2MAudioProfile pack parameters.

Parameters

pGenerator[optional]	• Generator – 0 - Voice

8.442.2 Field Documentation

8.442.2.1 uint8_t* pack_swiaudio_SLQSGetM2MAudioProfile_t::pGenerator

8.443 pack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator

8.443.1 Detailed Description

This structure contains the SLQSGetM2MAudioVolume pack parameters.

Parameters

Profile	• Audio Profile – 0-5
Generator	• Generator – 0 - Voice

8.443.2 Field Documentation

8.443.2.1 uint8_t pack_swiaudio_SLQSGetM2MAudioVolume_t::Generator

8.443.2.2 uint8_t pack_swiaudio_SLQSGetM2MAudioVolume_t::Profile

8.444 pack_swiaudio_SLQSGetM2MAVMute_t Struct Reference

Data Fields

- uint8_t Profile

8.444.1 Detailed Description

This structure contains the SLQSGetM2MAVMute pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5

8.444.2 Field Documentation

8.444.2.1 uint8_t pack_swiaudio_SLQSGetM2MAVMute_t::Profile

8.445 pack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference

Data Fields

- uint8_t Profile

8.445.1 Detailed Description

This structure contains the SLQSGetM2MSpkrGain pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5

8.445.2 Field Documentation

8.445.2.1 uint8_t pack_swiaudio_SLQSGetM2MSpkrGain_t::Profile

8.446 pack_swiaudio_SLQSSetM2MAudioAVCFG_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Device
- uint8_t PIFACEId
- swiaudio_PCMparams * pPCMParams

8.446.1 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

8.446.2 Field Documentation

8.446.2.1 uint8_t pack_swiaudio_SLQSSetM2MAudioAVCFG_t::Device

8.446.2.2 uint8_t pack_swiaudio_SLQSSetM2MAudioAVCFG_t::PIFACEId

8.446.2.3 swiaudio_PCMparams* pack_swiaudio_SLQSSetM2MAudioAVCFG_t::pPCMParams

8.446.2.4 uint8_t pack_swiaudio_SLQSSetM2MAudioAVCFG_t::Profile

8.447 pack_swiaudio_SLQSSetM2MAudioLPBK_t Struct Reference

Data Fields

- uint8_t Enable

8.447.1 Detailed Description

This structure contains the SLQSSetM2MAudioLPBK pack parameters.

Parameters

Enable	• Operation to be performed – 0 - stop – 1 - VOCODER loop – 2 - internal codec loop

8.447.2 Field Documentation

8.447.2.1 uint8_t pack_swiaudio_SLQSSetM2MAudioLPBK_t::Enable

8.448 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

8.448.1 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

8.448.2 Field Documentation

8.448.2.1 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pCwtMute

8.448.2.2 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pEarMute

8.448.2.3 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pGenerator

8.448.2.4 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pMicMute

8.448.2.5 uint8_t pack_swiaudio_SLQSSetM2MAudioProfile_t::Profile

8.448.2.6 uint8_t* pack_swiaudio_SLQSSetM2MAudioProfile_t::pVolume

8.449 pack_swiaudio_SLQSSetM2MAudioVolume_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t Generator
- uint8_t Level

8.449.1 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

8.449.2 Field Documentation

8.449.2.1 uint8_t pack_swiaudio_SLQSSetM2MAudioVolume_t::Generator

8.449.2.2 uint8_t pack_swiaudio_SLQSSetM2MAudioVolume_t::Level

8.449.2.3 uint8_t pack_swiaudio_SLQSSetM2MAudioVolume_t::Profile

8.450 pack_swiaudio_SLQSSetM2MAVMute_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t EarMute
- uint8_t MicMute
- uint8_t * pCwtMute

8.450.1 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

8.450.2 Field Documentation

8.450.2.1 uint8_t pack_swiaudio_SLQSSetM2MAVMute_t::EarMute

8.450.2.2 uint8_t pack_swiaudio_SLQSSetM2MAVMute_t::MicMute

8.450.2.3 uint8_t* pack_swiaudio_SLQSSetM2MAVMute_t::pCwtMute

8.450.2.4 uint8_t pack_swiaudio_SLQSSetM2MAVMute_t::Profile

8.451 pack_swiaudio_SLQSSetM2MSpkrGain_t Struct Reference

Data Fields

- uint8_t Profile
- uint16_t Value

8.451.1 Detailed Description

This structure contains the SLQSSetM2MSpkrGain pack parameters.

Parameters

Profile	• Audio Profile Number – 0-5
Value	• RX speakerphone gain – 0x0 - 0x7fff

8.451.2 Field Documentation

8.451.2.1 uint8_t pack_swiaudio_SLQSSetM2MSpkrGain_t::Profile

8.451.2.2 uint16_t pack_swiaudio_SLQSSetM2MSpkrGain_t::Value

8.452 pack_swiavms_SLQSAVMSSendSelection_t Struct Reference

Data Fields

- uint8_t selection
- uint32_t * pDeferTime
- uint8_t * pClientPerformOperationFlag
- uint8_t * pPackageID
- uint16_t * pRejectReason

8.452.1 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.

8.452.2 Field Documentation

8.452.2.1 uint8_t* pack_swiavms_SLQSAVMSSendSelection_t::pClientPerformOperationFlag

8.452.2.2 uint32_t* pack_swiavms_SLQSAVMSSendSelection_t::pDeferTime

8.452.2.3 uint8_t* pack_swiavms_SLQSAVMSSendSelection_t::pPackageID

8.452.2.4 uint16_t* pack_swiavms_SLQSAVMSSendSelection_t::pRejectReason

8.452.2.5 uint8_t pack_swiavms_SLQSAVMSSendSelection_t::selection

8.453 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

8.453.1 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

8.453.2 Field Documentation

8.453.2.1 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::AutoConnect

8.453.2.2 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::AutoReboot

8.453.2.3 PackSwiAvmsSetSettingsAPNInfo* pack_swiavms_SLQSAVMSSetSettings_t::pAPNInfo

8.453.2.4 PackSwiAvmsSetSettingsConnectionRetryTimers* pack_swiavms_SLQSAVMSSetSettings_t::pConnectionRetryTimers

8.453.2.5 uint8_t* pack_swiavms_SLQSAVMSSetSettings_t::pFwAutoSDM

8.453.2.6 uint8_t* pack_swiavms_SLQSAVMSSetSettings_t::pNotifStore

8.453.2.7 PackSwiAvmsSetSettingsPeriodInfo* pack_swiavms_SLQSAVMSSetSettings_t::pPeriodInfo

8.453.2.8 uint32_t* pack_swiavms_SLQSAVMSSetSettings_t::pPollingTimer

8.453.2.9 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::PromptFwDownload

8.453.2.10 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::PromptFwUpdate

8.454 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

8.454.1 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.

8.454.2 Field Documentation

8.454.2.1 uint8_t pack_swiavms_SLQSAVMSSetSettings_v2_t::AutoConnect

8.454.2.2 PackSwiAvmsSetSettingsAPNInfo* pack_swiavms_SLQSAVMSSetSettings_v2_t::pAPNInfo

8.454.2.3 uint8_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pAutoReboot

8.454.2.4 PackSwiAvmsSetSettingsConnectionRetryTimers* pack_swiavms_SLQSAVMSSetSettings_v2_t::pConnectionRetryTimers

8.454.2.5 uint8_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pFwAutoSDM

8.454.2.6 uint8_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pNotifStore

8.454.2.7 PackSwiAvmsSetSettingsPeriodInfo* pack_swiavms_SLQSAVMSSetSettings_v2_t::pPeriodInfo

8.454.2.8 uint32_t* pack_swiavms_SLQSAVMSSetSettings_v2_t::pPollingTimer

8.454.2.9 uint8_t pack_swiavms_SLQSAVMSSetSettings_v2_t::PromptFwDownload

8.454.2.10 uint8_t pack_swiavms_SLQSAVMSSetSettings_v2_t::PromptFwUpdate

8.455 pack_swiavms_SLQSAVMSStartSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.455.1 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.

8.455.2 Field Documentation

8.455.2.1 uint8_t pack_swiavms_SLQSAVMSStartSession_t::sessionType

8.456 pack_swiavms_SLQSAVMSStopSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.456.1 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

8.456.2 Field Documentation

8.456.2.1 uint8_t pack_swiavms_SLQSAVMSStopSession_t::sessionType

8.457 pack_swidms_SLQSSwiDmsSetMTU_t Struct Reference

Data Fields

- uint16_t MTUSize

8.457.1 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.

8.457.2 Field Documentation

8.457.2.1 uint16_t pack_swidms_SLQSSwiDmsSetMTU_t::MTUSize

8.458 pack_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference

Data Fields

- uint32_t CfgValue

8.458.1 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

8.458.2 Field Documentation

8.458.2.1 uint32_t pack_swidms_SLQSSwiDmsSetUsbComp_t::CfgValue

8.459 pack_swidms_SLQSSwiDmsSetUsbNetNum_t Struct Reference

Data Fields

- uint8_t nUsbNetNum

8.459.1 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

8.459.2 Field Documentation

8.459.2.1 uint8_t pack_swidms_SLQSSwiDmsSetUsbNetNum_t::nUsbNetNum

8.460 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

8.460.1 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

8.460.2 Field Documentation

8.460.2.1 uint32_t pack_swiloc_SwiLocSetAutoStart_t::fix_rate

8.460.2.2 uint8_t pack_swiloc_SwiLocSetAutoStart_t::fix_type

8.460.2.3 uint8_t pack_swiloc_SwiLocSetAutoStart_t::function

8.460.2.4 uint32_t pack_swiloc_SwiLocSetAutoStart_t::max_dist

8.460.2.5 uint8_t pack_swiloc_SwiLocSetAutoStart_t::max_time

8.460.2.6 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_rate

8.460.2.7 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_type

8.460.2.8 int pack_swiloc_SwiLocSetAutoStart_t::set_function

8.460.2.9 int pack_swiloc_SwiLocSetAutoStart_t::set_max_dist

8.460.2.10 int pack_swiloc_SwiLocSetAutoStart_t::set_max_time

8.461 pack_swioma_SLQSOMADMCancelSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.461.1 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

8.461.2 Field Documentation

8.461.2.1 uint32_t pack_swioma_SLQSOMADMCancelSession_t::sessionType

8.462 pack_swioma_SLQSOMADMCancelSessionExt_t Struct Reference

Data Fields

- uint32_t sessionType

8.462.1 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

8.462.2 Field Documentation

8.462.2.1 uint32_t pack_swioma_SLQSOMADMCancelSessionExt_t::sessionType

8.463 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference

Data Fields

- uint32_t SessionType

8.463.1 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

8.463.2 Field Documentation

8.463.2.1 uint32_t pack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.464 pack_swioma_SLQSOMADMSendSelection_t Struct Reference

Data Fields

- uint32_t selection
- uint32_t * pDeferTime
- uint32_t * pRejectReason

8.464.1 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.

8.464.2 Field Documentation

8.464.2.1 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pDeferTime

8.464.2.2 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pRejectReason

8.464.2.3 uint32_t pack_swioma_SLQSOMADMSendSelection_t::selection

8.465 pack_swioma_SLQSOMADMSendSelectionExt_t Struct Reference

Data Fields

- uint32_t selection

8.465.1 Detailed Description

Structure containing the OMA DM selection

Parameters

selection	[IN] • OMA-DM NIA Selection – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer

8.465.2 Field Documentation

8.465.2.1 uint32_t pack_swioma_SLQSOMADMSendSelectionExt_t::selection

8.466 pack_swioma_SLQSOMADMSetSettings_t Struct Reference

Data Fields

- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t * pAutosdm
- uint8_t * pFwAutoCheck

8.466.1 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

8.466.2 Field Documentation

8.466.2.1 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAdownload

8.466.2.2 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAUpdate

8.466.2.3 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pAutosdm

8.466.2.4 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pFwAutoCheck

8.467 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

8.467.1 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

8.467.2 Field Documentation

8.467.2.1 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::FOTAdownload

8.467.2.2 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::FOTAUpdate

8.467.2.3 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::FUMOEnable

8.467.2.4 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::OMADMEnable

8.467.2.5 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::OMADMLogEnable

8.467.2.6 uint8_t pack_swioma_SLQSOMADMSetSettingsExt_t::PRLEnable

8.468 pack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.468.1 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

8.468.2 Field Documentation

8.468.2.1 uint32_t pack_swioma_SLQSOMADMStartSession_t::sessionType

8.469 pack_swioma_SLQSOMADMStartSessionExt_t Struct Reference

Data Fields

- uint32_t sessionType

8.469.1 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

8.469.2 Field Documentation

8.469.2.1 uint32_t pack_swioma_SLQSOMADMStartSessionExt_t::sessionType

8.470 pack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference

Data Fields

- uint8_t mitigationDevIDLen
- char mitigationDevID [255]

8.470.1 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

8.470.2 Field Documentation

8.470.2.1 char pack_tmd_SLQSTmdDeRegNotMitigationLvl_t::mitigationDevID[255]

8.470.2.2 uint8_t pack_tmd_SLQSTmdDeRegNotMitigationLvl_t::mitigationDevIDLen

8.471 pack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference

Data Fields

- uint8_t mitigationDevIDLen
- char mitigationDevID [255]

8.471.1 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

8.471.2 Field Documentation

8.471.2.1 char pack_tmd_SLQSTmdGetMitigationLvl_t::mitigationDevID[255]

8.471.2.2 uint8_t pack_tmd_SLQSTmdGetMitigationLvl_t::mitigationDevIDLen

8.472 pack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference

Data Fields

- uint8_t mitigationDevIDLen
- char mitigationDevID [255]

8.472.1 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

8.472.2 Field Documentation

8.472.2.1 char pack_tmd_SLQSTmdRegNotMitigationLvl_t::mitigationDevID[255]

8.472.2.2 uint8_t pack_tmd_SLQSTmdRegNotMitigationLvl_t::mitigationDevIDLen

8.473 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

8.473.1 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.

8.473.2 Field Documentation

8.473.2.1 uim_changeUIMPIN pack_uim_ChangePin_t::changePIN

8.473.2.2 uim_encryptedPIN1 pack_uim_ChangePin_t::EncryptedPIN1

8.473.2.3 uint32_t* pack_uim_ChangePin_t::pIndicationToken

8.473.2.4 uint8_t* pack_uim_ChangePin_t::pKeyReferenceID

8.473.2.5 uim_sessionInformation pack_uim_ChangePin_t::sessionInfo

8.473.2.6 uint16_t pack_uim_ChangePin_t::TlvResult

8.474 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

8.474.1 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.

8.474.2 Field Documentation

8.474.2.1 uim_fileInfo pack_uim_ReadTransparent_t::fileIndex

8.474.2.2 uint8_t* pack_uim_ReadTransparent_t::pEncryptData

8.474.2.3 uint32_t* pack_uim_ReadTransparent_t::pIndicationToken

8.474.2.4 uim_readTransparentInfo pack_uim_ReadTransparent_t::readTransparent

8.474.2.5 uim_sessionInformation pack_uim_ReadTransparent_t::sessionInfo

8.474.2.6 uint16_t pack_uim_ReadTransparent_t::TlvResult

8.475 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

8.475.1 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.

8.475.2 Field Documentation

8.475.2.1 uim_encryptedPIN1 pack_uim_SetPinProtection_t::EncryptedPIN1

8.475.2.2 uint32_t* pack_uim_SetPinProtection_t::pIndicationToken

8.475.2.3 uim_setPINProtection pack_uim_SetPinProtection_t::pinProtection

8.475.2.4 uint8_t* pack_uim_SetPinProtection_t::pKeyReferenceID

8.475.2.5 uim_sessionInformation pack_uim_SetPinProtection_t::sessionInfo

8.475.2.6 uint16_t pack_uim_SetPinProtection_t::TlvResult

8.476 pack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_authenticationData authData
- uint32_t * pIndicationToken

8.476.1 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.

8.476.2 Field Documentation

8.476.2.1 uim_authenticationData pack_uim_SLQSUIMAuthenticate_t::authData

8.476.2.2 uint32_t* pack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.476.2.3 uim_sessionInformation pack_uim_SLQSUIMAuthenticate_t::sessionInfo

8.477 pack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_depersonalizationInformation depersonilisationInfo

8.477.1 Detailed Description

This structure contains information of the request parameters associated with a Depersonalization.

Parameters

depersonilisation-Info	• See uim_depersonalizationInformation for more information.

8.477.2 Field Documentation

8.477.2.1 uim_depersonalizationInformation pack_uim_SLQSUIMDepersonalization_t::depersonilisationInfo

8.478 pack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask

8.478.1 Detailed Description

This structure contains pack event register parameter.

Parameters

eventMask	- bit 1 - card status • bit 4 - physical slot status

8.478.2 Field Documentation

8.478.2.1 uint32_t pack_uim_SLQSUIMEventRegister_t::eventMask

8.479 pack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint32_t * pConfigurationMask

8.479.1 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.

8.479.2 Field Documentation

8.479.2.1 uint32_t* pack_uim_SLQSUIMGetConfiguration_t::pConfigurationMask

8.480 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_fileInfo fileIndex
- uint32_t * pIndicationToken

8.480.1 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.

8.480.2 Field Documentation

8.480.2.1 uim_fileInfo pack_uim_SLQSUIMGetFileAttributes_t::fileIndex

8.480.2.2 uint32_t* pack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.480.2.3 uim_sessionInformation pack_uim_SLQSUIMGetFileAttributes_t::sessionInfo

8.481 pack_uim_SLQSUIMPowerDown_t Struct Reference

Data Fields

- uint8_t slot

8.481.1 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

8.481.2 Field Documentation

8.481.2.1 uint8_t pack_uim_SLQSUIMPowerDown_t::slot

8.482 pack_uim_SLQSUIMPowerUp_t Struct Reference

Data Fields

- uint8_t slot
- uint8_t * pIgnoreHotSwapSwitch

8.482.1 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
pIgnoreHot-Swap-Switch(optional)	• Hot-swap switch status. – 0 - Checks the hot-swap switch status – 1 - Ignores the hot-swap switch status

8.482.2 Field Documentation

8.482.2.1 uint8_t* pack_uim_SLQSUIMPowerUp_t::pIgnoreHotSwapSwitch

8.482.2.2 uint8_t pack_uim_SLQSUIMPowerUp_t::slot

8.483 pack_uim_SLQSUIMRefreshComplete_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t refreshComplete

8.483.1 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

8.483.2 Field Documentation

8.483.2.1 uint8_t pack_uim_SLQSUIMRefreshComplete_t::refreshComplete

8.483.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshComplete_t::sessionInfo

8.484 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo

8.484.1 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.

8.484.2 Field Documentation

8.484.2.1 uim_sessionInformation pack_uim_SLQSUIMRefreshGetLastEvent_t::sessionInfo

8.485 pack_uim_SLQSUIMRefreshOK_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t OKtoRefresh

8.485.1 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

8.485.2 Field Documentation

8.485.2.1 uint8_t pack_uim_SLQSUIMRefreshOK_t::OKtoRefresh

8.485.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshOK_t::sessionInfo

8.486 pack_uim_SLQSUIMRefreshRegister_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_registerRefresh regRefresh

8.486.1 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

8.486.2 Field Documentation

8.486.2.1 uim_registerRefresh pack_uim_SLQSUIMRefreshRegister_t::regRefresh

8.486.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshRegister_t::sessionInfo

8.487 pack_uim_SLQSUIMSwitchSlot_t Struct Reference

Data Fields

- uint8_t bLogicalSlot
- uint32_t ulPhysicalSlot

8.487.1 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

8.487.2 Field Documentation

8.487.2.1 uint8_t pack_uim_SLQSUIMSwitchSlot_t::bLogicalSlot

8.487.2.2 uint32_t pack_uim_SLQSUIMSwitchSlot_t::ulPhysicalSlot

8.488 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

8.488.1 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.

pKeyReferenceI-D(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.
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

8.488.2 Field Documentation

8.488.2.1 uim_encryptedPIN1 pack_uim_UnblockPin_t::EncryptedPIN1

8.488.2.2 uint32_t* pack_uim_UnblockPin_t::pIndicationToken

8.488.2.3 uim_unblockUIMPIN pack_uim_UnblockPin_t::pinProtection

8.488.2.4 uint8_t* pack_uim_UnblockPin_t::pKeyReferenceID

8.488.2.5 uim_sessionInformation pack_uim_UnblockPin_t::sessionInfo

8.488.2.6 uint16_t pack_uim_UnblockPin_t::Tlvresult

8.489 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

8.489.1 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.

8.489.2 Field Documentation

8.489.2.1 uim_encryptedPIN1∗ pack_uim_VerifyPin_t::pEncryptedPIN1

8.489.2.2 uint32_t∗ pack_uim_VerifyPin_t::pIndicationToken

8.489.2.3 uint8_t∗ pack_uim_VerifyPin_t::pKeyReferenceID

8.489.2.4 uim_sessionInformation pack_uim_VerifyPin_t::sessionInfo

8.489.2.5 uint16_t pack_uim_VerifyPin_t::Tlvresult

8.489.2.6 uim_verifyUIMPIN pack_uim_VerifyPin_t::verifyPIN

8.490 pack_voice_AnswerUSSD_t Struct Reference

Data Fields

- uint8_t ∗ pInfo

8.490.1 Detailed Description

This structure contains pack anser USSD parameter.

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.490.2 Field Documentation

8.490.2.1 uint8_t* pack_voice_AnswerUSSD_t::pInfo

8.491 pack_voice_OriginateUSSD_t Struct Reference

Data Fields

- uint8_t * pInfo

8.491.1 Detailed Description

This structure contains pack orginate USSD parameter.

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.491.2 Field Documentation

8.491.2.1 uint8_t* pack_voice_OriginateUSSD_t::pInfo

8.492 pack_voice_SLQSOriginateUSSD_t Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.492.1 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

8.492.2 Field Documentation

8.492.2.1 uint8_t pack_voice_SLQSOriginateUSSD_t::ussData[182]

8.492.2.2 uint8_t pack_voice_SLQSOriginateUSSD_t::ussDCS

8.492.2.3 uint8_t pack_voice_SLQSOriginateUSSD_t::ussLen

8.493 pack_voice_SLQSVoiceALSSelectLine_t Struct Reference

Data Fields

- uint8_t lineValue

8.493.1 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

8.493.2 Field Documentation

8.493.2.1 uint8_t pack_voice_SLQSVoiceALSSelectLine_t::lineValue

8.494 pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference

Data Fields

- uint8_t switchOption

8.494.1 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

8.494.2 Field Documentation

8.494.2.1 uint8_t pack_voice_SLQSVoiceALSSetLineSwitching_t::switchOption

8.495 pack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.495.1 Detailed Description

Contains the parameters passed for pack voice Answer Call.

Parameters

pCallId	• Unique call identifier for the call that must be answered.

8.495.2 Field Documentation

8.495.2.1 uint8_t* pack_voice_SLQSVoiceAnswerCall_t::pCallId

8.496 pack_voice_SLQSVoiceBindSubscription_t Struct Reference

Data Fields

- uint8_t subsType

8.496.1 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

8.496.2 Field Documentation

8.496.2.1 uint8_t pack_voice_SLQSVoiceBindSubscription_t::subsType

8.497 pack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- voice_burstDTMFInfo BurstDTMFInfo
- voice_DTMFLengths * pBurstDTMFLengths

8.497.1 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.

8.497.2 Field Documentation

8.497.2.1 voice_burstDTMFInfo pack_voice_SLQSVoiceBurstDTMF_t::BurstDTMFInfo

8.497.2.2 voice_DTMFLengths∗ pack_voice_SLQSVoiceBurstDTMF_t::pBurstDTMFLengths

8.498 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

8.498.1 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

8.498.2 Field Documentation

8.498.2.1 uint8_t pack_voice_SLQSVoiceDialCall_t::callNumber[81]

8.498.2.2 voice_calledPartySubAdd* pack_voice_SLQSVoiceDialCall_t::pCallPartySubAdd

8.498.2.3 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCallType

8.498.2.4 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCLIRType

8.498.2.5 voice_CUGInfo* pack_voice_SLQSVoiceDialCall_t::pCUGInfo

8.498.2.6 uint8_t* pack_voice_SLQSVoiceDialCall_t::pEmergencyCategory

8.498.2.7 uint8_t* pack_voice_SLQSVoiceDialCall_t::pSvcType

8.498.2.8 voice_UUSInfo* pack_voice_SLQSVoiceDialCall_t::pUUSInFo

8.499 pack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.499.1 Detailed Description

This structure contain pack voice end call parameter.

Parameters

pCallId	• Unique call identifier for the call that must be ended

8.499.2 Field Documentation

8.499.2.1 uint8_t* pack_voice_SLQSVoiceEndCall_t::pCallId

8.500 pack_voice_SLQSVoiceGetCallBarring_t Struct Reference

Data Fields

- uint8_t reason
- uint8_t * pSvcClass

8.500.1 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

8.500.2 Field Documentation

8.500.2.1 uint8_t* pack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.500.2.2 uint8_t pack_voice_SLQSVoiceGetCallBarring_t::reason

8.501 pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t * pSvcClass

8.501.1 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.

8.501.2 Field Documentation

8.501.2.1 uint8_t* pack_voice_SLQSVoiceGetCallForwardingStatus_t::pSvcClass

8.501.2.2 uint8_t pack_voice_SLQSVoiceGetCallForwardingStatus_t::Reason

8.502 pack_voice_SLQSVoiceGetCallInfo_t Struct Reference

Data Fields

- uint8_t callID

8.502.1 Detailed Description

This structure contains information of the request parameters associated with a call.

Parameters

callID	• Call identifier for the call queried for information.

8.502.2 Field Documentation

8.502.2.1 uint8_t pack_voice_SLQSVoiceGetCallInfo_t::callID

8.503 pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference

Data Fields

- uint8_t * pSvcClass

8.503.1 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

8.503.2 Field Documentation

8.503.2.1 uint8_t* pack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.504 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

8.504.1 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.

8.504.2 Field Documentation

8.504.2.1 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAirTimer

8.504.2.2 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAMRStatus

8.504.2.3 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAutoAnswer

8.504.2.4 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pNamID

8.504.2.5 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoicePrivacy

8.504.2.6 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoiceSO

8.504.2.7 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pRoamTimer

8.504.2.8 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pTTYMode

8.504.2.9 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pVoiceDomainPref

8.505 pack_voice_SLQSVoiceIndicationRegister_t Struct Reference

Data Fields

- uint8_t * pRegDTMFEvents
- uint8_t * pRegVoicePrivacyEvents
- uint8_t * pSuppsNotifEvents

8.505.1 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.

8.505.2 Field Documentation

8.505.2.1 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegDTMFEvents

8.505.2.2 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegVoicePrivacyEvents

8.505.2.3 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pSuppsNotifEvents

8.506 pack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint8_t SUPSType

- uint8_t * pCallID

8.506.1 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.

8.506.2 Field Documentation

8.506.2.1 uint8_t* pack_voice_SLQSVoiceManageCalls_t::pCallID

8.506.2.2 uint8_t pack_voice_SLQSVoiceManageCalls_t::SUPSType

8.507 pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference

Data Fields

- struct voice_USSInfo USSInformation

8.507.1 Detailed Description

This structure contains Orig USSD No Wait Information Parameters.

Parameters

USSInformation	• See voice_USSInfo for more information.

8.507.2 Field Documentation

8.507.2.1 struct voice_USSInfo pack_voice_SLQSVoiceOrigUSSDNoWait_t::USSInformation

8.508 pack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t * pFlashPayLd
- uint8_t * pFlashType

8.508.1 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

8.508.2 Field Documentation

8.508.2.1 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pCallID

8.508.2.2 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashPayLd

8.508.2.3 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashType

8.509 pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t oldPasswd [4]
- uint8_t newPasswd [4]

- uint8_t newPasswdAgain [4]

8.509.1 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.

8.509.2 Field Documentation

8.509.2.1 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswd[4]

8.509.2.2 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswdAgain[4]

8.509.2.3 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::oldPasswd[4]

8.509.2.4 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::Reason

8.510 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

8.510.1 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.

8.510.2 Field Documentation

8.510.2.1 voice_airTimer* pack_voice_SLQSVoiceSetConfig_t::pAirTimerConfig

8.510.2.2 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pAutoAnswer

8.510.2.3 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceDomain

8.510.2.4 voice_prefVoiceSO* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSO

8.510.2.5 voice_roamTimer* pack_voice_SLQSVoiceSetConfig_t::pRoamTimerConfig

8.510.2.6 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pTTYMode

8.511 pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference

Data Fields

- uint8_t privacyPref

8.511.1 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

8.511.2 Field Documentation

8.511.2.1 uint8_t pack_voice_SLQSVoiceSetPreferredPrivacy_t::privacyPref

8.512 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

8.512.1 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

8.512.2 Field Documentation

8.512.2.1 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallBarringPasswd

8.512.2.2 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallForwardingNumber

8.512.2.3 voice_callFwdTypeAndPlan* pack_voice_SLQSVoiceSetSUPSService_t::pCallFwdTypeAndPlan

8.512.2.4 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pServiceClass

8.512.2.5 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pTimerVal

8.512.2.6 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::reason

8.512.2.7 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::voiceSvc

8.513 pack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t DTMFdigit

8.513.1 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.

8.513.2 Field Documentation

8.513.2.1 uint8_t pack_voice_SLQSVoiceStartContDTMF_t::DTMFdigit

8.513.2.2 uint8_t* pack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.514 pack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID

8.514.1 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

8.514.2 Field Documentation

8.514.2.1 uint8_t pack_voice_SLQSVoiceStopContDTMF_t::callID

8.515 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference

Data Fields

- uint8_t * pEnableNotification

8.515.1 Detailed Description

WDS SWI DHCPv4 Client Lease Change Structure

Parameters

pEnable-Notification	• Enable Notification or not

8.515.2 Field Documentation

8.515.2.1 uint8_t* pack_wds_DHCPv4ClientLeaseChange_t::pEnableNotification

8.516 pack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t profiletype

8.516.1 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

8.516.2 Field Documentation

8.516.2.1 uint32_t pack_wds_GetDefaultProfile_t::profiletype

8.517 pack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family

8.517.1 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

8.517.2 Field Documentation

8.517.2.1 uint8_t pack_wds_GetDefaultProfileNum_t::family

8.517.2.2 uint8_t pack_wds_GetDefaultProfileNum_t::type

8.518 pack_wds_GetDormancyState_t Struct Reference

8.518.1 Detailed Description

This structure contains pack get dormancy state information.

Parameters

NULL	• this is a dummy structure

8.519 pack_wds_GetLastMobileIPError_t Struct Reference

8.519.1 Detailed Description

This structure contains pack get last mobile IP error information.

Parameters

NULL	• this is a dummy structure

8.520 pack_wds_GetMobileIP_t Struct Reference

8.520.1 Detailed Description

This structure contains pack get mobile IP information.

Parameters

NULL	• this is a dummy structure

8.521 pack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t index

8.521.1 Detailed Description

This structure contains pack get mobile IP profile information.

Parameters

index	• Mobile IP profile ID

8.521.2 Field Documentation

8.521.2.1 uint8_t pack_wds_GetMobileIPProfile_t::index

8.522 pack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * pStatMask

8.522.1 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

8.522.2 Field Documentation

8.522.2.1 uint32_t* pack_wds_GetPacketStatistics_t::pStatMask

8.523 pack_wds_GetPacketStatus_t Struct Reference

Data Fields

- uint32_t statmask

8.523.1 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

8.523.2 Field Documentation

8.523.2.1 uint32_t pack_wds_GetPacketStatus_t::statmask

8.524 pack_wds_GetSessionDuration_t Struct Reference

8.524.1 Detailed Description

This structure contains pack get session duration information.

Parameters

NULL	• this is a dummy structure

8.525 pack_wds_RMSetTransferStatistics_t Struct Reference

Data Fields

- rmTrasnferStaticsReq RmTrasnferStaticsReq

8.525.1 Detailed Description

This structure contains pack fetch current data system transfer statistics information.

Parameters

RmTrasnfer-StaticsReq[IN]	• See rmTrasnferStaticsReq for more information

8.525.2 Field Documentation

8.525.2.1 rmTrasnferStaticsReq pack_wds_RMSetTransferStatistics_t::RmTrasnferStaticsReq

8.526 pack_wds_SetAutoconnect_t Struct Reference

Data Fields

- uint8_t acsetting
- uint8_t acroamsetting

8.526.1 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

8.526.2 Field Documentation

8.526.2.1 uint8_t pack_wds_SetAutoconnect_t::acroamsetting

8.526.2.2 uint8_t pack_wds_SetAutoconnect_t::acsetting

8.527 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

8.527.1 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)

8.527.2 Field Documentation

8.527.2.1 uint32_t pack_wds_SetDefaultProfile_t::authentication

8.527.2.2 uint32_t pack_wds_SetDefaultProfile_t::ipAddress

8.527.2.3 uint8_t* pack_wds_SetDefaultProfile_t::pApnname

8.527.2.4 uint32_t pack_wds_SetDefaultProfile_t::pdpType

8.527.2.5 uint8_t* pack_wds_SetDefaultProfile_t::pName

8.527.2.6 uint8_t* pack_wds_SetDefaultProfile_t::pPassword

8.527.2.7 uint32_t pack_wds_SetDefaultProfile_t::primaryDNS

8.527.2.8 uint32_t pack_wds_SetDefaultProfile_t::profileType

8.527.2.9 uint8_t* pack_wds_SetDefaultProfile_t::pUsername

8.527.2.10 uint32_t pack_wds_SetDefaultProfile_t::secondaryDNS

8.528 pack_wds_SetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family
- uint8_t index

8.528.1 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.

8.528.2 Field Documentation

8.528.2.1 uint8_t pack_wds_SetDefaultProfileNum_t::family

8.528.2.2 uint8_t pack_wds_SetDefaultProfileNum_t::index

8.528.2.3 uint8_t pack_wds_SetDefaultProfileNum_t::type

8.529 pack_wds_SetMobileIP_t Struct Reference

Data Fields

- uint32_t mode

8.529.1 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

8.529.2 Field Documentation

8.529.2.1 uint32_t pack_wds_SetMobileIP_t::mode

8.530 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

8.530.1 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

8.530.2 Field Documentation

8.530.2.1 uint8_t* pack_wds_SetMobileIPParameters_t::pHA2002bis

8.530.2.2 uint8_t* pack_wds_SetMobileIPParameters_t::pHAAuthenticator

8.530.2.3 uint32_t* pack_wds_SetMobileIPParameters_t::pMode

8.530.2.4 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegPeriod

8.530.2.5 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegTraffic

8.530.2.6 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryInterval

8.530.2.7 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryLimit

8.530.2.8 char* pack_wds_SetMobileIPParameters_t::pSPC

8.531 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

8.531.1 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

8.531.2 Field Documentation

8.531.2.1 uint8_t pack_wds_SetMobileIPProfile_t::index

8.531.2.2 uint32_t* pack_wds_SetMobileIPProfile_t::pAAASPI

8.531.2.3 uint32_t* pack_wds_SetMobileIPProfile_t::pAddress

8.531.2.4 uint8_t* pack_wds_SetMobileIPProfile_t::pEnabled

8.531.2.5 uint32_t* pack_wds_SetMobileIPProfile_t::pHASPI

8.531.2.6 int8_t* pack_wds_SetMobileIPProfile_t::pMNAAA

8.531.2.7 int8_t* pack_wds_SetMobileIPProfile_t::pMNHA

8.531.2.8 int8_t* pack_wds_SetMobileIPProfile_t::pNAI

8.531.2.9 uint32_t* pack_wds_SetMobileIPProfile_t::pPrimaryHA

8.531.2.10 uint8_t* pack_wds_SetMobileIPProfile_t::pRevTunneling

8.531.2.11 uint32_t* pack_wds_SetMobileIPProfile_t::pSecondaryHA

8.531.2.12 int8_t pack_wds_SetMobileIPProfile_t::spc[10]

8.532 pack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo * pCurProfile

8.532.1 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 (EM/MC74xx 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

8.532.2 Field Documentation

8.532.2.1 wds_profileInfo* pack_wds_SLQSCreateProfile_t::pCurProfile

8.532.2.2 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileId

8.532.2.3 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileType

8.533 pack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.533.1 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 (EM/MC74xx onwards)

8.533.2 Field Documentation

8.533.2.1 uint8_t pack_wds_SLQSDeleteProfile_t::profileIndex

8.533.2.2 uint8_t pack_wds_SLQSDeleteProfile_t::profileType

8.534 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

8.534.1 Detailed Description

This structure contains pack get current data system state information.

Parameters

NULL	• this is a dummy structure

8.535 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference

8.535.1 Detailed Description

This structure contains pack get data bearer technology information.

Parameters

NULL	• this is a dummy structure

8.536 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

8.536.1 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
pReportConn-Status	• 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

8.536.2 Field Documentation

8.536.2.1 uint32_t pack_wds_SLQSGetDUNCallInfo_t::Mask

8.536.2.2 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportChannelRate

8.536.2.3 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportConnStatus

8.536.2.4 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDataBearerTech

8.536.2.5 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDormStatus

8.536.2.6 wds_transferStatInd* pack_wds_SLQSGetDUNCallInfo_t::pTransferStatInd

8.537 pack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- uint8_t ProfileId
- uint8_t ProfileType

8.537.1 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 (EM/MC74xx 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

8.537.2 Field Documentation

8.537.2.1 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileId

8.537.2.2 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileType

8.538 pack_wds_SLQSGetRuntimeSettings_t Struct Reference

Data Fields

- uint32_t * pReqSettings

8.538.1 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)

8.538.2 Field Documentation

8.538.2.1 uint32_t* pack_wds_SLQSGetRuntimeSettings_t::pReqSettings

8.539 pack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo curProfile

8.539.1 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 (EM/MC74xx 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

8.539.2 Field Documentation

8.539.2.1 wds_profileInfo pack_wds_SLQSModifyProfile_t::curProfile

8.539.2.2 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileId

8.539.2.3 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileType

8.540 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

8.540.1 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 MC/EM74xx

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 (MC/EM74xx 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 MC/EM74xx 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 MC/EM74xx onwards

8.540.2 Field Documentation

8.540.2.1 uint16_t pack_wds_SLQSSet3GPPConfigItem_t::LTEAttachProfileListLen

8.540.2.2 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::p3gppRelease

8.540.2.3 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::pDefaultPDNEnabled

8.540.2.4 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfile

8.540.2.5 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfileList

8.540.2.6 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pProfileList

8.541 pack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint8_t IPFamilyPreference

8.541.1 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

8.541.2 Field Documentation

8.541.2.1 uint8_t pack_wds_SLQSSetIPFamilyPreference_t::IPFamilyPreference

8.542 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

8.542.1 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

8.542.2 Field Documentation

8.542.2.1 uint8_t pack_wds_SLQSSetWdsEventCallback_t::currentDataBearer

8.542.2.2 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearer

8.542.2.3 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearerTechExt

8.542.2.4 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataSystemStatus

8.542.2.5 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dormancyStatus

8.542.2.6 uint8_t pack_wds_SLQSSetWdsEventCallback_t::interval

8.542.2.7 uint8_t pack_wds_SLQSSetWdsEventCallback_t::mobileIP

8.542.2.8 uint8_t pack_wds_SLQSSetWdsEventCallback_t::transferStats

8.543 pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4ProfileId * pProfileId

8.543.1 Detailed Description

This structure contain get DHCPv4 client configure.

Parameters

pProfileId	pointer to Profile Id structure • See wdsDhcpv4ProfileId

8.543.2 Field Documentation

8.543.2.1 wdsDhcpv4ProfileId* pack_wds_SLQSSGetDHCPv4ClientConfig_t::pProfileId

8.544 pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wds_DHCPv4ProfileId * pProfileId
- wds_DHCPv4HWConfig * pHwConfig
- wds_DHCPv4OptionList * pRequestOptionList

8.544.1 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

8.544.2 Field Documentation

8.544.2.1 wds_DHCPv4HWConfig∗ pack_wds_SLQSSSetDHCPv4ClientConfig_t::pHwConfig

8.544.2.2 wds_DHCPv4ProfileId∗ pack_wds_SLQSSSetDHCPv4ClientConfig_t::pProfileId

8.544.2.3 wds_DHCPv4OptionList∗ pack_wds_SLQSSSetDHCPv4ClientConfig_t::pRequestOptionList

8.545 pack_wds_SLQSSSetLoopback_t Struct Reference

Data Fields

- uint8_t loopbackMode
- uint8_t loopbackMultiplier

8.545.1 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.

8.545.2 Field Documentation

8.545.2.1 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMode

8.545.2.2 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMultiplier

8.546 pack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint8_t ∗ pTech

- uint32_t * pprofileid3gpp
- uint32_t * pprofileid3gpp2
- uint32_t * pAuth
- char * pUser
- char * pPass

8.546.1 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

8.546.2 Field Documentation

8.546.2.1 uint32_t* pack_wds_SLQSStartDataSession_t::pAuth

8.546.2.2 char* pack_wds_SLQSStartDataSession_t::pPass

8.546.2.3 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp

8.546.2.4 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp2

8.546.2.5 uint8_t* pack_wds_SLQSStartDataSession_t::pTech

8.546.2.6 char* pack_wds_SLQSStartDataSession_t::pUser

8.547 pack_wds_SLQSStopDataSession_t Struct Reference

Data Fields

- uint32_t * psid

8.547.1 Detailed Description

This structure contains pack stop data session information.

Parameters

psid	session id

8.547.2 Field Documentation

8.547.2.1 uint32_t* pack_wds_SLQSStopDataSession_t::psid

8.548 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

8.548.1 Detailed Description

This structure contains the information about the Set Event Report Request parameters.

Parameters

pCurrChannel-RateInd	(optional) • Current Channel Rate Indicator. – 0 - Do not report – 1 - Report channel rate when it changes
pTransferStatInd	(optional) • See wds_TrStatInd for more information.
pDataBearer-TechInd	(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
pCurrPrefDataSysInd	(optional) • Current Preferred Data System Indicator. – 0 - Do not report – 1 - Report preferred data system when it changes
pEVDOPageMonPerChangeInd	(optional) • EV-DO Page Monitor Period Change Indicator. – 0 - Do not report – 1 - Report EV-DO page monitor period change event
pDataSystemStatusChangeInd	(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.

8.548.2 Field Documentation

8.548.2.1 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrChannelRateInd

8.548.2.2 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrDataBearerTechInd

8.548.2.3 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrPrefDataSysInd

8.548.2.4 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataBearerTechInd

8.548.2.5 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataCallStatusChangeInd

8.548.2.6 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataSystemStatusChangeInd

8.548.2.7 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDormancyStatusInd

8.548.2.8 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pEVDOPageMonPerChangeInd

8.548.2.9 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pMIPStatusInd

8.548.2.10 wds_TrStatInd* pack_wds_SLQSWdsSetEventReport_t::pTransferStatInd

8.549 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference

Data Fields

- uint8_t contextId
- uint8_t contextType

8.549.1 Detailed Description

This structure contains the PDP Runtime Settings Request parameters.

Parameters

contextId	• Context Identifier
contextType	• Identifies technology type – 0 - 3GPP – 1 - 3GPP2

8.549.2 Field Documentation

8.549.2.1 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.549.2.2 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextType

8.550 PackCreateProfileOut Struct Reference

Data Fields

- uint8_t ProfileType
- uint8_t ProfileIndex
- uint16_t ExtErrorCode

8.550.1 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**

8.550.2 Field Documentation

8.550.2.1 uint16_t PackCreateProfileOut::ExtErrorCode

8.550.2.2 uint8_t PackCreateProfileOut::ProfileIndex

8.550.2.3 uint8_t PackCreateProfileOut::ProfileType

8.551 packgetDyingGaspCfg Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.551.1 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.

8.551.2 Field Documentation

8.551.2.1 uint8_t* packgetDyingGaspCfg::pDestSMSContent

8.551.2.2 uint8_t* packgetDyingGaspCfg::pDestSMSNum

8.552 packgetDyingGaspStatistics Struct Reference

Data Fields

- uint32_t * pTimeStamp
- uint8_t * pSMSAttemptedFlag

8.552.1 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.

8.552.2 Field Documentation

8.552.2.1 uint8_t* packgetDyingGaspStatistics::pSMSAttemptedFlag

8.552.2.2 uint32_t* packgetDyingGaspStatistics::pTimeStamp

8.553 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]

8.553.1 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.

8.553.2 Field Documentation

8.553.2.1 uint8_t PackSwiAvmsSetSettingsAPNInfo::bAPNLength

8.553.2.2 uint8_t PackSwiAvmsSetSettingsAPNInfo::bPWDLength

8.553.2.3 uint8_t PackSwiAvmsSetSettingsAPNInfo::bUnameLength

8.553.2.4 uint8_t PackSwiAvmsSetSettingsAPNInfo::szAPN[49]

8.553.2.5 uint8_t PackSwiAvmsSetSettingsAPNInfo::szPWD[29]

8.553.2.6 uint8_t PackSwiAvmsSetSettingsAPNInfo::szUname[29]

8.554 PackSwiAvmsSetSettingsConnectionRetryTimers Struct Reference

Data Fields

- uint16_t Timers [8]

8.554.1 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)

8.554.2 Field Documentation

8.554.2.1 uint16_t PackSwiAvmsSetSettingsConnectionRetryTimers::Timers[8]

8.555 PackSwiAvmsSetSettingsPeriodInfo Struct Reference

Data Fields

- uint32_t ulMin
- uint32_t ulMax

8.555.1 Detailed Description

This structure contain pack set settings period information.

Parameters

ulMin	Min period in seconds.
ulMax	Max period in seconds.

8.555.2 Field Documentation

8.555.2.1 uint32_t PackSwiAvmsSetSettingsPeriodInfo::ulMax

8.555.2.2 uint32_t PackSwiAvmsSetSettingsPeriodInfo::ulMin

8.556 PackSwiAVMSSettingsAPNInfo Struct Reference

Data Fields

- uint8_t bAPNLength
- uint8_t * pAPN
- uint8_t bUnameLength
- uint8_t * pUname
- uint8_t bPWDLength
- uint8_t * pPWD

8.556.1 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.

8.556.2 Field Documentation

8.556.2.1 uint8_t PackSwiAVMSSettingsAPNInfo::bAPNLength

8.556.2.2 uint8_t PackSwiAVMSSettingsAPNInfo::bPWDLength

8.556.2.3 uint8_t PackSwiAVMSSettingsAPNInfo::bUnameLength

8.556.2.4 uint8_t* PackSwiAVMSSettingsAPNInfo::pAPN

8.556.2.5 uint8_t* PackSwiAVMSSettingsAPNInfo::pPWD

8.556.2.6 uint8_t* PackSwiAVMSSettingsAPNInfo::pUname

8.557 PackSwiAVMSSettingsConnectionRetryTimers Struct Reference

Data Fields

- uint16_t Timers [8]

8.557.1 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)

8.557.2 Field Documentation

8.557.2.1 uint16_t PackSwiAVMSSettingsConnectionRetryTimers::Timers[8]

8.558 PackSwiAVMSSettingsPeriodsInfo Struct Reference

Data Fields

- uint32_t min
- uint32_t max

8.558.1 Detailed Description

This structure contains the SLQSAVMSSettings Min/Max Period of an Observation.

Parameters

min	- Min period in seconds.
max	- Max period in seconds.

8.558.2 Field Documentation

8.558.2.1 uint32_t PackSwiAVMSSettingsPeriodsInfo::max

8.558.2.2 uint32_t PackSwiAVMSSettingsPeriodsInfo::min

8.559 qmiSmsMessageList Struct Reference

Data Fields

- uint32_t messageIndex
- uint32_t messageTag

8.559.1 Detailed Description

This structure contains SMS message list.

Parameters

messageIndex	• Message index of each matched message
messageTag	• Messagetag

8.559.2 Field Documentation

8.559.2.1 uint32_t qmiSmsMessageList::messageIndex

8.559.2.2 uint32_t qmiSmsMessageList::messageTag

8.560 qmiWDSDataBearerTechnology Struct Reference

Data Fields

- uint8_t currentNetwork
- uint32_t ratMask
- uint32_t soMask

8.560.1 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

8.560.2 Field Documentation

8.560.2.1 uint8_t qmiWDSDataBearerTechnology::currentNetwork

8.560.2.2 uint32_t qmiWDSDataBearerTechnology::ratMask

8.560.2.3 uint32_t qmiWDSDataBearerTechnology::soMask

8.561 qmTlvResult Struct Reference

Data Fields

- swi_uint256_t TlvPresenceMask
- qmulong TlvResultCode
- qmuint16 DeviceResult
- qmuint16 DeviceError

8.561.1 Field Documentation

8.561.1.1 qmuint16 qmTlvResult::DeviceError

8.561.1.2 qmuint16 qmTlvResult::DeviceResult

8.561.1.3 swi_uint256_t qmTlvResult::TlvPresenceMask

8.561.1.4 qmulong qmTlvResult::TlvResultCode

8.562 qos_BindDataPortMuxID_t Struct Reference

Data Fields

- uint8_t MuxID

8.562.1 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

8.562.2 Field Documentation

8.562.2.1 uint8_t qos_BindDataPortMuxID_t::MuxID

8.563 qos_BindDataPortPeripheralEndPointID_t Struct Reference

Data Fields

- uint32_t EndPointType
- uint32_t IfaceID

8.563.1 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.

8.563.2 Field Documentation

8.563.2.1 uint32_t qos_BindDataPortPeripheralEndPointID_t::EndPointType

8.563.2.2 uint32_t qos_BindDataPortPeripheralEndPointID_t::IfaceID

8.564 qos_BindDataPortSIODataPort_t Struct Reference

Data Fields

- uint16_t SIODataPort

8.564.1 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

8.564.2 Field Documentation

8.564.2.1 uint16_t qos_BindDataPortSIODataPort_t::SIODataPort

8.565 RFBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBandClass
- uint16_t activeChannel

8.565.1 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)

8.565.2 Field Documentation

8.565.2.1 uint16_t RFBandInfoElements::activeBandClass

8.565.2.2 uint16_t RFBandInfoElements::activeChannel

8.565.2.3 uint8_t RFBandInfoElements::radioInterface

8.566 rmTrasnferStaticsReq Struct Reference

Data Fields

- uint8_t bResetStatistics
- uint32_t ulMask

8.566.1 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

8.566.2 Field Documentation

8.566.2.1 uint8_t rmTrasnferStaticsReq::bResetStatistics

8.566.2.2 uint32_t rmTrasnferStaticsReq::ulMask

8.567 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]

8.567.1 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

8.567.2 Field Documentation

8.567.2.1 uint8_t sensorData_t::flags

8.567.2.2 uint8_t sensorData_t::sensorDataLen

8.567.2.3 uint32_t sensorData_t::timeOfFirstSample

8.567.2.4 uint16_t sensorData_t::timeOffset[64]

8.567.2.5 uint32_t sensorData_t::xAxis[64]

8.567.2.6 uint32_t sensorData_t::yAxis[64]

8.567.2.7 uint32_t sensorData_t::zAxis[64]

8.568 slot_t Struct Reference

Data Fields

- uint32_t uPhyCardStatus
- uint32_t uPhySlotStatus
- uint8_t bLogicalSlot
- uint8_t bICCIDLength
- uint8_t bICCID [255]

8.568.1 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.

8.568.2 Field Documentation

8.568.2.1 uint8_t slot_t::bICCID[255]

8.568.2.2 uint8_t slot_t::bICCIDLength

8.568.2.3 uint8_t slot_t::bLogicalSlot

8.568.2.4 uint32_t slot_t::uPhyCardStatus

8.568.2.5 uint32_t slot_t::uPhySlotStatus

8.569 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]

8.569.1 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.

8.569.2 Field Documentation

8.569.2.1 appStats slotInf::AppStatus[10]

8.569.2.2 uint8_t slotInf::cardState

8.569.2.3 uint8_t slotInf::errorState

8.569.2.4 uint8_t slotInf::numApp

8.569.2.5 uint8_t slotInf::upinRetries

8.569.2.6 uint8_t slotInf::upinState

8.569.2.7 uint8_t slotInf::upukRetries

8.570 slots_t Struct Reference

Data Fields

- slot_t uimSlotStatus [255]

8.570.1 Detailed Description

This structure contains slots status.

Parameters

uimSlotStatus	• See slot_t

8.570.2 Field Documentation

8.570.2.1 slot_t slots_t::uimSlotStatus[255]

8.571 sms_BroadcastConfig Struct Reference

Data Fields

- uint16_t fromServiceId
- uint16_t toServiceId
- uint8_t selected

8.571.1 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

8.571.2 Field Documentation

8.571.2.1 uint16_t sms_BroadcastConfig::fromServiceId

8.571.2.2 uint8_t sms_BroadcastConfig::selected

8.571.2.3 uint16_t sms_BroadcastConfig::toServiceId

8.572 sms_CDMABroadcastConfig Struct Reference

Data Fields

- uint16_t serviceCategory
- uint16_t language
- uint8_t selected

8.572.1 Detailed Description

This structure contains CDMABroadcastConfig parameters

Parameters

serviceCategory	• Service category
language	• Language
selected	• Specified service_category and language – 0x00 - Not selected – 0x01 - Selected

8.572.2 Field Documentation

8.572.2.1 uint16_t sms_CDMABroadcastConfig::language

8.572.2.2 uint8_t sms_CDMABroadcastConfig::selected

8.572.2.3 uint16_t sms_CDMABroadcastConfig::serviceCategory

8.573 sms_getIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.573.1 Detailed Description

This structure contains Get Indication Register Response parameters

Parameters

pRegTransLayerInfoEvt	- • Optional parameter indicating registration status of transport layer information events • Values: – 0x00 - Disabled – 0x01 - Enabled
pRegTransNWRegInfoEvt	- • 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

8.573.2 Field Documentation

8.573.2.1 uint8_t* sms_getIndicationReg::pRegCallStatInfoEvt

8.573.2.2 uint8_t* sms_getIndicationReg::pRegTransLayerInfoEvt

8.573.2.3 uint8_t* sms_getIndicationReg::pRegTransNWRegInfoEvt

8.574 sms_getMsgWaitingInfo Struct Reference

Data Fields

- uint8_t numInstances
- sms_messageWaitingInfoContent msgWaitInfo [0xFF]

8.574.1 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.

8.574.2 Field Documentation

8.574.2.1 sms_messageWaitingInfoContent sms_getMsgWaitingInfo::msgWaitInfo[0xFF]

8.574.2.2 uint8_t sms_getMsgWaitingInfo::numInstances

8.575 sms_getTransLayerInfo Struct Reference

Data Fields

- uint8_t * pRegInd
- sms_transLayerInfo * pTransLayerInfo

8.575.1 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.

8.575.2 Field Documentation

8.575.2.1 uint8_t* sms_getTransLayerInfo::pRegInd

8.575.2.2 sms_transLayerInfo* sms_getTransLayerInfo::pTransLayerInfo

8.576 sms_getTransNWRegInfo Struct Reference

Data Fields

- uint8_t * pRegStatus

8.576.1 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.

8.576.2 Field Documentation

8.576.2.1 uint8_t* sms_getTransNWRegInfo::pRegStatus

8.577 sms_maxStorageSizeReq Struct Reference

Data Fields

- uint8_t storageType
- uint8_t * pMessageMode

8.577.1 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

8.577.2 Field Documentation

8.577.2.1 uint8_t* sms_maxStorageSizeReq::pMessageMode

8.577.2.2 uint8_t sms_maxStorageSizeReq::storageType

8.578 sms_maxStorageSizeResp Struct Reference

Data Fields

- uint32_t maxStorageSize
- uint32_t freeSlots

8.578.1 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.

8.578.2 Field Documentation

8.578.2.1 uint32_t sms_maxStorageSizeResp::freeSlots

8.578.2.2 uint32_t sms_maxStorageSizeResp::maxStorageSize

8.579 sms_messageWaitingInfoContent Struct Reference

Data Fields

- uint8_t msgType
- uint8_t activeInd
- uint8_t msgCount

8.579.1 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

8.579.2 Field Documentation

8.579.2.1 uint8_t sms_messageWaitingInfoContent::activeInd

8.579.2.2 uint8_t sms_messageWaitingInfoContent::msgCount

8.579.2.3 uint8_t sms_messageWaitingInfoContent::msgType

8.580 sms_msgProtocolResp Struct Reference

Data Fields

- uint8_t msgProtocol

8.580.1 Detailed Description

This structure contains get message protocol response parameters

Parameters

msgProtocol	- • Message Protocol • Values: – 0x00 - MESSAGE_PROTOCOL_CDMA – 0x01 - MESSAGE_PROTOCOL_WCDMA

8.580.2 Field Documentation

8.580.2.1 uint8_t sms_msgProtocolResp::msgProtocol

8.581 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_CDMABroadcastConfig CDMABroadcastConfig [0x05]

8.581.1 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

8.581.2 Field Documentation

8.581.2.1 uint8_t sms_qaQmi3GPP2BroadcastCfgInfo::activated_ind

8.581.2.2 sms_CDMABroadcastConfig sms_qaQmi3GPP2BroadcastCfgInfo::CDMABroadcastConfig[0x05]

8.581.2.3 uint16_t sms_qaQmi3GPP2BroadcastCfgInfo::num_instances

8.582 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_BroadcastConfig broadcastConfig [0x05]

8.582.1 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

8.582.2 Field Documentation

8.582.2.1 uint8_t sms_qaQmi3GPPBroadcastCfgInfo::activated_ind

8.582.2.2 sms_BroadcastConfig sms_qaQmi3GPPBroadcastCfgInfo::broadcastConfig[0x05]

8.582.2.3 uint16_t sms_qaQmi3GPPBroadcastCfgInfo::num_instances

8.583 sms_routeEntry Struct Reference

Data Fields

- uint8_t messageType
- uint8_t messageClass

- uint8_t routeStorage
- uint8_t receiptAction

8.583.1 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)

8.583.2 Field Documentation

8.583.2.1 uint8_t sms_routeEntry::messageClass

8.583.2.2 uint8_t sms_routeEntry::messageType

8.583.2.3 uint8_t sms_routeEntry::receiptAction

8.583.2.4 uint8_t sms_routeEntry::routeStorage

8.584 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

8.584.1 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.

8.584.2 Field Documentation

8.584.2.1 uint32_t sms_sendAsyncsmsParams::messageFormat

8.584.2.2 uint32_t sms_sendAsyncsmsParams::messageSize

8.584.2.3 uint8_t* sms_sendAsyncsmsParams::pFollowOnDC

8.584.2.4 uint8_t* sms_sendAsyncsmsParams::pForceOnDC

8.584.2.5 uint8_t* sms_sendAsyncsmsParams::pLinktimer

8.584.2.6 uint8_t* sms_sendAsyncsmsParams::pMessage

8.584.2.7 uint8_t* sms_sendAsyncsmsParams::pRetryMessage

8.584.2.8 uint32_t* sms_sendAsyncsmsParams::pRetryMessageId

8.584.2.9 uint8_t* sms_sendAsyncsmsParams::pServiceOption

8.584.2.10 uint8_t* sms_sendAsyncsmsParams::pSmsOnIms

8.584.2.11 uint32_t* sms_sendAsyncsmsParams::pUserData

8.585 sms_setIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.585.1 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

8.585.2 Field Documentation

8.585.2.1 uint8_t* sms_setIndicationReg::pRegCallStatInfoEvt

8.585.2.2 uint8_t* sms_setIndicationReg::pRegTransLayerInfoEvt

8.585.2.3 uint8_t* sms_setIndicationReg::pRegTransNWRegInfoEvt

8.586 sms_setRoutesReq Struct Reference

Data Fields

- uint16_t numOfRoutes

- sms_routeEntry routeList [0x0A]
- uint8_t * pTransferStatusReport

8.586.1 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)

8.586.2 Field Documentation

8.586.2.1 uint16_t sms_setRoutesReq::numOfRoutes

8.586.2.2 uint8_t* sms_setRoutesReq::pTransferStatusReport

8.586.2.3 sms_routeEntry sms_setRoutesReq::routeList[0x0A]

8.587 sms_transLayerInfo Struct Reference

Data Fields

- uint8_t TransType
- uint8_t TransCap

8.587.1 Detailed Description

This structure contains Transport Layer Information

Parameters

TransType	• Transport Type – 0x00 - IMS
TransCap	• Transport Capability • Values: – 0x00 - CDMA – 0x01 - GW

8.587.2 Field Documentation

8.587.2.1 uint8_t sms_transLayerInfo::TransCap

8.587.2.2 uint8_t sms_transLayerInfo::TransType

8.588 sMSCAddressInfo Struct Reference

Data Fields

- uint8_t length
- uint8_t data [256]

8.588.1 Detailed Description

This structure contains SMSC information

Parameters

length	• Number of sets of following element
data	• SMSC address

8.588.2 Field Documentation

8.588.2.1 uint8_t sMSCAddressInfo::data[256]

8.588.2.2 uint8_t sMSCAddressInfo::length

8.589 sMSCAddressTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSCAddressInfo SMSCInfo

8.589.1 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

8.589.2 Field Documentation

8.589.2.1 sMSCAddressInfo sMSCAddressTlv::SMSCInfo

8.589.2.2 uint8_t sMSCAddressTlv::TlvPresent

8.590 sMSEtwsMessageInfo Struct Reference

Data Fields

- uint8_t notificationType
- uint16_t length
- uint8_t data [1254]

8.590.1 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

8.590.2 Field Documentation

8.590.2.1 uint8_t sMSEtwsMessageInfo::data[1254]

8.590.2.2 uint16_t sMSEtwsMessageInfo::length

8.590.2.3 uint8_t sMSEtwsMessageInfo::notificationType

8.591 sMSEtwsMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsMessageInfo EtwsMessageInfo

8.591.1 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

8.591.2 Field Documentation

8.591.2.1 sMSEtwsMessageInfo sMSEtwsMessageTlv::EtwsMessageInfo

8.591.2.2 uint8_t sMSEtwsMessageTlv::TlvPresent

8.592 sMSEtwsPlmnInfo Struct Reference

Data Fields

- uint16_t mobileCountryCode
- uint16_t mobileNetworkCode

8.592.1 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

8.592.2 Field Documentation

8.592.2.1 uint16_t sMSEtwsPlmnInfo::mobileCountryCode

8.592.2.2 uint16_t sMSEtwsPlmnInfo::mobileNetworkCode

8.593 sMSMessageModeInfo Struct Reference

Data Fields

- uint8_t messageMode

8.593.1 Detailed Description

This structure contains message mode information.

Parameters

messageMode	• Message Mode – 0x00 - CDMA – 0x01 - GW

8.593.2 Field Documentation

8.593.2.1 uint8_t sMSMessageModeInfo::messageMode

8.594 sMSMTMessageInfo Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex

8.594.1 Detailed Description

This structure contains MT message information.

Parameters

storageType	memory storage 0x00-UIM 0x01-NV
messageIndex	MT Message index

8.594.2 Field Documentation

8.594.2.1 uint32_t sMSMTMessageInfo::messageIndex

8.594.2.2 uint32_t sMSMTMessageInfo::storageType

8.595 sMSOnIMSInfo Struct Reference

Data Fields

- uint8_t smsOnIMS

8.595.1 Detailed Description

This structure contains SMS on IMS information

Parameters

smsOnIMS	SMS on IMS

8.595.2 Field Documentation

8.595.2.1 uint8_t sMSOnIMSInfo::smsOnIMS

8.596 sMSOnIMSTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSOnIMSInfo IMSInfo

8.596.1 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

8.596.2 Field Documentation

8.596.2.1 sMSOnIMSInfo sMSOnIMSTlv::IMSInfo

8.596.2.2 uint8_t sMSOnIMSTlv::TlvPresent

8.597 sMSTransferRouteMTMessageInfo Struct Reference

Data Fields

- uint8_t ackIndicator
- uint32_t transactionID
- uint8_t format
- uint16_t length
- uint8_t data [256]

8.597.1 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

8.597.2 Field Documentation

8.597.2.1 uint8_t sMSTransferRouteMTMessageInfo::ackIndicator

8.597.2.2 uint8_t sMSTransferRouteMTMessageInfo::data[256]

8.597.2.3 uint8_t sMSTransferRouteMTMessageInfo::format

8.597.2.4 uint16_t sMSTransferRouteMTMessageInfo::length

8.597.2.5 uint32_t sMSTransferRouteMTMessageInfo::transactionID

8.598 swi_uint256_t Struct Reference

Data Fields

- uint16_t word [16]

8.598.1 Detailed Description

SWI 256 bit data type

8.598.2 Field Documentation

8.598.2.1 uint16_t swi_uint256_t::word[16]

8.599 swiaudio_PCMparams Struct Reference

Data Fields

- uint8_t iFaceTabLen
- uint8_t iFaceTab [255]

8.599.1 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

8.599.2 Field Documentation

8.599.2.1 uint8_t swiaudio_PCMparams::iFaceTab[255]

8.599.2.2 uint8_t swiaudio_PCMparams::iFaceTabLen

8.600 swidms_ehrpdMTUSizeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ehrpdMTUSize

8.600.1 Detailed Description

This structure contains the parameters for EHRPD MTU Size.

Parameters

TlvPresent	• Tlv Present.
ehrpdMTUSize	EHRPD size • Size of EHRPD MTU

8.600.2 Field Documentation

8.600.2.1 uint16_t swidms_ehrpdMTUSizeTlv::ehrpdMTUSize

8.600.2.2 uint8_t swidms_ehrpdMTUSizeTlv::TlvPresent

8.601 swidms_hrpdMTUSizeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t hrpdMTUSize

8.601.1 Detailed Description

This structure contains the parameters for HRPD MTU Size.

Parameters

TlvPresent	• Tlv Present.
hrpdMTUSize	HRPD size • Size of HRPD MTU

8.601.2 Field Documentation

8.601.2.1 uint16_t swidms_hrpdMTUSizeTlv::hrpdMTUSize

8.601.2.2 uint8_t swidms_hrpdMTUSizeTlv::TlvPresent

8.602 swidms_intfaceCfgTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t CurrentCfgType
- uint32_t CfgValue

8.602.1 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

8.602.2 Field Documentation

8.602.2.1 uint32_t swidms_intfaceCfgTlv::CfgValue

8.602.2.2 uint32_t swidms_intfaceCfgTlv::CurrentCfgType

8.602.2.3 uint8_t swidms_intfaceCfgTlv::TlvPresent

8.603 swidms_mtuSize3gppTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t MTUSize3gpp

8.603.1 Detailed Description

This structure contains the parameters for 3GPP MTU Size.

Parameters

TlvPresent	• Tlv Present.

MTUSize3gpp	3GPP MTU size • Size of 3GPP MTU

8.603.2 Field Documentation

8.603.2.1 uint16_t swidms_mtuSize3gppTlv::MTUSize3gpp

8.603.2.2 uint8_t swidms_mtuSize3gppTlv::TlvPresent

8.604 swidms_supportedIntBitmaskTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ValidBitmasks

8.604.1 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.

8.604.2 Field Documentation

8.604.2.1 uint8_t swidms_supportedIntBitmaskTlv::TlvPresent

8.604.2.2 uint32_t swidms_supportedIntBitmaskTlv::ValidBitmasks

8.605 swidms_usbMTUSizeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t UsbMTUSize

8.605.1 Detailed Description

This structure contains the parameters for USB MTU Size

Parameters

TlvPresent	• Tlv Present.

UsbMTUSize	USB MTU size • Size of USB MTU

8.605.2 Field Documentation

8.605.2.1 uint8_t swidms_usbMTUSizeTlv::TlvPresent

8.605.2.2 uint16_t swidms_usbMTUSizeTlv::UsbMTUSize

8.606 tdscdmaSigInfoExt Struct Reference

Data Fields

- float rssi
- float rscp
- float ecio
- float sinr

8.606.1 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

8.606.2 Field Documentation

8.606.2.1 float tdscdmaSigInfoExt::ecio

8.606.2.2 float tdscdmaSigInfoExt::rscp

8.606.2.3 float tdscdmaSigInfoExt::rssi

8.606.2.4 float tdscdmaSigInfoExt::sinr

8.607 tempData_t Struct Reference

Data Fields

- uint32_t timeSource
- uint32_t timeOfFirstSample
- uint8_t temperatureDataLen
- uint16_t timeOffset [64]
- uint32_t temperature [64]

8.607.1 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

8.607.2 Field Documentation

8.607.2.1 uint32_t tempData_t::temperature[64]

8.607.2.2 uint8_t tempData_t::temperatureDataLen

8.607.2.3 uint32_t tempData_t::timeOfFirstSample

8.607.2.4 uint16_t tempData_t::timeOffset[64]

8.607.2.5 uint32_t tempData_t::timeSource

8.608 tmd_mitigationDevList Struct Reference

Data Fields

- uint8_t mitigationDevIdLen
- char mitigationDevId [255]
- uint8_t maxMitigationLevel

8.608.1 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.

8.608.2 Field Documentation

8.608.2.1 uint8_t tmd_mitigationDevList::maxMitigationLevel

8.608.2.2 char tmd_mitigationDevList::mitigationDevId[255]

8.608.2.3 uint8_t tmd_mitigationDevList::mitigationDevIdLen

8.609 transferRouteMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSTransferRouteMTMessageInfo TransferRouteMTMessageInfo

8.609.1 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

8.609.2 Field Documentation

8.609.2.1 uint8_t transferRouteMessageTlv::TlvPresent

8.609.2.2 sMSTransferRouteMTMessageInfo transferRouteMessageTlv::TransferRouteMTMessageInfo

8.610 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

8.610.1 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.
persoUnblockRetries	• 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.

8.610.2 Field Documentation

8.610.2.1 uint8_t uim_appStatus::aidLength

8.610.2.2 uint8_t uim_appStatus::aidVal[255]

8.610.2.3 uint8_t uim_appStatus::appState

8.610.2.4 uint8_t uim_appStatus::appType

8.610.2.5 uint8_t uim_appStatus::persoFeature

8.610.2.6 uint8_t uim_appStatus::persoRetries

8.610.2.7 uint8_t uim_appStatus::persoState

8.610.2.8 uint8_t uim_appStatus::persoUnblockRetries

8.610.2.9 uint8_t uim_appStatus::pin1Retries

8.610.2.10 uint8_t uim_appStatus::pin1State

8.610.2.11 uint8_t uim_appStatus::pin2Retries

8.610.2.12 uint8_t uim_appStatus::pin2State

8.610.2.13 uint8_t uim_appStatus::puk1Retries

8.610.2.14 uint8_t uim_appStatus::puk2Retries

8.610.2.15 uint8_t uim_appStatus::univPin

8.611 uim_authenticateResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [1024]

8.611.1 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.

8.611.2 Field Documentation

8.611.2.1 uint8_t uim_authenticateResult::content[1024]

8.611.2.2 uint16_t uim_authenticateResult::contentLen

8.612 uim_authenticationData Struct Reference

Data Fields

- uint8_t context
- uint16_t dataLen
- uint8_t data [1024]

8.612.1 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.

8.612.2 Field Documentation

8.612.2.1 uint8_t uim_authenticationData::context

8.612.2.2 uint8_t uim_authenticationData::data[1024]

8.612.2.3 uint16_t uim_authenticationData::dataLen

8.613 uim_cardResult Struct Reference

Data Fields

- uint8_t sw1
- uint8_t sw2

8.613.1 Detailed Description

This structure contains the information about the card result.

Parameters

sw1	• SW1 received from the card.
sw2	• SW2 received from the card.

8.613.2 Field Documentation

8.613.2.1 uint8_t uim_cardResult::sw1

8.613.2.2 uint8_t uim_cardResult::sw2

8.614 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]

8.614.1 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.

8.614.2 Field Documentation

8.614.2.1 uint16_t uim_cardStatus::index1xPri

8.614.2.2 uint16_t uim_cardStatus::index1xSec

8.614.2.3 uint16_t uim_cardStatus::indexGwPri

8.614.2.4 uint16_t uim_cardStatus::indexGwSec

8.614.2.5 uint8_t uim_cardStatus::numSlot

8.614.2.6 uim_slotInfo uim_cardStatus::SlotInfo[5]

8.615 uim_changeUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t oldPINLen
- uint8_t oldPINVal [255]
- uint8_t pinLen
- uint8_t pinVal [255]

8.615.1 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.

8.615.2 Field Documentation

8.615.2.1 uint8_t uim_changeUIMPIN::oldPINLen

8.615.2.2 uint8_t uim_changeUIMPIN::oldPINVal[255]

8.615.2.3 uint8_t uim_changeUIMPIN::pinID

8.615.2.4 uint8_t uim_changeUIMPIN::pinLen

8.615.2.5 uint8_t uim_changeUIMPIN::pinVal[255]

8.616 uim_depersonalizationInformation Struct Reference

Data Fields

- uint8_t feature
- uint8_t operation
- uint8_t ckLen
- uint8_t ckVal [255]

8.616.1 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.

8.616.2 Field Documentation

8.616.2.1 uint8_t uim_depersonalizationInformation::ckLen

8.616.2.2 uint8_t uim_depersonalizationInformation::ckVal[255]

8.616.2.3 uint8_t uim_depersonalizationInformation::feature

8.616.2.4 uint8_t uim_depersonalizationInformation::operation

8.617 uim_encryptedPIN1 Struct Reference

Data Fields

- uint8_t pin1Len
- uint8_t pin1Val [255]

8.617.1 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.

8.617.2 Field Documentation

8.617.2.1 uint8_t uim_encryptedPIN1::pin1Len

8.617.2.2 uint8_t uim_encryptedPIN1::pin1Val[255]

8.618 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]

8.618.1 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.

8.618.2 Field Documentation

8.618.2.1 uint16_t uim_fileAttributes::fileID

8.618.2.2 uint16_t uim_fileAttributes::fileSize

8.618.2.3 uint8_t uim_fileAttributes::fileType

8.618.2.4 uint16_t uim_fileAttributes::rawLen

8.618.2.5 uint8_t uim_fileAttributes::rawValue[255]

8.618.2.6 uint16_t uim_fileAttributes::recordCount

8.618.2.7 uint16_t uim_fileAttributes::recordSize

8.618.2.8 uint8_t uim_fileAttributes::secActivate

8.618.2.9 uint16_t uim_fileAttributes::secActivateMask

8.618.2.10 uint8_t uim_fileAttributes::secDeactivate

8.618.2.11 uint16_t uim_fileAttributes::secDeactivateMask

8.618.2.12 uint8_t uim_fileAttributes::secIncrease

8.618.2.13 uint16_t uim_fileAttributes::secIncreaseMask

8.618.2.14 uint8_t uim_fileAttributes::secRead

8.618.2.15 uint16_t uim_fileAttributes::secReadMask

8.618.2.16 uint8_t uim_fileAttributes::secWrite

8.618.2.17 uint16_t uim_fileAttributes::secWriteMask

8.619 uim_fileInfo Struct Reference

Data Fields

- uint16_t fileID
- uint8_t pathLen
- uint16_t path [255]

8.619.1 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.

8.619.2 Field Documentation

8.619.2.1 uint16_t uim_fileInfo::fileID

8.619.2.2 uint16_t uim_fileInfo::path[255]

8.619.2.3 uint8_t uim_fileInfo::pathLen

8.620 uim_hotSwapStatus Struct Reference

Data Fields

- uint8_t hotSwapLength
- uint8_t hotSwap [255]

8.620.1 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

8.620.2 Field Documentation

8.620.2.1 uint8_t uim_hotSwapStatus::hotSwap[255]

8.620.2.2 uint8_t uim_hotSwapStatus::hotSwapLength

8.621 uim_personalizationStatus Struct Reference

Data Fields

- uint8_t numFeatures
- uint8_t feature [12]
- uint8_t verifyLeft [12]
- uint8_t unblockLeft [12]

8.621.1 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.

8.621.2 Field Documentation

8.621.2.1 uint8_t uim_personalizationStatus::feature[12]

8.621.2.2 uint8_t uim_personalizationStatus::numFeatures

8.621.2.3 uint8_t uim_personalizationStatus::unblockLeft[12]

8.621.2.4 uint8_t uim_personalizationStatus::verifyLeft[12]

8.622 uim_readResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [255]

8.622.1 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.

8.622.2 Field Documentation

8.622.2.1 uint8_t uim_readResult::content[255]

8.622.2.2 uint16_t uim_readResult::contentLen

8.623 uim_readTransparentInfo Struct Reference

Data Fields

- uint16_t offset
- uint16_t length

8.623.1 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.

8.623.2 Field Documentation

8.623.2.1 uint16_t uim_readTransparentInfo::length

8.623.2.2 uint16_t uim_readTransparentInfo::offset

8.624 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]

8.624.1 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

8.624.2 Field Documentation

8.624.2.1 uint8_t uim_refreshevent::aid[255]

8.624.2.2 uint8_t uim_refreshevent::aidLength

8.624.2.3 uim_fileInfo uim_refreshevent::arrfileInfo[255]

8.624.2.4 uint8_t uim_refreshevent::mode

8.624.2.5 uint16_t uim_refreshevent::numOfFiles

8.624.2.6 uint8_t uim_refreshevent::sessionType

8.624.2.7 uint8_t uim_refreshevent::stage

8.625 uim_registerRefresh Struct Reference

Data Fields

- uint8_t registerFlag
- uint8_t voteForInit
- uint16_t numFiles
- uim_fileInfo arrfileInfo [255]

8.625.1 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

8.625.2 Field Documentation

8.625.2.1 uim_fileInfo uim_registerRefresh::arrfileInfo[255]

8.625.2.2 uint16_t uim_registerRefresh::numFiles

8.625.2.3 uint8_t uim_registerRefresh::registerFlag

8.625.2.4 uint8_t uim_registerRefresh::voteForInit

8.626 uim_remainingRetries Struct Reference

Data Fields

- uint8_t verifyLeft
- uint8_t unblockLeft

8.626.1 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.

8.626.2 Field Documentation

8.626.2.1 uint8_t uim_remainingRetries::unblockLeft

8.626.2.2 uint8_t uim_remainingRetries::verifyLeft

8.627 uim_sessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.627.1 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.

8.627.2 Field Documentation

8.627.2.1 uint8_t uim_sessionInformation::aid[255]

8.627.2.2 uint8_t uim_sessionInformation::aidLength

8.627.2.3 uint8_t uim_sessionInformation::sessionType

8.628 uim_setPINProtection Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinOperation

- uint8_t pinLength
- uint8_t pinValue [255]

8.628.1 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.

8.628.2 Field Documentation

8.628.2.1 uint8_t uim_setPINProtection::pinID

8.628.2.2 uint8_t uim_setPINProtection::pinLength

8.628.2.3 uint8_t uim_setPINProtection::pinOperation

8.628.2.4 uint8_t uim_setPINProtection::pinValue[255]

8.629 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]

8.629.1 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.

8.629.2 Field Documentation

8.629.2.1 uim_appStatus uim_slotInfo::AppStatus[10]

8.629.2.2 uint8_t uim_slotInfo::cardState

8.629.2.3 uint8_t uim_slotInfo::errorState

8.629.2.4 uint8_t uim_slotInfo::numApp

8.629.2.5 uint8_t uim_slotInfo::upinRetries

8.629.2.6 uint8_t uim_slotInfo::upinState

8.629.2.7 uint8_t uim_slotInfo::upukRetries

8.630 uim_UIMSessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.630.1 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.

8.630.2 Field Documentation

8.630.2.1 uint8_t uim_UIMSessionInformation::aid[255]

8.630.2.2 uint8_t uim_UIMSessionInformation::aidLength

8.630.2.3 uint8_t uim_UIMSessionInformation::sessionType

8.631 uim_unblockUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pukLen
- uint8_t pukVal [255]
- uint8_t newPINLen
- uint8_t newPINVal [255]

8.631.1 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.

8.631.2 Field Documentation

8.631.2.1 uint8_t uim_unblockUIMPIN::newPINLen

8.631.2.2 uint8_t uim_unblockUIMPIN::newPINVal[255]

8.631.2.3 uint8_t uim_unblockUIMPIN::pinID

8.631.2.4 uint8_t uim_unblockUIMPIN::pukLen

8.631.2.5 uint8_t uim_unblockUIMPIN::pukVal[255]

8.632 uim_verifyUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinLen
- uint8_t pinVal [255]

8.632.1 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.

8.632.2 Field Documentation

8.632.2.1 uint8_t uim_verifyUIMPIN::pinID

8.632.2.2 uint8_t uim_verifyUIMPIN::pinLen

8.632.2.3 uint8_t uim_verifyUIMPIN::pinVal[255]

8.633 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

8.633.1 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**

8.633.2 Field Documentation

8.633.2.1 swi_uint256_t unpack_audio_SLQSGetAudioPathConfig_t::ParamPresenceMask

8.633.2.2 uint16_t* unpack_audio_SLQSGetAudioPathConfig_t::pCodecSTGain

8.633.2.3 uint16_t* unpack_audio_SLQSGetAudioPathConfig_t::pDTMFTXGain

8.633.2.4 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pECMode

8.633.2.5 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pMICGainSelect

8.633.2.6 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pNSEnable

8.633.2.7 audio_RXAGCList* unpack_audio_SLQSGetAudioPathConfig_t::pRXAGCList

8.633.2.8 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pRXAVCAGCSwitch

8.633.2.9 audio_RXAVCList* unpack_audio_SLQSGetAudioPathConfig_t::pRXAVCList

8.633.2.10 audio_RXPCMIIRFltr* unpack_audio_SLQSGetAudioPathConfig_t::pRXPCMIIRFltr

8.633.2.11 audio_TXAGCList* unpack_audio_SLQSGetAudioPathConfig_t::pTXAGCList

8.633.2.12 uint8_t* unpack_audio_SLQSGetAudioPathConfig_t::pTXAVCSwitch

8.633.2.13 uint16_t* unpack_audio_SLQSGetAudioPathConfig_t::pTXGain

8.633.2.14 audio_TXPCMIIRFItr* unpack_audio_SLQSGetAudioPathConfig_t::pTXPCMIIRFItr

8.634 unpack_audio_SLQSGetAudioProfile_t Struct Reference

Data Fields

- uint8_t Profile
- uint8_t EarMute
- uint8_t MicMute
- uint8_t Volume
- swi_uint256_t ParamPresenceMask

8.634.1 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**

8.634.2 Field Documentation

8.634.2.1 uint8_t unpack_audio_SLQSGetAudioProfile_t::EarMute

8.634.2.2 uint8_t unpack_audio_SLQSGetAudioProfile_t::MicMute

8.634.2.3 swi_uint256_t unpack_audio_SLQSGetAudioProfile_t::ParamPresenceMask

8.634.2.4 uint8_t unpack_audio_SLQSGetAudioProfile_t::Profile

8.634.2.5 uint8_t unpack_audio_SLQSGetAudioProfile_t::Volume

8.635 unpack_audio_SLQSGetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint16_t ResCode
- swi_uint256_t ParamPresenceMask

8.635.1 Detailed Description

This structure contains the unpack parameters for SLQSGetAudioVolTLBConfig.

Parameters

ResCode	• Result of requested item • Bit to check in ParamPresenceMask - **33**

8.635.2 Field Documentation

8.635.2.1 swi_uint256_t unpack_audio_SLQSGetAudioVolTLBConfig_t::ParamPresenceMask

8.635.2.2 uint16_t unpack_audio_SLQSGetAudioVolTLBConfig_t::ResCode

8.636 unpack_audio_SLQSSetAudioVolTLBConfig_t Struct Reference

Data Fields

- uint16_t ResCode
- swi_uint256_t ParamPresenceMask

8.636.1 Detailed Description

This structure contains the unpack parameters for SLQSSetAudioVolTLBConfig.

Parameters

ResCode	• Result of requested item • Bit to check in ParamPresenceMask - **33**

8.636.2 Field Documentation

8.636.2.1 swi_uint256_t unpack_audio_SLQSSetAudioVolTLBConfig_t::ParamPresenceMask

8.636.2.2 uint16_t unpack_audio_SLQSSetAudioVolTLBConfig_t::ResCode

8.637 unpack_cat_SetCatEventCallback_ind_t Struct Reference

Data Fields

- uint8_t event_Index
- struct cat_commonEventTlv CCETlv [11]
- swi_uint256_t ParamPresenceMask

8.637.1 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**

8.637.2 Field Documentation

8.637.2.1 struct cat_commonEventTlv unpack_cat_SetCatEventCallback_ind_t::CCETlv[11]

8.637.2.2 uint8_t unpack_cat_SetCatEventCallback_ind_t::event_Index

8.637.2.3 swi_uint256_t unpack_cat_SetCatEventCallback_ind_t::ParamPresenceMask

8.638 unpack_cat_SetCATEventCallback_t Struct Reference

Data Fields

- uint32_t errorMask
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.638.1 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

8.638.2 Field Documentation

8.638.2.1 uint32_t unpack_cat_SetCATEventCallback_t::errorMask

8.638.2.2 swi_uint256_t unpack_cat_SetCATEventCallback_t::ParamPresenceMask

8.638.2.3 uint16_t unpack_cat_SetCATEventCallback_t::TlvResult

8.639 unpack_dms_GetActivationState_t Struct Reference

Data Fields

- uint8_t state
- swi_uint256_t ParamPresenceMask

8.639.1 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.

8.639.2 Field Documentation

8.639.2.1 swi_uint256_t unpack_dms_GetActivationState_t::ParamPresenceMask

8.639.2.2 uint8_t unpack_dms_GetActivationState_t::state

8.640 unpack_dms_GetBandCapability_t Struct Reference

Data Fields

- uint32_t BandCapability
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.640.1 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.

8.640.2 Field Documentation

8.640.2.1 uint32_t unpack_dms_GetBandCapability_t::BandCapability

8.640.2.2 swi_uint256_t unpack_dms_GetBandCapability_t::ParamPresenceMask

8.640.2.3 uint16_t unpack_dms_GetBandCapability_t::TlvResult

8.641 unpack_dms_GetCrashAction_t Struct Reference

Data Fields

- uint8_t DevCrashState
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.641.1 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.

8.641.2 Field Documentation

8.641.2.1 uint8_t unpack_dms_GetCrashAction_t::DevCrashState

8.641.2.2 swi_uint256_t unpack_dms_GetCrashAction_t::ParamPresenceMask

8.641.2.3 uint16_t unpack_dms_GetCrashAction_t::Tlvresult

8.642 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

8.642.1 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.

8.642.2 Field Documentation

8.642.2.1 uint8_t unpack_dms_GetCustFeature_t::DHCPRelayEnabled

8.642.2.2 uint8_t unpack_dms_GetCustFeature_t::DisableIMSI

8.642.2.3 uint32_t unpack_dms_GetCustFeature_t::GpsEnable

8.642.2.4 uint8_t unpack_dms_GetCustFeature_t::GPSLPM

8.642.2.5 uint8_t unpack_dms_GetCustFeature_t::GPSSel

8.642.2.6 uint16_t unpack_dms_GetCustFeature_t::IPFamSupport

8.642.2.7 uint8_t unpack_dms_GetCustFeature_t::IsVoiceEnabled

8.642.2.8 swi_uint256_t unpack_dms_GetCustFeature_t::ParamPresenceMask

8.642.2.9 uint8_t unpack_dms_GetCustFeature_t::RMAutoConnect

8.642.2.10 uint8_t unpack_dms_GetCustFeature_t::SMSSupport

8.642.2.11 uint16_t unpack_dms_GetCustFeature_t::Tlvresult

8.643 unpack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- DMSgetCustomFeatureV2 GetCustomFeatureV2
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.643.1 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.

8.643.2 Field Documentation

8.643.2.1 DMSgetCustomFeatureV2 unpack_dms_GetCustFeaturesV2_t::GetCustomFeatureV2

8.643.2.2 swi_uint256_t unpack_dms_GetCustFeaturesV2_t::ParamPresenceMask

8.643.2.3 uint16_t unpack_dms_GetCustFeaturesV2_t::Tlvresult

8.644 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 [64]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.644.1 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.

8.644.2 Field Documentation

8.644.2.1 uint32_t unpack_dms_GetDeviceCap_t::DataServiceCapability

8.644.2.2 uint32_t unpack_dms_GetDeviceCap_t::MaxRXChannelRate

8.644.2.3 uint32_t unpack_dms_GetDeviceCap_t::MaxTXChannelRate

8.644.2.4 swi_uint256_t unpack_dms_GetDeviceCap_t::ParamPresenceMask

8.644.2.5 uint8_t unpack_dms_GetDeviceCap_t::RadioIfaces[64]

8.644.2.6 uint32_t unpack_dms_GetDeviceCap_t::RadioIfacesSize

8.644.2.7 uint32_t unpack_dms_GetDeviceCap_t::SimCapability

8.644.2.8 uint16_t unpack_dms_GetDeviceCap_t::Tlvresult

8.645 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

8.645.1 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.

8.645.2 Field Documentation

8.645.2.1 uint32_t unpack_dms_GetDeviceCapabilities_t::dataServiceCaCapability

8.645.2.2 uint32_t unpack_dms_GetDeviceCapabilities_t::maxRxChannelRate

8.645.2.3 uint32_t unpack_dms_GetDeviceCapabilities_t::maxTxChannelRate

8.645.2.4 swi_uint256_t unpack_dms_GetDeviceCapabilities_t::ParamPresenceMask

8.645.2.5 uint8_t unpack_dms_GetDeviceCapabilities_t::RadioIfaces[255]

8.645.2.6 uint32_t unpack_dms_GetDeviceCapabilities_t::radioIfacesSize

8.645.2.7 uint32_t unpack_dms_GetDeviceCapabilities_t::simCapability

8.646 unpack_dms_GetDeviceHardwareRev_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.646.1 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
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.646.2 Field Documentation

8.646.2.1 swi_uint256_t unpack_dms_GetDeviceHardwareRev_t::ParamPresenceMask

8.646.2.2 char unpack_dms_GetDeviceHardwareRev_t::String[255]

8.646.2.3 uint8_t unpack_dms_GetDeviceHardwareRev_t::stringSize

8.646.2.4 uint16_t unpack_dms_GetDeviceHardwareRev_t::Tlvresult

8.647 unpack_dms_GetDeviceMfr_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.647.1 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.

8.647.2 Field Documentation

8.647.2.1 swi_uint256_t unpack_dms_GetDeviceMfr_t::ParamPresenceMask

8.647.2.2 char unpack_dms_GetDeviceMfr_t::String[255]

8.647.2.3 uint8_t unpack_dms_GetDeviceMfr_t::stringSize

8.647.2.4 uint16_t unpack_dms_GetDeviceMfr_t::Tlvresult

8.648 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

8.648.1 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.

8.648.2 Field Documentation

8.648.2.1 uint8_t unpack_dms_GetDeviceSerialNumbers_t::esnSize

8.648.2.2 char unpack_dms_GetDeviceSerialNumbers_t::ESNString[255]

8.648.2.3 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSize

8.648.2.4 char unpack_dms_GetDeviceSerialNumbers_t::IMEIString[255]

8.648.2.5 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSvnSize

8.648.2.6 char unpack_dms_GetDeviceSerialNumbers_t::ImeiSvnString[255]

8.648.2.7 uint8_t unpack_dms_GetDeviceSerialNumbers_t::meidSize

8.648.2.8 char unpack_dms_GetDeviceSerialNumbers_t::MEIDString[255]

8.648.2.9 swi_uint256_t unpack_dms_GetDeviceSerialNumbers_t::ParamPresenceMask

8.648.2.10 uint16_t unpack_dms_GetDeviceSerialNumbers_t::Tlvresult

8.649 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

8.649.1 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 EM/MC74xx(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.

8.649.2 Field Documentation

8.649.2.1 char unpack_dms_GetFirmwareInfo_t::appversion_str[85]

8.649.2.2 char unpack_dms_GetFirmwareInfo_t::bootversion_str[85]

8.649.2.3 char unpack_dms_GetFirmwareInfo_t::carrier_str[20]

8.649.2.4 char unpack_dms_GetFirmwareInfo_t::cur_carr_name[17]

8.649.2.5 char unpack_dms_GetFirmwareInfo_t::cur_carr_rev[13]

8.649.2.6 char unpack_dms_GetFirmwareInfo_t::modelid_str[20]

8.649.2.7 char unpack_dms_GetFirmwareInfo_t::packageid_str[85]

8.649.2.8 swi_uint256_t unpack_dms_GetFirmwareInfo_t::ParamPresenceMask

8.649.2.9 char unpack_dms_GetFirmwareInfo_t::priversion_str[16]

8.649.2.10 char unpack_dms_GetFirmwareInfo_t::sku_str[15]

8.649.2.11 uint16_t unpack_dms_GetFirmwareInfo_t::Tlvresult

8.650 unpack_dms_GetFirmwareRevision_t Struct Reference

Data Fields

- uint8_t amssSize
- char AMSSString [255]
- char PRIString [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.650.1 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.

8.650.2 Field Documentation

8.650.2.1 uint8_t unpack_dms_GetFirmwareRevision_t::amssSize

8.650.2.2 char unpack_dms_GetFirmwareRevision_t::AMSSString[255]

8.650.2.3 swi_uint256_t unpack_dms_GetFirmwareRevision_t::ParamPresenceMask

8.650.2.4 char unpack_dms_GetFirmwareRevision_t::PRIString[255]

8.650.2.5 uint16_t unpack_dms_GetFirmwareRevision_t::Tlvresult

8.651 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

8.651.1 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.

8.651.2 Field Documentation

8.651.2.1 uint8_t unpack_dms_GetFirmwareRevisions_t::amssSize

8.651.2.2 char unpack_dms_GetFirmwareRevisions_t::AMSSString[255]

8.651.2.3 uint8_t unpack_dms_GetFirmwareRevisions_t::bootSize

8.651.2.4 char unpack_dms_GetFirmwareRevisions_t::BootString[255]

8.651.2.5 swi_uint256_t unpack_dms_GetFirmwareRevisions_t::ParamPresenceMask

8.651.2.6 uint8_t unpack_dms_GetFirmwareRevisions_t::priSize

8.651.2.7 char unpack_dms_GetFirmwareRevisions_t::PRIString[255]

8.651.2.8 uint16_t unpack_dms_GetFirmwareRevisions_t::Tlvresult

8.652 unpack_dms_GetFSN_t Struct Reference

Data Fields

- char String [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.652.1 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.

8.652.2 Field Documentation

8.652.2.1 swi_uint256_t unpack_dms_GetFSN_t::ParamPresenceMask

8.652.2.2 char unpack_dms_GetFSN_t::String[255]

8.652.2.3 uint16_t unpack_dms_GetFSN_t::Tlvresult

8.653 unpack_dms_GetHardwareRevision_t Struct Reference

Data Fields

- char hwVer [255]
- swi_uint256_t ParamPresenceMask

8.653.1 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.

8.653.2 Field Documentation

8.653.2.1 char unpack_dms_GetHardwareRevision_t::hwVer[255]

8.653.2.2 swi_uint256_t unpack_dms_GetHardwareRevision_t::ParamPresenceMask

8.654 unpack_dms_GetIMSI_t Struct Reference

Data Fields

- char imsi [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.654.1 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.

8.654.2 Field Documentation

8.654.2.1 char unpack_dms_GetIMSI_t::imsi[255]

8.654.2.2 swi_uint256_t unpack_dms_GetIMSI_t::ParamPresenceMask

8.654.2.3 uint16_t unpack_dms_GetIMSI_t::Tlvresult

8.655 unpack_dms_GetManufacturer_t Struct Reference

Data Fields

- char manufacturer [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.655.1 Detailed Description

This structure is used to store device manufacturer information.

Parameters

manufacturer[O-UT]	• NULL terminated string • Bit to check in ParamPresenceMask - **1**
Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.655.2 Field Documentation

8.655.2.1 char unpack_dms_GetManufacturer_t::manufacturer[255]

8.655.2.2 swi_uint256_t unpack_dms_GetManufacturer_t::ParamPresenceMask

8.655.2.3 uint16_t unpack_dms_GetManufacturer_t::Tlvresult

8.656 unpack_dms_GetModelID_t Struct Reference

Data Fields

- char modelid [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.656.1 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.

8.656.2 Field Documentation

8.656.2.1 char unpack_dms_GetModelID_t::modelid[255]

8.656.2.2 swi_uint256_t unpack_dms_GetModelID_t::ParamPresenceMask

8.656.2.3 uint16_t unpack_dms_GetModelID_t::Tlvresult

8.657 unpack_dms_GetNetworkTime_t Struct Reference

Data Fields

- uint16_t source
- uint64_t timestamp
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.657.1 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.

8.657.2 Field Documentation

8.657.2.1 swi_uint256_t unpack_dms_GetNetworkTime_t::ParamPresenceMask

8.657.2.2 uint16_t unpack_dms_GetNetworkTime_t::source

8.657.2.3 uint64_t unpack_dms_GetNetworkTime_t::timestamp

8.657.2.4 uint16_t unpack_dms_GetNetworkTime_t::TlvResult

8.658 unpack_dms_GetOfflineReason_t Struct Reference

Data Fields

- uint32_t * pReasonMask
- uint32_t * pbPlatform
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.658.1 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.

8.658.2 Field Documentation

8.658.2.1 swi_uint256_t unpack_dms_GetOfflineReason_t::ParamPresenceMask

8.658.2.2 uint32_t* unpack_dms_GetOfflineReason_t::pbPlatform

8.658.2.3 uint32_t* unpack_dms_GetOfflineReason_t::pReasonMask

8.658.2.4 uint16_t unpack_dms_GetOfflineReason_t::Tlvresult

8.659 unpack_dms_GetPower_t Struct Reference

Data Fields

- uint32_t OperationMode
- uint32_t OfflineReason
- uint32_t HardwareControlledMode
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.659.1 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.

8.659.2 Field Documentation

8.659.2.1 uint32_t unpack_dms_GetPower_t::HardwareControlledMode

8.659.2.2 uint32_t unpack_dms_GetPower_t::OfflineReason

8.659.2.3 uint32_t unpack_dms_GetPower_t::OperationMode

8.659.2.4 swi_uint256_t unpack_dms_GetPower_t::ParamPresenceMask

8.659.2.5 uint16_t unpack_dms_GetPower_t::Tlvresult

8.660 unpack_dms_GetPRLVersion_t Struct Reference

Data Fields

- uint8_t u8PRLPreference
- uint16_t u16PRLVersion
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.660.1 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.

8.660.2 Field Documentation

8.660.2.1 swi_uint256_t unpack_dms_GetPRLVersion_t::ParamPresenceMask

8.660.2.2 uint16_t unpack_dms_GetPRLVersion_t::Tlvresult

8.660.2.3 uint16_t unpack_dms_GetPRLVersion_t::u16PRLVersion

8.660.2.4 uint8_t unpack_dms_GetPRLVersion_t::u8PRLPreference

8.661 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

8.661.1 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.

8.661.2 Field Documentation

8.661.2.1 char unpack_dms_GetSerialNumbers_t::esn[255]

8.661.2.2 char unpack_dms_GetSerialNumbers_t::imei_no[255]

8.661.2.3 char unpack_dms_GetSerialNumbers_t::imeisv_svn[255]

8.661.2.4 char unpack_dms_GetSerialNumbers_t::meid[255]

8.661.2.5 swi_uint256_t unpack_dms_GetSerialNumbers_t::ParamPresenceMask

8.662 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

8.662.1 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.

8.662.2 Field Documentation

8.662.2.1 uint8_t unpack_dms_GetUSBComp_t::NumSupUSBComps

8.662.2.2 swi_uint256_t unpack_dms_GetUSBComp_t::ParamPresenceMask

8.662.2.3 uint8_t unpack_dms_GetUSBComp_t::SupUSBComps[255]

8.662.2.4 uint16_t unpack_dms_GetUSBComp_t::Tlvresult

8.662.2.5 uint8_t unpack_dms_GetUSBComp_t::USBComp

8.663 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

8.663.1 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.

8.663.2 Field Documentation

8.663.2.1 char unpack_dms_GetVoiceNumber_t::MIN[255]

8.663.2.2 uint8_t unpack_dms_GetVoiceNumber_t::minSize

8.663.2.3 swi_uint256_t unpack_dms_GetVoiceNumber_t::ParamPresenceMask

8.663.2.4 uint16_t unpack_dms_GetVoiceNumber_t::Tlvresult

8.663.2.5 char unpack_dms_GetVoiceNumber_t::VoiceNumber[255]

8.663.2.6 uint8_t unpack_dms_GetVoiceNumber_t::voiceNumberSize

8.664 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

8.664.1 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.

8.664.2 Field Documentation

8.664.2.1 dms_PSMActiveTimerIndTlv unpack_dms_PSMCfgChange_ind_t::ActiveTimer

8.664.2.2 dms_PSMEnableStateIndTlv unpack_dms_PSMCfgChange_ind_t::EnableState

8.664.2.3 swi_uint256_t unpack_dms_PSMCfgChange_ind_t::ParamPresenceMask

8.664.2.4 dms_PSMPeriodicUpdateTimerIndTlv unpack_dms_PSMCfgChange_ind_t::PeriodicUpdateTimer

8.664.2.5 uint16_t unpack_dms_PSMCfgChange_ind_t::Tlvresult

8.665 unpack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.665.1 Detailed Description

This structure contains reset to factory default unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.665.2 Field Documentation

8.665.2.1 swi_uint256_t unpack_dms_ResetToFactoryDefaults_t::ParamPresenceMask

8.665.2.2 uint16_t unpack_dms_ResetToFactoryDefaults_t::Tlvresult

8.666 unpack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.666.1 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.

8.666.2 Field Documentation

8.666.2.1 swi_uint256_t unpack_dms_SetActivationStatusCallback_t::ParamPresenceMask

8.666.2.2 uint16_t unpack_dms_SetActivationStatusCallback_t::Tlvresult

8.667 unpack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t notused
- swi_uint256_t ParamPresenceMask

8.667.1 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.

8.667.2 Field Documentation

8.667.2.1 uint8_t unpack_dms_SetCrashAction_t::notused

8.667.2.2 swi_uint256_t unpack_dms_SetCrashAction_t::ParamPresenceMask

8.668 unpack_dms_SetCustFeature_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.668.1 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.

8.668.2 Field Documentation

8.668.2.1 swi_uint256_t unpack_dms_SetCustFeature_t::ParamPresenceMask

8.668.2.2 uint16_t unpack_dms_SetCustFeature_t::Tlvresult

8.669 unpack_dms_SetCustFeaturesV2_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.669.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.669.2 Field Documentation

8.669.2.1 swi_uint256_t unpack_dms_SetCustFeaturesV2_t::ParamPresenceMask

8.669.2.2 uint16_t unpack_dms_SetCustFeaturesV2_t::Tlvresult

8.670 unpack_dms_SetEventReport_ind_t Struct Reference

Data Fields

- dms_ActivationStatusTlv ActivationStatusTlv
- dms_OperatingModeTlv OperatingModeTlv
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.670.1 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.

8.670.2 Field Documentation

8.670.2.1 dms_ActivationStatusTlv unpack_dms_SetEventReport_ind_t::ActivationStatusTlv

8.670.2.2 dms_OperatingModeTlv unpack_dms_SetEventReport_ind_t::OperatingModeTlv

8.670.2.3 swi_uint256_t unpack_dms_SetEventReport_ind_t::ParamPresenceMask

8.670.2.4 uint16_t unpack_dms_SetEventReport_ind_t::Tlvresult

8.671 unpack_dms_SetEventReport_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.671.1 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.

8.671.2 Field Documentation

8.671.2.1 swi_uint256_t unpack_dms_SetEventReport_t::ParamPresenceMask

8.671.2.2 uint16_t unpack_dms_SetEventReport_t::Tlvresult

8.672 unpack_dms_SetFirmwarePreference_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.672.1 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.

8.672.2 Field Documentation

8.672.2.1 swi_uint256_t unpack_dms_SetFirmwarePreference_t::ParamPresenceMask

8.672.2.2 uint16_t unpack_dms_SetFirmwarePreference_t::Tlvresult

8.673 unpack_dms_SetIndicationRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.673.1 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.

8.673.2 Field Documentation

8.673.2.1 swi_uint256_t unpack_dms_SetIndicationRegister_t::ParamPresenceMask

8.673.2.2 uint16_t unpack_dms_SetIndicationRegister_t::Tlvresult

8.674 unpack_dms_SetPower_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.674.1 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.

8.674.2 Field Documentation

8.674.2.1 swi_uint256_t unpack_dms_SetPower_t::ParamPresenceMask

8.674.2.2 uint16_t unpack_dms_SetPower_t::Tlvresult

8.675 unpack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.675.1 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.

8.675.2 Field Documentation

8.675.2.1 swi_uint256_t unpack_dms_SetUSBComp_t::ParamPresenceMask

8.675.2.2 uint16_t unpack_dms_SetUSBComp_t::Tlvresult

8.676 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

8.676.1 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.

8.676.2 Field Documentation

8.676.2.1 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_LpmFlag

8.676.2.2 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_PersistentLpm

8.676.2.3 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_PowerOffMode

8.676.2.4 int unpack_dms_SLQSDmsSwiGetPCInfo_t::has_Wdisable

8.676.2.5 uint32_t unpack_dms_SLQSDmsSwiGetPCInfo_t::LpmFlag

8.676.2.6 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::opMode

8.676.2.7 swi_uint256_t unpack_dms_SLQSDmsSwiGetPCInfo_t::ParamPresenceMask

8.676.2.8 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::PersistentLpm

8.676.2.9 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::PowerOffMode

8.676.2.10 uint8_t unpack_dms_SLQSDmsSwiGetPCInfo_t::Wdisable

8.677 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.677.1 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.

8.677.2 Field Documentation

8.677.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::ParamPresenceMask

8.677.2.2 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::source

8.677.2.3 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::Tlvresult

8.677.2.4 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::type

8.678 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.678.1 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.

8.678.2 Field Documentation

8.678.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiGetResetInfo_t::ParamPresenceMask

8.678.2.2 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::source

8.678.2.3 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_t::Tlvresult

8.678.2.4 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::type

8.679 unpack_dms_SLQSDmsSwiGetUimSelection_t Struct Reference

Data Fields

- uint8_t uimSelect
- dms_UimAutoSwitchActSlotTlv * pUimAutoSwitchActSlot
- swi_uint256_t ParamPresenceMask

8.679.1 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

8.679.2 Field Documentation

8.679.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiGetUimSelection_t::ParamPresenceMask

8.679.2.2 dms_UimAutoSwitchActSlotTlv* unpack_dms_SLQSDmsSwiGetUimSelection_t::pUimAutoSwitchActSlot

8.679.2.3 uint8_t unpack_dms_SLQSDmsSwiGetUimSelection_t::uimSelect

8.680 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.680.1 Detailed Description

This structure contains set registration state for different indication unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.680.2 Field Documentation

8.680.2.1 swi_uint256_t unpack_dms_SLQSDmsSwiIndicationRegister_t::ParamPresenceMask

8.680.2.2 uint16_t unpack_dms_SLQSDmsSwiIndicationRegister_t::Tlvresult

8.681 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

8.681.1 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.

8.681.2 Field Documentation

8.681.2.1 uint64_t unpack_dms_SLQSGetBandCapability_t::bandCapability

8.681.2.2 int unpack_dms_SLQSGetBandCapability_t::is_LteBandCapability_Available

8.681.2.3 int unpack_dms_SLQSGetBandCapability_t::is_TdsBandCapability_Available

8.681.2.4 uint64_t unpack_dms_SLQSGetBandCapability_t::LteBandCapability

8.681.2.5 swi_uint256_t unpack_dms_SLQSGetBandCapability_t::ParamPresenceMask

8.681.2.6 uint64_t unpack_dms_SLQSGetBandCapability_t::TdsBandCapability

8.682 unpack_dms_SLQSGetERIFile_t Struct Reference

Data Fields

- eriDataparams eriFile
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.682.1 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.

8.682.2 Field Documentation

8.682.2.1 eriDataparams unpack_dms_SLQSGetERIFile_t::eriFile

8.682.2.2 swi_uint256_t unpack_dms_SLQSGetERIFile_t::ParamPresenceMask

8.682.2.3 uint16_t unpack_dms_SLQSGetERIFile_t::Tlvresult

8.683 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

8.683.1 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.

8.683.2 Field Documentation

8.683.2.1 dms_PSMActiveTimerTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pActiveTimer

8.683.2.2 swi_uint256_t unpack_dms_SLQSGetPowerSaveModeConfig_t::ParamPresenceMask

8.683.2.3 dms_PSMDurationDueToOOSTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pDurationDueToOOS

8.683.2.4 dms_PSMDurationThresholdTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pDurationThreshold

8.683.2.5 dms_PSMEarlyWakeupTimeTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pEarlyWakeupTime

8.683.2.6 dms_PSMPeriodicUpdateTimerTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pPeriodicUpdateTimer

8.683.2.7 dms_PSMEnableStateTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pPsmEnableState

8.683.2.8 dms_PSMRandomizationWindowTlv* unpack_dms_SLQSGetPowerSaveModeConfig_t::pRandomization-Window

8.684 unpack_dms_SLQSSetPowerSaveModeConfig_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.684.1 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.

8.684.2 Field Documentation

8.684.2.1 swi_uint256_t unpack_dms_SLQSSetPowerSaveModeConfig_t::ParamPresenceMask

8.684.2.2 uint16_t unpack_dms_SLQSSetPowerSaveModeConfig_t::Tlvresult

8.685 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.685.1 Detailed Description

This structure contains Clear Dying GASP unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.685.2 Field Documentation

8.685.2.1 swi_uint256_t unpack_dms_SLQSSwiClearDyingGaspStatistics_t::ParamPresenceMask

8.685.2.2 uint16_t unpack_dms_SLQSSwiClearDyingGaspStatistics_t::Tlvresult

8.686 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult
- crashInfoParams crashInfoParam
- swi_uint256_t ParamPresenceMask

8.686.1 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.

8.686.2 Field Documentation

8.686.2.1 crashInfoParams unpack_dms_SLQSSwiGetCrashInfo_t::crashInfoParam

8.686.2.2 swi_uint256_t unpack_dms_SLQSSwiGetCrashInfo_t::ParamPresenceMask

8.686.2.3 uint16_t unpack_dms_SLQSSwiGetCrashInfo_t::TlvResult

8.687 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference

Data Fields

- packgetDyingGaspCfg * pGetDyingGaspCfg
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.687.1 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.

8.687.2 Field Documentation

8.687.2.1 swi_uint256_t unpack_dms_SLQSSwiGetDyingGaspCfg_t::ParamPresenceMask

8.687.2.2 packgetDyingGaspCfg* unpack_dms_SLQSSwiGetDyingGaspCfg_t::pGetDyingGaspCfg

8.687.2.3 uint16_t unpack_dms_SLQSSwiGetDyingGaspCfg_t::Tlvresult

8.688 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference

Data Fields

- packgetDyingGaspStatistics * pGetDyingGaspStatistics
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.688.1 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

8.688.2 Field Documentation

8.688.2.1 swi_uint256_t unpack_dms_SLQSSwiGetDyingGaspStatistics_t::ParamPresenceMask

8.688.2.2 packgetDyingGaspStatistics* unpack_dms_SLQSSwiGetDyingGaspStatistics_t::pGetDyingGaspStatistics

8.688.2.3 uint16_t unpack_dms_SLQSSwiGetDyingGaspStatistics_t::Tlvresult

8.689 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

8.689.1 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.

8.689.2 Field Documentation

8.689.2.1 char unpack_dms_SLQSSwiGetFirmwareCurr_t::carrier[16]

8.689.2.2 char unpack_dms_SLQSSwiGetFirmwareCurr_t::fwvers[16]

8.689.2.3 uint8_t unpack_dms_SLQSSwiGetFirmwareCurr_t::numEntries

8.689.2.4 swi_uint256_t unpack_dms_SLQSSwiGetFirmwareCurr_t::ParamPresenceMask

8.689.2.5 image_info_t* unpack_dms_SLQSSwiGetFirmwareCurr_t::pCurrImgInfo

8.689.2.6 char unpack_dms_SLQSSwiGetFirmwareCurr_t::pkgver[16]

8.689.2.7 char unpack_dms_SLQSSwiGetFirmwareCurr_t::priver[16]

8.690 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

8.690.1 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.

8.690.2 Field Documentation

8.690.2.1 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::imgType

8.690.2.2 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::logString[255]

8.690.2.3 swi_uint256_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::ParamPresenceMask

8.690.2.4 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refData

8.690.2.5 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refString[15]

8.690.2.6 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::ResCode

8.690.2.7 uint16_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::TlvResult

8.691 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

8.691.1 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.

8.691.2 Field Documentation

8.691.2.1 char unpack_dms_SLQSSwiGetHostDevInfo_t::hostID[255]

8.691.2.2 char unpack_dms_SLQSSwiGetHostDevInfo_t::manString[255]

8.691.2.3 char unpack_dms_SLQSSwiGetHostDevInfo_t::modelString[255]

8.691.2.4 swi_uint256_t unpack_dms_SLQSSwiGetHostDevInfo_t::ParamPresenceMask

8.691.2.5 char unpack_dms_SLQSSwiGetHostDevInfo_t::plasmaIDString[255]

8.691.2.6 char unpack_dms_SLQSSwiGetHostDevInfo_t::swVerString[255]

8.691.2.7 uint16_t unpack_dms_SLQSSwiGetHostDevInfo_t::TlvResult

8.692 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference

Data Fields

- uint16_t TlvResult
- char nameString [255]
- char versionString [255]

- swi_uint256_t ParamPresenceMask

8.692.1 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.

8.692.2 Field Documentation

8.692.2.1 char unpack_dms_SLQSSwiGetOSInfo_t::nameString[255]

8.692.2.2 swi_uint256_t unpack_dms_SLQSSwiGetOSInfo_t::ParamPresenceMask

8.692.2.3 uint16_t unpack_dms_SLQSSwiGetOSInfo_t::TlvResult

8.692.2.4 char unpack_dms_SLQSSwiGetOSInfo_t::versionString[255]

8.693 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference

Data Fields

- uint16_t TlvResult
- char meidString [8]
- swi_uint256_t ParamPresenceMask

8.693.1 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.

8.693.2 Field Documentation

8.693.2.1 char unpack_dms_SLQSSwiGetSerialNoExt_t::meidString[8]

8.693.2.2 swi_uint256_t unpack_dms_SLQSSwiGetSerialNoExt_t::ParamPresenceMask

8.693.2.3 uint16_t unpack_dms_SLQSSwiGetSerialNoExt_t::Tlvresult

8.694 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.694.1 Detailed Description

This structure contains set Dying GASP Config unpack

Parameters

Tlvresult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.694.2 Field Documentation

8.694.2.1 swi_uint256_t unpack_dms_SLQSSwiSetDyingGaspCfg_t::ParamPresenceMask

8.694.2.2 uint16_t unpack_dms_SLQSSwiSetDyingGaspCfg_t::Tlvresult

8.695 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.695.1 Detailed Description

This structure contains SWI set host dev info unpack

Parameters

TlvresultParamPresence-Mask	• Unpack Result • bitmask representation to indicate valid parameters.

8.695.2 Field Documentation

8.695.2.1 swi_uint256_t unpack_dms_SLQSSwiSetHostDevInfo_t::ParamPresenceMask

8.695.2.2 uint16_t unpack_dms_SLQSSwiSetHostDevInfo_t::Tlvresult

8.696 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.696.1 Detailed Description

This structure contains SWI set host OS info unpack

Parameters

TlvresultParamPresence-Mask	• Unpack Result • bitmask representation to indicate valid parameters.

8.696.2 Field Documentation

8.696.2.1 swi_uint256_t unpack_dms_SLQSSwiSetOSInfo_t::ParamPresenceMask

8.696.2.2 uint16_t unpack_dms_SLQSSwiSetOSInfo_t::Tlvresult

8.697 unpack_dms_SLQSUIMGetState_t Struct Reference

Data Fields

- uint8_t state
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.697.1 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.

8.697.2 Field Documentation

8.697.2.1 swi_uint256_t unpack_dms_SLQSUIMGetState_t::ParamPresenceMask

8.697.2.2 uint8_t unpack_dms_SLQSUIMGetState_t::state

8.697.2.3 uint16_t unpack_dms_SLQSUIMGetState_t::Tlvresult

8.698 unpack_dms_SwiEventReportCallBack_ind_t Struct Reference

Data Fields

- dms_TemperatureTlv TempTlv
- dms_VoltageTlv VoltTlv
- dms_UimStatusTlv UimStatusTlv
- swi_uint256_t ParamPresenceMask

8.698.1 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

8.698.2 Field Documentation

8.698.2.1 swi_uint256_t unpack_dms_SwiEventReportCallBack_ind_t::ParamPresenceMask

8.698.2.2 dms_TemperatureTlv unpack_dms_SwiEventReportCallBack_ind_t::TempTlv

8.698.2.3 dms_UimStatusTlv unpack_dms_SwiEventReportCallBack_ind_t::UimStatusTlv

8.698.2.4 dms_VoltageTlv unpack_dms_SwiEventReportCallBack_ind_t::VoltTlv

8.699 unpack_dms_SwiSetEventReport_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.699.1 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.

8.699.2 Field Documentation

8.699.2.1 swi_uint256_t unpack_dms_SwiSetEventReport_t::ParamPresenceMask

8.699.2.2 uint16_t unpack_dms_SwiSetEventReport_t::Tlvresult

8.700 unpack_dms_SwiUimSelect_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.700.1 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.

8.700.2 Field Documentation

8.700.2.1 swi_uint256_t unpack_dms_SwiUimSelect_t::ParamPresenceMask

8.700.2.2 uint16_t unpack_dms_SwiUimSelect_t::TlvResult

8.701 unpack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facilityState
- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.701.1 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.

8.701.2 Field Documentation

8.701.2.1 uint8_t unpack_dms_UIMGetControlKeyStatus_t::facilityState

8.701.2.2 swi_uint256_t unpack_dms_UIMGetControlKeyStatus_t::ParamPresenceMask

8.701.2.3 uint16_t unpack_dms_UIMGetControlKeyStatus_t::Tlvresult

8.701.2.4 uint8_t unpack_dms_UIMGetControlKeyStatus_t::unblockRetriesLeft

8.701.2.5 uint8_t unpack_dms_UIMGetControlKeyStatus_t::verifyRetriesLeft

8.702 unpack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint8_t stringSize
- uint8_t String [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.702.1 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.

8.702.2 Field Documentation

8.702.2.1 swi_uint256_t unpack_dms_UIMGetICCID_t::ParamPresenceMask

8.702.2.2 uint8_t unpack_dms_UIMGetICCID_t::String[255]

8.702.2.3 uint8_t unpack_dms_UIMGetICCID_t::stringSize

8.702.2.4 uint16_t unpack_dms_UIMGetICCID_t::Tlvresult

8.703 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

8.703.1 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.

8.703.2 Field Documentation

8.703.2.1 uint8_t unpack_dms_UIMGetPINStatus_t::p1Status

8.703.2.2 uint8_t unpack_dms_UIMGetPINStatus_t::p1UnblockRetriesLeft

8.703.2.3 uint8_t unpack_dms_UIMGetPINStatus_t::p1VerifyRetriesLeft

8.703.2.4 uint8_t unpack_dms_UIMGetPINStatus_t::p2Status

8.703.2.5 uint8_t unpack_dms_UIMGetPINStatus_t::p2UnblockRetriesLeft

8.703.2.6 uint8_t unpack_dms_UIMGetPINStatus_t::p2VerifyRetriesLeft

8.703.2.7 swi_uint256_t unpack_dms_UIMGetPINStatus_t::ParamPresenceMask

8.703.2.8 uint16_t unpack_dms_UIMGetPINStatus_t::Tlvresult

8.704 unpack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.704.1 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.

8.704.2 Field Documentation

8.704.2.1 swi_uint256_t unpack_dms_UIMSetControlKeyProtection_t::ParamPresenceMask

8.704.2.2 uint16_t unpack_dms_UIMSetControlKeyProtection_t::Tlvresult

8.704.2.3 uint8_t unpack_dms_UIMSetControlKeyProtection_t::verifyRetriesLeft

8.705 unpack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.705.1 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.

8.705.2 Field Documentation

8.705.2.1 swi_uint256_t unpack_dms_UIMSetPINProtection_t::ParamPresenceMask

8.705.2.2 uint16_t unpack_dms_UIMSetPINProtection_t::Tlvresult

8.705.2.3 uint8_t unpack_dms_UIMSetPINProtection_t::unblockRetriesLeft

8.705.2.4 uint8_t unpack_dms_UIMSetPINProtection_t::verifyRetriesLeft

8.706 unpack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t unblockRetriesLeft
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.706.1 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.

8.706.2 Field Documentation

8.706.2.1 swi_uint256_t unpack_dms_UIMUnblockControlKey_t::ParamPresenceMask

8.706.2.2 uint16_t unpack_dms_UIMUnblockControlKey_t::Tlvresult

8.706.2.3 uint8_t unpack_dms_UIMUnblockControlKey_t::unblockRetriesLeft

8.707 unpack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageListSize
- FMSPrefImageList * pImageList
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.707.1 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

8.707.2 Field Documentation

8.707.2.1 uint32_t unpack_fms_GetImagesPreference_t::ImageListSize

8.707.2.2 swi_uint256_t unpack_fms_GetImagesPreference_t::ParamPresenceMask

8.707.2.3 FMSPrefImageList* unpack_fms_GetImagesPreference_t::pImageList

8.707.2.4 uint16_t unpack_fms_GetImagesPreference_t::Tlvresult

8.708 unpack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint32_t imagelistSize
- FMSImageList imageList
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.708.1 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

8.708.2 Field Documentation

8.708.2.1 FMSImageList unpack_fms_GetStoredImages_t::imageList

8.708.2.2 uint32_t unpack_fms_GetStoredImages_t::imagelistSize

8.708.2.3 swi_uint256_t unpack_fms_GetStoredImages_t::ParamPresenceMask

8.708.2.4 uint16_t unpack_fms_GetStoredImages_t::Tlvresult

8.709 unpack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageTypesSize
- uint8_t ImageTypes [255]
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.709.1 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

8.709.2 Field Documentation

8.709.2.1 uint8_t unpack_fms_SetImagesPreference_t::ImageTypes[255]

8.709.2.2 uint32_t unpack_fms_SetImagesPreference_t::ImageTypesSize

8.709.2.3 swi_uint256_t unpack_fms_SetImagesPreference_t::ParamPresenceMask

8.709.2.4 uint16_t unpack_fms_SetImagesPreference_t::Tlvresult

8.710 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

8.710.1 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**
pSMSOverIPNwInd	• SMS over IP Network Indication Flag – TRUE - Turn on mobile-originated SMS – FALSE - Turn off mobile-originated SMS • Bit to check in ParamPresenceMask - **18**
pPhoneCtxtURILen[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**

8.710.2 Field Documentation

8.710.2.1 swi_uint256_t unpack_ims_SLQSGetIMSSMSConfig_t::ParamPresenceMask

8.710.2.2 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pPhoneCtxtURI

8.710.2.3 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pPhoneCtxtURILen

8.710.2.4 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pSettingResp

8.710.2.5 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pSMSFormat

8.710.2.6 uint8_t* unpack_ims_SLQSGetIMSSMSConfig_t::pSMSOverIPNwInd

8.711 unpack_ims_SLQSGetIMSUserConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- uint8_t * pIMSDomainLen
- uint8_t * pIMSDomain
- swi_uint256_t ParamPresenceMask

8.711.1 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**

8.711.2 Field Documentation

8.711.2.1 swi_uint256_t unpack_ims_SLQSGetIMSUserConfig_t::ParamPresenceMask

8.711.2.2 uint8_t* unpack_ims_SLQSGetIMSUserConfig_t::pIMSDomain

8.711.2.3 uint8_t* unpack_ims_SLQSGetIMSUserConfig_t::pIMSDomainLen

8.711.2.4 uint8_t* unpack_ims_SLQSGetIMSUserConfig_t::pSettingResp

8.712 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

8.712.1 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**

8.712.2 Field Documentation

8.712.2.1 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrMode

8.712.2.2 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrOctetAligned

8.712.2.3 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrWbEnable

8.712.2.4 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrWBMode

8.712.2.5 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pAmrWBOctetAligned

8.712.2.6 swi_uint256_t unpack_ims_SLQSGetIMSVoIPConfig_t::ParamPresenceMask

8.712.2.7 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pMinSessionExpiryTimer

8.712.2.8 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pRingBackTimer

8.712.2.9 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pRingingTimer

8.712.2.10 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pRTPRTCPInactTimer

8.712.2.11 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pScrAmrEnable

8.712.2.12 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pScrAmrWbEnable

8.712.2.13 uint16_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pSessionExpiryTimer

8.712.2.14 uint8_t* unpack_ims_SLQSGetIMSVoIPConfig_t::pSettingResp

8.713 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

8.713.1 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.

8.713.2 Field Documentation

8.713.2.1 swi_uint256_t unpack_ims_SLQSGetRegMgrConfig_t::ParamPresenceMask

8.713.2.2 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pIMSTestMode

8.713.2.3 uint16_t* unpack_ims_SLQSGetRegMgrConfig_t::pPCSCFPort

8.713.2.4 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pPriCSCFPortName

8.713.2.5 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pPriCSCFPortNameLen

8.713.2.6 uint8_t* unpack_ims_SLQSGetRegMgrConfig_t::pSettingResp

8.714 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

8.714.1 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**

8.714.2 Field Documentation

8.714.2.1 swi_uint256_t unpack_ims_SLQSGetSIPConfig_t::ParamPresenceMask

8.714.2.2 uint8_t* unpack_ims_SLQSGetSIPConfig_t::pSettingResp

8.714.2.3 uint8_t* unpack_ims_SLQSGetSIPConfig_t::pSigCompEnabled

8.714.2.4 uint16_t* unpack_ims_SLQSGetSIPConfig_t::pSIPLocalPort

8.714.2.5 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pSubscribeTimer

8.714.2.6 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerSIPReg

8.714.2.7 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerT1

8.714.2.8 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerT2

8.714.2.9 uint32_t* unpack_ims_SLQSGetSIPConfig_t::pTimerTf

8.715 unpack_ims_SLQSRegMgrCfgCallBack_ind_t Struct Reference

Data Fields

- ims_PCSCFPortInfo PCTlv

- ims_CSCFPortNameInfo PNTlv
- ims_IMSTestModeInfo TMTlv
- swi_uint256_t ParamPresenceMask

8.715.1 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

8.715.2 Field Documentation

8.715.2.1 swi_uint256_t unpack_ims_SLQSRegMgrCfgCallBack_ind_t::ParamPresenceMask

8.715.2.2 ims_PCSCFPortInfo unpack_ims_SLQSRegMgrCfgCallBack_ind_t::PCTlv

8.715.2.3 ims_CSCFPortNameInfo unpack_ims_SLQSRegMgrCfgCallBack_ind_t::PNTlv

8.715.2.4 ims_IMSTestModeInfo unpack_ims_SLQSRegMgrCfgCallBack_ind_t::TMTlv

8.716 unpack_ims_SLQSSetIMSSMSConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.716.1 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**

8.716.2 Field Documentation

8.716.2.1 swi_uint256_t unpack_ims_SLQSSetIMSSMSConfig_t::ParamPresenceMask

8.716.2.2 uint8_t* unpack_ims_SLQSSetIMSSMSConfig_t::pSettingResp

8.717 unpack_ims_SLQSSetIMSUserConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.717.1 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**

8.717.2 Field Documentation

8.717.2.1 swi_uint256_t unpack_ims_SLQSSetIMSUserConfig_t::ParamPresenceMask

8.717.2.2 uint8_t* unpack_ims_SLQSSetIMSUserConfig_t::pSettingResp

8.718 unpack_ims_SLQSSetIMSVoIPConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.718.1 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**

8.718.2 Field Documentation

8.718.2.1 swi_uint256_t unpack_ims_SLQSSetIMSVoIPConfig_t::ParamPresenceMask

8.718.2.2 uint8_t* unpack_ims_SLQSSetIMSVoIPConfig_t::pSettingResp

8.719 unpack_ims_SLQSSetRegMgrConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.719.1 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**

8.719.2 Field Documentation

8.719.2.1 swi_uint256_t unpack_ims_SLQSSetRegMgrConfig_t::ParamPresenceMask

8.719.2.2 uint8_t* unpack_ims_SLQSSetRegMgrConfig_t::pSettingResp

8.720 unpack_ims_SLQSSetSIPConfig_t Struct Reference

Data Fields

- uint8_t * pSettingResp
- swi_uint256_t ParamPresenceMask

8.720.1 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**

8.720.2 Field Documentation

8.720.2.1 swi_uint256_t unpack_ims_SLQSSetSIPConfig_t::ParamPresenceMask

8.720.2.2 uint8_t* unpack_ims_SLQSSetSIPConfig_t::pSettingResp

8.721 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

8.721.1 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

8.721.2 Field Documentation

8.721.2.1 swi_uint256_t unpack_ims_SLQSSIPCfgCallBack_ind_t::ParamPresenceMask

8.721.2.2 ims_SigCompEnInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::SCTlv

8.721.2.3 ims_SIPPortInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::SPTlv

8.721.2.4 ims_SIPRegnTmrInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::SRTlv

8.721.2.5 ims_SubscrTmrInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::STTlv

8.721.2.6 ims_TmrT1Info unpack_ims_SLQSSIPCfgCallBack_ind_t::TT1Tlv

8.721.2.7 ims_TmrT2Info unpack_ims_SLQSSIPCfgCallBack_ind_t::TT2Tlv

8.721.2.8 ims_TmrTfInfo unpack_ims_SLQSSIPCfgCallBack_ind_t::TTfTlv

8.722 unpack_ims_SLQSSMSCfgCallBack_ind_t Struct Reference

Data Fields

- ims_SMSFmtInfo SFTlv
- ims_SMSoIPNwInfo SINTlv
- ims_PhCtxtURIInfo PCURTlv
- swi_uint256_t ParamPresenceMask

8.722.1 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

8.722.2 Field Documentation

8.722.2.1 swi_uint256_t unpack_ims_SLQSSMSCfgCallBack_ind_t::ParamPresenceMask

8.722.2.2 ims_PhCtxtURIInfo unpack_ims_SLQSSMSCfgCallBack_ind_t::PCURTlv

8.722.2.3 ims_SMSFmtInfo unpack_ims_SLQSSMSCfgCallBack_ind_t::SFTlv

8.722.2.4 ims_SMSoIPNwInfo unpack_ims_SLQSSMSCfgCallBack_ind_t::SINTlv

8.723 unpack_ims_SLQSUserCfgCallBack_ind_t Struct Reference

Data Fields

- ims_IMSDomainInfo IDTlv
- swi_uint256_t ParamPresenceMask

8.723.1 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

8.723.2 Field Documentation

8.723.2.1 ims_IMSDomainInfo unpack_ims_SLQSUserCfgCallBack_ind_t::IDTlv

8.723.2.2 swi_uint256_t unpack_ims_SLQSUserCfgCallBack_ind_t::ParamPresenceMask

8.724 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

8.724.1 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

8.724.2 Field Documentation

8.724.2.1 ims_AMRModeInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AMTlv

8.724.2.2 ims_AMROctAlgnInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AOATlv

8.724.2.3 ims_AMRWBModeInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AWMTlv

8.724.2.4 ims_AMRWBOctAlgnInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::AWOATlv

8.724.2.5 ims_EnabAMRWBInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::EAWTlv

8.724.2.6 ims_EnabSCRAMRInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::ESATlv

8.724.2.7 ims_EnabSCRAMRWBInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::ESAWTlv

8.724.2.8 ims_MinSessExpInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::MSETlv

8.724.2.9 swi_uint256_t unpack_ims_SLQSVoIPCfgCallBack_ind_t::ParamPresenceMask

8.724.2.10 ims_RngBkTmrInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::RBTTlv

8.724.2.11 ims_RTPRTCPInactTmrDurInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::RTIDTlv

8.724.2.12 ims_RngTmrInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::RTTlv

8.724.2.13 ims_SessDurInfo unpack_ims_SLQSVoIPCfgCallBack_ind_t::SDTlv

8.725 unpack_imsa_SLQSGetIMSARegStatus_t Struct Reference

Data Fields

- uint8_t ImsRegStatus
- uint16_t ImsRegErrCode
- uint32_t NewImsRegStatus
- swi_uint256_t ParamPresenceMask

8.725.1 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**

NewImsReg-Status	• New IMS Registration Status • Values – 0 - Not registered for IMS – 1 - Registering for IMS – 2 - Registered for IMS • Bit to check in ParamPresenceMask - **18**

8.725.2 Field Documentation

8.725.2.1 uint16_t unpack_imsa_SLQSGetIMSARegStatus_t::ImsRegErrCode

8.725.2.2 uint8_t unpack_imsa_SLQSGetIMSARegStatus_t::ImsRegStatus

8.725.2.3 uint32_t unpack_imsa_SLQSGetIMSARegStatus_t::NewImsRegStatus

8.725.2.4 swi_uint256_t unpack_imsa_SLQSGetIMSARegStatus_t::ParamPresenceMask

8.726 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

8.726.1 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**

8.726.2 Field Documentation

8.726.2.1 swi_uint256_t unpack_imsa_SLQSGetIMSAServiceStatus_t::ParamPresenceMask

8.726.2.2 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::SmsServiceRat

8.726.2.3 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::SmsServiceStatus

8.726.2.4 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::UtServiceRat

8.726.2.5 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::UtServiceStatus

8.726.2.6 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VoipServiceRat

8.726.2.7 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VoipServiceStatus

8.726.2.8 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VsServiceRat

8.726.2.9 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VsServiceStatus

8.726.2.10 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VtServiceRat

8.726.2.11 uint32_t unpack_imsa_SLQSGetIMSAServiceStatus_t::VtServiceStatus

8.727 unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t Struct Reference

Data Fields

- uint8_t PdpConnState
- imsa_IMSFailErrCodeTlv FailErrCode
- swi_uint256_t ParamPresenceMask

8.727.1 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

8.727.2 Field Documentation

8.727.2.1 imsa_IMSFailErrCodeTlv unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t::FailErrCode

8.727.2.2 swi_uint256_t unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t::ParamPresenceMask

8.727.2.3 uint8_t unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t::PdpConnState

8.728 unpack_imsa_SLQSImsaRatStatusCallBack_ind_t Struct Reference

Data Fields

- imsa_RatHandoverStatusInfo RatHandover
- swi_uint256_t ParamPresenceMask

8.728.1 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

8.728.2 Field Documentation

8.728.2.1 swi_uint256_t unpack_imsa_SLQSImsaRatStatusCallBack_ind_t::ParamPresenceMask

8.728.2.2 imsa_RatHandoverStatusInfo unpack_imsa_SLQSImsaRatStatusCallBack_ind_t::RatHandover

8.729 unpack_imsa_SLQSImsaRegStatusCallBack_ind_t Struct Reference

Data Fields

- imsa_IMSRegStatusInfo IMSRegistration
- imsa_IMSRegStatusErrorCodeInfo IMSRegistrationError
- imsa_NewIMSRegStatusInfo NewIMSRegistration
- swi_uint256_t ParamPresenceMask

8.729.1 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

8.729.2 Field Documentation

8.729.2.1 imsa_IMSRegStatusInfo unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::IMSRegistration

8.729.2.2 imsa_IMSRegStatusErrorCodeInfo unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::IMSRegistrationError

8.729.2.3 imsa_NewIMSRegStatusInfo unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::NewIMSRegistration

8.729.2.4 swi_uint256_t unpack_imsa_SLQSImsaRegStatusCallBack_ind_t::ParamPresenceMask

8.730 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

8.730.1 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

8.730.2 Field Documentation

8.730.2.1 swi_uint256_t unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::ParamPresenceMask

8.730.2.2 imsa_SmsRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::SmsRat

8.730.2.3 imsa_SmsSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::SmsService

8.730.2.4 imsa_UtRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::UtRat

8.730.2.5 imsa_UtSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::UtService

8.730.2.6 imsa_VoipRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VoipRat

8.730.2.7 imsa_VoipSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VoipService

8.730.2.8 imsa_VtRatInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VtRat

8.730.2.9 imsa_VtSvcStatusInfo unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t::VtService

8.731 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

8.731.1 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**

8.731.2 Field Documentation

8.731.2.1 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtEllipsoid

8.731.2.2 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtMeanSeaLevel

8.731.2.3 swi_uint256_t unpack_loc_BestAvailPos_Ind_t::ParamPresenceMask

8.731.2.4 loc_gpsTime* unpack_loc_BestAvailPos_Ind_t::pGpsTime

8.731.2.5 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeading

8.731.2.6 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeadingUnc

8.731.2.7 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorCirConf

8.731.2.8 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorEllpConf

8.731.2.9 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorReliability

8.731.2.10 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncCircular

8.731.2.11 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseOrientAzimuth

8.731.2.12 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMajor

8.731.2.13 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMinor

8.731.2.14 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLatitude

8.731.2.15 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLongitude

8.731.2.16 uint32_t* unpack_loc_BestAvailPos_Ind_t::pMagneticDeviation

8.731.2.17 loc_precisionDilution* unpack_loc_BestAvailPos_Ind_t::pPrecisionDilution

8.731.2.18 loc_sensorDataUsage* unpack_loc_BestAvailPos_Ind_t::pSensorDataUsage

8.731.2.19 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedHorizontal

8.731.2.20 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedUnc

8.731.2.21 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVertical

8.731.2.22 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVerticalUnc

8.731.2.23 loc_svUsedforFix* unpack_loc_BestAvailPos_Ind_t::pSvUsedforFix

8.731.2.24 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTechnologyMask

8.731.2.25 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeSrc

8.731.2.26 uint64_t* unpack_loc_BestAvailPos_Ind_t::pTimestampUtc

8.731.2.27 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeUnc

8.731.2.28 uint8_t* unpack_loc_BestAvailPos_Ind_t::pVertConfidence

8.731.2.29 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertReliability

8.731.2.30 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertUnc

8.731.2.31 uint32_t* unpack_loc_BestAvailPos_Ind_t::pXid

8.731.2.32 uint32_t unpack_loc_BestAvailPos_Ind_t::status

8.731.2.33 uint16_t unpack_loc_BestAvailPos_Ind_t::Tlvresult

8.732 unpack_loc_CradleMountCallback_Ind_t Struct Reference

Data Fields

- uint32_t cradleMountConfigStatus
- swi_uint256_t ParamPresenceMask

8.732.1 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**

8.732.2 Field Documentation

8.732.2.1 uint32_t unpack_loc_CradleMountCallback_Ind_t::cradleMountConfigStatus

8.732.2.2 swi_uint256_t unpack_loc_CradleMountCallback_Ind_t::ParamPresenceMask

8.733 unpack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- uint16_t Tlvresult

- swi_uint256_t ParamPresenceMask

8.733.1 Detailed Description

This structure contains LOC delete assist data unpack

Parameters

TlvResult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

8.733.2 Field Documentation

8.733.2.1 swi_uint256_t unpack_loc_Delete_Assist_Data_t::ParamPresenceMask

8.733.2.2 uint16_t unpack_loc_Delete_Assist_Data_t::TlvResult

8.734 unpack_loc_DeleteAssistData_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.734.1 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

8.734.2 Field Documentation

8.734.2.1 swi_uint256_t unpack_loc_DeleteAssistData_Ind_t::ParamPresenceMask

8.734.2.2 uint32_t unpack_loc_DeleteAssistData_Ind_t::status

8.734.2.3 uint16_t unpack_loc_DeleteAssistData_Ind_t::TlvResult

8.735 unpack_loc_EngineState_Ind_t Struct Reference

Data Fields

- uint32_t engineState
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.735.1 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

8.735.2 Field Documentation

8.735.2.1 uint32_t unpack_loc_EngineState_Ind_t::engineState

8.735.2.2 swi_uint256_t unpack_loc_EngineState_Ind_t::ParamPresenceMask

8.735.2.3 uint16_t unpack_loc_EngineState_Ind_t::TlvResult

8.736 unpack_loc_EventNMEA_Ind_t Struct Reference

Data Fields

- char NMEAData [201]
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.736.1 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

8.736.2 Field Documentation

8.736.2.1 char unpack_loc_EventNMEA_Ind_t::NMEAData[201]

8.736.2.2 swi_uint256_t unpack_loc_EventNMEA_Ind_t::ParamPresenceMask

8.736.2.3 uint16_t unpack_loc_EventNMEA_Ind_t::TlvResult

8.737 unpack_loc_EventRegister_t Struct Reference

Data Fields

- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.737.1 Detailed Description

This structure contains Event Register unpack

Parameters

TlvResult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

8.737.2 Field Documentation

8.737.2.1 swi_uint256_t unpack_loc_EventRegister_t::ParamPresenceMask

8.737.2.2 uint16_t unpack_loc_EventRegister_t::TlvResult

8.738 unpack_loc_EventTimeSyncCallback_Ind_t Struct Reference

Data Fields

- uint32_t timeSyncRefCounter
- swi_uint256_t ParamPresenceMask

8.738.1 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**

8.738.2 Field Documentation

8.738.2.1 swi_uint256_t unpack_loc_EventTimeSyncCallback_Ind_t::ParamPresenceMask

8.738.2.2 uint32_t unpack_loc_EventTimeSyncCallback_Ind_t::timeSyncRefCounter

8.739 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

8.739.1 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**

8.739.2 Field Documentation

8.739.2.1 swi_uint256_t unpack_loc_GetServer_Ind_t::ParamPresenceMask

8.739.2.2 loc_IPv4Info* unpack_loc_GetServer_Ind_t::pIPv4AddrInfo

8.739.2.3 loc_IPv6Info* unpack_loc_GetServer_Ind_t::pIPv6AddrInfo

8.739.2.4 loc_urlAddr* unpack_loc_GetServer_Ind_t::pURL

8.739.2.5 uint32_t unpack_loc_GetServer_Ind_t::serverStatus

8.739.2.6 uint32_t unpack_loc_GetServer_Ind_t::serverType

8.739.2.7 uint16_t unpack_loc_GetServer_Ind_t::Tlvresult

8.740 unpack_loc_GnssSvInfo_Ind_t Struct Reference

Data Fields

- uint8_t altitudeAssumed
- loc_satelliteInfo * pSatelliteInfo
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.740.1 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

8.740.2 Field Documentation

8.740.2.1 uint8_t unpack_loc_GnssSvInfo_Ind_t::altitudeAssumed

8.740.2.2 swi_uint256_t unpack_loc_GnssSvInfo_Ind_t::ParamPresenceMask

8.740.2.3 loc_satelliteInfo* unpack_loc_GnssSvInfo_Ind_t::pSatelliteInfo

8.740.2.4 uint16_t unpack_loc_GnssSvInfo_Ind_t::TlvResult

8.741 unpack_loc_InjectPositionCallback_Ind_t Struct Reference

Data Fields

- uint32_t status
- swi_uint256_t ParamPresenceMask

8.741.1 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**

Note

None

8.741.2 Field Documentation

8.741.2.1 swi_uint256_t unpack_loc_InjectPositionCallback_Ind_t::ParamPresenceMask

8.741.2.2 uint32_t unpack_loc_InjectPositionCallback_Ind_t::status

8.742 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

8.742.1 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**

pGyroTemp-Samples-Accepted[OPTI-ONAL]	• 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**

8.742.2 Field Documentation

8.742.2.1 uint32_t unpack_loc_InjectSensorDataCallback_Ind_t::injectSensorDataStatus

8.742.2.2 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pAccelSamplesAccepted

8.742.2.3 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pAccelTempSamplesAccepted

8.742.2.4 swi_uint256_t unpack_loc_InjectSensorDataCallback_Ind_t::ParamPresenceMask

8.742.2.5 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pGyroSamplesAccepted

8.742.2.6 uint8_t* unpack_loc_InjectSensorDataCallback_Ind_t::pGyroTempSamplesAccepted

8.742.2.7 uint32_t* unpack_loc_InjectSensorDataCallback_Ind_t::pOpaqueIdentifier

8.743 unpack_loc_InjectTimeSyncDataCallback_Ind_t Struct Reference

Data Fields

- uint32_t injectTimeSyncStatus
- swi_uint256_t ParamPresenceMask

8.743.1 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**

8.743.2 Field Documentation

8.743.2.1 uint32_t unpack_loc_InjectTimeSyncDataCallback_Ind_t::injectTimeSyncStatus

8.743.2.2 swi_uint256_t unpack_loc_InjectTimeSyncDataCallback_Ind_t::ParamPresenceMask

8.744 unpack_loc_InjectUTCTimeCallback_Ind_t Struct Reference

Data Fields

- uint32_t status
- swi_uint256_t ParamPresenceMask

8.744.1 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**

Note

None

8.744.2 Field Documentation

8.744.2.1 swi_uint256_t unpack_loc_InjectUTCTimeCallback_Ind_t::ParamPresenceMask

8.744.2.2 uint32_t unpack_loc_InjectUTCTimeCallback_Ind_t::status

8.745 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

8.745.1 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**

8.745.2 Field Documentation

8.745.2.1 uint8_t* unpack_loc_PositionRpt_Ind_t::pAltitudeAssumed

8.745.2.2 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtEllipsoid

8.745.2.3 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtMeanSeaLevel

8.745.2.4 swi_uint256_t unpack_loc_PositionRpt_Ind_t::ParamPresenceMask

8.745.2.5 uint32_t* unpack_loc_PositionRpt_Ind_t::pFixId

8.745.2.6 loc_gpsTime* unpack_loc_PositionRpt_Ind_t::pGpsTime

8.745.2.7 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeading

8.745.2.8 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeadingUnc

8.745.2.9 uint8_t* unpack_loc_PositionRpt_Ind_t::pHorConfidence

8.745.2.10 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorReliability

8.745.2.11 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncCircular

8.745.2.12 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseOrientAzimuth

8.745.2.13 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMajor

8.745.2.14 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMinor

8.745.2.15 uint64_t* unpack_loc_PositionRpt_Ind_t::pLatitude

8.745.2.16 uint8_t* unpack_loc_PositionRpt_Ind_t::pLeapSeconds

8.745.2.17 uint64_t* unpack_loc_PositionRpt_Ind_t::pLongitude

8.745.2.18 uint32_t* unpack_loc_PositionRpt_Ind_t::pMagneticDeviation

8.745.2.19 loc_precisionDilution* unpack_loc_PositionRpt_Ind_t::pPrecisionDilution

8.745.2.20 loc_sensorDataUsage* unpack_loc_PositionRpt_Ind_t::pSensorDataUsage

8.745.2.21 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedHorizontal

8.745.2.22 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedUnc

8.745.2.23 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedVertical

8.745.2.24 loc_svUsedforFix* unpack_loc_PositionRpt_Ind_t::pSvUsedforFix

8.745.2.25 uint32_t* unpack_loc_PositionRpt_Ind_t::pTechnologyMask

8.745.2.26 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeSrc

8.745.2.27 uint64_t* unpack_loc_PositionRpt_Ind_t::pTimestampUtc

8.745.2.28 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeUnc

8.745.2.29 uint8_t* unpack_loc_PositionRpt_Ind_t::pVertConfidence

8.745.2.30 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertReliability

8.745.2.31 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertUnc

8.745.2.32 uint8_t unpack_loc_PositionRpt_Ind_t::sessionId

8.745.2.33 uint32_t unpack_loc_PositionRpt_Ind_t::sessionStatus

8.745.2.34 uint16_t unpack_loc_PositionRpt_Ind_t::TIvresult

8.746 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

8.746.1 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**

8.746.2 Field Documentation

8.746.2.1 loc_accelAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pAccelAcceptReady

8.746.2.2 loc_accelTempAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pAccelTempAcceptReady

8.746.2.3 swi_uint256_t unpack_loc_SensorStreamingCallback_Ind_t::ParamPresenceMask

8.746.2.4 loc_gyroAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pGyroAcceptReady

8.746.2.5 loc_gyroTempAcceptReady∗ unpack_loc_SensorStreamingCallback_Ind_t::pGyroTempAcceptReady

8.747 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.747.1 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

8.747.2 Field Documentation

8.747.2.1 swi_uint256_t unpack_loc_SetExtPowerConfig_Ind_t::ParamPresenceMask

8.747.2.2 uint32_t unpack_loc_SetExtPowerConfig_Ind_t::status

8.747.2.3 uint16_t unpack_loc_SetExtPowerConfig_Ind_t::TlvResult

8.748 unpack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.748.1 Detailed Description

This structure contains Set Ext Power State unpack

Parameters

TlvResult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

8.748.2 Field Documentation

8.748.2.1 swi_uint256_t unpack_loc_SetExtPowerState_t::ParamPresenceMask

8.748.2.2 uint16_t unpack_loc_SetExtPowerState_t::TlvResult

8.749 unpack_loc_SetOperationMode_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.749.1 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

8.749.2 Field Documentation

8.749.2.1 swi_uint256_t unpack_loc_SetOperationMode_Ind_t::ParamPresenceMask

8.749.2.2 uint32_t unpack_loc_SetOperationMode_Ind_t::status

8.749.2.3 uint16_t unpack_loc_SetOperationMode_Ind_t::Tlvresult

8.750 unpack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.750.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

TlvResult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

8.750.2 Field Documentation

8.750.2.1 swi_uint256_t unpack_loc_SetOperationMode_t::ParamPresenceMask

8.750.2.2 uint16_t unpack_loc_SetOperationMode_t::TlvResult

8.751 unpack_loc_SetServer_Ind_t Struct Reference

Data Fields

- uint32_t serverStatus
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.751.1 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.

8.751.2 Field Documentation

8.751.2.1 swi_uint256_t unpack_loc_SetServer_Ind_t::ParamPresenceMask

8.751.2.2 uint32_t unpack_loc_SetServer_Ind_t::serverStatus

8.751.2.3 uint16_t unpack_loc_SetServer_Ind_t::Tlvresult

8.752 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.752.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

8.752.2 Field Documentation

8.752.2.1 swi_uint256_t unpack_loc_SLQSLOCGetBestAvailPos_t::ParamPresenceMask

8.752.2.2 uint16_t unpack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.753 unpack_loc_Start_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.753.1 Detailed Description

This structure contains Start LOC unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

8.753.2 Field Documentation

8.753.2.1 swi_uint256_t unpack_loc_Start_t::ParamPresenceMask

8.753.2.2 uint16_t unpack_loc_Start_t::Tlvresult

8.754 unpack_loc_Stop_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.754.1 Detailed Description

This structure contains Stop LOC unpack

Parameters

Tlvresult	• Unpack result. • Bit to check in ParamPresenceMask - **2**

8.754.2 Field Documentation

8.754.2.1 swi_uint256_t unpack_loc_Stop_t::ParamPresenceMask

8.754.2.2 uint16_t unpack_loc_Stop_t::Tlvresult

8.755 unpack_nas_GetACCOLC_t Struct Reference

Data Fields

- uint8_t * pAccolc
- swi_uint256_t ParamPresenceMask

8.755.1 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.

8.755.2 Field Documentation

8.755.2.1 uint8_t* unpack_nas_GetACCOLC_t::pAccolc

8.755.2.2 swi_uint256_t unpack_nas_GetACCOLC_t::ParamPresenceMask

8.756 unpack_nas_GetANAAAAuthenticationStatus_t Struct Reference

Data Fields

- uint32_t * pAuthStatus
- swi_uint256_t ParamPresenceMask

8.756.1 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.

8.756.2 Field Documentation

8.756.2.1 swi_uint256_t unpack_nas_GetANAAAAuthenticationStatus_t::ParamPresenceMask

8.756.2.2 uint32_t* unpack_nas_GetANAAAAuthenticationStatus_t::pAuthStatus

8.757 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

8.757.1 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.

8.757.2 Field Documentation

8.757.2.1 uint32_t unpack_nas_GetCDMANetworkParameters_t::Application

8.757.2.2 uint32_t unpack_nas_GetCDMANetworkParameters_t::Broadcast

8.757.2.3 uint8_t unpack_nas_GetCDMANetworkParameters_t::CustomSCP

8.757.2.4 uint8_t unpack_nas_GetCDMANetworkParameters_t::ForceRev0

8.757.2.5 swi_uint256_t unpack_nas_GetCDMANetworkParameters_t::ParamPresenceMask

8.757.2.6 uint32_t unpack_nas_GetCDMANetworkParameters_t::Protocol

8.757.2.7 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignNID

8.757.2.8 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignSID

8.757.2.9 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegHomeSID

8.757.2.10 uint32_t unpack_nas_GetCDMANetworkParameters_t::Roaming

8.757.2.11 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCI

8.757.2.12 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCM

8.758 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

8.758.1 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.

8.758.2 Field Documentation

8.758.2.1 uint8_t unpack_nas_GetHomeNetwork3GPP2_t::nameSize

8.758.2.2 swi_uint256_t unpack_nas_GetHomeNetwork3GPP2_t::ParamPresenceMask

8.758.2.3 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMCC

8.758.2.4 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMNC

8.758.2.5 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pName

8.758.2.6 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNID

8.758.2.7 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescDisp

8.758.2.8 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescEnc

8.758.2.9 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescLen

8.758.2.10 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MCC

8.758.2.11 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MNC

8.758.2.12 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2Name

8.758.2.13 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pSID

8.759 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

8.759.1 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.

8.759.2 Field Documentation

8.759.2.1 uint16_t unpack_nas_GetHomeNetwork_t::mcc

8.759.2.2 uint16_t unpack_nas_GetHomeNetwork_t::mnc

8.759.2.3 char unpack_nas_GetHomeNetwork_t::name[255]

8.759.2.4 uint16_t unpack_nas_GetHomeNetwork_t::nid

8.759.2.5 swi_uint256_t unpack_nas_GetHomeNetwork_t::ParamPresenceMask

8.759.2.6 uint16_t unpack_nas_GetHomeNetwork_t::sid

8.760 unpack_nas_GetNetworkPreference_t Struct Reference

Data Fields

- uint32_t ActiveTechPref
- uint32_t Duration
- uint32_t PersistentTechPref
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.760.1 Detailed Description

Returns the network registration preference. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx 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.

8.760.2 Field Documentation

8.760.2.1 uint32_t unpack_nas_GetNetworkPreference_t::ActiveTechPref

8.760.2.2 uint32_t unpack_nas_GetNetworkPreference_t::Duration

8.760.2.3 swi_uint256_t unpack_nas_GetNetworkPreference_t::ParamPresenceMask

8.760.2.4 uint32_t unpack_nas_GetNetworkPreference_t::PersistentTechPref

8.760.2.5 uint16_t unpack_nas_GetNetworkPreference_t::Tlvresult

8.761 unpack_nas_GetRFInfo_t Struct Reference

Data Fields

- uint8_t instancesSize
- RFBandInfoElements RFBandInfoElements [255]
- swi_uint256_t ParamPresenceMask

8.761.1 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.

8.761.2 Field Documentation

8.761.2.1 uint8_t unpack_nas_GetRFInfo_t::instancesSize

8.761.2.2 swi_uint256_t unpack_nas_GetRFInfo_t::ParamPresenceMask

8.761.2.3 RFBandInfoElements unpack_nas_GetRFInfo_t::RFBandInfoElements[255]

8.762 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

8.762.1 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.

8.762.2 Field Documentation

8.762.2.1 uint32_t unpack_nas_GetServingNetwork_t::CSDomain

8.762.2.2 uint8_t unpack_nas_GetServingNetwork_t::DataCaps[255]

8.762.2.3 uint8_t unpack_nas_GetServingNetwork_t::DataCapsLen

8.762.2.4 uint16_t unpack_nas_GetServingNetwork_t::MCC

8.762.2.5 uint16_t unpack_nas_GetServingNetwork_t::MNC

8.762.2.6 uint8_t unpack_nas_GetServingNetwork_t::Name[255]

8.762.2.7 uint8_t unpack_nas_GetServingNetwork_t::nameSize

8.762.2.8 swi_uint256_t unpack_nas_GetServingNetwork_t::ParamPresenceMask

8.762.2.9 uint32_t unpack_nas_GetServingNetwork_t::PSDomain

8.762.2.10 uint8_t unpack_nas_GetServingNetwork_t::RadioIfaces[255]

8.762.2.11 uint8_t unpack_nas_GetServingNetwork_t::RadioIfacesSize

8.762.2.12 uint32_t unpack_nas_GetServingNetwork_t::RAN

8.762.2.13 uint32_t unpack_nas_GetServingNetwork_t::RegistrationState

8.762.2.14 uint32_t unpack_nas_GetServingNetwork_t::Roaming

8.763 unpack_nas_GetServingNetworkCapabilities_t Struct Reference

Data Fields

- uint8_t DataCapsLen
- uint8_t DataCaps [255]
- swi_uint256_t ParamPresenceMask

8.763.1 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.

8.763.2 Field Documentation

8.763.2.1 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCaps[255]

8.763.2.2 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCapsLen

8.763.2.3 swi_uint256_t unpack_nas_GetServingNetworkCapabilities_t::ParamPresenceMask

8.764 unpack_nas_GetSignalStrengths_t Struct Reference

Data Fields

- uint32_t len
- signed char rssi [8]
- uint32_t radio [8]
- swi_uint256_t ParamPresenceMask

8.764.1 Detailed Description

This structure contains unpack get signal strengths parameters.

Parameters

len	• number of rssi & 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.

8.764.2 Field Documentation

8.764.2.1 uint32_t unpack_nas_GetSignalStrengths_t::len

8.764.2.2 swi_uint256_t unpack_nas_GetSignalStrengths_t::ParamPresenceMask

8.764.2.3 uint32_t unpack_nas_GetSignalStrengths_t::radio[8]

8.764.2.4 signed char unpack_nas_GetSignalStrengths_t::rssi[8]

8.765 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

8.765.1 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.

8.765.2 Field Documentation

8.765.2.1 nas_QmiNas3GppNetworkInfo* unpack_nas_PerformNetworkScan_t::p3GppNetworkInfoInstances

8.765.2.2 uint8_t* unpack_nas_PerformNetworkScan_t::p3GppNetworkInstanceSize

8.765.2.3 swi_uint256_t unpack_nas_PerformNetworkScan_t::ParamPresenceMask

8.765.2.4 nas_lteOpModeTlv* unpack_nas_PerformNetworkScan_t::pLteOpModeTlv

8.765.2.5 nas_QmisNasSlqsNasPCIInfo* unpack_nas_PerformNetworkScan_t::pPCIInfo

8.765.2.6 nas_QmisNasPcsDigit* unpack_nas_PerformNetworkScan_t::pPCSInstance

8.765.2.7 uint8_t* unpack_nas_PerformNetworkScan_t::pPCSInstanceSize

8.765.2.8 nas_QmiNas3GppNetworkRAT* unpack_nas_PerformNetworkScan_t::pRATINstance

8.765.2.9 uint8_t* unpack_nas_PerformNetworkScan_t::pRATInstanceSize

8.765.2.10 uint32_t* unpack_nas_PerformNetworkScan_t::pScanResult

8.766 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference

Data Fields

- uint8_t dataCapsSize
- uint8_t dataCaps [255]
- swi_uint256_t ParamPresenceMask

8.766.1 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.

8.766.2 Field Documentation

8.766.2.1 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCaps[255]

8.766.2.2 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCapsSize

8.766.2.3 swi_uint256_t unpack_nas_SetDataCapabilitiesCallback_ind_t::ParamPresenceMask

8.767 unpack_nas_SetEventReportInd_t Struct Reference

Data Fields

- nas_SignalStrengthTlv SSTlv
- nas_RFInfoTlv RFTlv
- nas_RejectReasonTlv RRTlv
- nas_SLQSSignalStrengthsTlv SLQSSSTlv
- swi_uint256_t ParamPresenceMask

8.767.1 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.

8.767.2 Field Documentation

8.767.2.1 swi_uint256_t unpack_nas_SetEventReportInd_t::ParamPresenceMask

8.767.2.2 nas_RFInfoTlv unpack_nas_SetEventReportInd_t::RFTlv

8.767.2.3 nas_RejectReasonTlv unpack_nas_SetEventReportInd_t::RRTlv

8.767.2.4 nas_SLQSSignalStrengthsTlv unpack_nas_SetEventReportInd_t::SLQSSSTlv

8.767.2.5 nas_SignalStrengthTlv unpack_nas_SetEventReportInd_t::SSTlv

8.768 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference

Data Fields

- nas_PhyCaAggScellIndType sPhyCaAggScellIndType
- nas_PhyCaAggScellDlBw sPhyCaAggScellDlBw
- nas_PhyCaAggScellInfo sPhyCaAggScellInfo
- nas_PhyCaAggPcellInfo sPhyCaAggPcellInfo
- nas_PhyCaAggScellIndex sPhyCaAggScellIndex
- swi_uint256_t ParamPresenceMask

8.768.1 Detailed Description

Structure for storing the LTE PHY CA indication parameters.

Parameters

pPhyCaAgg-ScellIndType	• See nas_PhyCaAggScellIndType for more information. • Bit to check in ParamPresenceMask - **1**
sPhyCaAgg-ScellDlBw	• See nas_PhyCaAggScellDlBw for more information. • Bit to check in ParamPresenceMask - **16**

sPhyCaAgg-ScellInfo	• See nas_PhyCaAggScellInfo for more information. • Bit to check in ParamPresenceMask - **17**
sPhyCaAgg-PcellInfo	• See nas_PhyCaAggPcellInfo for more information. • Bit to check in ParamPresenceMask - **18**
sPhyCaAgg-ScellIndex	• See nas_PhyCaAggScellIndex for more information. • Bit to check in ParamPresenceMask - **19**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.768.2 Field Documentation

8.768.2.1 swi_uint256_t unpack_nas_SetNasLTECphyCaIndCallback_ind_t::ParamPresenceMask

8.768.2.2 nas_PhyCaAggPcellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggPcellInfo

8.768.2.3 nas_PhyCaAggScellDlBw unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellDlBw

8.768.2.4 nas_PhyCaAggScellIndex unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndex

8.768.2.5 nas_PhyCaAggScellIndType unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndType

8.768.2.6 nas_PhyCaAggScellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellInfo

8.769 unpack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.769.1 Detailed Description

This structure contains unpack set network preference parameters.

Parameters

Tlvresult	• unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.769.2 Field Documentation

8.769.2.1 swi_uint256_t unpack_nas_SetNetworkPreference_t::ParamPresenceMask

8.769.2.2 uint16_t unpack_nas_SetNetworkPreference_t::Tlvresult

8.770 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference

Data Fields

- uint8_t roaming
- swi_uint256_t ParamPresenceMask

8.770.1 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.

8.770.2 Field Documentation

8.770.2.1 swi_uint256_t unpack_nas_SetRoamingIndicatorCallback_ind_t::ParamPresenceMask

8.770.2.2 uint8_t unpack_nas_SetRoamingIndicatorCallback_ind_t::roaming

8.771 unpack_nas_SetServingSystemCallback_ind_t Struct Reference

Data Fields

- NASServingSystemInfo SSInfo
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.771.1 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.

8.771.2 Field Documentation

8.771.2.1 swi_uint256_t unpack_nas_SetServingSystemCallback_ind_t::ParamPresenceMask

8.771.2.2 NASServingSystemInfo unpack_nas_SetServingSystemCallback_ind_t::SSInfo

8.771.2.3 uint16_t unpack_nas_SetServingSystemCallback_ind_t::Tlvresult

8.772 unpack_nas_SLQSGetErrorRate_t Struct Reference

Data Fields

- uint16_t * pCDMAFrameErrRate
- uint16_t * pHDRPackErrRate
- uint8_t * pGSMBER
- uint8_t * pWCDMABER
- swi_uint256_t ParamPresenceMask

8.772.1 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.

8.772.2 Field Documentation

8.772.2.1 swi_uint256_t unpack_nas_SLQSGetErrorRate_t::ParamPresenceMask

8.772.2.2 uint16_t* unpack_nas_SLQSGetErrorRate_t::pCDMAFrameErrRate

8.772.2.3 uint8_t* unpack_nas_SLQSGetErrorRate_t::pGSMBER

8.772.2.4 uint16_t* unpack_nas_SLQSGetErrorRate_t::pHDRPackErrRate

8.772.2.5 uint8_t* unpack_nas_SLQSGetErrorRate_t::pWCDMABER

8.773 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference

Data Fields

- NasGetLTECphyCaInfo LTECphyCaInfo
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.773.1 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.

8.773.2 Field Documentation

8.773.2.1 NasGetLTECphyCaInfo unpack_nas_SlqsGetLTECphyCAInfo_t::LTECphyCaInfo

8.773.2.2 swi_uint256_t unpack_nas_SlqsGetLTECphyCAInfo_t::ParamPresenceMask

8.773.2.3 uint16_t unpack_nas_SlqsGetLTECphyCAInfo_t::Tlvresult

8.774 unpack_nas_SLQSGetNetworkTime_t Struct Reference

Data Fields

- nas_timeInfo * p3GPP2TimeInfo
- nas_timeInfo * p3GPPTimeInfo
- swi_uint256_t ParamPresenceMask

8.774.1 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**

Parameters

p3GPPTimeInfo	[Optional] – See nas_timeInfo for more information

- * - Bit to check in ParamPresenceMask - **17**

Parameters

ParamPresenceMask	– bitmask representation to indicate valid parameters.

8.774.2 Field Documentation

8.774.2.1 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPP2TimeInfo

8.774.2.2 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPPTimeInfo

8.774.2.3 swi_uint256_t unpack_nas_SLQSGetNetworkTime_t::ParamPresenceMask

8.775 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

8.775.1 Detailed Description

Structure for Operator Name unpack.

Parameters

pSrvcProvider-Name	• Refer nas_serviceProviderName for details (Optional). • Can provide NULL if this parameter is not required. • Bit to check in ParamPresenceMask - **16**
pOperatorPLM-NList	• 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.

8.775.2 Field Documentation

8.775.2.1 swi_uint256_t unpack_nas_SLQSGetOperatorNameData_t::ParamPresenceMask

8.775.2.2 nas_PLMNNetworkNameData* unpack_nas_SLQSGetOperatorNameData_t::pNITZInformation

8.775.2.3 nas_operatorNameString* unpack_nas_SLQSGetOperatorNameData_t::pOperatorNameString

8.775.2.4 nas_operatorPLMNList* unpack_nas_SLQSGetOperatorNameData_t::pOperatorPLMNList

8.775.2.5 nas_PLMNNetworkName* unpack_nas_SLQSGetOperatorNameData_t::pPLMNNetworkName

8.775.2.6 nas_serviceProviderName* unpack_nas_SLQSGetOperatorNameData_t::pSrvcProviderName

8.776 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

8.776.1 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.

8.776.2 Field Documentation

8.776.2.1 char unpack_nas_SLQSGetPLMNName_t::longName[255]

8.776.2.2 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameCI

8.776.2.3 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameEn

8.776.2.4 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameLen

8.776.2.5 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameSB

8.776.2.6 swi_uint256_t unpack_nas_SLQSGetPLMNName_t::ParamPresenceMask

8.776.2.7 uint8_t unpack_nas_SLQSGetPLMNName_t::shortName[255]

8.776.2.8 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameCI

8.776.2.9 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameEn

8.776.2.10 char unpack_nas_SLQSGetPLMNName_t::shortNameLen

8.776.2.11 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameSB

8.776.2.12 char unpack_nas_SLQSGetPLMNName_t::spn[255]

8.776.2.13 uint8_t unpack_nas_SLQSGetPLMNName_t::spnEncoding

8.776.2.14 uint8_t unpack_nas_SLQSGetPLMNName_t::spnLength

8.777 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

8.777.1 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.

8.777.2 Field Documentation

8.777.2.1 uint16_t unpack_nas_SLQSGetServingSystem_t::BasestationID

8.777.2.2 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLatitude

8.777.2.3 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLongitude

8.777.2.4 nas_callBarStatus unpack_nas_SLQSGetServingSystem_t::CallBarStatus

8.777.2.5 uint8_t unpack_nas_SLQSGetServingSystem_t::CDMA_P_Rev

8.777.2.6 nas_CDMASysInfoExt unpack_nas_SLQSGetServingSystem_t::CDMASystemInfoExt

8.777.2.7 uint32_t unpack_nas_SLQSGetServingSystem_t::CellID

8.777.2.8 uint8_t unpack_nas_SLQSGetServingSystem_t::ConcSvcInfo

8.777.2.9 nas_currentPLMN unpack_nas_SLQSGetServingSystem_t::CurrentPLMN

8.777.2.10 nas_dataSrvCapabilities unpack_nas_SLQSGetServingSystem_t::DataSrvCapabilities

8.777.2.11 uint8_t unpack_nas_SLQSGetServingSystem_t::DefaultRoamInd

8.777.2.12 nas_detailSvcInfo unpack_nas_SLQSGetServingSystem_t::DetailedSvcInfo

8.777.2.13 uint8_t unpack_nas_SLQSGetServingSystem_t::DTMInd

8.777.2.14 nas_qaQmi3Gpp2TimeZone unpack_nas_SLQSGetServingSystem_t::Gpp2TimeZone

8.777.2.15 uint8_t unpack_nas_SLQSGetServingSystem_t::GppNetworkDSTAdjustment

8.777.2.16 uint8_t unpack_nas_SLQSGetServingSystem_t::GppTimeZone

8.777.2.17 uint8_t unpack_nas_SLQSGetServingSystem_t::HdrPersonality

8.777.2.18 uint16_t unpack_nas_SLQSGetServingSystem_t::Lac

8.777.2.19 uint16_t unpack_nas_SLQSGetServingSystem_t::NetworkID

8.777.2.20 swi_uint256_t unpack_nas_SLQSGetServingSystem_t::ParamPresenceMask

8.777.2.21 uint8_t unpack_nas_SLQSGetServingSystem_t::PRLInd

8.777.2.22 uint8_t unpack_nas_SLQSGetServingSystem_t::RoamIndicatorVal

8.777.2.23 nas_roamIndList unpack_nas_SLQSGetServingSystem_t::RoamingIndicatorList

8.777.2.24 nas_servSystem unpack_nas_SLQSGetServingSystem_t::ServingSystem

8.777.2.25 uint16_t unpack_nas_SLQSGetServingSystem_t::SystemID

8.777.2.26 uint16_t unpack_nas_SLQSGetServingSystem_t::TrackAreaCode

8.778 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

8.778.1 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.

8.778.2 Field Documentation

8.778.2.1 nas_ecioListElement unpack_nas_SLQSGetSignalStrength_t::ecioList[18]

8.778.2.2 uint16_t unpack_nas_SLQSGetSignalStrength_t::ecioListLen

8.778.2.3 nas_errorRateListElement unpack_nas_SLQSGetSignalStrength_t::errorRateList[18]

8.778.2.4 uint16_t unpack_nas_SLQSGetSignalStrength_t::errorRateListLen

8.778.2.5 int32_t unpack_nas_SLQSGetSignalStrength_t::lo

8.778.2.6 int16_t unpack_nas_SLQSGetSignalStrength_t::ltersrp

8.778.2.7 int16_t unpack_nas_SLQSGetSignalStrength_t::ltesnr

8.778.2.8 swi_uint256_t unpack_nas_SLQSGetSignalStrength_t::ParamPresenceMask

8.778.2.9 nas_rsrqInformation unpack_nas_SLQSGetSignalStrength_t::rsrqInfo

8.778.2.10 nas_rxSignalStrengthListElement unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthList[18]

8.778.2.11 uint16_t unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthListLen

8.778.2.12 uint16_t unpack_nas_SLQSGetSignalStrength_t::signalStrengthReqMask

8.778.2.13 uint8_t unpack_nas_SLQSGetSignalStrength_t::sinr

8.779 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

8.779.1 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**
pLTEVoiceSupportSysInfo	• Indicates voice support status on LTE. – 0x00 - Voice is not supported – 0x01 - Voice is supported • Bit to check in ParamPresenceMask - **33**
pGSMCipherDomainSysInfo	• 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**
pWCDMACipherDomainSysInfo	• 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.

8.779.2 Field Documentation

8.779.2.1 nas_AddCDMASysInfo∗ unpack_nas_SLQSGetSysInfo_t::pAddCDMASysInfo

8.779.2.2 nas_AddSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pAddGSMSysInfo

8.779.2.3 uint16_t∗ unpack_nas_SLQSGetSysInfo_t::pAddHDRSysInfo

8.779.2.4 uint16_t∗ unpack_nas_SLQSGetSysInfo_t::pAddLTESysInfo

8.779.2.5 nas_AddSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pAddWCDMASysInfo

8.779.2.6 swi_uint256_t unpack_nas_SLQSGetSysInfo_t::ParamPresenceMask

8.779.2.7 uint32_t∗ unpack_nas_SLQSGetSysInfo_t::pCampedCiotLteOpMode

8.779.2.8 nas_SrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pCDMASrvStatusInfo

8.779.2.9 nas_CDMASysInfo∗ unpack_nas_SLQSGetSysInfo_t::pCDMASysInfo

8.779.2.10 nas_CallBarringSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pGSMCallBarringSysInfo

8.779.2.11 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pGSMCipherDomainSysInfo

8.779.2.12 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pGSMSrvStatusInfo

8.779.2.13 nas_GSMSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pGSMSysInfo

8.779.2.14 nas_SrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pHDRSrvStatusInfo

8.779.2.15 nas_HDRSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pHDRSysInfo

8.779.2.16 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pLTESrvStatusInfo

8.779.2.17 nas_LTESysInfo∗ unpack_nas_SLQSGetSysInfo_t::pLTESysInfo

8.779.2.18 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pLTEVoiceSupportSysInfo

8.779.2.19 nas_CallBarringSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pWCDMACallBarringSysInfo

8.779.2.20 uint8_t* unpack_nas_SLQSGetSysInfo_t::pWCDMACipherDomainSysInfo

8.779.2.21 nas_GSMSrvStatusInfo* unpack_nas_SLQSGetSysInfo_t::pWCDMASrvStatusInfo

8.779.2.22 nas_WCDMASysInfo* unpack_nas_SLQSGetSysInfo_t::pWCDMASysInfo

8.780 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

8.780.1 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**

Parameters

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**

Parameters

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**

Parameters

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**

Parameters

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**

Parameters

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**

Parameters

pNetSelPref	– Optional parameter indicating network selection preference – Values: * 0x00 - Automatic network selection * 0x01 - Manual network selection.

- $*$ - Bit to check in ParamPresenceMask - **22**

Parameters

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**

Parameters

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**

Parameters

ParamPresence-Mask	– bitmask representation to indicate valid parameters.

8.780.2 Field Documentation

8.780.2.1 swi_uint256_t unpack_nas_SLQSGetSysSelectionPref_t::ParamPresenceMask

8.780.2.2 uint64_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pBandPref

8.780.2.3 uint8_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pEmerMode

8.780.2.4 uint32_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pGWAcqOrderPref

8.780.2.5 uint64_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pLTEBandPref

8.780.2.6 uint16_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pModePref

8.780.2.7 uint8_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pNetSelPref

8.780.2.8 uint16_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pPRLPref

8.780.2.9 uint16_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pRoamPref

8.780.2.10 uint32_t∗ unpack_nas_SLQSGetSysSelectionPref_t::pSrvDomainPref

8.781 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

8.781.1 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
pGWAcqOrder-Pref	• 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
pRatDisabled-Mask	• Optional parameter representing the radio technologies that are disabled. • See nas_RatDisabledMaskTlv for more information
pCiotLteOp-ModePref	• 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
pLteNb1Band-Pref	• Optional parameter representing the LTE NB1 band preference. • See nas_LteNb1BandPrefTlv for more information
pCiotAcqOrder-Pref	• 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
pLTEBandPref-Ext	• Optional parameter indicating LTE Band Preference Extended • See nas_LTEBandPrefExtTlv for more information

8.781.2 Field Documentation

8.781.2.1 nas_AcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pAcqOrderPref

8.781.2.2 nas_BandPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pBandPref

8.781.2.3 nas_CiotAcqOrderPrefTlv* unpack_nas_SLQSGetSysSelectionPrefExt_t::pCiotAcqOrderPref

8.781.2.4 **nas_CiotLteOpModePrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pCiotLteOpModePref

8.781.2.5 **nas_EmerModeTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pEmerMode

8.781.2.6 **nas_GWAcqOrderPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pGWAcqOrderPref

8.781.2.7 **nas_LTEBandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLTEBandPref

8.781.2.8 **nas_LTEBandPrefExtTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLTEBandPrefExt

8.781.2.9 **nas_LteM1BandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLteM1BandPref

8.781.2.10 **nas_LteNb1BandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pLteNb1BandPref

8.781.2.11 **nas_ModePrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pModePref

8.781.2.12 **nas_NetSelPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pNetSelPref

8.781.2.13 **nas_NR5gBandPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pNr5gBandPref

8.781.2.14 **nas_PRLPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pPRLPref

8.781.2.15 **nas_RatDisabledMaskTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pRatDisabledMask

8.781.2.16 **nas_RoamPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pRoamPref

8.781.2.17 **nas_SrvDomainPrefTlv**∗ unpack_nas_SLQSGetSysSelectionPrefExt_t::pSrvDomainPref

8.782 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

8.782.1 Detailed Description

Structure for storing the eDRX Change Info indication parameters.

Parameters

edrxEnableType-Tlv	• See nas_EdrxEnableType for more information. • Bit to check in ParamPresenceMask - **16**
edrxCycle-LengthTlv	• See nas_EdrxCycleLength for more information. • Bit to check in ParamPresenceMask - **17**

edrxPagingTime-WindowTlv	• 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**
edrxCiotLte-ModeTlv	• See nas_EdrxCiotLteMode for more information. • Bit to check in ParamPresenceMask - **20**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.782.2 Field Documentation

8.782.2.1 nas_EdrxCiotLteMode unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxCiotLteModeTlv

8.782.2.2 nas_EdrxCycleLength unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxCycleLengthTlv

8.782.2.3 nas_EdrxEnableType unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxEnableTypeTlv

8.782.2.4 nas_EdrxPagingTimeWindow unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxPagingTimeWindow-Tlv

8.782.2.5 nas_EdrxRatType unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::edrxRatTypeTlv

8.782.2.6 swi_uint256_t unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t::ParamPresenceMask

8.783 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
- swi_uint256_t ParamPresenceMask

8.783.1 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**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.783.2 Field Documentation

8.783.2.1 swi_uint256_t unpack_nas_SLQSNasGet3GPP2Subscription_t::ParamPresenceMask

8.783.2.2 nas_CDMAChannel* unpack_nas_SLQSNasGet3GPP2Subscription_t::pCDMAChannel

8.783.2.3 nas_dirNum* unpack_nas_SLQSNasGet3GPP2Subscription_t::pDirNum

8.783.2.4 nas_homeSIDNID* unpack_nas_SLQSNasGet3GPP2Subscription_t::pHomeSIDNID

8.783.2.5 nas_minBasedIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pMinBasedIMSI

8.783.2.6 nas_namName* unpack_nas_SLQSNasGet3GPP2Subscription_t::pNAMNameInfo

8.783.2.7 nas_trueIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pTrueIMSI

8.784 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

8.784.1 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-NeighboringGSM	• See nas_LTEInfoNeighboringGSM for more information. • Bit to check in ParamPresenceMask - **21**
pLTEInfo-NeighboringWCDMA	• 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**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.784.2 Field Documentation

8.784.2.1 **swi_uint256_t unpack_nas_SLQSNasGetCellLocationInfo_t::ParamPresenceMask**

8.784.2.2 **nas_CDMAInfo*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pCDMAInfo**

8.784.2.3 **nas_GERANInfo*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pGERANInfo**

8.784.2.4 **nas_LTEInfoInterfreq*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoInterfreq**

8.784.2.5 **nas_LTEInfoIntrafreq*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoIntrafreq**

8.784.2.6 **nas_LTEInfoNeighboringGSM*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringGSM**

8.784.2.7 **nas_LTEInfoNeighboringWCDMA*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringWCDMA**

8.784.2.8 **uint32_t*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSCellID**

8.784.2.9 **nas_UMTSInfo*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSInfo**

8.784.2.10 **nas_WCDMAInfoLTENeighborCell*** **unpack_nas_SLQSNasGetCellLocationInfo_t::pWCDMAInfoLTENeighborCell**

8.785 unpack_nas_SLQSNASGeteDRXParams_t Struct Reference

Data Fields

- uint8_t * pCycleLen
- uint8_t * pPagingTimeWindow
- uint8_t * pEdrxEnable
- swi_uint256_t ParamPresenceMask

8.785.1 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.

8.785.2 Field Documentation

8.785.2.1 swi_uint256_t unpack_nas_SLQSNASGeteDRXParams_t::ParamPresenceMask

8.785.2.2 uint8_t* unpack_nas_SLQSNASGeteDRXParams_t::pCycleLen

8.785.2.3 uint8_t* unpack_nas_SLQSNASGeteDRXParams_t::pEdrxEnable

8.785.2.4 uint8_t* unpack_nas_SLQSNASGeteDRXParams_t::pPagingTimeWindow

8.786 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

8.786.1 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.

8.786.2 Field Documentation

8.786.2.1 swi_uint256_t unpack_nas_SLQSNASGeteDRXParamsExt_t::ParamPresenceMask

8.786.2.2 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pCycleLen

8.786.2.3 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pEdrxEnable

8.786.2.4 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pEdrxRAT

8.786.2.5 uint32_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pLteOpMode

8.786.2.6 uint8_t* unpack_nas_SLQSNASGeteDRXParamsExt_t::pPagingTimeWindow

8.787 unpack_nas_SLQSNASGetForbiddenNetworks_t Struct Reference

Data Fields

- nas_ForbiddenNetworks3GPP * pForbiddenNetworks3GPP
- swi_uint256_t ParamPresenceMask

8.787.1 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.

8.787.2 Field Documentation

8.787.2.1 swi_uint256_t unpack_nas_SLQSNASGetForbiddenNetworks_t::ParamPresenceMask

8.787.2.2 nas_ForbiddenNetworks3GPP* unpack_nas_SLQSNASGetForbiddenNetworks_t::pForbiddenNetworks3GPP

8.788 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference

Data Fields

- uint8_t * pColorCode
- swi_uint256_t ParamPresenceMask

8.788.1 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.

8.788.2 Field Documentation

8.788.2.1 swi_uint256_t unpack_nas_SLQSNasGetHDRColorCode_t::ParamPresenceMask

8.788.2.2 uint8_t* unpack_nas_SLQSNasGetHDRColorCode_t::pColorCode

8.789 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

8.789.1 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

8.789.2 Field Documentation

8.789.2.1 nas_LTEOperationMode* unpack_nas_SLQSNasGetRFInfo_t::pLTEOperationMode

8.789.2.2 nas_RfBandInfoExtFormat* unpack_nas_SLQSNasGetRFInfo_t::pRfBandInfoExtFormat

8.789.2.3 nas_RfBandwidthInfo* unpack_nas_SLQSNasGetRFInfo_t::pRfBandwidthInfo

8.789.2.4 nas_RfDedicatedBandInfo* unpack_nas_SLQSNasGetRFInfo_t::pRfDedicatedBandInfo

8.789.2.5 nas_RfBandInfoList unpack_nas_SLQSNasGetRFInfo_t::rfbandInfoList

8.789.2.6 uint16_t unpack_nas_SLQSNasGetRFInfo_t::Tlvresult

8.790 unpack_nas_SLQSNasGetSigInfo_t Struct Reference

Data Fields

- cdmaSSInfo CDMASSInfo
- hdrSSInfo HDRSSInfo
- int8_t GSMSSInfo
- cdmaSSInfo WCDMASSInfo
- lteSSInfo LTESSInfo

- swi_uint256_t ParamPresenceMask

8.790.1 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.

8.790.2 Field Documentation

8.790.2.1 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::CDMASSInfo

8.790.2.2 int8_t unpack_nas_SLQSNasGetSigInfo_t::GSMSSInfo

8.790.2.3 hdrSSInfo unpack_nas_SLQSNasGetSigInfo_t::HDRSSInfo

8.790.2.4 lteSSInfo unpack_nas_SLQSNasGetSigInfo_t::LTESSInfo

8.790.2.5 swi_uint256_t unpack_nas_SLQSNasGetSigInfo_t::ParamPresenceMask

8.790.2.6 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::WCDMASSInfo

8.791 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- nas_rxInfo * pRXChain0Info
- nas_rxInfo * pRXChain1Info

- nas_txInfo * pTXInfo
- swi_uint256_t ParamPresenceMask

8.791.1 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.

8.791.2 Field Documentation

8.791.2.1 swi_uint256_t unpack_nas_SLQSNasGetTxRxInfo_t::ParamPresenceMask

8.791.2.2 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain0Info

8.791.2.3 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain1Info

8.791.2.4 nas_txInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pTXInfo

8.792 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

8.792.1 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 – 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 • Bit to check in ParamPresenceMask - **3**

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.

8.792.2 Field Documentation

8.792.2.1 swi_uint256_t unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::ParamPresenceMask

8.792.2.2 nas_CsgId* unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::pCsgId

8.792.2.3 nas_LteOpMode* unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::pLteOpMode

8.792.2.4 nas_PlmnID* unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::pPlmnId

8.792.2.5 uint8_t unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::radioIf

8.792.2.6 uint8_t unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::rejectCause

8.792.2.7 LITE_TYPE_OF_SERVICE_DOMAIN unpack_nas_SLQSNasNetworkRejectCallback_Ind_t::serviceDomain

8.793 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

8.793.1 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.

8.793.2 Field Documentation

8.793.2.1 swi_uint256_t unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::ParamPresenceMask

8.793.2.2 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pDayltSavAdj

8.793.2.3 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pRadioInterface

8.793.2.4 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pTimeZone

8.793.2.5 nas_UniversalTime unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::universalTime

8.794 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

8.794.1 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**
pTDSCDMASigInfoExt	• See tdscdmaSigInfoExt for more information. • Bit to check in ParamPresenceMask - **22**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.794.2 Field Documentation

8.794.2.1 swi_uint256_t unpack_nas_SLQSNasSigInfoCallback_ind_t::ParamPresenceMask

8.794.2.2 cdmaSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pCDMASigInfo

8.794.2.3 int8_t* unpack_nas_SLQSNasSigInfoCallback_ind_t::pGSMSigInfo

8.794.2.4 hdrSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pHDRSigInfo

8.794.2.5 lteSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pLTESigInfo

8.794.2.6 int8_t* unpack_nas_SLQSNasSigInfoCallback_ind_t::pRscp

8.794.2.7 tdscdmaSigInfoExt∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pTDSCDMASigInfoExt

8.794.2.8 cdmaSSInfo∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pWCDMASigInfo

8.795 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN ∗ pWcdmaUARFCN
- nas_lteEARFCN ∗ pLteEARFCN
- nas_ltePCI ∗ pLtePCI
- swi_uint256_t ParamPresenceMask

8.795.1 Detailed Description

Structure for Get Channel Lock unpack.

Parameters

pWcdmaUARFCN	[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**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.795.2 Field Documentation

8.795.2.1 swi_uint256_t unpack_nas_SLQSNASSwiGetChannelLock_t::ParamPresenceMask

8.795.2.2 nas_lteEARFCN∗ unpack_nas_SLQSNASSwiGetChannelLock_t::pLteEARFCN

8.795.2.3 nas_ltePCI∗ unpack_nas_SLQSNASSwiGetChannelLock_t::pLtePCI

8.795.2.4 nas_wcdmaUARFCN∗ unpack_nas_SLQSNASSwiGetChannelLock_t::pWcdmaUARFCN

8.796 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference

Data Fields

- nas_CommInfo commonInfo
- nas_LTEInfo ∗ pLTEInfo
- swi_uint256_t ParamPresenceMask

8.796.1 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.

8.796.2 Field Documentation

8.796.2.1 nas_CommInfo unpack_nas_SLQSNasSwiModemStatus_t::commonInfo

8.796.2.2 swi_uint256_t unpack_nas_SLQSNasSwiModemStatus_t::ParamPresenceMask

8.796.2.3 nas_LTEInfo* unpack_nas_SLQSNasSwiModemStatus_t::pLTEInfo

8.797 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSwiOTAMessageInd Info
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.797.1 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.

8.797.2 Field Documentation

8.797.2.1 NASQmiCbkNasSwiOTAMessageInd unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Info

8.797.2.2 swi_uint256_t unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::ParamPresenceMask

8.797.2.3 uint16_t unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Tlvresult

8.798 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

8.798.1 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.

8.798.2 Field Documentation

8.798.2.1 swi_uint256_t unpack_nas_SLQSNasTimerCallback_ind_t::ParamPresenceMask

8.798.2.2 char unpack_nas_SLQSNasTimerCallback_ind_t::t3396_apn[101]

8.798.2.3 uint8_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_plmn_id[3]

8.798.2.4 uint32_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_val

8.799 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSystemSelPrefInd Info
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.799.1 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.

8.799.2 Field Documentation

8.799.2.1 NASQmiCbkNasSystemSelPrefInd unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Info

8.799.2.2 swi_uint256_t unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::ParamPresenceMask

8.799.2.3 uint16_t unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::TlvResult

8.800 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference

Data Fields

- uint16_t * pCurrentPersonality
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtocolSubtypeElement
- swi_uint256_t ParamPresenceMask

8.800.1 Detailed Description

Structure for HDRPersonality unpack.

Parameters

pCurrent-Personality[Out]	• Current active personality index. • Bit to check in ParamPresenceMask - **16**

pPersonalityList-Length[In/Out]	• Number of Personality Protocol Subtype contains in this response. • maximum input value is 3 • Bit to check in ParamPresenceMask - **17**
pProtocol-Subtype-Element[Out]	• See nas_protocolSubtypeElement for more information. • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.800.2 Field Documentation

8.800.2.1 swi_uint256_t unpack_nas_SLQSSwiGetHDRPersonality_t::ParamPresenceMask

8.800.2.2 uint16_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pCurrentPersonality

8.800.2.3 uint8_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pPersonalityListLength

8.800.2.4 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRPersonality_t::pProtocolSubtypeElement

8.801 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference

Data Fields

- uint16_t * pCurrentPrsnlty
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtoSubTypElmnt
- uint64_t * pAppSubType
- swi_uint256_t ParamPresenceMask

8.801.1 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.

8.801.2 Field Documentation

8.801.2.1 uint64_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pAppSubType

8.801.2.2 swi_uint256_t unpack_nas_SLQSSwiGetHDRProtSubtype_t::ParamPresenceMask

8.801.2.3 uint16_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pCurrentPrsnlty

8.801.2.4 uint8_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pPersonalityListLength

8.801.2.5 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pProtoSubTypElmnt

8.802 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference

Data Fields

- nas_DRCParams * pDRCParams
- uint8_t * pUATI
- nas_PilotSetData * pPilotSetData
- swi_uint256_t ParamPresenceMask

8.802.1 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.

8.802.2 Field Documentation

8.802.2.1 swi_uint256_t unpack_nas_SLQSSwiGetHRPDStats_t::ParamPresenceMask

8.802.2.2 nas_DRCParams* unpack_nas_SLQSSwiGetHRPDStats_t::pDRCParams

8.802.2.3 nas_PilotSetData* unpack_nas_SLQSSwiGetHRPDStats_t::pPilotSetData

8.802.2.4 uint8_t* unpack_nas_SLQSSwiGetHRPDStats_t::pUATI

8.803 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference

Data Fields

- uint8_t ValidityCW0
- uint8_t CQIValueCW0
- uint8_t ValidityCW1
- uint8_t CQIValueCW1
- swi_uint256_t ParamPresenceMask

8.803.1 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.

8.803.2 Field Documentation

8.803.2.1 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW0

8.803.2.2 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW1

8.803.2.3 swi_uint256_t unpack_nas_SLQSSwiGetLteCQI_t::ParamPresenceMask

8.803.2.4 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW0

8.803.2.5 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW1

8.804 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference

Data Fields

- nas_SccRxInfo * pSccRxInfo
- swi_uint256_t ParamPresenceMask

8.804.1 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**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.804.2 Field Documentation

8.804.2.1 swi_uint256_t unpack_nas_SLQSSwiGetLteSccRxInfo_t::ParamPresenceMask

8.804.2.2 nas_SccRxInfo* unpack_nas_SLQSSwiGetLteSccRxInfo_t::pSccRxInfo

8.805 unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t Struct Reference

Data Fields

- nas_HDRPersonality_Ind_Data * pHDRPersInd
- swi_uint256_t ParamPresenceMask

8.805.1 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.

8.805.2 Field Documentation

8.805.2.1 swi_uint256_t unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t::ParamPresenceMask

8.805.2.2 nas_HDRPersonality_Ind_Data* unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t::pHDRPersInd

8.806 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

8.806.1 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.

8.806.2 Field Documentation

8.806.2.1 swi_uint256_t unpack_nas_SLQSSwiNetworkDebug_t::ParamPresenceMask

8.806.2.2 nas_DataStatusDetail* unpack_nas_SLQSSwiNetworkDebug_t::pDataStatusDetail

8.806.2.3 nas_DeviceConfigDetail* unpack_nas_SLQSSwiNetworkDebug_t::pDeviceConfigDetail

8.806.2.4 nas_NetworkStat1x* unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStat1x

8.806.2.5 nas_NetworkStatEVDO* unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStatEVDO

8.806.2.6 uint8_t* unpack_nas_SLQSSwiNetworkDebug_t::pObjectVer

8.807 unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t Struct Reference

Data Fields

- nas_RankIndicatorTlv rankIndicatorTlv
- swi_uint256_t ParamPresenceMask

8.807.1 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.

8.807.2 Field Documentation

8.807.2.1 swi_uint256_t unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t::ParamPresenceMask

8.807.2.2 nas_RankIndicatorTlv unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t::rankIndicatorTlv

8.808 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

8.808.1 Detailed Description

This structure contains unpack system information callback indication 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**
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.

8.808.2 Field Documentation

8.808.2.1 **nas_AddCDMASysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddCDMASysInfo

8.808.2.2 **nas_AddSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddGSMSysInfo

8.808.2.3 uint16_t* unpack_nas_SLQSSysInfoCallback_ind_t::pAddHDRSysInfo

8.808.2.4 uint16_t* unpack_nas_SLQSSysInfoCallback_ind_t::pAddLTESysInfo

8.808.2.5 **nas_AddSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pAddWCDMASysInfo

8.808.2.6 **swi_uint256_t** unpack_nas_SLQSSysInfoCallback_ind_t::ParamPresenceMask

8.808.2.7 **nas_SrvStatusInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASrvStatusInfo

8.808.2.8 **nas_CDMASysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASysInfo

8.808.2.9 **nas_CallBarringSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCallBarringSysInfo

8.808.2.10 uint8_t* unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCipherDomainSysInfo

8.808.2.11 **nas_GSMSrvStatusInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSrvStatusInfo

8.808.2.12 **nas_GSMSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSysInfo

8.808.2.13 **nas_SrvStatusInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSrvStatusInfo

8.808.2.14 **nas_HDRSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSysInfo

8.808.2.15 **nas_LteCiotOpModeTlv*** unpack_nas_SLQSSysInfoCallback_ind_t::pLteCiotOpModeTlv

8.808.2.16 **nas_GSMSrvStatusInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pLTESrvStatusInfo

8.808.2.17 **nas_LTESysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pLTESysInfo

8.808.2.18 uint8_t* unpack_nas_SLQSSysInfoCallback_ind_t::pLTEVoiceSupportSysInfo

8.808.2.19 **nas_NR5GCellStatusTlv*** unpack_nas_SLQSSysInfoCallback_ind_t::pNR5GCellStatus

8.808.2.20 **nas_NR5GSerStatTlv*** unpack_nas_SLQSSysInfoCallback_ind_t::pNR5GSerStatTlv

8.808.2.21 **nas_NR5GSystemInfoTlv*** unpack_nas_SLQSSysInfoCallback_ind_t::pNR5GSystemInfoTlv

8.808.2.22 uint8_t* unpack_nas_SLQSSysInfoCallback_ind_t::pSysInfoNoChange

8.808.2.23 **nas_CallBarringSysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACallBarringSysInfo

8.808.2.24 uint8_t* unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACipherDomainSysInfo

8.808.2.25 **nas_GSMSrvStatusInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASrvStatusInfo

8.808.2.26 **nas_WCDMASysInfo*** unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASysInfo

8.809 unpack_omaDmConfigTlv_t Struct Reference

Data Fields

- uint8_t state
- uint8_t userInputReq
- uint16_t userInputTimeout
- uint16_t alertmsglength
- uint8_t alertmsg [256]

8.809.1 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)

8.809.2 Field Documentation

8.809.2.1 uint8_t unpack_omaDmConfigTlv_t::alertmsg[256]

8.809.2.2 uint16_t unpack_omaDmConfigTlv_t::alertmsglength

8.809.2.3 uint8_t unpack_omaDmConfigTlv_t::state

8.809.2.4 uint8_t unpack_omaDmConfigTlv_t::userInputReq

8.809.2.5 uint16_t unpack_omaDmConfigTlv_t::userInputTimeout

8.810 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

8.810.1 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

8.810.2 Field Documentation

8.810.2.1 uint8_t unpack_omaDmFotaTlv_t::description[256]

8.810.2.2 uint16_t unpack_omaDmFotaTlv_t::descriptionlength

8.810.2.3 uint32_t unpack_omaDmFotaTlv_t::fwdloadsize

8.810.2.4 uint32_t unpack_omaDmFotaTlv_t::fwloadComplete

8.810.2.5 uint16_t unpack_omaDmFotaTlv_t::namelength

8.810.2.6 uint8_t unpack_omaDmFotaTlv_t::package_name[256]

8.810.2.7 uint8_t unpack_omaDmFotaTlv_t::sessionType

8.810.2.8 uint8_t unpack_omaDmFotaTlv_t::severity

8.810.2.9 uint8_t unpack_omaDmFotaTlv_t::state

8.810.2.10 uint16_t unpack_omaDmFotaTlv_t::updateCompleteStatus

8.810.2.11 uint8_t unpack_omaDmFotaTlv_t::userInputReq

8.810.2.12 uint16_t unpack_omaDmFotaTlv_t::userInputTimeout

8.810.2.13 uint8_t unpack_omaDmFotaTlv_t::version[256]

8.810.2.14 uint16_t unpack_omaDmFotaTlv_t::versionlength

8.811 unpack_omaDmNotificationsTlv_t Struct Reference

Data Fields

- uint8_t notification
- uint16_t sessionStatus

8.811.1 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.

8.811.2 Field Documentation

8.811.2.1 uint8_t unpack_omaDmNotificationsTlv_t::notification

8.811.2.2 uint16_t unpack_omaDmNotificationsTlv_t::sessionStatus

8.812 unpack_pds_GetPDSDefaults_t Struct Reference

Data Fields

- uint32_t * pOperation
- uint8_t * pTimeout
- uint32_t * pInterval
- uint32_t * pAccuracy
- swi_uint256_t ParamPresenceMask

8.812.1 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.

8.812.2 Field Documentation

8.812.2.1 uint32_t* unpack_pds_GetPDSDefaults_t::pAccuracy

8.812.2.2 swi_uint256_t unpack_pds_GetPDSDefaults_t::ParamPresenceMask

8.812.2.3 uint32_t* unpack_pds_GetPDSDefaults_t::pInterval

8.812.2.4 uint32_t* unpack_pds_GetPDSDefaults_t::pOperation

8.812.2.5 uint8_t* unpack_pds_GetPDSDefaults_t::pTimeout

8.813 unpack_pds_GetPDSState_t Struct Reference

Data Fields

- uint32_t * pEnabledStatus
- uint32_t * pTrackingStatus
- swi_uint256_t ParamPresenceMask

8.813.1 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.

8.813.2 Field Documentation

8.813.2.1 swi_uint256_t unpack_pds_GetPDSState_t::ParamPresenceMask

8.813.2.2 uint32_t* unpack_pds_GetPDSState_t::pEnabledStatus

8.813.2.3 uint32_t* unpack_pds_GetPDSState_t::pTrackingStatus

8.814 unpack_pds_GetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t * pbAuto
- swi_uint256_t ParamPresenceMask

8.814.1 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.

8.814.2 Field Documentation

8.814.2.1 swi_uint256_t unpack_pds_GetPortAutomaticTracking_t::ParamPresenceMask

8.814.2.2 uint32_t* unpack_pds_GetPortAutomaticTracking_t::pbAuto

8.815 unpack_pds_GetServiceAutomaticTracking_t Struct Reference

Data Fields

- uint32_t * pbAuto
- swi_uint256_t ParamPresenceMask

8.815.1 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.

8.815.2 Field Documentation

8.815.2.1 swi_uint256_t unpack_pds_GetServiceAutomaticTracking_t::ParamPresenceMask

8.815.2.2 uint32_t* unpack_pds_GetServiceAutomaticTracking_t::pbAuto

8.816 unpack_pds_GetXTRAAutomaticDownload_t Struct Reference

Data Fields

- uint32_t * pbEnabled
- uint16_t * pInterval
- swi_uint256_t ParamPresenceMask

8.816.1 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.

8.816.2 Field Documentation

8.816.2.1 swi_uint256_t unpack_pds_GetXTRAAutomaticDownload_t::ParamPresenceMask

8.816.2.2 uint32_t* unpack_pds_GetXTRAAutomaticDownload_t::pbEnabled

8.816.2.3 uint16_t* unpack_pds_GetXTRAAutomaticDownload_t::pInterval

8.817 unpack_pds_GetXTRANetwork_t Struct Reference

Data Fields

- uint32_t * pPreference
- swi_uint256_t ParamPresenceMask

8.817.1 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.

8.817.2 Field Documentation

8.817.2.1 swi_uint256_t unpack_pds_GetXTRANetwork_t::ParamPresenceMask

8.817.2.2 uint32_t* unpack_pds_GetXTRANetwork_t::pPreference

8.818 unpack_pds_GetXTRAValidity_t Struct Reference

Data Fields

- uint16_t * pGPSWeek

- uint16_t * pGPSWeekOffset
- uint16_t * pDuration
- swi_uint256_t ParamPresenceMask

8.818.1 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.

8.818.2 Field Documentation

8.818.2.1 swi_uint256_t unpack_pds_GetXTRAValidity_t::ParamPresenceMask

8.818.2.2 uint16_t* unpack_pds_GetXTRAValidity_t::pDuration

8.818.2.3 uint16_t* unpack_pds_GetXTRAValidity_t::pGPSWeek

8.818.2.4 uint16_t* unpack_pds_GetXTRAValidity_t::pGPSWeekOffset

8.819 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

8.819.1 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.

8.819.2 Field Documentation

8.819.2.1 double unpack_pds_SetEventReport_Ind_t::dLatitude

8.819.2.2 double unpack_pds_SetEventReport_Ind_t::dLongitude

8.819.2.3 uint8_t unpack_pds_SetEventReport_Ind_t::has_dLatitude

8.819.2.4 uint8_t unpack_pds_SetEventReport_Ind_t::has_dLongitude

8.819.2.5 uint8_t unpack_pds_SetEventReport_Ind_t::has_PositionDataNMEA

8.819.2.6 uint8_t unpack_pds_SetEventReport_Ind_t::has_posSrc

8.819.2.7 uint8_t unpack_pds_SetEventReport_Ind_t::has_SessionStatus

8.819.2.8 swi_uint256_t unpack_pds_SetEventReport_Ind_t::ParamPresenceMask

8.819.2.9 char unpack_pds_SetEventReport_Ind_t::PositionDataNMEA[200]

8.819.2.10 uint32_t unpack_pds_SetEventReport_Ind_t::posSrc

8.819.2.11 uint8_t unpack_pds_SetEventReport_Ind_t::SessionStatus

8.820 unpack_pds_SetPdsState_Ind_t Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t EnabledStatus
- uint32_t TrackingStatus
- swi_uint256_t ParamPresenceMask

8.820.1 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.

8.820.2 Field Documentation

8.820.2.1 uint32_t unpack_pds_SetPdsState_Ind_t::EnabledStatus

8.820.2.2 swi_uint256_t unpack_pds_SetPdsState_Ind_t::ParamPresenceMask

8.820.2.3 uint16_t unpack_pds_SetPdsState_Ind_t::TlvPresent

8.820.2.4 uint32_t unpack_pds_SetPdsState_Ind_t::TrackingStatus

8.821 unpack_pds_SLQSGetAGPSConfig_t Struct Reference

Data Fields

- uint32_t * pServerAddress
- uint32_t * pServerPort
- uint8_t * pServerURL
- uint8_t * pServerURLLength
- swi_uint256_t ParamPresenceMask

8.821.1 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.

8.821.2 Field Documentation

8.821.2.1 swi_uint256_t unpack_pds_SLQSGetAGPSConfig_t::ParamPresenceMask

8.821.2.2 uint32_t* unpack_pds_SLQSGetAGPSConfig_t::pServerAddress

8.821.2.3 uint32_t* unpack_pds_SLQSGetAGPSConfig_t::pServerPort

8.821.2.4 uint8_t* unpack_pds_SLQSGetAGPSConfig_t::pServerURL

8.821.2.5 uint8_t* unpack_pds_SLQSGetAGPSConfig_t::pServerURLLength

8.822 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

8.822.1 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**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.822.2 Field Documentation

8.822.2.1 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::Altitude

8.822.2.2 uint8_t unpack_pds_SLQSGetGPSStateInfo_t::EngineState

8.822.2.3 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_almanac_sv_msk

8.822.2.4 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_ephemeris_sv_msk

8.822.2.5 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_health_sv_msk

8.822.2.6 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_visible_sv_msk

8.822.2.7 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_almanac_sv_msk

8.822.2.8 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_ephemeris_sv_msk

8.822.2.9 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_health_sv_msk

8.822.2.10 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_visible_sv_msk

8.822.2.11 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::HorizontalUncertainty

8.822.2.12 uint8_t unpack_pds_SLQSGetGPSStateInfo_t::Iono_valid

8.822.2.13 uint64_t unpack_pds_SLQSGetGPSStateInfo_t::Latitude

8.822.2.14 uint64_t unpack_pds_SLQSGetGPSStateInfo_t::Longitude

8.822.2.15 swi_uint256_t unpack_pds_SLQSGetGPSStateInfo_t::ParamPresenceMask

8.822.2.16 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_almanac_sv_msk

8.822.2.17 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_ephemeris_sv_msk

8.822.2.18 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_health_sv_msk

8.822.2.19 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_visible_sv_msk

8.822.2.20 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::Time_uncert_ms

8.822.2.21 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::TimeStmp_gps_week

8.822.2.22 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::TimeStmp_tow_ms

8.822.2.23 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::ValidMask

8.822.2.24 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::VerticalUncertainty

8.822.2.25 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_start_gps_minutes

8.822.2.26 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_start_gps_week

8.822.2.27 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_valid_duration_hours

8.823 unpack_qmi_t Struct Reference

Data Fields

- enum msgtype type
- uint16_t msgid
- uint16_t xid

8.823.1 Detailed Description

qmi response context

Parameters

out	*type*	message type
out	*msgid*	message id
out	*xid*	transaction id

8.823.2 Field Documentation

8.823.2.1 uint16_t unpack_qmi_t::msgid

8.823.2.2 enum msgtype unpack_qmi_t::type

8.823.2.3 uint16_t unpack_qmi_t::xid

8.824 unpack_qos_BindDataPort_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.824.1 Detailed Description

This structure contains unpack Qos BindDataPort.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.824.2 Field Documentation

8.824.2.1 swi_uint256_t unpack_qos_BindDataPort_t::ParamPresenceMask

8.825 unpack_qos_dataRate_t Struct Reference

Data Fields

- uint32_t dataRateMax
- uint32_t guaranteedRate

8.825.1 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)

8.825.2 Field Documentation

8.825.2.1 uint32_t unpack_qos_dataRate_t::dataRateMax

8.825.2.2 uint32_t unpack_qos_dataRate_t::guaranteedRate

8.826 unpack_qos_IPv4Addr_t Struct Reference

Data Fields

- uint32_t addr
- uint32_t subnetMask

8.826.1 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

8.826.2 Field Documentation

8.826.2.1 uint32_t unpack_qos_IPv4Addr_t::addr

8.826.2.2 uint32_t unpack_qos_IPv4Addr_t::subnetMask

8.827 unpack_qos_IPv6Addr_t Struct Reference

Data Fields

- uint8_t addr [16]

- uint8_t prefixLen

8.827.1 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.

8.827.2 Field Documentation

8.827.2.1 uint8_t unpack_qos_IPv6Addr_t::addr[16]

8.827.2.2 uint8_t unpack_qos_IPv6Addr_t::prefixLen

8.828 unpack_qos_IPv6TrafCls_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.828.1 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

8.828.2 Field Documentation

8.828.2.1 uint8_t unpack_qos_IPv6TrafCls_t::mask

8.828.2.2 uint8_t unpack_qos_IPv6TrafCls_t::val

8.829 unpack_qos_pktErrRate_t Struct Reference

Data Fields

- uint16_t multiplier

- uint16_t exponent

8.829.1 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)

8.829.2 Field Documentation

8.829.2.1 uint16_t unpack_qos_pktErrRate_t::exponent

8.829.2.2 uint16_t unpack_qos_pktErrRate_t::multiplier

8.830 unpack_qos_Port_t Struct Reference

Data Fields

- uint16_t port
- uint16_t range

8.830.1 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

8.830.2 Field Documentation

8.830.2.1 uint16_t unpack_qos_Port_t::port

8.830.2.2 uint16_t unpack_qos_Port_t::range

8.831 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

8.831.1 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_TxQFlow-Granted_-Available	• TRUE if optional TxQFlowGranted is available
TxQFlow-Granted	• The Tx Qos flow granted, please check unpack_qos_swiQosFlow_t for more information
is_RxQFlow-Granted_-Available	• TRUE if optional RxQFlowGranted is available
RxQFlow-Granted	• 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.

8.831.2 Field Documentation

8.831.2.1 uint8_t unpack_qos_QosFlowInfo_t::BearerID

8.831.2.2 uint8_t unpack_qos_QosFlowInfo_t::is_RxQFlowGranted_Available

8.831.2.3 uint8_t unpack_qos_QosFlowInfo_t::is_TxQFlowGranted_Available

8.831.2.4 uint8_t unpack_qos_QosFlowInfo_t::NumRxFilters

8.831.2.5 uint8_t unpack_qos_QosFlowInfo_t::NumTxFilters

8.831.2.6 unpack_qos_QosFlowInfoState_t unpack_qos_QosFlowInfo_t::QFlowState

8.831.2.7 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::RxQFilter[25]

8.831.2.8 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::RxQFlowGranted

8.831.2.9 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::TxQFilter[25]

8.831.2.10 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::TxQFlowGranted

8.832 unpack_qos_QosFlowInfoState_t Struct Reference

Data Fields

- uint32_t id
- uint8_t isNewFlow
- uint8_t state

8.832.1 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

8.832.2 Field Documentation

8.832.2.1 uint32_t unpack_qos_QosFlowInfoState_t::id

8.832.2.2 uint8_t unpack_qos_QosFlowInfoState_t::isNewFlow

8.832.2.3 uint8_t unpack_qos_QosFlowInfoState_t::state

8.833 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference

Data Fields

- uint8_t NWQoSStatus
- swi_uint256_t ParamPresenceMask

8.833.1 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.

8.833.2 Field Documentation

8.833.2.1 uint8_t unpack_qos_SLQSQosGetNetworkStatus_t::NWQoSStatus

8.833.2.2 swi_uint256_t unpack_qos_SLQSQosGetNetworkStatus_t::ParamPresenceMask

8.834 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

8.834.1 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.

8.834.2 Field Documentation

8.834.2.1 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl

8.834.2.2 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext

8.834.2.3 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext2

8.834.2.4 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul

8.834.2.5 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext

8.834.2.6 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext2

8.834.2.7 uint32_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.834.2.8 swi_uint256_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ParamPresenceMask

8.835 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

8.835.1 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.

8.835.2 Field Documentation

8.835.2.1 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::apnId

8.835.2.2 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::numQosFlow

8.835.2.3 swi_uint256_t unpack_qos_SLQSQosSwiReadDataStats_t::ParamPresenceMask

8.835.2.4 unpack_QosFlowStat_t unpack_qos_SLQSQosSwiReadDataStats_t::qosFlow[10]

8.835.2.5 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_bytes

8.835.2.6 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_pkt

8.835.2.7 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes

8.835.2.8 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes_drp

8.835.2.9 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt

8.835.2.10 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt_drp

8.836 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference

Data Fields

- uint8_t NumFlows
- unpack_qos_QosFlowInfo_t QosFlowInfo [8]
- swi_uint256_t ParamPresenceMask

8.836.1 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.

8.836.2 Field Documentation

8.836.2.1 uint8_t unpack_qos_SLQSSetQosEventCallback_ind_t::NumFlows

8.836.2.2 swi_uint256_t unpack_qos_SLQSSetQosEventCallback_ind_t::ParamPresenceMask

8.836.2.3 unpack_qos_QosFlowInfo_t unpack_qos_SLQSSetQosEventCallback_ind_t::QosFlowInfo[8]

8.837 unpack_qos_SLQSSetQosEventCallback_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.837.1 Detailed Description

This structure contains unpack QOS SLQSSetQosEventCallback. param ParamPresenceMask

- bitmask representation to indicate valid parameters.

8.837.2 Field Documentation

8.837.2.1 swi_uint256_t unpack_qos_SLQSSetQosEventCallback_t::ParamPresenceMask

8.838 unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference

Data Fields

- uint8_t status
- swi_uint256_t ParamPresenceMask

8.838.1 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

8.838.2 Field Documentation

8.838.2.1 swi_uint256_t unpack_qos_SLQSSetQosNWStatusCallback_ind_t::ParamPresenceMask

8.838.2.2 uint8_t unpack_qos_SLQSSetQosNWStatusCallback_ind_t::status

8.839 unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference

Data Fields

- uint16_t event
- swi_uint256_t ParamPresenceMask

8.839.1 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.

8.839.2 Field Documentation

8.839.2.1 uint16_t unpack_qos_SLQSSetQosPriEventCallback_ind_t::event

8.839.2.2 swi_uint256_t unpack_qos_SLQSSetQosPriEventCallback_ind_t::ParamPresenceMask

8.840 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

8.840.1 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.

8.840.2 Field Documentation

8.840.2.1 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::event

8.840.2.2 uint32_t unpack_qos_SLQSSetQosStatusCallback_ind_t::id

8.840.2.3 swi_uint256_t unpack_qos_SLQSSetQosStatusCallback_ind_t::ParamPresenceMask

8.840.2.4 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::reason

8.840.2.5 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::status

8.841 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

8.841.1 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

8.841.2 Field Documentation

8.841.2.1 uint32_t unpack_qos_swiQosFilter_t::EspSpi

8.841.2.2 uint16_t unpack_qos_swiQosFilter_t::Id

8.841.2.3 uint8_t unpack_qos_swiQosFilter_t::index

8.841.2.4 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4DstAddr

8.841.2.5 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4SrcAddr

8.841.2.6 unpack_qos_Tos_t unpack_qos_swiQosFilter_t::IPv4Tos

8.841.2.7 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6DstAddr

8.841.2.8 uint32_t unpack_qos_swiQosFilter_t::IPv6Label

8.841.2.9 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6SrcAddr

8.841.2.10 unpack_qos_IPv6TrafCls_t unpack_qos_swiQosFilter_t::IPv6TrafCls

8.841.2.11 uint8_t unpack_qos_swiQosFilter_t::is_EspSpi_Available

8.841.2.12 uint8_t unpack_qos_swiQosFilter_t::is_Id_Available

8.841.2.13 uint8_t unpack_qos_swiQosFilter_t::is_IPv4DstAddr_Available

8.841.2.14 uint8_t unpack_qos_swiQosFilter_t::is_IPv4SrcAddr_Available

8.841.2.15 uint8_t unpack_qos_swiQosFilter_t::is_IPv4Tos_Available

8.841.2.16 uint8_t unpack_qos_swiQosFilter_t::is_IPv6DstAddr_Available

8.841.2.17 uint8_t unpack_qos_swiQosFilter_t::is_IPv6Label_Available

8.841.2.18 uint8_t unpack_qos_swiQosFilter_t::is_IPv6SrcAddr_Available

8.841.2.19 uint8_t unpack_qos_swiQosFilter_t::is_IPv6TrafCls_Available

8.841.2.20 uint8_t unpack_qos_swiQosFilter_t::is_NxtHdrProto_Available

8.841.2.21 uint8_t unpack_qos_swiQosFilter_t::is_Precedence_Available

8.841.2.22 uint8_t unpack_qos_swiQosFilter_t::is_TCPDstPort_Available

8.841.2.23 uint8_t unpack_qos_swiQosFilter_t::is_TCPSrcPort_Available

8.841.2.24 uint8_t unpack_qos_swiQosFilter_t::is_TranDstPort_Available

8.841.2.25 uint8_t unpack_qos_swiQosFilter_t::is_TranSrcPort_Available

8.841.2.26 uint8_t unpack_qos_swiQosFilter_t::is_UDPDstPort_Available

8.841.2.27 uint8_t unpack_qos_swiQosFilter_t::is_UDPSrcPort_Available

8.841.2.28 uint8_t unpack_qos_swiQosFilter_t::NxtHdrProto

8.841.2.29 uint16_t unpack_qos_swiQosFilter_t::Precedence

8.841.2.30 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPDstPort

8.841.2.31 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPSrcPort

8.841.2.32 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranDstPort

8.841.2.33 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranSrcPort

8.841.2.34 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPDstPort

8.841.2.35 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPSrcPort

8.841.2.36 uint8_t unpack_qos_swiQosFilter_t::version

8.842 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

8.842.1 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

8.842.2 Field Documentation

8.842.2.1 unpack_qos_dataRate_t unpack_qos_swiQosFlow_t::DataRate

8.842.2.2 uint8_t unpack_qos_swiQosFlow_t::index

8.842.2.3 uint8_t unpack_qos_swiQosFlow_t::is_DataRate_Available

8.842.2.4 uint8_t unpack_qos_swiQosFlow_t::is_Jitter_Available

8.842.2.5 uint8_t unpack_qos_swiQosFlow_t::is_Latency_Available

8.842.2.6 uint8_t unpack_qos_swiQosFlow_t::is_LteQci_Available

8.842.2.7 uint8_t unpack_qos_swiQosFlow_t::is_MaxAllowedPktSz_Available

8.842.2.8 uint8_t unpack_qos_swiQosFlow_t::is_MinPolicedPktSz_Available

8.842.2.9 uint8_t unpack_qos_swiQosFlow_t::is_PktErrRate_Available

8.842.2.10 uint8_t unpack_qos_swiQosFlow_t::is_ProfileId3GPP2_Available

8.842.2.11 uint8_t unpack_qos_swiQosFlow_t::is_TokenBucket_Available

8.842.2.12 uint8_t unpack_qos_swiQosFlow_t::is_TrafficClass_Available

8.842.2.13 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPP2Pri_Available

8.842.2.14 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPImCn_Available

8.842.2.15 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPResResidualBER_Available

8.842.2.16 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPSigInd_Available

8.842.2.17 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPTraHdlPri_Available

8.842.2.18 uint32_t unpack_qos_swiQosFlow_t::Jitter

8.842.2.19 uint32_t unpack_qos_swiQosFlow_t::Latency

8.842.2.20 uint8_t unpack_qos_swiQosFlow_t::LteQci

8.842.2.21 uint32_t unpack_qos_swiQosFlow_t::MaxAllowedPktSz

8.842.2.22 uint32_t unpack_qos_swiQosFlow_t::MinPolicedPktSz

8.842.2.23 unpack_qos_pktErrRate_t unpack_qos_swiQosFlow_t::PktErrRate

8.842.2.24 uint16_t unpack_qos_swiQosFlow_t::ProfileId3GPP2

8.842.2.25 unpack_qos_tokenBucket_t unpack_qos_swiQosFlow_t::TokenBucket

8.842.2.26 uint8_t unpack_qos_swiQosFlow_t::TrafficClass

8.842.2.27 uint8_t unpack_qos_swiQosFlow_t::val_3GPP2Pri

8.842.2.28 uint8_t unpack_qos_swiQosFlow_t::val_3GPPImCn

8.842.2.29 uint16_t unpack_qos_swiQosFlow_t::val_3GPPResResidualBER

8.842.2.30 uint8_t unpack_qos_swiQosFlow_t::val_3GPPSigInd

8.842.2.31 uint8_t unpack_qos_swiQosFlow_t::val_3GPPTraHdlPri

8.843 unpack_qos_tokenBucket_t Struct Reference

Data Fields

- uint32_t peakRate
- uint32_t tokenRate
- uint32_t bucketSz

8.843.1 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

8.843.2 Field Documentation

8.843.2.1 uint32_t unpack_qos_tokenBucket_t::bucketSz

8.843.2.2 uint32_t unpack_qos_tokenBucket_t::peakRate

8.843.2.3 uint32_t unpack_qos_tokenBucket_t::tokenRate

8.844 unpack_qos_Tos_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.844.1 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.

8.844.2 Field Documentation

8.844.2.1 uint8_t unpack_qos_Tos_t::mask

8.844.2.2 uint8_t unpack_qos_Tos_t::val

8.845 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

8.845.1 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

8.845.2 Field Documentation

8.845.2.1 uint32_t unpack_QosFlowStat_t::bearerId

8.845.2.2 uint64_t unpack_QosFlowStat_t::tx_bytes

8.845.2.3 uint64_t unpack_QosFlowStat_t::tx_bytes_drp

8.845.2.4 uint32_t unpack_QosFlowStat_t::tx_pkt

8.845.2.5 uint32_t unpack_QosFlowStat_t::tx_pkt_drp

8.846 unpack_result_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.846.1 Detailed Description

unpack result struct

Parameters

`out`	*Tlvresult*	
`out`	*ParamPresence-Mask*	Parameter presence bit mask

8.846.2 Field Documentation

8.846.2.1 swi_uint256_t unpack_result_t::ParamPresenceMask

8.846.2.2 uint16_t unpack_result_t::Tlvresult

8.847 unpack_rms_GetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask
- swi_uint256_t ParamPresenceMask

8.847.1 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.

8.847.2 Field Documentation

8.847.2.1 uint32_t unpack_rms_GetSMSWake_t::enabled

8.847.2.2 swi_uint256_t unpack_rms_GetSMSWake_t::ParamPresenceMask

8.847.2.3 uint32_t unpack_rms_GetSMSWake_t::wake_mask

8.848 unpack_rms_SetSMSWake_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.848.1 Detailed Description

This structure contains unpack Set SMS Wake.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.848.2 Field Documentation

8.848.2.1 swi_uint256_t unpack_rms_SetSMSWake_t::ParamPresenceMask

8.849 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

8.849.1 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**

8.849.2 Field Documentation

8.849.2.1 swi_uint256_t unpack_RMTransferStatistics_ind_t::ParamPresenceMask

8.849.2.2 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxDropConutTlv

8.849.2.3 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::RxOkByteCountTlv

8.849.2.4 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxOkConutTlv

8.849.2.5 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxDropConutTlv

8.849.2.6 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::TxOkByteCountTlv

8.849.2.7 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxOkConutTlv

8.850 unpack_sar_SLQSGetRfSarState_t Struct Reference

Data Fields

- uint32_t * pSarRFState
- swi_uint256_t ParamPresenceMask

8.850.1 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.

8.850.2 Field Documentation

8.850.2.1 swi_uint256_t unpack_sar_SLQSGetRfSarState_t::ParamPresenceMask

8.850.2.2 uint32_t* unpack_sar_SLQSGetRfSarState_t::pSarRFState

8.851 unpack_sms_GetSMSCAddress_t Struct Reference

Data Fields

- uint8_t addressSize
- uint8_t * pSMSCAddress

- uint8_t typeSize
- uint8_t * pSMSCType
- swi_uint256_t ParamPresenceMask

8.851.1 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.

8.851.2 Field Documentation

8.851.2.1 uint8_t unpack_sms_GetSMSCAddress_t::addressSize

8.851.2.2 swi_uint256_t unpack_sms_GetSMSCAddress_t::ParamPresenceMask

8.851.2.3 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCAddress

8.851.2.4 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCType

8.851.2.5 uint8_t unpack_sms_GetSMSCAddress_t::typeSize

8.852 unpack_sms_SaveSMS_t Struct Reference

Data Fields

- uint32_t * pMessageIndex
- swi_uint256_t ParamPresenceMask

8.852.1 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.

8.852.2 Field Documentation

8.852.2.1 swi_uint256_t unpack_sms_SaveSMS_t::ParamPresenceMask

8.852.2.2 uint32_t* unpack_sms_SaveSMS_t::pMessageIndex

8.853 unpack_sms_SendSMS_t Struct Reference

Data Fields

- uint16_t messageID
- uint32_t messageFailureCode
- swi_uint256_t ParamPresenceMask

8.853.1 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.

8.853.2 Field Documentation

8.853.2.1 uint32_t unpack_sms_SendSMS_t::messageFailureCode

8.853.2.2 uint16_t unpack_sms_SendSMS_t::messageID

8.853.2.3 swi_uint256_t unpack_sms_SendSMS_t::ParamPresenceMask

8.854 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

8.854.1 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.

8.854.2 Field Documentation

8.854.2.1 eTWSPLMNInfoTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSPLMNTlv

8.854.2.2 sMSEtwsMessageTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSTlv

8.854.2.3 sMSOnIMSTlv unpack_sms_SetNewSMSCallback_ind_t::IMSTlv

8.854.2.4 messageModeTlv unpack_sms_SetNewSMSCallback_ind_t::MMTlv

8.854.2.5 newMTMessageTlv unpack_sms_SetNewSMSCallback_ind_t::NewMMTlv

8.854.2.6 swi_uint256_t unpack_sms_SetNewSMSCallback_ind_t::ParamPresenceMask

8.854.2.7 sMSCAddressTlv unpack_sms_SetNewSMSCallback_ind_t::SMSCTlv

8.854.2.8 transferRouteMessageTlv unpack_sms_SetNewSMSCallback_ind_t::TRMessageTlv

8.855 unpack_sms_SetNewSMSCallback_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.855.1 Detailed Description

This structure contains unpack set new SMS callback.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.855.2 Field Documentation

8.855.2.1 swi_uint256_t unpack_sms_SetNewSMSCallback_t::ParamPresenceMask

8.856 unpack_sms_SetSMSCAddress_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.856.1 Detailed Description

This structure contains unpack Set SMS Address parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.856.2 Field Documentation

8.856.2.1 swi_uint256_t unpack_sms_SetSMSCAddress_t::ParamPresenceMask

8.857 unpack_sms_SLQSDeleteSMS_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.857.1 Detailed Description

This structure contains unpack delete SMS parameters.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.857.2 Field Documentation

8.857.2.1 swi_uint256_t unpack_sms_SLQSDeleteSMS_t::ParamPresenceMask

8.858 unpack_sms_SLQSGetIndicationRegister_t Struct Reference

Data Fields

- sms_getIndicationReg * pGetIndicationRegInfo
- swi_uint256_t ParamPresenceMask

8.858.1 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.

8.858.2 Field Documentation

8.858.2.1 swi_uint256_t unpack_sms_SLQSGetIndicationRegister_t::ParamPresenceMask

8.858.2.2 sms_getIndicationReg* unpack_sms_SLQSGetIndicationRegister_t::pGetIndicationRegInfo

8.859 unpack_sms_SLQSGetMessageWaiting_t Struct Reference

Data Fields

- sms_getMsgWaitingInfo * pGetMsgWaitingInfoResp
- swi_uint256_t ParamPresenceMask

8.859.1 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.

8.859.2 Field Documentation

8.859.2.1 swi_uint256_t unpack_sms_SLQSGetMessageWaiting_t::ParamPresenceMask

8.859.2.2 sms_getMsgWaitingInfo* unpack_sms_SLQSGetMessageWaiting_t::pGetMsgWaitingInfoResp

8.860 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

8.860.1 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.

8.860.2 Field Documentation

8.860.2.1 uint8_t unpack_sms_SLQSGetSMS_t::message[2048]

8.860.2.2 uint32_t unpack_sms_SLQSGetSMS_t::messageFormat

8.860.2.3 uint32_t unpack_sms_SLQSGetSMS_t::messageSize

8.860.2.4 uint32_t unpack_sms_SLQSGetSMS_t::messageTag

8.860.2.5 swi_uint256_t unpack_sms_SLQSGetSMS_t::ParamPresenceMask

8.861 unpack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- sms_qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig
- sms_qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig
- swi_uint256_t ParamPresenceMask

8.861.1 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.

8.861.2 Field Documentation

8.861.2.1 swi_uint256_t unpack_sms_SLQSGetSmsBroadcastConfig_t::ParamPresenceMask

8.861.2.2 sms_qaQmi3GPPBroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pBroadcastConfig

8.861.2.3 sms_qaQmi3GPP2BroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pCDMABroadcastConfig

8.862 unpack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t messageListSize
- qmiSmsMessageList messageList [255]
- swi_uint256_t ParamPresenceMask

8.862.1 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.

8.862.2 Field Documentation

8.862.2.1 qmiSmsMessageList unpack_sms_SLQSGetSMSList_t::messageList[255]

8.862.2.2 uint32_t unpack_sms_SLQSGetSMSList_t::messageListSize

8.862.2.3 swi_uint256_t unpack_sms_SLQSGetSMSList_t::ParamPresenceMask

8.863 unpack_sms_SLQSGetTransLayerInfo_t Struct Reference

Data Fields

- sms_getTransLayerInfo * pGetTransLayerInfo
- swi_uint256_t ParamPresenceMask

8.863.1 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.

8.863.2 Field Documentation

8.863.2.1 swi_uint256_t unpack_sms_SLQSGetTransLayerInfo_t::ParamPresenceMask

8.863.2.2 sms_getTransLayerInfo* unpack_sms_SLQSGetTransLayerInfo_t::pGetTransLayerInfo

8.864 unpack_sms_SLQSGetTransNWRegInfo_t Struct Reference

Data Fields

- sms_getTransNWRegInfo * pGetTransNWRegInfo
- swi_uint256_t ParamPresenceMask

8.864.1 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.

8.864.2 Field Documentation

8.864.2.1 swi_uint256_t unpack_sms_SLQSGetTransNWRegInfo_t::ParamPresenceMask

8.864.2.2 sms_getTransNWRegInfo* unpack_sms_SLQSGetTransNWRegInfo_t::pGetTransNWRegInfo

8.865 unpack_sms_SLQSModifySMSStatus_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.865.1 Detailed Description

This structure contains unpack modify SMS status parameters.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.865.2 Field Documentation

8.865.2.1 swi_uint256_t unpack_sms_SLQSModifySMSStatus_t::ParamPresenceMask

8.866 unpack_sms_SLQSNWRegInfoCallback_ind_t Struct Reference

Data Fields

- uint8_t NWRegStat
- swi_uint256_t ParamPresenceMask

8.866.1 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

8.866.2 Field Documentation

8.866.2.1 uint8_t unpack_sms_SLQSNWRegInfoCallback_ind_t::NWRegStat

8.866.2.2 swi_uint256_t unpack_sms_SLQSNWRegInfoCallback_ind_t::ParamPresenceMask

8.867 unpack_sms_SLQSSendAsyncSMS_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.867.1 Detailed Description

This structure contains unpack SLQSSendAsyncSMS parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.867.2 Field Documentation

8.867.2.1 swi_uint256_t unpack_sms_SLQSSendAsyncSMS_t::ParamPresenceMask

8.868 unpack_sms_SLQSSetIndicationRegister_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.868.1 Detailed Description

This structure contains unpack SLQSSetIndicationRegister parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.868.2 Field Documentation

8.868.2.1 swi_uint256_t unpack_sms_SLQSSetIndicationRegister_t::ParamPresenceMask

8.869 unpack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.869.1 Detailed Description

This structure contains unpack SLQSSetSmsBroadcastActivation parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.869.2 Field Documentation

8.869.2.1 swi_uint256_t unpack_sms_SLQSSetSmsBroadcastActivation_t::ParamPresenceMask

8.870 unpack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.870.1 Detailed Description

This structure contains unpack SLQSSetSmsBroadcastConfig parameters.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.870.2 Field Documentation

8.870.2.1 swi_uint256_t unpack_sms_SLQSSetSmsBroadcastConfig_t::ParamPresenceMask

8.871 unpack_sms_SLQSSetSmsStorage_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.871.1 Detailed Description

This structure contains unpack SLQSSetSmsStorage parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.871.2 Field Documentation

8.871.2.1 swi_uint256_t unpack_sms_SLQSSetSmsStorage_t::ParamPresenceMask

8.872 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeResp * pMaxStorageSizeResp
- swi_uint256_t ParamPresenceMask

8.872.1 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.

8.872.2 Field Documentation

8.872.2.1 swi_uint256_t unpack_sms_SLQSSmsGetMaxStorageSize_t::ParamPresenceMask

8.872.2.2 sms_maxStorageSizeResp* unpack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeResp

8.873 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference

Data Fields

- sms_msgProtocolResp * pMessageProtocol
- swi_uint256_t ParamPresenceMask

8.873.1 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.

8.873.2 Field Documentation

8.873.2.1 swi_uint256_t unpack_sms_SLQSSmsGetMessageProtocol_t::ParamPresenceMask

8.873.2.2 sms_msgProtocolResp* unpack_sms_SLQSSmsGetMessageProtocol_t::pMessageProtocol

8.874 unpack_sms_SLQSSmsSetRoutes_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.874.1 Detailed Description

This structure contains unpack SLQSSmsSetRoutes parameter.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.874.2 Field Documentation

8.874.2.1 swi_uint256_t unpack_sms_SLQSSmsSetRoutes_t::ParamPresenceMask

8.875 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference

Data Fields

- uint32_t * pSmsStorage
- swi_uint256_t ParamPresenceMask

8.875.1 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.

8.875.2 Field Documentation

8.875.2.1 swi_uint256_t unpack_sms_SLQSSwiGetSMSStorage_t::ParamPresenceMask

8.875.2.2 uint32_t* unpack_sms_SLQSSwiGetSMSStorage_t::pSmsStorage

8.876 unpack_sms_SLQSTransLayerInfoCallback_ind_t Struct Reference

Data Fields

- uint8_t regInd
- sms_transLayerInfo * pTransLayerInfo
- swi_uint256_t ParamPresenceMask

8.876.1 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

8.876.2 Field Documentation

8.876.2.1 swi_uint256_t unpack_sms_SLQSTransLayerInfoCallback_ind_t::ParamPresenceMask

8.876.2.2 sms_transLayerInfo* unpack_sms_SLQSTransLayerInfoCallback_ind_t::pTransLayerInfo

8.876.2.3 uint8_t unpack_sms_SLQSTransLayerInfoCallback_ind_t::regInd

8.877 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

8.877.1 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.

8.877.2 Field Documentation

8.877.2.1 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::alphaIDLen

8.877.2.2 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::causeCode

8.877.2.3 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::errorClass

8.877.2.4 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::messageID

8.877.2.5 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::msgDelFailureCause

8.877.2.6 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::msgDelFailureType

8.877.2.7 uint8_t* unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::pAlphaID

8.877.2.8 swi_uint256_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::ParamPresenceMask

8.877.2.9 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::RPCause

8.877.2.10 uint16_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::sendStatus

8.877.2.11 uint8_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::TPCause

8.877.2.12 uint32_t unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t::userData

8.878 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference

Data Fields

- uint8_t storageType
- uint8_t messageMode
- swi_uint256_t ParamPresenceMask

8.878.1 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.

8.878.2 Field Documentation

8.878.2.1 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::messageMode

8.878.2.2 swi_uint256_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::ParamPresenceMask

8.878.2.3 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::storageType

8.879 unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t Struct Reference

Data Fields

- uint8_t numInstances
- sms_messageWaitingInfoContent msgWaitInfo [0xFF]
- swi_uint256_t ParamPresenceMask

8.879.1 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.

8.879.2 Field Documentation

8.879.2.1 sms_messageWaitingInfoContent unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t::msgWaitInfo[0xFF]

8.879.2.2 uint8_t unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t::numInstances

8.879.2.3 swi_uint256_t unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t::ParamPresenceMask

8.880 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

8.880.1 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.

8.880.2 Field Documentation

8.880.2.1 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::CwtMute

8.880.2.2 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::EarMute

8.880.2.3 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::Generator

8.880.2.4 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::MicMute

8.880.2.5 swi_uint256_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::ParamPresenceMask

8.880.2.6 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::Profile

8.880.2.7 uint8_t unpack_swiaudio_SLQSGetM2MAudioProfile_t::Volume

8.881 unpack_swiaudio_SLQSGetM2MAudioVolume_t Struct Reference

Data Fields

- uint8_t Level
- swi_uint256_t ParamPresenceMask

8.881.1 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.

8.881.2 Field Documentation

8.881.2.1 uint8_t unpack_swiaudio_SLQSGetM2MAudioVolume_t::Level

8.881.2.2 swi_uint256_t unpack_swiaudio_SLQSGetM2MAudioVolume_t::ParamPresenceMask

8.882 unpack_swiaudio_SLQSGetM2MAVMute_t Struct Reference

Data Fields

- uint8_t EarMute
- uint8_t MicMute
- uint8_t CwtMute
- swi_uint256_t ParamPresenceMask

8.882.1 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.

8.882.2 Field Documentation

8.882.2.1 uint8_t unpack_swiaudio_SLQSGetM2MAVMute_t::CwtMute

8.882.2.2 uint8_t unpack_swiaudio_SLQSGetM2MAVMute_t::EarMute

8.882.2.3 uint8_t unpack_swiaudio_SLQSGetM2MAVMute_t::MicMute

8.882.2.4 swi_uint256_t unpack_swiaudio_SLQSGetM2MAVMute_t::ParamPresenceMask

8.883 unpack_swiaudio_SLQSGetM2MSpkrGain_t Struct Reference

Data Fields

- uint16_t Value
- swi_uint256_t ParamPresenceMask

8.883.1 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.

8.883.2 Field Documentation

8.883.2.1 swi_uint256_t unpack_swiaudio_SLQSGetM2MSpkrGain_t::ParamPresenceMask

8.883.2.2 uint16_t unpack_swiaudio_SLQSGetM2MSpkrGain_t::Value

8.884 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

8.884.1 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.

8.884.2 Field Documentation

8.884.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSEventReportInd_t::ParamPresenceMask

8.884.2.2 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo* unpack_swiavms_SLQSAVMSEventReportInd_t::pBinaryUpdateSessionInfo

8.884.2.3 UnpackSwiAvmsEventReportConfig* unpack_swiavms_SLQSAVMSEventReportInd_t::pConfig

8.884.2.4 UnpackSwiAvmsEventReportConnectionRequest* unpack_swiavms_SLQSAVMSEventReportInd_t::pConnectionRequest

8.884.2.5 UnpackSwiAvmsEventReportDataSessionStatus* unpack_swiavms_SLQSAVMSEventReportInd_t::pDataSessionStatus

8.884.2.6 UnpackSwiAvmsEventReportHTTPStatus* unpack_swiavms_SLQSAVMSEventReportInd_t::pHTTPStatus

8.884.2.7 UnpackSwiAvmsEventReportNotification* unpack_swiavms_SLQSAVMSEventReportInd_t::pNotification

8.884.2.8 UnpackSwiAvmsEventReportPackageID* unpack_swiavms_SLQSAVMSEventReportInd_t::pPackageID

8.884.2.9 UnpackSwiAvmsEventReportRegStatus* unpack_swiavms_SLQSAVMSEventReportInd_t::pRegStatus

8.884.2.10 UnpackSwiAvmsEventReportSessionType* unpack_swiavms_SLQSAVMSEventReportInd_t::pSessionType

8.884.2.11 UnpackSwiAvmsEventReportWAMSParamChange* unpack_swiavms_SLQSAVMSEventReportInd_t::pWAMSParaChanged

8.884.2.12 uint32_t unpack_swiavms_SLQSAVMSEventReportInd_t::resultcode

8.885 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

8.885.1 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.

8.885.2 Field Documentation

8.885.2.1 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::AutoConnect

8.885.2.2 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::AutoReboot

8.885.2.3 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwAutoSDM

8.885.2.4 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwPromptdownload

8.885.2.5 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwPromptUpdate

8.885.2.6 uint32_t unpack_swiavms_SLQSAVMSGetSettings_t::OMADMEnabled

8.885.2.7 PackSwiAVMSSettingsAPNInfo* unpack_swiavms_SLQSAVMSGetSettings_t::pAPNInfo

8.885.2.8 swi_uint256_t unpack_swiavms_SLQSAVMSGetSettings_t::ParamPresenceMask

8.885.2.9 PackSwiAVMSSettingsConnectionRetryTimers* unpack_swiavms_SLQSAVMSGetSettings_t::pConnectionRetryTimers

8.885.2.10 uint8_t* unpack_swiavms_SLQSAVMSGetSettings_t::pNotificationStore

8.885.2.11 PackSwiAVMSSettingsPeriodsInfo* unpack_swiavms_SLQSAVMSGetSettings_t::pPeroidsInfo

8.885.2.12 uint32_t* unpack_swiavms_SLQSAVMSGetSettings_t::pPollingTimer

8.885.2.13 uint32_t unpack_swiavms_SLQSAVMSGetSettings_t::resultcode

8.886 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

8.886.1 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

FwPromptdownload	- Firmware Prompt download. • 0 - Disabled. • 1 - Enabled. – Bit to check in ParamPresenceMask - **3**
FwPromptUpdate	- 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**
pConnectionRetryTimers[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**
pNotificationStore[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.

8.886.2 Field Documentation

8.886.2.1 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::AutoConnect

8.886.2.2 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::FwAutoSDM

8.886.2.3 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::FwPromptdownload

8.886.2.4 uint8_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::FwPromptUpdate

8.886.2.5 uint32_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::OMADMEnabled

8.886.2.6 PackSwiAVMSSettingsAPNInfo* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pAPNInfo

8.886.2.7 swi_uint256_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::ParamPresenceMask

8.886.2.8 uint8_t* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pAutoReboot

8.886.2.9 PackSwiAVMSSettingsConnectionRetryTimers* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pConnectionRetryTimers

8.886.2.10 uint8_t* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pNotificationStore

8.886.2.11 PackSwiAVMSSettingsPeriodsInfo* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pPeroidsInfo

8.886.2.12 uint32_t* unpack_swiavms_SLQSAVMSGetSettings_v2_t::pPollingTimer

8.886.2.13 uint32_t unpack_swiavms_SLQSAVMSGetSettings_v2_t::resultcode

8.887 unpack_swiavms_SLQSAVMSSendSelection_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.887.1 Detailed Description

Structure that contains the session type response for AVMS Selection command

Parameters

resultcode	- Tlv Result Code. • Bit to check in ParamPresenceMask - **2**

8.887.2 Field Documentation

8.887.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSendSelection_t::ParamPresenceMask

8.887.2.2 uint32_t unpack_swiavms_SLQSAVMSSendSelection_t::resultcode

8.888 unpack_swiavms_SLQSAVMSSessionGetInfo_t Struct Reference

Data Fields

- UnpackSwiAvmsEventReportBinaryUpdateSessionInfo * pBinaryUpdateSessionInfo
- UnpackSwiAvmsEventReportConfig * pConfig
- UnpackSwiAvmsEventReportNotification * pNotification
- UnpackSwiAvmsEventReportPackageID * pPackageID
- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.888.1 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.

8.888.2 Field Documentation

8.888.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSessionGetInfo_t::ParamPresenceMask

8.888.2.2 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo* unpack_swiavms_SLQSAVMSSessionGetInfo_t::pBinaryUpdateSessionInfo

8.888.2.3 UnpackSwiAvmsEventReportConfig* unpack_swiavms_SLQSAVMSSessionGetInfo_t::pConfig

8.888.2.4 UnpackSwiAvmsEventReportNotification* unpack_swiavms_SLQSAVMSSessionGetInfo_t::pNotification

8.888.2.5 UnpackSwiAvmsEventReportPackageID* unpack_swiavms_SLQSAVMSSessionGetInfo_t::pPackageID

8.888.2.6 uint32_t unpack_swiavms_SLQSAVMSSessionGetInfo_t::resultcode

8.889 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.889.1 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**

8.889.2 Field Documentation

8.889.2.1 swi_uint256_t unpack_swiavms_SLQSAvmsSetEventReport_t::ParamPresenceMask

8.889.2.2 uint32_t unpack_swiavms_SLQSAvmsSetEventReport_t::resultcode

8.890 unpack_swiavms_SLQSAVMSSetSettings_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.890.1 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**

8.890.2 Field Documentation

8.890.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSetSettings_t::ParamPresenceMask

8.890.2.2 uint32_t unpack_swiavms_SLQSAVMSSetSettings_t::resultcode

8.891 unpack_swiavms_SLQSAVMSSetSettings_v2_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.891.1 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**

8.891.2 Field Documentation

8.891.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSSetSettings_v2_t::ParamPresenceMask

8.891.2.2 uint32_t unpack_swiavms_SLQSAVMSSetSettings_v2_t::resultcode

8.892 unpack_swiavms_SLQSAVMSStartSession_t Struct Reference

Data Fields

- uint32_t sessionResponse
- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.892.1 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.

8.892.2 Field Documentation

8.892.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSStartSession_t::ParamPresenceMask

8.892.2.2 uint32_t unpack_swiavms_SLQSAVMSStartSession_t::resultcode

8.892.2.3 uint32_t unpack_swiavms_SLQSAVMSStartSession_t::sessionResponse

8.893 unpack_swiavms_SLQSAVMSStopSession_t Struct Reference

Data Fields

- uint32_t resultcode
- swi_uint256_t ParamPresenceMask

8.893.1 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**

8.893.2 Field Documentation

8.893.2.1 swi_uint256_t unpack_swiavms_SLQSAVMSStopSession_t::ParamPresenceMask

8.893.2.2 uint32_t unpack_swiavms_SLQSAVMSStopSession_t::resultcode

8.894 unpack_swidms_SLQSSwiDmsGetMTU_t Struct Reference

Data Fields

- swidms_mtuSize3gppTlv * pMTUSize3gpp
- swidms_hrpdMTUSizeTlv * pHrpdMTUSize
- swidms_ehrpdMTUSizeTlv * pEhrpdMTUSize
- swidms_usbMTUSizeTlv * pUsbMTUSize
- swi_uint256_t ParamPresenceMask

8.894.1 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.

8.894.2 Field Documentation

8.894.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsGetMTU_t::ParamPresenceMask

8.894.2.2 swidms_ehrpdMTUSizeTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pEhrpdMTUSize

8.894.2.3 swidms_hrpdMTUSizeTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pHrpdMTUSize

8.894.2.4 swidms_mtuSize3gppTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pMTUSize3gpp

8.894.2.5 swidms_usbMTUSizeTlv* unpack_swidms_SLQSSwiDmsGetMTU_t::pUsbMTUSize

8.895 unpack_swidms_SLQSSwiDmsGetUsbComp_t Struct Reference

Data Fields

- swidms_intfaceCfgTlv * pInterfaceCfg
- swidms_supportedIntBitmaskTlv * pSupportedBitmasks
- swi_uint256_t ParamPresenceMask

8.895.1 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.

8.895.2 Field Documentation

8.895.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsGetUsbComp_t::ParamPresenceMask

8.895.2.2 swidms_intfaceCfgTlv* unpack_swidms_SLQSSwiDmsGetUsbComp_t::pInterfaceCfg

8.895.2.3 swidms_supportedIntBitmaskTlv* unpack_swidms_SLQSSwiDmsGetUsbComp_t::pSupportedBitmasks

8.896 unpack_swidms_SLQSSwiDmsGetUsbNetNum_t Struct Reference

Data Fields

- uint8_t usbNetNum
- swi_uint256_t ParamPresenceMask

8.896.1 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.

8.896.2 Field Documentation

8.896.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsGetUsbNetNum_t::ParamPresenceMask

8.896.2.2 uint8_t unpack_swidms_SLQSSwiDmsGetUsbNetNum_t::usbNetNum

8.897 unpack_swidms_SLQSSwiDmsSetMTU_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.897.1 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.

8.897.2 Field Documentation

8.897.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsSetMTU_t::ParamPresenceMask

8.897.2.2 uint16_t unpack_swidms_SLQSSwiDmsSetMTU_t::Tlvresult

8.898 unpack_swidms_SLQSSwiDmsSetUsbComp_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.898.1 Detailed Description

This structure is used to store unpack_swidms_SLQSSwiDmsSetUsbComp_t parameters.

Parameters

Tlvresult	• unpack Tlv result • Bit to check in ParamPresenceMask - **2**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.898.2 Field Documentation

8.898.2.1 swi_uint256_t unpack_swidms_SLQSSwiDmsSetUsbComp_t::ParamPresenceMask

8.898.2.2 uint16_t unpack_swidms_SLQSSwiDmsSetUsbComp_t::Tlvresult

8.899 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

8.899.1 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 • I - 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 • I - reported by modem

8.899.2 Field Documentation

8.899.2.1 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::fix_rate

8.899.2.2 int unpack_swiloc_SwiLocGetAutoStart_t::fix_rate_reported

8.899.2.3 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::fix_type

8.899.2.4 int unpack_swiloc_SwiLocGetAutoStart_t::fix_type_reported

8.899.2.5 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::function

8.899.2.6 int unpack_swiloc_SwiLocGetAutoStart_t::function_reported

8.899.2.7 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::max_dist

8.899.2.8 int unpack_swiloc_SwiLocGetAutoStart_t::max_dist_reported

8.899.2.9 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::max_time

8.899.2.10 int unpack_swiloc_SwiLocGetAutoStart_t::max_time_reported

8.899.2.11 swi_uint256_t unpack_swiloc_SwiLocGetAutoStart_t::ParamPresenceMask

8.900 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

8.900.1 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.

8.900.2 Field Documentation

8.900.2.1 uint32_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::eventType

8.900.2.2 swi_uint256_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::ParamPresenceMask

8.900.2.3 unpack_omaDmConfigTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoConfig

8.900.2.4 unpack_omaDmFotaTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoFota

8.900.2.5 unpack_omaDmNotificationsTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoNotification

8.901 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

8.901.1 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.

8.901.2 Field Documentation

8.901.2.1 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Date[255]

8.901.2.2 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::DateLength

8.901.2.3 swi_uint256_t unpack_swioma_SLQSOMADMGetSessionInfo_t::ParamPresenceMask

8.901.2.4 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescLength

8.901.2.5 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescription[255]

8.901.2.6 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgName[255]

8.901.2.7 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgNameLength

8.901.2.8 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::RetryCount

8.901.2.9 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionState

8.901.2.10 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.901.2.11 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Severity

8.901.2.12 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Source[255]

8.901.2.13 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SourceLength

8.901.2.14 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Status

8.901.2.15 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Time[255]

8.901.2.16 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::TimeLength

8.901.2.17 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::UpdateCompleteStatus

8.902 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

8.902.1 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.

8.902.2 Field Documentation

8.902.2.1 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::Autosdm

8.902.2.2 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAdownload

8.902.2.3 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAUpdate

8.902.2.4 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FwAutoCheck

8.902.2.5 uint32_t unpack_swioma_SLQSOMADMGetSettings_t::OMADMEnabled

8.902.2.6 swi_uint256_t unpack_swioma_SLQSOMADMGetSettings_t::ParamPresenceMask

8.903 unpack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t FwAvailability
- swi_uint256_t ParamPresenceMask

8.903.1 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.

8.903.2 Field Documentation

8.903.2.1 uint32_t unpack_swioma_SLQSOMADMStartSession_t::FwAvailability

8.903.2.2 swi_uint256_t unpack_swioma_SLQSOMADMStartSession_t::ParamPresenceMask

8.904 unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.904.1 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.

8.904.2 Field Documentation

8.904.2.1 swi_uint256_t unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t::ParamPresenceMask

8.904.2.2 uint16_t unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t::TlvResult

8.905 unpack_tmd_SLQSTmdGetMitigationDevList_t Struct Reference

Data Fields

- uint8_t MitigationDevListLen
- tmd_mitigationDevList MitigationDevList [255]
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.905.1 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**
pMitigationDev-List	• Mitigation Device List (Optional) • See tmd_mitigationDevList for more information. • Bit to check in ParamPresenceMask - **16**
TlvResult	• Unpack Result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.905.2 Field Documentation

8.905.2.1 tmd_mitigationDevList unpack_tmd_SLQSTmdGetMitigationDevList_t::MitigationDevList[255]

8.905.2.2 uint8_t unpack_tmd_SLQSTmdGetMitigationDevList_t::MitigationDevListLen

8.905.2.3 swi_uint256_t unpack_tmd_SLQSTmdGetMitigationDevList_t::ParamPresenceMask

8.905.2.4 uint16_t unpack_tmd_SLQSTmdGetMitigationDevList_t::Tlvresult

8.906 unpack_tmd_SLQSTmdGetMitigationLvl_t Struct Reference

Data Fields

- uint8_t CurrentmitigationLvl
- uint8_t ReqMitigationLvl
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.906.1 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.

8.906.2 Field Documentation

8.906.2.1 uint8_t unpack_tmd_SLQSTmdGetMitigationLvl_t::CurrentmitigationLvl

8.906.2.2 swi_uint256_t unpack_tmd_SLQSTmdGetMitigationLvl_t::ParamPresenceMask

8.906.2.3 uint8_t unpack_tmd_SLQSTmdGetMitigationLvl_t::ReqMitigationLvl

8.906.2.4 uint16_t unpack_tmd_SLQSTmdGetMitigationLvl_t::Tlvresult

8.907 unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t Struct Reference

Data Fields

- uint8_t deviceIdLen

- char deviceID [255]
- uint8_t lvl
- swi_uint256_t ParamPresenceMask

8.907.1 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.

8.907.2 Field Documentation

8.907.2.1 char unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::deviceID[255]

8.907.2.2 uint8_t unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::deviceIdLen

8.907.2.3 uint8_t unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::lvl

8.907.2.4 swi_uint256_t unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t::ParamPresenceMask

8.908 unpack_tmd_SLQSTmdRegNotMitigationLvl_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.908.1 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.

8.908.2 Field Documentation

8.908.2.1 swi_uint256_t unpack_tmd_SLQSTmdRegNotMitigationLvl_t::ParamPresenceMask

8.908.2.2 uint16_t unpack_tmd_SLQSTmdRegNotMitigationLvl_t::Tlvresult

8.909 unpack_uim_ChangePin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.909.1 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.

8.909.2 Field Documentation

8.909.2.1 swi_uint256_t unpack_uim_ChangePin_t::ParamPresenceMask

8.909.2.2 uim_encryptedPIN1* unpack_uim_ChangePin_t::pEncryptedPIN1

8.909.2.3 uint32_t* unpack_uim_ChangePin_t::pIndicationToken

8.909.2.4 uim_remainingRetries∗ unpack_uim_ChangePin_t::pRemainingRetries

8.909.2.5 uint16_t unpack_uim_ChangePin_t::Tlvresult

8.910 unpack_uim_GetCardStatus_t Struct Reference

Data Fields

- uim_cardStatus ∗ pCardStatus
- uim_hotSwapStatus ∗ pHotSwapStatus
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.910.1 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.

8.910.2 Field Documentation

8.910.2.1 swi_uint256_t unpack_uim_GetCardStatus_t::ParamPresenceMask

8.910.2.2 uim_cardStatus∗ unpack_uim_GetCardStatus_t::pCardStatus

8.910.2.3 uim_hotSwapStatus∗ unpack_uim_GetCardStatus_t::pHotSwapStatus

8.910.2.4 uint16_t unpack_uim_GetCardStatus_t::Tlvresult

8.911 unpack_uim_ReadTransparent_t Struct Reference

Data Fields

- uim_cardResult ∗ pCardResult
- uim_readResult ∗ pReadResult
- uint32_t ∗ pIndicationToken
- uint8_t ∗ pEncryptedData
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.911.1 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.

8.911.2 Field Documentation

8.911.2.1 swi_uint256_t unpack_uim_ReadTransparent_t::ParamPresenceMask

8.911.2.2 uim_cardResult* unpack_uim_ReadTransparent_t::pCardResult

8.911.2.3 uint8_t* unpack_uim_ReadTransparent_t::pEncryptedData

8.911.2.4 uint32_t* unpack_uim_ReadTransparent_t::pIndicationToken

8.911.2.5 uim_readResult* unpack_uim_ReadTransparent_t::pReadResult

8.911.2.6 uint16_t unpack_uim_ReadTransparent_t::Tlvresult

8.912 unpack_uim_SetPinProtection_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.912.1 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.

8.912.2 Field Documentation

8.912.2.1 swi_uint256_t unpack_uim_SetPinProtection_t::ParamPresenceMask

8.912.2.2 uim_encryptedPIN1* unpack_uim_SetPinProtection_t::pEncryptedPIN1

8.912.2.3 uint32_t* unpack_uim_SetPinProtection_t::pIndicationToken

8.912.2.4 uim_remainingRetries* unpack_uim_SetPinProtection_t::pRemainingRetries

8.912.2.5 uint16_t unpack_uim_SetPinProtection_t::Tlvresult

8.913 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference

Data Fields

- slots_t slotsstatusChange
- uint8_t bNumberOfPhySlots
- swi_uint256_t ParamPresenceMask

8.913.1 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.

8.913.2 Field Documentation

8.913.2.1 uint8_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::bNumberOfPhySlots

8.913.2.2 swi_uint256_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::ParamPresenceMask

8.913.2.3 slots_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::slotsstatusChange

8.914 unpack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_authenticateResult * pAuthenticateResult
- uint32_t * pIndicationToken
- swi_uint256_t ParamPresenceMask

8.914.1 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.

8.914.2 Field Documentation

8.914.2.1 swi_uint256_t unpack_uim_SLQSUIMAuthenticate_t::ParamPresenceMask

8.914.2.2 uim_authenticateResult∗ unpack_uim_SLQSUIMAuthenticate_t::pAuthenticateResult

8.914.2.3 uim_cardResult∗ unpack_uim_SLQSUIMAuthenticate_t::pCardResult

8.914.2.4 uint32_t∗ unpack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.915 unpack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_remainingRetries ∗ pRemainingRetries
- swi_uint256_t ParamPresenceMask

8.915.1 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.

8.915.2 Field Documentation

8.915.2.1 swi_uint256_t unpack_uim_SLQSUIMDepersonalization_t::ParamPresenceMask

8.915.2.2 uim_remainingRetries∗ unpack_uim_SLQSUIMDepersonalization_t::pRemainingRetries

8.916 unpack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask
- swi_uint256_t ParamPresenceMask

8.916.1 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.

8.916.2 Field Documentation

8.916.2.1 uint32_t unpack_uim_SLQSUIMEventRegister_t::eventMask

8.916.2.2 swi_uint256_t unpack_uim_SLQSUIMEventRegister_t::ParamPresenceMask

8.917 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint8_t * pAutoSelection
- uim_personalizationStatus * pPersonalizationStatus
- uint8_t * pHaltSubscription
- swi_uint256_t ParamPresenceMask

8.917.1 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.

8.917.2 Field Documentation

8.917.2.1 swi_uint256_t unpack_uim_SLQSUIMGetConfiguration_t::ParamPresenceMask

8.917.2.2 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pAutoSelection

8.917.2.3 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pHaltSubscription

8.917.2.4 uim_personalizationStatus* unpack_uim_SLQSUIMGetConfiguration_t::pPersonalizationStatus

8.918 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_fileAttributes * pFileAttributes
- uint32_t * pIndicationToken
- swi_uint256_t ParamPresenceMask

8.918.1 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.

8.918.2 Field Documentation

8.918.2.1 swi_uint256_t unpack_uim_SLQSUIMGetFileAttributes_t::ParamPresenceMask

8.918.2.2 uim_cardResult* unpack_uim_SLQSUIMGetFileAttributes_t::pCardResult

8.918.2.3 uim_fileAttributes* unpack_uim_SLQSUIMGetFileAttributes_t::pFileAttributes

8.918.2.4 uint32_t* unpack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.919 unpack_uim_SLQSUIMGetSlotsStatus_t Struct Reference

Data Fields

- uint8_t * pNumberOfPhySlot
- slots_t * pUimSlotsStatus
- swi_uint256_t ParamPresenceMask

8.919.1 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.

8.919.2 Field Documentation

8.919.2.1 swi_uint256_t unpack_uim_SLQSUIMGetSlotsStatus_t::ParamPresenceMask

8.919.2.2 uint8_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pNumberOfPhySlot

8.919.2.3 slots_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pUimSlotsStatus

8.920 unpack_uim_SLQSUIMRefreshCallback_Ind_t Struct Reference

Data Fields

- uint8_t TlvPresent
- uim_refreshevent refreshEvent
- swi_uint256_t ParamPresenceMask

8.920.1 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**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.920.2 Field Documentation

8.920.2.1 swi_uint256_t unpack_uim_SLQSUIMRefreshCallback_Ind_t::ParamPresenceMask

8.920.2.2 uim_refreshevent unpack_uim_SLQSUIMRefreshCallback_Ind_t::refreshEvent

8.920.2.3 uint8_t unpack_uim_SLQSUIMRefreshCallback_Ind_t::TlvPresent

8.921 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_refreshevent * pRefreshEvent
- swi_uint256_t ParamPresenceMask

8.921.1 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.

8.921.2 Field Documentation

8.921.2.1 swi_uint256_t unpack_uim_SLQSUIMRefreshGetLastEvent_t::ParamPresenceMask

8.921.2.2 uim_refreshevent* unpack_uim_SLQSUIMRefreshGetLastEvent_t::pRefreshEvent

8.922 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference

Data Fields

- uim_cardStatus * pCardStatus
- swi_uint256_t ParamPresenceMask

8.922.1 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.

8.922.2 Field Documentation

8.922.2.1 swi_uint256_t unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t::ParamPresenceMask

8.922.2.2 uim_cardStatus* unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t::pCardStatus

8.923 unpack_uim_UnblockPin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.923.1 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.

8.923.2 Field Documentation

8.923.2.1 swi_uint256_t unpack_uim_UnblockPin_t::ParamPresenceMask

8.923.2.2 uim_encryptedPIN1∗ unpack_uim_UnblockPin_t::pEncryptedPIN1

8.923.2.3 uint32_t∗ unpack_uim_UnblockPin_t::pIndicationToken

8.923.2.4 uim_remainingRetries∗ unpack_uim_UnblockPin_t::pRemainingRetries

8.923.2.5 uint16_t unpack_uim_UnblockPin_t::TlvResult

8.924 unpack_uim_VerifyPin_t Struct Reference

Data Fields

- uim_remainingRetries ∗ pRemainingRetries
- uim_encryptedPIN1 ∗ pEncryptedPIN1
- uint32_t ∗ pIndicationToken
- uint16_t TlvResult
- swi_uint256_t ParamPresenceMask

8.924.1 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.

8.924.2 Field Documentation

8.924.2.1 swi_uint256_t unpack_uim_VerifyPin_t::ParamPresenceMask

8.924.2.2 uim_encryptedPIN1∗ unpack_uim_VerifyPin_t::pEncryptedPIN1

8.924.2.3 uint32_t∗ unpack_uim_VerifyPin_t::pIndicationToken

8.924.2.4 uim_remainingRetries∗ unpack_uim_VerifyPin_t::pRemainingRetries

8.924.2.5 uint16_t unpack_uim_VerifyPin_t::TlvResult

8.925 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

8.925.1 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**

pArrConnectPartyNum(optional)	• See voice_arrConnectPartyNum for more information. • Bit to check in ParamPresenceMask - **22**
pArrDiagInfo(optional)	• See voice_arrDiagInfo for more information. • Bit to check in ParamPresenceMask - **23**
pArrCalledPartyNum(optional)	• See voice_arrCalledPartyNum for more information. • Bit to check in ParamPresenceMask - **24**
pArrRedirPartyNum(optional)	• See voice_arrRedirPartyNum for more information. • Bit to check in ParamPresenceMask - **25**
pArrAlertingPattern(optional)	• See voice_arrAlertingPattern for more information. • Bit to check in ParamPresenceMask - **26**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

8.925.2 Field Documentation

8.925.2.1 voice_arrCallInfo unpack_voice_allCallStatusCallback_ind_t::arrCallInfomation

8.925.2.2 swi_uint256_t unpack_voice_allCallStatusCallback_ind_t::ParamPresenceMask

8.925.2.3 voice_arrAlertingPattern* unpack_voice_allCallStatusCallback_ind_t::pArrAlertingPattern

8.925.2.4 voice_arrAlertingType* unpack_voice_allCallStatusCallback_ind_t::pArrAlertingType

8.925.2.5 voice_arrAlphaID* unpack_voice_allCallStatusCallback_ind_t::pArrAlphaID

8.925.2.6 voice_arrCalledPartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrCalledPartyNum

8.925.2.7 voice_arrCallEndReason* unpack_voice_allCallStatusCallback_ind_t::pArrCallEndReason

8.925.2.8 voice_arrConnectPartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrConnectPartyNum

8.925.2.9 voice_arrDiagInfo* unpack_voice_allCallStatusCallback_ind_t::pArrDiagInfo

8.925.2.10 voice_arrRedirPartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrRedirPartyNum

8.925.2.11 voice_arrRemotePartyName* unpack_voice_allCallStatusCallback_ind_t::pArrRemotePartyName

8.925.2.12 voice_arrRemotePartyNum* unpack_voice_allCallStatusCallback_ind_t::pArrRemotePartyNum

8.925.2.13 voice_arrSvcOption* unpack_voice_allCallStatusCallback_ind_t::pArrSvcOption

8.926 unpack_voice_DTMFEventCallback_ind_t Struct Reference

Data Fields

- voice_DTMFInfo DTMFInformation
- uint8_t * pOnLength
- uint8_t * pOffLength
- swi_uint256_t ParamPresenceMask

8.926.1 Detailed Description

This structure contains the parameters passed for DTMF event indication by the device.

Parameters

DTMFInformation(mandatory)	See voice_DTMFInfo for more information. • Bit to check in ParamPresenceMask - **1**
pOnLength(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**
pOffLength(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**
ParamPresenceMask	• bitmask representation to indicate valid parameters.

Note

None

8.926.2 Field Documentation

8.926.2.1 voice_DTMFInfo unpack_voice_DTMFEventCallback_ind_t::DTMFInformation

8.926.2.2 swi_uint256_t unpack_voice_DTMFEventCallback_ind_t::ParamPresenceMask

8.926.2.3 uint8_t* unpack_voice_DTMFEventCallback_ind_t::pOffLength

8.926.2.4 uint8_t* unpack_voice_DTMFEventCallback_ind_t::pOnLength

8.927 unpack_voice_OTASPStatusCallback_ind_t Struct Reference

Data Fields

- uint8_t callID
- uint8_t OTASPStatus
- swi_uint256_t ParamPresenceMask

8.927.1 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.

8.927.2 Field Documentation

8.927.2.1 uint8_t unpack_voice_OTASPStatusCallback_ind_t::callID

8.927.2.2 uint8_t unpack_voice_OTASPStatusCallback_ind_t::OTASPStatus

8.927.2.3 swi_uint256_t unpack_voice_OTASPStatusCallback_ind_t::ParamPresenceMask

8.928 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

8.928.1 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

Generated by Doxygen 1.8.14

AlphaIdentifier	• see voice_alphaIDInfo definition • Bit to check in ParamPresenceMask - **17**
pUSSDInfo	• USS Data from Network (See structure 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 structure 'voice_ccSUPSType' definition • Bit to check in ParamPresenceMask - **21**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.928.2 Field Documentation

8.928.2.1 voice_alphaIDInfo* unpack_voice_SLQSOriginateUSSD_t::pAlphaIDInfo

8.928.2.2 swi_uint256_t unpack_voice_SLQSOriginateUSSD_t::ParamPresenceMask

8.928.2.3 uint8_t* unpack_voice_SLQSOriginateUSSD_t::pCallId

8.928.2.4 uint8_t* unpack_voice_SLQSOriginateUSSD_t::pCcResultType

8.928.2.5 voice_ccSUPSType* unpack_voice_SLQSOriginateUSSD_t::pCCSuppsType

8.928.2.6 uint16_t* unpack_voice_SLQSOriginateUSSD_t::pfailureCause

8.928.2.7 struct voice_USSInfo* unpack_voice_SLQSOriginateUSSD_t::pUSSDInfo

8.929 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId
- swi_uint256_t ParamPresenceMask

8.929.1 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.

8.929.2 Field Documentation

8.929.2.1 swi_uint256_t unpack_voice_SLQSVoiceAnswerCall_t::ParamPresenceMask

8.929.2.2 uint8_t* unpack_voice_SLQSVoiceAnswerCall_t::pCallId

8.930 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- swi_uint256_t ParamPresenceMask

8.930.1 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.

8.930.2 Field Documentation

8.930.2.1 swi_uint256_t unpack_voice_SLQSVoiceBurstDTMF_t::ParamPresenceMask

8.930.2.2 uint8_t* unpack_voice_SLQSVoiceBurstDTMF_t::pCallID

8.931 unpack_voice_SLQSVoiceDialCall_t Struct Reference

Data Fields

- uint8_t * pCallID
- voice_alphaIDInfo * pAlphaIDInfo

- uint8_t * pCCResultType
- voice_ccSUPSType * pCCSUPSType
- swi_uint256_t ParamPresenceMask

8.931.1 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.

8.931.2 Field Documentation

8.931.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceDialCall_t::pAlphaIDInfo

8.931.2.2 swi_uint256_t unpack_voice_SLQSVoiceDialCall_t::ParamPresenceMask

8.931.2.3 uint8_t* unpack_voice_SLQSVoiceDialCall_t::pCallID

8.931.2.4 uint8_t* unpack_voice_SLQSVoiceDialCall_t::pCCResultType

8.931.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceDialCall_t::pCCSUPSType

8.932 unpack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t * pCallId

- swi_uint256_t ParamPresenceMask

8.932.1 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.

8.932.2 Field Documentation

8.932.2.1 swi_uint256_t unpack_voice_SLQSVoiceEndCall_t::ParamPresenceMask

8.932.2.2 uint8_t* unpack_voice_SLQSVoiceEndCall_t::pCallId

8.933 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

8.933.1 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.

8.933.2 Field Documentation

8.933.2.1 swi_uint256_t unpack_voice_SLQSVoiceGetAllCallInfo_t::ParamPresenceMask

8.933.2.2 voice_arrAlertingPattern* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingPattern

8.933.2.3 voice_arrAlertingType* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingType

8.933.2.4 voice_arrAlphaID* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlphaID

8.933.2.5 voice_arrCalledPartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCalledPartyNum

8.933.2.6 voice_arrCallEndReason* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallEndReason

8.933.2.7 voice_arrCallInfo* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallInfo

8.933.2.8 voice_arrConnectPartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrConnectPartyNum

8.933.2.9 voice_arrDiagInfo* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrDiagInfo

8.933.2.10 voice_arrRedirPartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRedirPartyNum

8.933.2.11 voice_arrRemotePartyName* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyName

8.933.2.12 voice_arrRemotePartyNum* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyNum

8.933.2.13 voice_arrSvcOption* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrSvcOption

8.933.2.14 voice_arrUUSInfo* unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrUUSInfo

8.933.2.15 uint8_t* unpack_voice_SLQSVoiceGetAllCallInfo_t::pOTASPStatus

8.933.2.16 uint8_t* unpack_voice_SLQSVoiceGetAllCallInfo_t::pVoicePrivacy

8.934 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

8.934.1 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.

8.934.2 Field Documentation

8.934.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallBarring_t::pAlphaIDInfo

8.934.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCallBarring_t::ParamPresenceMask

8.934.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCallID

8.934.2.4 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCCResType

8.934.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallBarring_t::pCCSUPSType

8.934.2.6 uint16_t* unpack_voice_SLQSVoiceGetCallBarring_t::pFailCause

8.934.2.7 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.935 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

8.935.1 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.

8.935.2 Field Documentation

8.935.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pAlphaIDInfo

8.935.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCallForwardingStatus_t::ParamPresenceMask

8.935.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCallID

8.935.2.4 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCResType

8.935.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCSUPSType

8.935.2.6 uint16_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pFailCause

8.935.2.7 voice_getCallFWExtInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWExtInfo

8.935.2.8 voice_getCallFWInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWInfo

8.936 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

8.936.1 Detailed Description

This structure contains information of the unpack parameters associated with a call.

Parameters

pCall-Info(optional)	• See voice_callInfo for more information. • Bit to check in ParamPresenceMask - **16**
pRemoteParty-Num(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.

8.936.2 Field Documentation

8.936.2.1 uint32_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertingPattern

8.936.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertType

8.936.2.3 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pAlphaIDInfo

8.936.2.4 swi_uint256_t unpack_voice_SLQSVoiceGetCallInfo_t::ParamPresenceMask

8.936.2.5 voice_callInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pCallInfo

8.936.2.6 voice_connectNumInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pConnectNumInfo

8.936.2.7 voice_diagInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pDiagInfo

8.936.2.8 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pOTASPStatus

8.936.2.9 voice_remotePartyName* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyName

8.936.2.10 voice_remotePartyNum* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyNum

8.936.2.11 uint16_t* unpack_voice_SLQSVoiceGetCallInfo_t::pSrvOpt

8.936.2.12 voice_UUSInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pUUSInfo

8.936.2.13 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pVoicePrivacy

8.937 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

8.937.1 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.

8.937.2 Field Documentation

8.937.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallWaiting_t::pAlphaIDInfo

8.937.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCallWaiting_t::ParamPresenceMask

8.937.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCallID

8.937.2.4 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCResType

8.937.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCSUPSType

8.937.2.6 uint16_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pFailCause

8.937.2.7 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.938 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

8.938.1 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.

8.938.2 Field Documentation

8.938.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIP_t::pAlphaIDInfo

8.938.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCLIP_t::ParamPresenceMask

8.938.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCallID

8.938.2.4 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCCResType

8.938.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIP_t::pCCSUPSType

8.938.2.6 voice_CLIPResp* unpack_voice_SLQSVoiceGetCLIP_t::pCLIPResp

8.938.2.7 uint16_t* unpack_voice_SLQSVoiceGetCLIP_t::pFailCause

8.939 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

8.939.1 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.

8.939.2 Field Documentation

8.939.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIR_t::pAlphaIDInfo

8.939.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCLIR_t::ParamPresenceMask

8.939.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCallID

8.939.2.4 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCCResType

8.939.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIR_t::pCCSUPSType

8.939.2.6 voice_CLIRResp* unpack_voice_SLQSVoiceGetCLIR_t::pCLIRResp

8.939.2.7 uint16_t* unpack_voice_SLQSVoiceGetCLIR_t::pFailCause

8.940 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

8.940.1 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.

8.940.2 Field Documentation

8.940.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCNAP_t::pAlphaIDInfo

8.940.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCNAP_t::ParamPresenceMask

8.940.2.3 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCallID

8.940.2.4 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCCResType

8.940.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCNAP_t::pCCSUPSType

8.940.2.6 voice_CNAPResp* unpack_voice_SLQSVoiceGetCNAP_t::pCNAPResp

8.940.2.7 uint16_t* unpack_voice_SLQSVoiceGetCNAP_t::pFailCause

8.941 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

8.941.1 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.

8.941.2 Field Documentation

8.941.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLP_t::pAlphaIDInfo

8.941.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCOLP_t::ParamPresenceMask

8.941.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCallID

8.941.2.4 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCCResType

8.941.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLP_t::pCCSUPSType

8.941.2.6 voice_COLPResp* unpack_voice_SLQSVoiceGetCOLP_t::pCOLPResp

8.941.2.7 uint16_t* unpack_voice_SLQSVoiceGetCOLP_t::pFailCause

8.942 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

8.942.1 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.

8.942.2 Field Documentation

8.942.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLR_t::pAlphaIDInfo

8.942.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetCOLR_t::ParamPresenceMask

8.942.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCallID

8.942.2.4 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCCResType

8.942.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLR_t::pCCSUPSType

8.942.2.6 voice_COLRResp* unpack_voice_SLQSVoiceGetCOLR_t::pCOLRResp

8.942.2.7 uint16_t* unpack_voice_SLQSVoiceGetCOLR_t::pFailCause

8.943 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

8.943.1 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.

8.943.2 Field Documentation

8.943.2.1 voice_airTimer* unpack_voice_SLQSVoiceGetConfig_t::pAirTimerCnt

8.943.2.2 swi_uint256_t unpack_voice_SLQSVoiceGetConfig_t::ParamPresenceMask

8.943.2.3 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pAutoAnswerStat

8.943.2.4 voice_curAMRConfig* unpack_voice_SLQSVoiceGetConfig_t::pCurAMRConfig

8.943.2.5 voice_prefVoiceSO* unpack_voice_SLQSVoiceGetConfig_t::pCurPrefVoiceSO

8.943.2.6 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurrTTYMode

8.943.2.7 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoiceDomainPref

8.943.2.8 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoicePrivacyPref

8.943.2.9 voice_roamTimer* unpack_voice_SLQSVoiceGetConfig_t::pRoamTimerCnt

8.944 unpack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint16_t * pFailCause
- swi_uint256_t ParamPresenceMask

8.944.1 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.

8.944.2 Field Documentation

8.944.2.1 swi_uint256_t unpack_voice_SLQSVoiceManageCalls_t::ParamPresenceMask

8.944.2.2 uint16_t* unpack_voice_SLQSVoiceManageCalls_t::pFailCause

8.945 unpack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID
- swi_uint256_t ParamPresenceMask

8.945.1 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.

8.945.2 Field Documentation

8.945.2.1 swi_uint256_t unpack_voice_SLQSVoiceSendFlash_t::ParamPresenceMask

8.945.2.2 uint8_t* unpack_voice_SLQSVoiceSendFlash_t::pCallID

8.946 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

8.946.1 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.

8.946.2 Field Documentation

8.946.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pAlphaIDInfo

8.946.2.2 swi_uint256_t unpack_voice_SLQSVoiceSetCallBarringPassword_t::ParamPresenceMask

8.946.2.3 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCallID

8.946.2.4 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCResType

8.946.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCSUPSType

8.946.2.6 uint16_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pFailCause

8.947 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

8.947.1 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.

8.947.2 Field Documentation

8.947.2.1 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAirTimerStatus

8.947.2.2 swi_uint256_t unpack_voice_SLQSVoiceSetConfig_t::ParamPresenceMask

8.947.2.3 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAutoAnsStatus

8.947.2.4 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSOStatus

8.947.2.5 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pRoamTimerStatus

8.947.2.6 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pTTYConfigStatus

8.947.2.7 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pVoiceDomainPrefStatus

8.948 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

8.948.1 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.

8.948.2 Field Documentation

8.948.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetSUPSService_t::pAlphaIDInfo

8.948.2.2 swi_uint256_t unpack_voice_SLQSVoiceSetSUPSService_t::ParamPresenceMask

8.948.2.3 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCallID

8.948.2.4 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCCResultType

8.948.2.5 voice_ccSUPSType* unpack_voice_SLQSVoiceSetSUPSService_t::pCCSUPSType

8.948.2.6 uint16_t* unpack_voice_SLQSVoiceSetSUPSService_t::pFailCause

8.949 unpack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- swi_uint256_t ParamPresenceMask

8.949.1 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.

8.949.2 Field Documentation

8.949.2.1 swi_uint256_t unpack_voice_SLQSVoiceStartContDTMF_t::ParamPresenceMask

8.949.2.2 uint8_t* unpack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.950 unpack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID
- swi_uint256_t ParamPresenceMask

8.950.1 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.

8.950.2 Field Documentation

8.950.2.1 uint8_t unpack_voice_SLQSVoiceStopContDTMF_t::callID

8.950.2.2 swi_uint256_t unpack_voice_SLQSVoiceStopContDTMF_t::ParamPresenceMask

8.951 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

8.951.1 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

8.951.2 Field Documentation

8.951.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pAlphaIDInfo

8.951.2.2 swi_uint256_t unpack_voice_SLQSVoiceSUPSCallback_ind_t::ParamPresenceMask

8.951.2.3 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallBarPasswd

8.951.2.4 **voice_getCallFWInfo*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallFwdInfo

8.951.2.5 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallFWNum

8.951.2.6 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallFWTimerVal

8.951.2.7 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCallID

8.951.2.8 **voice_CLIPResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCLIPstatus

8.951.2.9 **voice_CLIRResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCLIRstatus

8.951.2.10 **voice_CNAPResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCNAPstatus

8.951.2.11 **voice_COLPResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCOLPstatus

8.951.2.12 **voice_COLRResp*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pCOLRstatus

8.951.2.13 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pDataSrc

8.951.2.14 uint16_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pFailCause

8.951.2.15 **voice_newPwdData*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pNewPwdData

8.951.2.16 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pReason

8.951.2.17 uint8_t* unpack_voice_SLQSVoiceSUPSCallback_ind_t::pSvcClass

8.951.2.18 struct **voice_USSInfo*** unpack_voice_SLQSVoiceSUPSCallback_ind_t::pUSSInfo

8.951.2.19 **voice_SUPSInfo** unpack_voice_SLQSVoiceSUPSCallback_ind_t::SUPSInformation

8.952 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

8.952.1 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

8.952.2 Field Documentation

8.952.2.1 uint8_t unpack_voice_SUPSNotificationCallback_ind_t::callID

8.952.2.2 uint8_t unpack_voice_SUPSNotificationCallback_ind_t::notifType

8.952.2.3 swi_uint256_t unpack_voice_SUPSNotificationCallback_ind_t::ParamPresenceMask

8.952.2.4 uint16_t* unpack_voice_SUPSNotificationCallback_ind_t::pCUGIndex

8.952.2.5 voice_ECTNum* unpack_voice_SUPSNotificationCallback_ind_t::pECTNum

8.953 unpack_voice_USSDNotificationCallback_ind_t Struct Reference

Data Fields

- uint8_t notification_Type
- voice_USSDNotificationNetworkInfo USSDNotificationNetworkInfo
- swi_uint256_t ParamPresenceMask

8.953.1 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.

8.953.2 Field Documentation

8.953.2.1 uint8_t unpack_voice_USSDNotificationCallback_ind_t::notification_Type

8.953.2.2 swi_uint256_t unpack_voice_USSDNotificationCallback_ind_t::ParamPresenceMask

8.953.2.3 voice_USSDNotificationNetworkInfo unpack_voice_USSDNotificationCallback_ind_t::USSDNotificationNetworkInfo

8.954 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

8.954.1 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**
pConnectNum-Info[Optional]	• Connected Number Information • see voice_connectNumInfo for more information • Bit to check in ParamPresenceMask - **22**
pCallingParty-Info[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**
pCalledParty-Info[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**
pExtDispRec-Info[Optional]	• Extended Display Record Information • see voice_extDispRecInfo for more information • Bit to check in ParamPresenceMask - **30**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.954.2 Field Documentation

8.954.2.1 uint8_t unpack_voice_VoiceInfoRecCallback_ind_t::callID

8.954.2.2 swi_uint256_t unpack_voice_VoiceInfoRecCallback_ind_t::ParamPresenceMask

8.954.2.3 voice_calledPartyInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pCalledPartyInfo

8.954.2.4 voice_callerIDInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pCallerIDInfo

8.954.2.5 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pCallerNameInfo

8.954.2.6 voice_connectNumInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pCallingPartyInfo

8.954.2.7 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pCallWaitInd

8.954.2.8 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pCLIRCause

8.954.2.9 voice_connectNumInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pConnectNumInfo

8.954.2.10 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pDispInfo

8.954.2.11 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pExtDispInfo

8.954.2.12 voice_extDispRecInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pExtDispRecInfo

8.954.2.13 voice_lineCtrlInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pLineCtrlInfo

8.954.2.14 voice_NSSAudioCtrl* unpack_voice_VoiceInfoRecCallback_ind_t::pNSSAudioCtrl

8.954.2.15 uint8_t* unpack_voice_VoiceInfoRecCallback_ind_t::pNSSRelease

8.954.2.16 voice_redirNumInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pRedirNumInfo

8.954.2.17 voice_signalInfo* unpack_voice_VoiceInfoRecCallback_ind_t::pSignalInfo

8.955 unpack_voice_voicePrivacyChangeCallback_ind_t Struct Reference

Data Fields

- uint8_t callID
- uint8_t voicePrivacy
- swi_uint256_t ParamPresenceMask

8.955.1 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

8.955.2 Field Documentation

8.955.2.1 uint8_t unpack_voice_voicePrivacyChangeCallback_ind_t::callID

8.955.2.2 swi_uint256_t unpack_voice_voicePrivacyChangeCallback_ind_t::ParamPresenceMask

8.955.2.3 uint8_t unpack_voice_voicePrivacyChangeCallback_ind_t::voicePrivacy

8.956 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

8.956.1 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.

8.956.2 Field Documentation

8.956.2.1 wds_DHCPLeaseOptTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseOptTlv

8.956.2.2 wds_DHCPLeaseStateTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseStateTlv

8.956.2.3 wds_IPv4AdTlv unpack_wds_DHCPv4ClientLease_ind_t::IPv4AddrTlv

8.956.2.4 swi_uint256_t unpack_wds_DHCPv4ClientLease_ind_t::ParamPresenceMask

8.956.2.5 wds_DHCPProfileIdTlv unpack_wds_DHCPv4ClientLease_ind_t::ProfileIdTlv

8.957 unpack_wds_GetAutoconnect_t Struct Reference

Data Fields

- uint32_t * psetting
- swi_uint256_t ParamPresenceMask

8.957.1 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.

8.957.2 Field Documentation

8.957.2.1 swi_uint256_t unpack_wds_GetAutoconnect_t::ParamPresenceMask

8.957.2.2 uint32_t* unpack_wds_GetAutoconnect_t::psetting

8.958 unpack_wds_GetByteTotals_t Struct Reference

Data Fields

- uint64_t * pTXTotalBytes
- uint64_t * pRXTotalBytes
- swi_uint256_t ParamPresenceMask

8.958.1 Detailed Description

This structure contains unpack get byte totals information.

Parameters

pTXTotalBytes	• Bytes transmitted without error • Bit to check in ParamPresenceMask - **16**
pRXTotalBytes	• Bytes received without error • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.958.2 Field Documentation

8.958.2.1 swi_uint256_t unpack_wds_GetByteTotals_t::ParamPresenceMask

8.958.2.2 uint64_t* unpack_wds_GetByteTotals_t::pRXTotalBytes

8.958.2.3 uint64_t* unpack_wds_GetByteTotals_t::pTXTotalBytes

8.959 unpack_wds_GetConnectionRate_t Struct Reference

Data Fields

- uint32_t currentChannelTXRate
- uint32_t currentChannelRXRate
- uint32_t maxChannelTXRate
- uint32_t maxChannelRXRate
- swi_uint256_t ParamPresenceMask

8.959.1 Detailed Description

This structure contains unpack get connection rate information.

Parameters

currentChannel-TXRate	• Current channel Tx rate (in bps) • Bit to check in ParamPresenceMask - **1**
currentChannel-RXRate	• Current channel Rx rate (in bps) • Bit to check in ParamPresenceMask - **1**
maxChannelTX-Rate	• Maximum Tx rate (bps) that may be assigned to device by serving system. • Bit to check in ParamPresenceMask - **1**
maxChannelRX-Rate	• Maximum Rx rate (bps) that may be assigned to device by serving system. • Bit to check in ParamPresenceMask - **1**

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.959.2 Field Documentation

8.959.2.1 uint32_t unpack_wds_GetConnectionRate_t::currentChannelRXRate

8.959.2.2 uint32_t unpack_wds_GetConnectionRate_t::currentChannelTXRate

8.959.2.3 uint32_t unpack_wds_GetConnectionRate_t::maxChannelRXRate

8.959.2.4 uint32_t unpack_wds_GetConnectionRate_t::maxChannelTXRate

8.959.2.5 swi_uint256_t unpack_wds_GetConnectionRate_t::ParamPresenceMask

8.960 unpack_wds_GetDataBearerTechnology_t Struct Reference

Data Fields

- uint32_t * pDataBearer
- swi_uint256_t ParamPresenceMask

8.960.1 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.

8.960.2 Field Documentation

8.960.2.1 swi_uint256_t unpack_wds_GetDataBearerTechnology_t::ParamPresenceMask

8.960.2.2 uint32_t* unpack_wds_GetDataBearerTechnology_t::pDataBearer

8.961 unpack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t pdptype
- uint32_t ipaddr
- uint32_t pridns
- uint32_t secdns
- uint16_t ipaddrv6
- uint16_t pridnsv6
- uint16_t secdnsv6
- 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

8.961.1 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.

8.961.2 Field Documentation

8.961.2.1 int8_t unpack_wds_GetDefaultProfile_t::apnname[255]

8.961.2.2 uint8_t unpack_wds_GetDefaultProfile_t::apnsize

8.961.2.3 uint32_t unpack_wds_GetDefaultProfile_t::auth

8.961.2.4 uint32_t unpack_wds_GetDefaultProfile_t::ipaddr

8.961.2.5 uint16_t unpack_wds_GetDefaultProfile_t::ipaddrv6

8.961.2.6 int8_t unpack_wds_GetDefaultProfile_t::name[255]

8.961.2.7 uint8_t unpack_wds_GetDefaultProfile_t::namesize

8.961.2.8 swi_uint256_t unpack_wds_GetDefaultProfile_t::ParamPresenceMask

8.961.2.9 uint32_t unpack_wds_GetDefaultProfile_t::pdptype

8.961.2.10 uint32_t unpack_wds_GetDefaultProfile_t::pridns

8.961.2.11 uint16_t unpack_wds_GetDefaultProfile_t::pridnsv6

8.961.2.12 uint32_t unpack_wds_GetDefaultProfile_t::secdns

8.961.2.13 uint16_t unpack_wds_GetDefaultProfile_t::secdnsv6

8.961.2.14 int8_t unpack_wds_GetDefaultProfile_t::username[255]

8.961.2.15 uint8_t unpack_wds_GetDefaultProfile_t::usersize

8.962 unpack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t index
- swi_uint256_t ParamPresenceMask

8.962.1 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.

8.962.2 Field Documentation

8.962.2.1 uint8_t unpack_wds_GetDefaultProfileNum_t::index

8.962.2.2 swi_uint256_t unpack_wds_GetDefaultProfileNum_t::ParamPresenceMask

8.963 unpack_wds_GetDormancyState_t Struct Reference

Data Fields

- uint32_t dormancyState
- swi_uint256_t ParamPresenceMask

8.963.1 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.

8.963.2 Field Documentation

8.963.2.1 uint32_t unpack_wds_GetDormancyState_t::dormancyState

8.963.2.2 swi_uint256_t unpack_wds_GetDormancyState_t::ParamPresenceMask

8.964 unpack_wds_GetLastMobileIPError_t Struct Reference

Data Fields

- uint32_t error
- swi_uint256_t ParamPresenceMask

8.964.1 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.

8.964.2 Field Documentation

8.964.2.1 uint32_t unpack_wds_GetLastMobileIPError_t::error

8.964.2.2 swi_uint256_t unpack_wds_GetLastMobileIPError_t::ParamPresenceMask

8.965 unpack_wds_GetMobileIP_t Struct Reference

Data Fields

- uint32_t mipMode
- swi_uint256_t ParamPresenceMask

8.965.1 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.

8.965.2 Field Documentation

8.965.2.1 uint32_t unpack_wds_GetMobileIP_t::mipMode

8.965.2.2 swi_uint256_t unpack_wds_GetMobileIP_t::ParamPresenceMask

8.966 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

8.966.1 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.

8.966.2 Field Documentation

8.966.2.1 uint32_t unpack_wds_GetMobileIPProfile_t::AAASPI

8.966.2.2 uint32_t unpack_wds_GetMobileIPProfile_t::AAAState

8.966.2.3 uint32_t unpack_wds_GetMobileIPProfile_t::address

8.966.2.4 uint8_t unpack_wds_GetMobileIPProfile_t::enabled

8.966.2.5 uint32_t unpack_wds_GetMobileIPProfile_t::HASPI

8.966.2.6 uint32_t unpack_wds_GetMobileIPProfile_t::HAState

8.966.2.7 int8_t unpack_wds_GetMobileIPProfile_t::NAI[255]

8.966.2.8 uint8_t unpack_wds_GetMobileIPProfile_t::naiSize

8.966.2.9 swi_uint256_t unpack_wds_GetMobileIPProfile_t::ParamPresenceMask

8.966.2.10 uint32_t unpack_wds_GetMobileIPProfile_t::primaryHA

8.966.2.11 uint8_t unpack_wds_GetMobileIPProfile_t::revTunneling

8.966.2.12 uint32_t unpack_wds_GetMobileIPProfile_t::secondaryHA

8.967 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

8.967.1 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.

8.967.2 Field Documentation

8.967.2.1 swi_uint256_t unpack_wds_GetPacketStatistics_t::ParamPresenceMask

8.967.2.2 uint32_t* unpack_wds_GetPacketStatistics_t::pRXDroppedCount

8.967.2.3 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOkBytesCount

8.967.2.4 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOKBytesLastCall

8.967.2.5 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketErrors

8.967.2.6 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketOverflows

8.967.2.7 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketSuccesses

8.967.2.8 uint32_t* unpack_wds_GetPacketStatistics_t::pTXDroppedCount

8.967.2.9 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOkBytesCount

8.967.2.10 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOKBytesLastCall

8.967.2.11 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketErrors

8.967.2.12 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketOverflows

8.967.2.13 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketSuccesses

8.968 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

8.968.1 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.

8.968.2 Field Documentation

8.968.2.1 swi_uint256_t unpack_wds_GetPacketStatus_t::ParamPresenceMask

8.968.2.2 uint32_t unpack_wds_GetPacketStatus_t::rXDroppedCount

8.968.2.3 uint64_t unpack_wds_GetPacketStatus_t::rXOkBytesCount

8.968.2.4 uint64_t unpack_wds_GetPacketStatus_t::rXOKBytesLastCall

8.968.2.5 uint32_t unpack_wds_GetPacketStatus_t::rXPacketErrors

8.968.2.6 uint32_t unpack_wds_GetPacketStatus_t::rXPacketOverflows

8.968.2.7 uint32_t unpack_wds_GetPacketStatus_t::rXPacketSuccesses

8.968.2.8 uint32_t unpack_wds_GetPacketStatus_t::tXDroppedCount

8.968.2.9 uint64_t unpack_wds_GetPacketStatus_t::tXOkBytesCount

8.968.2.10 uint64_t unpack_wds_GetPacketStatus_t::tXOKBytesLastCall

8.968.2.11 uint32_t unpack_wds_GetPacketStatus_t::tXPacketErrors

8.968.2.12 uint32_t unpack_wds_GetPacketStatus_t::tXPacketOverflows

8.968.2.13 uint32_t unpack_wds_GetPacketStatus_t::tXPacketSuccesses

8.969 unpack_wds_GetSessionDuration_t Struct Reference

Data Fields

- uint64_t callDuration
- swi_uint256_t ParamPresenceMask

8.969.1 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.

8.969.2 Field Documentation

8.969.2.1 uint64_t unpack_wds_GetSessionDuration_t::callDuration

8.969.2.2 swi_uint256_t unpack_wds_GetSessionDuration_t::ParamPresenceMask

8.970 unpack_wds_GetSessionState_t Struct Reference

Data Fields

- uint32_t connectionStatus
- swi_uint256_t ParamPresenceMask

8.970.1 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.

8.970.2 Field Documentation

8.970.2.1 uint32_t unpack_wds_GetSessionState_t::connectionStatus

8.970.2.2 swi_uint256_t unpack_wds_GetSessionState_t::ParamPresenceMask

8.971 unpack_wds_RMSetTransferStatistics_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.971.1 Detailed Description

This structure contains unpack fetch current data system transfer statistics information.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.971.2 Field Documentation

8.971.2.1 swi_uint256_t unpack_wds_RMSetTransferStatistics_t::ParamPresenceMask

8.972 unpack_wds_SetMobileIPProfile_t Struct Reference

Data Fields

- swi_uint256_t ParamPresenceMask

8.972.1 Detailed Description

This structure contains set mobile IP profile information.

Parameters

ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.972.2 Field Documentation

8.972.2.1 swi_uint256_t unpack_wds_SetMobileIPProfile_t::ParamPresenceMask

8.973 unpack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- PackCreateProfileOut * pCreateProfileOut
- uint8_t * pProfileID
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.973.1 Detailed Description

This structure contains unpack create profile information.

Parameters

pCreateProfile-Out	• SLQS Create profile Information • See PackCreateProfileOut

pProfileID	• SLQS profile identifier information
TlvresultI	• TLV present flag, TURE if presented, otherwise FALSE
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.973.2 Field Documentation

8.973.2.1 swi_uint256_t unpack_wds_SLQSCreateProfile_t::ParamPresenceMask

8.973.2.2 PackCreateProfileOut* unpack_wds_SLQSCreateProfile_t::pCreateProfileOut

8.973.2.3 uint8_t* unpack_wds_SLQSCreateProfile_t::pProfileID

8.973.2.4 uint16_t unpack_wds_SLQSCreateProfile_t::TlvresultI

8.974 unpack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint16_t extendedErrorCode
- swi_uint256_t ParamPresenceMask

8.974.1 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.

8.974.2 Field Documentation

8.974.2.1 uint16_t unpack_wds_SLQSDeleteProfile_t::extendedErrorCode

8.974.2.2 swi_uint256_t unpack_wds_SLQSDeleteProfile_t::ParamPresenceMask

8.975 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

8.975.1 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

8.975.2 Field Documentation

8.975.2.1 wds_ChannelRateTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::CRTlv

8.975.2.2 wds_ConnStatusTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::CSTlv

8.975.2.3 wds_DataBearTechTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::DBTTlv

8.975.2.4 wds_DormStatTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::DSTlv

8.975.2.5 wds_LastMdmCallEndRsnTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::LMCERTlv

8.975.2.6 swi_uint256_t unpack_wds_SLQSDUNCallInfoCallBack_ind_t::ParamPresenceMask

8.975.2.7 wds_RXBytesOKTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::RXBOTlv

8.975.2.8 wds_TXBytesOKTlv unpack_wds_SLQSDUNCallInfoCallBack_ind_t::TXBOTlv

8.976 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

8.976.1 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 MC/EM74xx • 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 (MC/EM74xx 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 MC/EM74xx 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 MC/EM74xx onwards • Bit to check in ParamPresenceMask - **20**
	ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.976.2 Field Documentation

8.976.2.1 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::_3gppRelease

8.976.2.2 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::defaultPDNEnabled

8.976.2.3 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfile

8.976.2.4 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileList[24]

8.976.2.5 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileListLen

8.976.2.6 swi_uint256_t unpack_wds_SLQSGet3GPPConfigItem_t::ParamPresenceMask

8.976.2.7 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::profileList[5]

8.977 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

Data Fields

- uint8_t prefNetwork
- uint8_t networkInfoLen
- currNetworkInfo currNetworkInfo [255]
- swi_uint256_t ParamPresenceMask

8.977.1 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.

8.977.2 Field Documentation

8.977.2.1 currNetworkInfo unpack_wds_SLQSGetCurrDataSystemStat_t::currNetworkInfo[255]

8.977.2.2 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::networkInfoLen

8.977.2.3 swi_uint256_t unpack_wds_SLQSGetCurrDataSystemStat_t::ParamPresenceMask

8.977.2.4 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::prefNetwork

8.978 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

8.978.1 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**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.978.2 Field Documentation

8.978.2.1 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_rx_rate

8.978.2.2 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_tx_rate

8.978.2.3 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_rx_rate

8.978.2.4 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_tx_rate

8.978.2.5 swi_uint256_t unpack_wds_SLQSGetCurrentChannelRate_t::ParamPresenceMask

8.979 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference

Data Fields

- uint8_t dataBearerMask
- qmiWDSDataBearerTechnology curDataBearerTechnology
- qmiWDSDataBearerTechnology lastCallDataBearerTechnology
- swi_uint256_t ParamPresenceMask

8.979.1 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.

8.979.2 Field Documentation

8.979.2.1 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::curDataBearerTechnology

8.979.2.2 uint8_t unpack_wds_SLQSGetDataBearerTechnology_t::dataBearerMask

8.979.2.3 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::lastCallDataBearerTechnology

8.979.2.4 swi_uint256_t unpack_wds_SLQSGetDataBearerTechnology_t::ParamPresenceMask

8.980 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

8.980.1 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.

8.980.2 Field Documentation

8.980.2.1 uint16_t unpack_wds_SLQSGetDUNCallInfo_t::callEndReason

8.980.2.2 dunchannelRate unpack_wds_SLQSGetDUNCallInfo_t::channelRate

8.980.2.3 connectionStatus unpack_wds_SLQSGetDUNCallInfo_t::connectionStatus

8.980.2.4 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dataBearerTech

8.980.2.5 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dormancyStatus

8.980.2.6 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallDataBearerTech

8.980.2.7 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallRXOKBytesCnt

8.980.2.8 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallTXOKBytesCnt

8.980.2.9 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::mdmCallDurationActive

8.980.2.10 swi_uint256_t unpack_wds_SLQSGetDUNCallInfo_t::ParamPresenceMask

8.980.2.11 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::rxOKBytesCount

8.980.2.12 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::txOKBytesCount

8.981 unpack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- UnPackGetProfileSettingOut * pProfileSettings
- uint8_t ProfileType
- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.981.1 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.

8.981.2 Field Documentation

8.981.2.1 swi_uint256_t unpack_wds_SLQSGetProfileSettings_t::ParamPresenceMask

8.981.2.2 UnPackGetProfileSettingOut* unpack_wds_SLQSGetProfileSettings_t::pProfileSettings

8.981.2.3 uint8_t unpack_wds_SLQSGetProfileSettings_t::ProfileType

8.981.2.4 uint16_t unpack_wds_SLQSGetProfileSettings_t::Tlvresult

8.982 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

8.982.1 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**
PCSCFAddrPCO	PCSCF address PCO • Bit to check in ParamPresenceMask - **34**
ServerAddrList	PCSCF server address list IPV4 • Bit to check in ParamPresenceMask - **35**
PCSCFFQDNAddrList	PCSCF FQDN address list IPV4 • Bit to check in ParamPresenceMask - **36**
PrimaryDNSV6	Primary DNS IPV6 • Bit to check in ParamPresenceMask - **39**
SecondryDNSV6	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**
IPFamilyPreference	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.

8.982.2 Field Documentation

8.982.2.1 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::APNName[128]

8.982.2.2 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Authentication

8.982.2.3 struct **wds_DomainNameList** unpack_wds_SLQSGetRuntimeSettings_t::DomainList

8.982.2.4 struct **wds_GPRSQoS** unpack_wds_SLQSGetRuntimeSettings_t::GPRSGrantedQoS

8.982.2.5 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::GWAddressV4

8.982.2.6 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IMCNflag

8.982.2.7 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IPFamilyPreference

8.982.2.8 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::IPv4

8.982.2.9 struct **wds_IPV6AddressInfo** unpack_wds_SLQSGetRuntimeSettings_t::IPV6AddrInfo

8.982.2.10 struct **wds_IPV6GWAddressInfo** unpack_wds_SLQSGetRuntimeSettings_t::IPV6GWAddrInfo

8.982.2.11 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Mtu

8.982.2.12 **swi_uint256_t** unpack_wds_SLQSGetRuntimeSettings_t::ParamPresenceMask

8.982.2.13 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::PCSCFAddrPCO

8.982.2.14 struct **wds_PCSCFFQDNAddressList** unpack_wds_SLQSGetRuntimeSettings_t::PCSCFFQDNAddrList

8.982.2.15 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PDPType

8.982.2.16 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV4

8.982.2.17 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV6[8]

8.982.2.18 struct **wds_ProfileIdentifier** unpack_wds_SLQSGetRuntimeSettings_t::ProfileID

8.982.2.19 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::ProfileName[128]

8.982.2.20 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV4

8.982.2.21 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV6[8]

8.982.2.22 struct **wds_PCSCFIPv4ServerAddressList** unpack_wds_SLQSGetRuntimeSettings_t::ServerAddrList

8.982.2.23 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SubnetMaskV4

8.982.2.24 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::Technology

8.982.2.25 LibPackUMTSQoS unpack_wds_SLQSGetRuntimeSettings_t::UMTSGrantedQoS

8.982.2.26 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::Username[128]

8.983 unpack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint16_t * pExtErrorCode
- swi_uint256_t ParamPresenceMask

8.983.1 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.

8.983.2 Field Documentation

8.983.2.1 swi_uint256_t unpack_wds_SLQSModifyProfile_t::ParamPresenceMask

8.983.2.2 uint16_t* unpack_wds_SLQSModifyProfile_t::pExtErrorCode

8.984 unpack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult
- swi_uint256_t ParamPresenceMask

8.984.1 Detailed Description

This structure contains unpack Set IP family preference information

Parameters

Tlvresult	unpack result
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.984.2 Field Documentation

8.984.2.1 swi_uint256_t unpack_wds_SLQSSetIPFamilyPreference_t::ParamPresenceMask

8.984.2.2 uint16_t unpack_wds_SLQSSetIPFamilyPreference_t::Tlvresult

8.985 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

8.985.1 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.

8.985.2 Field Documentation

8.985.2.1 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::bearerID

8.985.2.2 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::conn_status

8.985.2.3 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::ipFamily

8.985.2.4 swi_uint256_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::ParamPresenceMask

8.985.2.5 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::reconfigReqd

8.985.2.6 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::sessionEndReason

8.985.2.7 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::techName

8.985.2.8 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReason

8.985.2.9 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReasonType

8.986 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

8.986.1 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.

8.986.2 Field Documentation

8.986.2.1 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::currDBTechAvail

8.986.2.2 wds_currNetworkInfo unpack_wds_SLQSSetWdsEventCallback_ind_t::currNWInfo[255]

8.986.2.3 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dataSysStatAvail

8.986.2.4 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechAvail

8.986.2.5 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechExtAvail

8.986.2.6 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtRatValue

8.986.2.7 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtSoMask

8.986.2.8 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechnology

8.986.2.9 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechnologyExt

8.986.2.10 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatAvail

8.986.2.11 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatus

8.986.2.12 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipstatAvail

8.986.2.13 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipStatus

8.986.2.14 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::netInfoLen

8.986.2.15 swi_uint256_t unpack_wds_SLQSSetWdsEventCallback_ind_t::ParamPresenceMask

8.986.2.16 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::prefNetwork

8.986.2.17 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::ratMask

8.986.2.18 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_bytes

8.986.2.19 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_pkts

8.986.2.20 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::soMask

8.986.2.21 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_bytes

8.986.2.22 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_pkts

8.986.2.23 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::xferStatAvail

8.987 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4HwConfig * pHwConfig
- wdsDhcpv4OptionList * pRequestOptionList
- swi_uint256_t ParamPresenceMask

8.987.1 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.

8.987.2 Field Documentation

8.987.2.1 swi_uint256_t unpack_wds_SLQSSGetDHCPv4ClientConfig_t::ParamPresenceMask

8.987.2.2 wdsDhcpv4HwConfig* unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pHwConfig

8.987.2.3 wdsDhcpv4OptionList* unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pRequestOptionList

8.988 unpack_wds_SLQSSGetLoopback_t Struct Reference

Data Fields

- uint8_t ByteLoopbackMode
- uint8_t ByteLoopbackMultiplier
- swi_uint256_t ParamPresenceMask

8.988.1 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.

8.988.2 Field Documentation

8.988.2.1 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMode

8.988.2.2 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMultiplier

8.988.2.3 swi_uint256_t unpack_wds_SLQSSGetLoopback_t::ParamPresenceMask

8.989 unpack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint32_t * psid
- uint32_t * pFailureReason
- uint32_t * pVerboseFailReasonType
- uint32_t * pVerboseFailureReason
- swi_uint256_t ParamPresenceMask

8.989.1 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**
pVerboseFail-ReasonType	• Parameter describing type of verbose failure reason • See qaGobiApiTableCallEndReasons.h for Call End Reason Type • Bit to check in ParamPresenceMask - **17**
pVerboseFailure-Reason	• Verbose reason explaining why call failed. Depends on verbFailReasonType parameter • See qaGobiApiTableCallEndReasons.h for Call End Reason • Bit to check in ParamPresenceMask - **17**
ParamPresence-Mask	• bitmask representation to indicate valid parameters.

8.989.2 Field Documentation

8.989.2.1 swi_uint256_t unpack_wds_SLQSStartDataSession_t::ParamPresenceMask

8.989.2.2 uint32_t* unpack_wds_SLQSStartDataSession_t::pFailureReason

8.989.2.3 uint32_t* unpack_wds_SLQSStartDataSession_t::psid

8.989.2.4 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailReasonType

8.989.2.5 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailureReason

8.990 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

8.990.1 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.

8.990.2 Field Documentation

8.990.2.1 int8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::apnName[100]

8.990.2.2 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::bearerId

8.990.2.3 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.990.2.4 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4Address

8.990.2.5 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4GWAddress

8.990.2.6 ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6Address

8.990.2.7 ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6GWAddress

8.990.2.8 swi_uint256_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ParamPresenceMask

8.990.2.9 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv4Address

8.990.2.10 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv6Address[8]

8.990.2.11 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv4Address

8.990.2.12 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv6Address[8]

8.990.2.13 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv4Address

8.990.2.14 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv6Address[8]

8.990.2.15 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv4Address

8.990.2.16 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv6Address[8]

8.991 UnPackGetProfileSettingOut Struct Reference

Data Fields

- UnpackQmiProfileInfo curProfile
- uint16_t * pExtErrCode

8.991.1 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**

8.991.2 Field Documentation

8.991.2.1 UnpackQmiProfileInfo UnPackGetProfileSettingOut::curProfile

8.991.2.2 uint16_t* UnPackGetProfileSettingOut::pExtErrCode

8.992 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

8.992.1 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

8.992.2 Field Documentation

8.992.2.1 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bBinaryType

8.992.2.2 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bSerity

8.992.2.3 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bState

8.992.2.4 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bUserInputRequest

8.992.2.5 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szDescription[1024]

8.992.2.6 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szName[128]

8.992.2.7 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szVersion[128]

8.992.2.8 uint8_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::TlvPresent

8.992.2.9 uint32_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::ulPkgDownloadComplete

8.992.2.10 uint32_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::ulPkgDownloadSize

8.992.2.11 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wDescriptionLength

8.992.2.12 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wNameLength

8.992.2.13 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wUpdateCompeteStatus

8.992.2.14 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wUserInputTimeout

8.992.2.15 uint16_t UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wVersionLength

8.993 UnpackSwiAvmsEventReportConfig Struct Reference

Data Fields

- uint8_t bState
- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint16_t wAlertMsgLength
- uint8_t szAlertMsg [200]
- uint8_t TlvPresent

8.993.1 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

8.993.2 Field Documentation

8.993.2.1 uint8_t UnpackSwiAvmsEventReportConfig::bState

8.993.2.2 uint8_t UnpackSwiAvmsEventReportConfig::bUserInputRequest

8.993.2.3 uint8_t UnpackSwiAvmsEventReportConfig::szAlertMsg[200]

8.993.2.4 uint8_t UnpackSwiAvmsEventReportConfig::TlvPresent

8.993.2.5 uint16_t UnpackSwiAvmsEventReportConfig::wAlertMsgLength

8.993.2.6 uint16_t UnpackSwiAvmsEventReportConfig::wUserInputTimeout

8.994 UnpackSwiAvmsEventReportConnectionRequest Struct Reference

Data Fields

- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint8_t TlvPresent

8.994.1 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

8.994.2 Field Documentation

8.994.2.1 uint8_t UnpackSwiAvmsEventReportConnectionRequest::bUserInputRequest

8.994.2.2 uint8_t UnpackSwiAvmsEventReportConnectionRequest::TlvPresent

8.994.2.3 uint16_t UnpackSwiAvmsEventReportConnectionRequest::wUserInputTimeout

8.995 UnpackSwiAvmsEventReportDataSessionStatus Struct Reference

Data Fields

- uint8_t bType
- uint16_t wErrorCode
- uint8_t TlvPresent

8.995.1 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

8.995.2 Field Documentation

8.995.2.1 uint8_t UnpackSwiAvmsEventReportDataSessionStatus::bType

8.995.2.2 uint8_t UnpackSwiAvmsEventReportDataSessionStatus::TlvPresent

8.995.2.3 uint16_t UnpackSwiAvmsEventReportDataSessionStatus::wErrorCode

8.996 UnpackSwiAvmsEventReportHTTPStatus Struct Reference

Data Fields

- uint16_t wHTTPStatus
- uint8_t TlvPresent

8.996.1 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

8.996.2 Field Documentation

8.996.2.1 uint8_t UnpackSwiAvmsEventReportHTTPStatus::TlvPresent

8.996.2.2 uint16_t UnpackSwiAvmsEventReportHTTPStatus::wHTTPStatus

8.997 UnpackSwiAvmsEventReportNotification Struct Reference

Data Fields

- uint8_t bNotification
- uint16_t wSessionStatus
- uint8_t TlvPresent

8.997.1 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

8.997.2 Field Documentation

8.997.2.1 uint8_t UnpackSwiAvmsEventReportNotification::bNotification

8.997.2.2 uint8_t UnpackSwiAvmsEventReportNotification::TlvPresent

8.997.2.3 uint16_t UnpackSwiAvmsEventReportNotification::wSessionStatus

8.998 UnpackSwiAvmsEventReportPackageID Struct Reference

Data Fields

- uint8_t bPackageID
- uint8_t TlvPresent

8.998.1 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

8.998.2 Field Documentation

8.998.2.1 uint8_t UnpackSwiAvmsEventReportPackageID::bPackageID

8.998.2.2 uint8_t UnpackSwiAvmsEventReportPackageID::TlvPresent

8.999 UnpackSwiAvmsEventReportRegStatus Struct Reference

Data Fields

- uint8_t bRegStatus
- uint8_t TlvPresent

8.999.1 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

8.999.2 Field Documentation

8.999.2.1 uint8_t UnpackSwiAvmsEventReportRegStatus::bRegStatus

8.999.2.2 uint8_t UnpackSwiAvmsEventReportRegStatus::TlvPresent

8.1000 UnpackSwiAvmsEventReportSessionType Struct Reference

Data Fields

- uint8_t bType
- uint8_t TlvPresent

8.1000.1 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

8.1000.2 Field Documentation

8.1000.2.1 uint8_t UnpackSwiAvmsEventReportSessionType::bType

8.1000.2.2 uint8_t UnpackSwiAvmsEventReportSessionType::TlvPresent

8.1001 UnpackSwiAvmsEventReportWAMSParamChange Struct Reference

Data Fields

- uint16_t wWamsChangeMask
- uint8_t TlvPresent

8.1001.1 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

8.1001.2 Field Documentation

8.1001.2.1 uint8_t UnpackSwiAvmsEventReportWAMSParamChange::TlvPresent

8.1001.2.2 uint16_t UnpackSwiAvmsEventReportWAMSParamChange::wWamsChangeMask

8.1002 unpackWdsProfileParam Union Reference

Data Fields

- LibpackProfile3GPP SlqsProfile3GPP
- LibpackProfile3GPP2 SlqsProfile3GPP2

8.1002.1 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

8.1002.2 Field Documentation

8.1002.2.1 LibpackProfile3GPP unpackWdsProfileParam::SlqsProfile3GPP

8.1002.2.2 LibpackProfile3GPP2 unpackWdsProfileParam::SlqsProfile3GPP2

8.1003 voice_airTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t airTimerValue

8.1003.1 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.

8.1003.2 Field Documentation

8.1003.2.1 uint32_t voice_airTimer::airTimerValue

8.1003.2.2 uint8_t voice_airTimer::namID

8.1004 voice_allCallsAlphaIDInfo Struct Reference

Data Fields

- uint8_t callID
- voice_alphaIDInfo AlphaIDInfo

8.1004.1 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.

8.1004.2 Field Documentation

8.1004.2.1 voice_alphaIDInfo voice_allCallsAlphaIDInfo::AlphaIDInfo

8.1004.2.2 uint8_t voice_allCallsAlphaIDInfo::callID

8.1005 voice_allCallsDiagInfo Struct Reference

Data Fields

- uint8_t callID
- voice_diagInfo DiagInfo

8.1005.1 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.

8.1005.2 Field Documentation

8.1005.2.1 uint8_t voice_allCallsDiagInfo::callID

8.1005.2.2 voice_diagInfo voice_allCallsDiagInfo::DiagInfo

8.1006 voice_allCallsUUSInfo Struct Reference

Data Fields

- uint8_t callID
- voice_UUSInfo uusInfo

8.1006.1 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.

8.1006.2 Field Documentation

8.1006.2.1 uint8_t voice_allCallsUUSInfo::callID

8.1006.2.2 voice_UUSInfo voice_allCallsUUSInfo::uusInfo

8.1007 voice_alphaIDInfo Struct Reference

Data Fields

- uint8_t alphaDcs
- uint8_t alphaLen
- uint8_t alphaText [255]

8.1007.1 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

8.1007.2 Field Documentation

8.1007.2.1 uint8_t voice_alphaIDInfo::alphaDcs

8.1007.2.2 uint8_t voice_alphaIDInfo::alphaLen

8.1007.2.3 uint8_t voice_alphaIDInfo::alphaText[255]

8.1008 voice_arrAlertingPattern Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint32_t alertingPattern [20]

8.1008.1 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

8.1008.2 Field Documentation

8.1008.2.1 uint32_t voice_arrAlertingPattern::alertingPattern[20]

8.1008.2.2 uint8_t voice_arrAlertingPattern::callID[20]

8.1008.2.3 uint8_t voice_arrAlertingPattern::numInstances

8.1009 voice_arrAlertingType Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint8_t AlertingType [20]

8.1009.1 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

8.1009.2 Field Documentation

8.1009.2.1 uint8_t voice_arrAlertingType::AlertingType[20]

8.1009.2.2 uint8_t voice_arrAlertingType::callID[20]

8.1009.2.3 uint8_t voice_arrAlertingType::numInstances

8.1010 voice_arrAlphaID Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsAlphaIDInfo allCallsAlphaIDInfoArr [20]

8.1010.1 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.

8.1010.2 Field Documentation

8.1010.2.1 voice_allCallsAlphaIDInfo voice_arrAlphaID::allCallsAlphaIDInfoArr[20]

8.1010.2.2 uint8_t voice_arrAlphaID::numInstances

8.1011 voice_arrCalledPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo CalledPartyNum [20]

8.1011.1 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.

8.1011.2 Field Documentation

8.1011.2.1 voice_peerNumberInfo voice_arrCalledPartyNum::CalledPartyNum[20]

8.1011.2.2 uint8_t voice_arrCalledPartyNum::numInstances

8.1012 voice_arrCallEndReason Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t callEndReason [20]

8.1012.1 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

8.1012.2 Field Documentation

8.1012.2.1 uint16_t voice_arrCallEndReason::callEndReason[20]

8.1012.2.2 uint8_t voice_arrCallEndReason::callID[20]

8.1012.2.3 uint8_t voice_arrCallEndReason::numInstances

8.1013 voice_arrCallInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallInformation getAllCallInfo [20]

8.1013.1 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.

8.1013.2 Field Documentation

8.1013.2.1 voice_getAllCallInformation voice_arrCallInfo::getAllCallInfo[20]

8.1013.2.2 uint8_t voice_arrCallInfo::numInstances

8.1014 voice_arrConnectPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo ConnectedPartyNum [20]

8.1014.1 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.

8.1014.2 Field Documentation

8.1014.2.1 voice_peerNumberInfo voice_arrConnectPartyNum::ConnectedPartyNum[20]

8.1014.2.2 uint8_t voice_arrConnectPartyNum::numInstances

8.1015 voice_arrDiagInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsDiagInfo DiagInfo [20]

8.1015.1 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.

8.1015.2 Field Documentation

8.1015.2.1 voice_allCallsDiagInfo voice_arrDiagInfo::DiagInfo[20]

8.1015.2.2 uint8_t voice_arrDiagInfo::numInstances

8.1016 voice_arrRedirPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo RedirPartyNum [20]

8.1016.1 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.

8.1016.2 Field Documentation

8.1016.2.1 uint8_t voice_arrRedirPartyNum::numInstances

8.1016.2.2 voice_peerNumberInfo voice_arrRedirPartyNum::RedirPartyNum[20]

8.1017 voice_arrRemotePartyName Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyName GetAllCallRmtPtyName [20]

8.1017.1 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.

8.1017.2 Field Documentation

8.1017.2.1 voice_getAllCallRmtPtyName voice_arrRemotePartyName::GetAllCallRmtPtyName[20]

8.1017.2.2 uint8_t voice_arrRemotePartyName::numInstances

8.1018 voice_arrRemotePartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyNum RmtPtyNum [20]

8.1018.1 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.

8.1018.2 Field Documentation

8.1018.2.1 uint8_t voice_arrRemotePartyNum::numInstances

8.1018.2.2 voice_getAllCallRmtPtyNum voice_arrRemotePartyNum::RmtPtyNum[20]

8.1019 voice_arrSvcOption Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t srvOption [20]

8.1019.1 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.

8.1019.2 Field Documentation

8.1019.2.1 uint8_t voice_arrSvcOption::callID[20]

8.1019.2.2 uint8_t voice_arrSvcOption::numInstances

8.1019.2.3 uint16_t voice_arrSvcOption::srvOption[20]

8.1020 voice_arrUUSInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsUUSInfo AllCallsUUSInfo [20]

8.1020.1 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.

8.1020.2 Field Documentation

8.1020.2.1 voice_allCallsUUSInfo voice_arrUUSInfo::AllCallsUUSInfo[20]

8.1020.2.2 uint8_t voice_arrUUSInfo::numInstances

8.1021 voice_burstDTMFInfo Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t digitCnt
- uint8_t pDigitBuff [255]

8.1021.1 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

8.1021.2 Field Documentation

8.1021.2.1 uint8_t voice_burstDTMFInfo::digitCnt

8.1021.2.2 uint8_t* voice_burstDTMFInfo::pCallID

8.1021.2.3 uint8_t voice_burstDTMFInfo::pDigitBuff[255]

8.1022 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]

8.1022.1 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.

8.1022.2 Field Documentation

8.1022.2.1 uint8_t voice_calledPartyInfo::number[255]

8.1022.2.2 uint8_t voice_calledPartyInfo::numLen

8.1022.2.3 uint8_t voice_calledPartyInfo::numPlan

8.1022.2.4 uint8_t voice_calledPartyInfo::numType

8.1022.2.5 uint8_t voice_calledPartyInfo::PI

8.1022.2.6 uint8_t voice_calledPartyInfo::SI

8.1023 voice_calledPartySubAdd Struct Reference

Data Fields

- uint8_t extBit

- uint8_t subAddrType
- uint8_t oddEvenInd
- uint8_t subAddrLen
- uint8_t subAddr [255]

8.1023.1 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.

8.1023.2 Field Documentation

8.1023.2.1 uint8_t voice_calledPartySubAdd::extBit

8.1023.2.2 uint8_t voice_calledPartySubAdd::oddEvenInd

8.1023.2.3 uint8_t voice_calledPartySubAdd::subAddr[255]

8.1023.2.4 uint8_t voice_calledPartySubAdd::subAddrLen

8.1023.2.5 uint8_t voice_calledPartySubAdd::subAddrType

8.1024 voice_callerIDInfo Struct Reference

Data Fields

- uint8_t PI
- uint8_t callerIDLen
- uint8_t callerID [255]

8.1024.1 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.

8.1024.2 Field Documentation

8.1024.2.1 uint8_t voice_callerIDInfo::callerID[255]

8.1024.2.2 uint8_t voice_callerIDInfo::callerIDLen

8.1024.2.3 uint8_t voice_callerIDInfo::PI

8.1025 voice_callFwdTypeAndPlan Struct Reference

Data Fields

- uint8_t numberType
- uint8_t numberPlan

8.1025.1 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

8.1025.2 Field Documentation

8.1025.2.1 uint8_t voice_callFwdTypeAndPlan::numberPlan

8.1025.2.2 uint8_t voice_callFwdTypeAndPlan::numberType

8.1026 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]

8.1026.1 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.

8.1026.2 Field Documentation

8.1026.2.1 uint8_t voice_callFWExtInfo::noReplyTimer

8.1026.2.2 uint8_t voice_callFWExtInfo::number[255]

8.1026.2.3 uint8_t voice_callFWExtInfo::numLen

8.1026.2.4 uint8_t voice_callFWExtInfo::numPlan

8.1026.2.5 uint8_t voice_callFWExtInfo::numType

8.1026.2.6 uint8_t voice_callFWExtInfo::PI

8.1026.2.7 uint8_t voice_callFWExtInfo::SI

8.1026.2.8 uint8_t voice_callFWExtInfo::SvcClass

8.1026.2.9 uint8_t voice_callFWExtInfo::SvcStatus

8.1027 voice_callFWInfo Struct Reference

Data Fields

- uint8_t SvcStatus
- uint8_t SvcClass
- uint8_t numLen
- uint8_t number [255]
- uint8_t noReplyTimer

8.1027.1 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.

8.1027.2 Field Documentation

8.1027.2.1 uint8_t voice_callFWInfo::noReplyTimer

8.1027.2.2 uint8_t voice_callFWInfo::number[255]

8.1027.2.3 uint8_t voice_callFWInfo::numLen

8.1027.2.4 uint8_t voice_callFWInfo::SvcClass

8.1027.2.5 uint8_t voice_callFWInfo::SvcStatus

8.1028 voice_callInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t callState
- uint8_t callType
- uint8_t direction
- uint8_t mode

8.1028.1 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

8.1028.2 Field Documentation

8.1028.2.1 uint8_t voice_callInfo::callID

8.1028.2.2 uint8_t voice_callInfo::callState

8.1028.2.3 uint8_t voice_callInfo::callType

8.1028.2.4 uint8_t voice_callInfo::direction

8.1028.2.5 uint8_t voice_callInfo::mode

8.1029 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]

8.1029.1 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.

8.1029.2 Field Documentation

8.1029.2.1 uint8_t voice_callingPartyInfo::number[255]

8.1029.2.2 uint8_t voice_callingPartyInfo::numLen

8.1029.2.3 uint8_t voice_callingPartyInfo::numPlan

8.1029.2.4 uint8_t voice_callingPartyInfo::numType

8.1029.2.5 uint8_t voice_callingPartyInfo::PI

8.1029.2.6 uint8_t voice_callingPartyInfo::SI

8.1030 voice_ccSUPSType Struct Reference

Data Fields

- uint8_t svcType
- uint8_t reason

8.1030.1 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.

8.1030.2 Field Documentation

8.1030.2.1 uint8_t voice_ccSUPSType::reason

8.1030.2.2 uint8_t voice_ccSUPSType::svcType

8.1031 voice_CLIPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1031.1 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

8.1031.2 Field Documentation

8.1031.2.1 uint8_t voice_CLIPResp::ActiveStatus

8.1031.2.2 uint8_t voice_CLIPResp::ProvisionStatus

8.1032 voice_CLIRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1032.1 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

8.1032.2 Field Documentation

8.1032.2.1 uint8_t voice_CLIRResp::ActiveStatus

8.1032.2.2 uint8_t voice_CLIRResp::ProvisionStatus

8.1033 voice_CNAPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1033.1 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

8.1033.2 Field Documentation

8.1033.2.1 uint8_t voice_CNAPResp::ActiveStatus

8.1033.2.2 uint8_t voice_CNAPResp::ProvisionStatus

8.1034 voice_COLPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1034.1 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

8.1034.2 Field Documentation

8.1034.2.1 uint8_t voice_COLPResp::ActiveStatus

8.1034.2.2 uint8_t voice_COLPResp::ProvisionStatus

8.1035 voice_COLRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.1035.1 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

8.1035.2 Field Documentation

8.1035.2.1 uint8_t voice_COLRResp::ActiveStatus

8.1035.2.2 uint8_t voice_COLRResp::ProvisionStatus

8.1036 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]

8.1036.1 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.

8.1036.2 Field Documentation

8.1036.2.1 uint8_t voice_connectNumInfo::callerID[81]

8.1036.2.2 uint8_t voice_connectNumInfo::callerIDLen

8.1036.2.3 uint8_t voice_connectNumInfo::numPlan

8.1036.2.4 uint8_t voice_connectNumInfo::numPresInd

8.1036.2.5 uint8_t voice_connectNumInfo::numType

8.1036.2.6 uint8_t voice_connectNumInfo::screeningInd

8.1037 voice_CUGInfo Struct Reference

Data Fields

- uint16_t CUGIndex
- uint8_t SuppPrefCUG
- uint8_t SuppOA

8.1037.1 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

8.1037.2 Field Documentation

8.1037.2.1 uint16_t voice_CUGInfo::CUGIndex

8.1037.2.2 uint8_t voice_CUGInfo::SuppOA

8.1037.2.3 uint8_t voice_CUGInfo::SuppPrefCUG

8.1038 voice_curAMRConfig Struct Reference

Data Fields

- uint8_t gsmAmrStat
- uint8_t wcdmaAmrStat

8.1038.1 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

8.1038.2 Field Documentation

8.1038.2.1 uint8_t voice_curAMRConfig::gsmAmrStat

8.1038.2.2 uint8_t voice_curAMRConfig::wcdmaAmrStat

8.1039 voice_diagInfo Struct Reference

Data Fields

- uint8_t diagInfoLen
- uint8_t diagnosticInfo [255]

8.1039.1 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.

8.1039.2 Field Documentation

8.1039.2.1 uint8_t voice_diagInfo::diagInfoLen

8.1039.2.2 uint8_t voice_diagInfo::diagnosticInfo[255]

8.1040 voice_DTMFInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t DTMFEvent
- uint8_t digitCnt
- uint8_t digitBuff [255]

8.1040.1 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

8.1040.2 Field Documentation

8.1040.2.1 uint8_t voice_DTMFInfo::callID

8.1040.2.2 uint8_t voice_DTMFInfo::digitBuff[255]

8.1040.2.3 uint8_t voice_DTMFInfo::digitCnt

8.1040.2.4 uint8_t voice_DTMFInfo::DTMFEvent

8.1041 voice_DTMFLengths Struct Reference

Data Fields

- uint8_t DTMFPulseWidth
- uint8_t DTMFInterdigitInterval

8.1041.1 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

8.1041.2 Field Documentation

8.1041.2.1 uint8_t voice_DTMFLengths::DTMFInterdigitInterval

8.1041.2.2 uint8_t voice_DTMFLengths::DTMFPulseWidth

8.1042 voice_ECTNum Struct Reference

Data Fields

- uint8_t ECTCallState
- uint8_t presentationInd
- uint8_t number [81]

8.1042.1 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

8.1042.2 Field Documentation

8.1042.2.1 uint8_t voice_ECTNum::ECTCallState

8.1042.2.2 uint8_t voice_ECTNum::number[81]

8.1042.2.3 uint8_t voice_ECTNum::presentationInd

8.1043 voice_extDispRecInfo Struct Reference

Data Fields

- uint8_t dispType
- uint8_t extDispInfoLen
- uint8_t extDispInfo [255]

8.1043.1 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.

8.1043.2 Field Documentation

8.1043.2.1 uint8_t voice_extDispRecInfo::dispType

8.1043.2.2 uint8_t voice_extDispRecInfo::extDispInfo[255]

8.1043.2.3 uint8_t voice_extDispRecInfo::extDispInfoLen

8.1044 voice_getAllCallInformation Struct Reference

Data Fields

- voice_callInfo CallInfo
- uint8_t isEmpty
- uint8_t ALS

8.1044.1 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

8.1044.2 Field Documentation

8.1044.2.1 uint8_t voice_getAllCallInformation::ALS

8.1044.2.2 voice_callInfo voice_getAllCallInformation::CallInfo

8.1044.2.3 uint8_t voice_getAllCallInformation::isEmpty

8.1045 voice_getAllCallRmtPtyName Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyName RemotePartyName

8.1045.1 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.

8.1045.2 Field Documentation

8.1045.2.1 uint8_t voice_getAllCallRmtPtyName::callID

8.1045.2.2 voice_remotePartyName voice_getAllCallRmtPtyName::RemotePartyName

8.1046 voice_getAllCallRmtPtyNum Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyNum RemotePartyNum

8.1046.1 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.

8.1046.2 Field Documentation

8.1046.2.1 uint8_t voice_getAllCallRmtPtyNum::callID

8.1046.2.2 voice_remotePartyNum voice_getAllCallRmtPtyNum::RemotePartyNum

8.1047 voice_getCallFWExtInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWExtInfo CallFWExtInfo [20]

8.1047.1 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.

8.1047.2 Field Documentation

8.1047.2.1 voice_callFWExtInfo voice_getCallFWExtInfo::CallFWExtInfo[20]

8.1047.2.2 uint8_t voice_getCallFWExtInfo::numInstances

8.1048 voice_getCallFWInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWInfo CallFWInfo [20]

8.1048.1 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.

8.1048.2 Field Documentation

8.1048.2.1 voice_callFWInfo voice_getCallFWInfo::CallFWInfo[20]

8.1048.2.2 uint8_t voice_getCallFWInfo::numInstances

8.1049 voice_lineCtrlInfo Struct Reference

Data Fields

- uint8_t polarityIncluded
- uint8_t toggleMode
- uint8_t revPolarity
- uint8_t pwrDenialTime

8.1049.1 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

8.1049.2 Field Documentation

8.1049.2.1 uint8_t voice_lineCtrlInfo::polarityIncluded

8.1049.2.2 uint8_t voice_lineCtrlInfo::pwrDenialTime

8.1049.2.3 uint8_t voice_lineCtrlInfo::revPolarity

8.1049.2.4 uint8_t voice_lineCtrlInfo::toggleMode

8.1050 voice_newPwdData Struct Reference

Data Fields

- uint8_t newPwd [4]
- uint8_t newPwdAgain [4]

8.1050.1 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.

8.1050.2 Field Documentation

8.1050.2.1 uint8_t voice_newPwdData::newPwd[4]

8.1050.2.2 uint8_t voice_newPwdData::newPwdAgain[4]

8.1051 voice_NSSAudioCtrl Struct Reference

Data Fields

- uint8_t upLink
- uint8_t downLink

8.1051.1 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].

8.1051.2 Field Documentation

8.1051.2.1 uint8_t voice_NSSAudioCtrl::downLink

8.1051.2.2 uint8_t voice_NSSAudioCtrl::upLink

8.1052 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]

8.1052.1 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.

8.1052.2 Field Documentation

8.1052.2.1 uint8_t voice_peerNumberInfo::callID

8.1052.2.2 uint8_t voice_peerNumberInfo::number[81]

8.1052.2.3 uint8_t voice_peerNumberInfo::numLen

8.1052.2.4 uint8_t voice_peerNumberInfo::numPI

8.1052.2.5 uint8_t voice_peerNumberInfo::numPlan

8.1052.2.6 uint8_t voice_peerNumberInfo::numSI

8.1052.2.7 uint8_t voice_peerNumberInfo::numType

8.1053 voice_prefVoiceSO Struct Reference

Data Fields

- uint8_t namID
- uint8_t evrcCapability
- uint16_t homePageVoiceSO
- uint16_t homeOrigVoiceSO
- uint16_t roamOrigVoiceSO

8.1053.1 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

8.1053.2 Field Documentation

8.1053.2.1 uint8_t voice_prefVoiceSO::evrcCapability

8.1053.2.2 uint16_t voice_prefVoiceSO::homeOrigVoiceSO

8.1053.2.3 uint16_t voice_prefVoiceSO::homePageVoiceSO

8.1053.2.4 uint8_t voice_prefVoiceSO::namID

8.1053.2.5 uint16_t voice_prefVoiceSO::roamOrigVoiceSO

8.1054 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]

8.1054.1 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.

8.1054.2 Field Documentation

8.1054.2.1 uint8_t voice_redirNumInfo::number[255]

8.1054.2.2 uint8_t voice_redirNumInfo::numLen

8.1054.2.3 uint8_t voice_redirNumInfo::numPlan

8.1054.2.4 uint8_t voice_redirNumInfo::numType

8.1054.2.5 uint8_t voice_redirNumInfo::PI

8.1054.2.6 uint8_t voice_redirNumInfo::reason

8.1054.2.7 uint8_t voice_redirNumInfo::SI

8.1055 voice_remotePartyName Struct Reference

Data Fields

- uint8_t namePI
- uint8_t codingScheme
- uint8_t nameLen
- uint8_t callerName [255]

8.1055.1 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.

8.1055.2 Field Documentation

8.1055.2.1 uint8_t voice_remotePartyName::callerName[255]

8.1055.2.2 uint8_t voice_remotePartyName::codingScheme

8.1055.2.3 uint8_t voice_remotePartyName::nameLen

8.1055.2.4 uint8_t voice_remotePartyName::namePI

8.1056 voice_remotePartyNum Struct Reference

Data Fields

- uint8_t presentationInd
- uint8_t numLen
- uint8_t remPartyNumber [81]

8.1056.1 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.

8.1056.2 Field Documentation

8.1056.2.1 uint8_t voice_remotePartyNum::numLen

8.1056.2.2 uint8_t voice_remotePartyNum::presentationInd

8.1056.2.3 uint8_t voice_remotePartyNum::remPartyNumber[81]

8.1057 voice_roamTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t roamTimerValue

8.1057.1 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.

8.1057.2 Field Documentation

8.1057.2.1 uint8_t voice_roamTimer::namID

8.1057.2.2 uint32_t voice_roamTimer::roamTimerValue

8.1058 voice_signalInfo Struct Reference

Data Fields

- uint8_t signalType
- uint8_t alertPitch
- uint8_t signal

8.1058.1 Detailed Description

This structure contains Signal Information

Parameters

signalType	• Call identifier for the call.
alertPitch	• Signal Information
signal	• Caller ID Information

8.1058.2 Field Documentation

8.1058.2.1 uint8_t voice_signalInfo::alertPitch

8.1058.2.2 uint8_t voice_signalInfo::signal

8.1058.2.3 uint8_t voice_signalInfo::signalType

8.1059 voice_SUPSInfo Struct Reference

Data Fields

- uint8_t svcType
- uint8_t isModByCC

8.1059.1 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

8.1059.2 Field Documentation

8.1059.2.1 uint8_t voice_SUPSInfo::isModByCC

8.1059.2.2 uint8_t voice_SUPSInfo::svcType

8.1060 voice_USSDNotificationNetworkInfo Struct Reference

Data Fields

- uint8_t tlvPresent
- struct voice_USSInfo networkInfo

8.1060.1 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.

8.1060.2 Field Documentation

8.1060.2.1 struct voice_USSInfo voice_USSDNotificationNetworkInfo::networkInfo

8.1060.2.2 uint8_t voice_USSDNotificationNetworkInfo::tlvPresent

8.1061 voice_USSInfo Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.1061.1 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

8.1061.2 Field Documentation

8.1061.2.1 uint8_t voice_USSInfo::ussData[182]

8.1061.2.2 uint8_t voice_USSInfo::ussDCS

8.1061.2.3 uint8_t voice_USSInfo::ussLen

8.1062 voice_UUSInfo Struct Reference

Data Fields

- uint8_t UUSType
- uint8_t UUSDcs

- uint8_t UUSDatalen
- uint8_t UUSData [255]

8.1062.1 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

8.1062.2 Field Documentation

8.1062.2.1 uint8_t voice_UUSInfo::UUSData[255]

8.1062.2.2 uint8_t voice_UUSInfo::UUSDatalen

8.1062.2.3 uint8_t voice_UUSInfo::UUSDcs

8.1062.2.4 uint8_t voice_UUSInfo::UUSType

8.1063 wds_channelRate Struct Reference

Data Fields

- uint32_t CurrChanTxRate
- uint32_t CurrChanRxRate

8.1063.1 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

8.1063.2 Field Documentation

8.1063.2.1 uint32_t wds_channelRate::CurrChanRxRate

8.1063.2.2 uint32_t wds_channelRate::CurrChanTxRate

8.1064 wds_ChannelRateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- wds_channelRate ChannelRate

8.1064.1 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

8.1064.2 Field Documentation

8.1064.2.1 wds_channelRate wds_ChannelRateTlv::ChannelRate

8.1064.2.2 uint8_t wds_ChannelRateTlv::TlvPresent

8.1065 wds_ConnStatusTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t MDMConnStatus

8.1065.1 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

8.1065.2 Field Documentation

8.1065.2.1 uint8_t wds_ConnStatusTlv::MDMConnStatus

8.1065.2.2 uint8_t wds_ConnStatusTlv::TlvPresent

8.1066 wds_currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.1066.1 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

8.1066.2 Field Documentation

8.1066.2.1 uint8_t wds_currNetworkInfo::NetworkType

8.1066.2.2 uint32_t wds_currNetworkInfo::RATMask

8.1066.2.3 uint32_t wds_currNetworkInfo::SOMask

8.1067 wds_DataBearTechTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t DataBearerTech

8.1067.1 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

8.1067.2 Field Documentation

8.1067.2.1 uint8_t wds_DataBearTechTlv::DataBearerTech

8.1067.2.2 uint8_t wds_DataBearTechTlv::TlvPresent

8.1068 wds_DataULongLongTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t ullData

8.1068.1 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

8.1068.2 Field Documentation

8.1068.2.1 uint8_t wds_DataULongLongTlv::TlvPresent

8.1068.2.2 uint64_t wds_DataULongLongTlv::ullData

8.1069 wds_DataULongTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ulData

8.1069.1 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

8.1069.2 Field Documentation

8.1069.2.1 uint8_t wds_DataULongTlv::TlvPresent

8.1069.2.2 uint32_t wds_DataULongTlv::ulData

8.1070 wds_DHCPLeaseOptTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t numOpt
- wds_DHCPOpt optList [30]
- uint8_t optListData [2048]

8.1070.1 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

8.1070.2 Field Documentation

8.1070.2.1 uint8_t wds_DHCPLeaseOptTlv::numOpt

8.1070.2.2 wds_DHCPOpt wds_DHCPLeaseOptTlv::optList[30]

8.1070.2.3 uint8_t wds_DHCPLeaseOptTlv::optListData[2048]

8.1070.2.4 uint8_t wds_DHCPLeaseOptTlv::TlvPresent

8.1071 wds_DHCPLeaseStateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t leaseState

8.1071.1 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

8.1071.2 Field Documentation

8.1071.2.1 uint8_t wds_DHCPLeaseStateTlv::leaseState

8.1071.2.2 uint8_t wds_DHCPLeaseStateTlv::TlvPresent

8.1072 wds_DHCPOpt Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t * pOptVal

8.1072.1 Detailed Description

DHCP option code information

Parameters

optCode	option code • values: – 0 - 255
optValLen	length of option code • values: – 0 - 255
pOptVal	option value

8.1072.2 Field Documentation

8.1072.2.1 uint8_t wds_DHCPOpt::optCode

8.1072.2.2 uint8_t wds_DHCPOpt::optValLen

8.1072.2.3 uint8_t* wds_DHCPOpt::pOptVal

8.1073 wds_DHCPProfileIdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t profileType
- uint8_t profileId

8.1073.1 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)

8.1073.2 Field Documentation

8.1073.2.1 uint8_t wds_DHCPProfileIdTlv::profileId

8.1073.2.2 uint8_t wds_DHCPProfileIdTlv::profileType

8.1073.2.3 uint8_t wds_DHCPProfileIdTlv::TlvPresent

8.1074 wds_DHCPv4HWConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.1074.1 Detailed Description

WDS SWI DHCPv4 HW Config Structure.

Parameters

hwType	• HW Type 1 - Ethernet 20 - Serial
chaddrlen	• chaddrlen
chaddr	• chaddr. Max size 16 bytes

8.1074.2 Field Documentation

8.1074.2.1 uint8_t wds_DHCPv4HWConfig::chaddr[16]

8.1074.2.2 uint8_t wds_DHCPv4HWConfig::chaddrLen

8.1074.2.3 uint8_t wds_DHCPv4HWConfig::hwType

8.1075 wds_DHCPv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.1075.1 Detailed Description

WDS SWI DHCPv4 Option Structure

Parameters

optCode	• Option code – 0 - 255
optValLen	• Option value length – 0 - 255

optVal	• Option value

8.1075.2 Field Documentation

8.1075.2.1 uint8_t wds_DHCPv4Option::optCode

8.1075.2.2 uint8_t wds_DHCPv4Option::optVal[255]

8.1075.2.3 uint8_t wds_DHCPv4Option::optValLen

8.1076 wds_DHCPv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wds_DHCPv4Option * pOptList

8.1076.1 Detailed Description

WDS SWI DHCPv4 Option List Structure

Parameters

numOpt	• number of options – 0 - 255
pOptList	• pointer to list of DHCP Options

8.1076.2 Field Documentation

8.1076.2.1 uint8_t wds_DHCPv4OptionList::numOpt

8.1076.2.2 wds_DHCPv4Option* wds_DHCPv4OptionList::pOptList

8.1077 wds_DHCPv4ProfileId Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileId

8.1077.1 Detailed Description

WDS SWI DHCPv4 Profile Identifier Structure

Parameters

profileType	• 0 for 3GPP
profileId	• 1 to 24 for 3GPP profile

8.1077.2 Field Documentation

8.1077.2.1 uint8_t wds_DHCPv4ProfileId::profileId

8.1077.2.2 uint8_t wds_DHCPv4ProfileId::profileType

8.1078 wds_Domain Struct Reference

Data Fields

- uint16_t domainLen
- uint8_t domainName [256]

8.1078.1 Detailed Description

This structure contains the DomainName Information

Parameters

domainLen	• length of the recieved Domain name
domainName	• Domain name(Max 256 characters)

8.1078.2 Field Documentation

8.1078.2.1 uint16_t wds_Domain::domainLen

8.1078.2.2 uint8_t wds_Domain::domainName[256]

8.1079 wds_DomainNameList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_Domain domain [10]

8.1079.1 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.

8.1079.2 Field Documentation

8.1079.2.1 struct wds_Domain wds_DomainNameList::domain[10]

8.1079.2.2 uint8_t wds_DomainNameList::numInstances

8.1080 wds_DormStatTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t DormancyStat

8.1080.1 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

8.1080.2 Field Documentation

8.1080.2.1 uint8_t wds_DormStatTlv::DormancyStat

8.1080.2.2 uint8_t wds_DormStatTlv::TlvPresent

8.1081 wds_GPRSQoS Struct Reference

Data Fields

- uint32_t precedenceClass
- uint32_t delayClass
- uint32_t reliabilityClass

- uint32_t peakThroughputClass
- uint32_t meanThroughputClass

8.1081.1 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.

8.1081.2 Field Documentation

8.1081.2.1 uint32_t wds_GPRSQoS::delayClass

8.1081.2.2 uint32_t wds_GPRSQoS::meanThroughputClass

8.1081.2.3 uint32_t wds_GPRSQoS::peakThroughputClass

8.1081.2.4 uint32_t wds_GPRSQoS::precedenceClass

8.1081.2.5 uint32_t wds_GPRSQoS::reliabilityClass

8.1082 wds_IPv4AdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t IPv4Addr

8.1082.1 Detailed Description

IPv4 address TLV information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
IPv4Addr	IPv4 address

8.1082.2 Field Documentation

8.1082.2.1 uint32_t wds_IPv4AdTlv::IPv4Addr

8.1082.2.2 uint8_t wds_IPv4AdTlv::TlvPresent

8.1083 wds_IPV6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.1083.1 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.

8.1083.2 Field Documentation

8.1083.2.1 uint16_t wds_IPV6AddressInfo::IPAddressV6[8]

8.1083.2.2 uint8_t wds_IPV6AddressInfo::IPV6PrefixLen

8.1084 wds_IPV6GWAddressInfo Struct Reference

Data Fields

- uint8_t gwV6PrefixLen
- uint16_t gwAddressV6 [8]

8.1084.1 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.

8.1084.2 Field Documentation

8.1084.2.1 uint16_t wds_IPV6GWAddressInfo::gwAddressV6[8]

8.1084.2.2 uint8_t wds_IPV6GWAddressInfo::gwV6PrefixLen

8.1085 wds_LastMdmCallEndRsnTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t CallEndReason

8.1085.1 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

8.1085.2 Field Documentation

8.1085.2.1 uint16_t wds_LastMdmCallEndRsnTlv::CallEndReason

8.1085.2.2 uint8_t wds_LastMdmCallEndRsnTlv::TlvPresent

8.1086 wds_PCSCFFQDNAddress Struct Reference

Data Fields

- uint16_t fqdnLen
- uint8_t fqdnAddr [256]

8.1086.1 Detailed Description

This structure contains the PCSCFFQDNAddress Information

Parameters

fqdnLen	• length of the received FQDN address
fqdnAddr	• FQDN address(Max 256 characters)

8.1086.2 Field Documentation

8.1086.2.1 uint8_t wds_PCSCFFQDNAddress::fqdnAddr[256]

8.1086.2.2 uint16_t wds_PCSCFFQDNAddress::fqdnLen

8.1087 wds_PCSCFFQDNAddressList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_PCSCFFQDNAddress pcsfFQDNAddress [10]

8.1087.1 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.

8.1087.2 Field Documentation

8.1087.2.1 uint8_t wds_PCSCFFQDNAddressList::numInstances

8.1087.2.2 struct wds_PCSCFFQDNAddress wds_PCSCFFQDNAddressList::pcsfFQDNAddress[10]

8.1088 wds_PCSCFIPv4ServerAddressList Struct Reference

Data Fields

- uint8_t numInstances

- uint32_t pscsfIPv4Addr [64]

8.1088.1 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.

8.1088.2 Field Documentation

8.1088.2.1 uint8_t wds_PCSCFIPv4ServerAddressList::numInstances

8.1088.2.2 uint32_t wds_PCSCFIPv4ServerAddressList::pscsfIPv4Addr[64]

8.1089 wds_ProfileIdentifier Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.1089.1 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.

8.1089.2 Field Documentation

8.1089.2.1 uint8_t wds_ProfileIdentifier::profileIndex

8.1089.2.2 uint8_t wds_ProfileIdentifier::profileType

8.1090 wds_profileInfo Union Reference

Data Fields

- LibPackprofile_3GPP SlqsProfile3GPP
- LibPackprofile_3GPP2 SlqsProfile3GPP2

8.1090.1 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

8.1090.2 Field Documentation

8.1090.2.1 LibPackprofile_3GPP wds_profileInfo::SlqsProfile3GPP

8.1090.2.2 LibPackprofile_3GPP2 wds_profileInfo::SlqsProfile3GPP2

8.1091 wds_RXBytesOKTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t RxByteOKCnt

8.1091.1 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

8.1091.2 Field Documentation

8.1091.2.1 uint64_t wds_RXBytesOKTlv::RxByteOKCnt

8.1091.2.2 uint8_t wds_RXBytesOKTlv::TlvPresent

8.1092 wds_transferStatInd Struct Reference

Data Fields

- uint8_t StatsPeriod
- uint32_t StatsMask

8.1092.1 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

8.1092.2 Field Documentation

8.1092.2.1 uint32_t wds_transferStatInd::StatsMask

8.1092.2.2 uint8_t wds_transferStatInd::StatsPeriod

8.1093 wds_TrStatInd Struct Reference

Data Fields

- uint8_t statsPeriod
- uint32_t statsMask

8.1093.1 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 SLQSWdsEventReportCallBack. • All unlisted bits are reserved for future use and must be set to zero.

8.1093.2 Field Documentation

8.1093.2.1 uint32_t wds_TrStatInd::statsMask

8.1093.2.2 uint8_t wds_TrStatInd::statsPeriod

8.1094 wds_TXBytesOKTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t TxByteOKCnt

8.1094.1 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

8.1094.2 Field Documentation

8.1094.2.1 uint8_t wds_TXBytesOKTlv::TlvPresent

8.1094.2.2 uint64_t wds_TXBytesOKTlv::TxByteOKCnt

8.1095 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

8.1095.1 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.

8.1095.2 Field Documentation

8.1095.2.1 uint8_t wds_UMTSMinQoS::deliveryErrSDU

8.1095.2.2 uint32_t wds_UMTSMinQoS::grntDownlinkBitrate

8.1095.2.3 uint32_t wds_UMTSMinQoS::grntUplinkBitrate

8.1095.2.4 uint32_t wds_UMTSMinQoS::maxDownlinkBitrate

8.1095.2.5 uint32_t wds_UMTSMinQoS::maxSDUSize

8.1095.2.6 uint32_t wds_UMTSMinQoS::maxUplinkBitrate

8.1095.2.7 uint8_t wds_UMTSMinQoS::qosDeliveryOrder

8.1095.2.8 uint8_t wds_UMTSMinQoS::resBerRatio

8.1095.2.9 uint8_t wds_UMTSMinQoS::sduErrorRatio

8.1095.2.10 uint8_t wds_UMTSMinQoS::trafficClass

8.1095.2.11 uint32_t wds_UMTSMinQoS::trafficPriority

8.1095.2.12 uint32_t wds_UMTSMinQoS::transferDelay

8.1096 wdsDhcpv4HwConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.1096.1 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

8.1096.2 Field Documentation

8.1096.2.1 uint8_t wdsDhcpv4HwConfig::chaddr[16]

8.1096.2.2 uint8_t wdsDhcpv4HwConfig::chaddrLen

8.1096.2.3 uint8_t wdsDhcpv4HwConfig::hwType

8.1097 wdsDhcpv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.1097.1 Detailed Description

This structure contains DHCP V4 Option information.

Parameters

optCode	Option code • 0 - 255
optValLen	Option value length • 0 - 255
optVal	Option Value

8.1097.2 Field Documentation

8.1097.2.1 uint8_t wdsDhcpv4Option::optCode

8.1097.2.2 uint8_t wdsDhcpv4Option::optVal[255]

8.1097.2.3 uint8_t wdsDhcpv4Option::optValLen

8.1098 wdsDhcpv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wdsDhcpv4Option * pOptList

8.1098.1 Detailed Description

Structure contain DHCP V4 Option List

Parameters

numOpt	number of options • 0 - 255
pOptList	pointer to list of DHCP Options • See wdsDhcpv4Option

8.1098.2 Field Documentation

8.1098.2.1 uint8_t wdsDhcpv4OptionList::numOpt

8.1098.2.2 wdsDhcpv4Option* wdsDhcpv4OptionList::pOptList

8.1099 wdsDhcpv4ProfileId Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileId

8.1099.1 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 (EM74xx and onwards)

8.1099.2 Field Documentation

8.1099.2.1 uint8_t wdsDhcpv4ProfileId::profileId

8.1099.2.2 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 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)

9.3.1 Macro Definition Documentation

9.3.1.1 #define CAN_COMMON_EVENT_TLV_NUMBER 11

9.3.1.2 #define CAT_EVENT_DATA_MAX_LENGTH 255

9.3.2 Typedef Documentation

9.3.2.1 typedef unpack_result_t unpack_cat_CATSendEnvelopeCommand_t

9.3.2.2 typedef unpack_result_t unpack_cat_CATSendTerminalResponse_t

9.3.3 Function Documentation

9.3.3.1 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

9.3.3.2 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

9.3.3.3 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

9.3.3.4 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

9.3.3.5 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

9.3.3.6 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

9.3.3.7 int unpack_cat_SetCatEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_cat_SetCatEvent-Callback_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

9.4 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}

Typedefs

- typedef void(∗ logger)(uint8_t lvl, const char ∗buff)

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,
  eWDS,
  eDMS,
  eNAS =3,
  eQOS,
  eSMS =5,
  ePDS =0x06,
  eVOICE =0x09,
  eCAT =0x0A,
  eUIM =0x0B,
  eLOC =0x10,
  eSAR =0x11,
  eIMS =0x12,
  eTMD =0x18,
  eIMSA =0x21,
  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

9.4.1 Macro Definition Documentation

9.4.1.1 #define DEAULT_LOC_TIMEOUT_IN_SEC 2

9.4.1.2 #define MINREQBKLEN (2048)

9.4.1.3 #define MSGID_AND_LEN (4)

9.4.1.4 #define MSGID_DONT_CARE (0xffff)

9.4.1.5 #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; \
        }
```

9.4.1.6 #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;
```

9.4.1.7 #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; \
        }
```

9.4.1.8 #define SDU_HDR_LEN (3)

9.4.1.9 #define SWI_INIT_UNPACK_RESULT_VALUE {0, SWI_UINT256_INT_VALUE}

9.4.1.10 #define UNUSEDPARAM(*x*) (void)x

9.4.2 Typedef Documentation

9.4.2.1 typedef void(∗ logger)(uint8_t lvl, const char ∗buff)

9.4.3 Enumeration Type Documentation

9.4.3.1 enum eLOG_LEVEL

log levels

Enumerator

eLOG_INFO

eLOG_DEBUG

eLOG_WARN

eLOG_FATAL

9.4.3.2 enum eQMI_SVC

qmi service

Enumerator

eCTL

eWDS

eDMS

eNAS
eQOS
eSMS
ePDS
eVOICE
eCAT
eUIM
eLOC
eSAR
eIMS
eTMD
eIMSA
eRMS
eSWIOMA
eAUDIO
eSWIM2MCMD
eSWIM2MCMD_AVC2
eSWIOMAEXT
eSWIAUDIO
eSWILOC
eSWIDMS

9.4.3.3 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

9.4.3.4 enum msgtype

qmi message type

Enumerator

eREQ
eRSP
eIND

9.4.4 Function Documentation

9.4.4.1 void fill_pack_ctx (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* uint8_t *svc,* int *timeout*)

9.4.4.2 void fill_sdu_hdr (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf*)

9.4.4.3 char* get_version ( )

Returns

version string

9.4.4.4 uint16_t helper_get_error_code (uint8_t * *qmi_resp*)

9.4.4.5 char* helper_get_error_reason (uint16_t *retVal*)

9.4.4.6 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

9.4.4.7 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

9.4.4.8 uint16_t helper_get_xid (uint8_t * *qmi_resp*)

9.4.4.9 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

9.4.4.10 int helper_set_log_func (logger * *func*)

set log function

9.4.4.11 int helper_set_log_lvl (uint8_t *lvl*)

set log level

9.4.4.12 char* liteqmi_GetVersion ( )

Returns

version string

9.4.4.13 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

9.4.4.14 void liteqmi_log (uint8_t *lvl,* const char * *fmt,* ...)

9.4.4.15 unsigned unpack_result_code_only (uint8_t * *pMdmResp*)

common handler for unpacking response with TLV type 0x02 only

9.4.5 Variable Documentation

9.4.5.1 logger* glog

9.4.5.2 uint8_t gloglvl

9.5 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 unpack_dms_GetDeviceCapabilities_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_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 64
- #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 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_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_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)

9.5.1 Macro Definition Documentation

9.5.1.1 #define ACT_CODE_MAX_SIZE 81

9.5.1.2 #define CK_MAX_SIZE 8

9.5.1.3 #define DMS_IMGDETAILS_LEN 16

9.5.1.4 #define DMS_MAX_CUST_ID_LEN 64

9.5.1.5 #define DMS_MAX_CUST_VALUE_LEN 8

9.5.1.6 #define DMS_MAX_FWUPDATE_LOG_STR_SZ 255

9.5.1.7 #define DMS_MAX_FWUPDATE_REF_STR_SZ 15

9.5.1.8 #define DMS_MAX_RADIO_IFCS_SIZE 64

9.5.1.9 #define DMS_PM_FACTORY 0x02 /* Factory Test Mode */

9.5.1.10 #define DMS_PM_LOW 0x01 /* Low Power */

9.5.1.11 #define DMS_PM_OFFLINE 0x03 /* Offline */

9.5.1.12 #define DMS_PM_ONLINE 0x00 /* Online */

9.5.1.13 #define DMS_PM_PERSISTENT_LOW 0x06 /* Persistent Low Power */

9.5.1.14 #define DMS_PM_RESET 0x04 /* Reset */

9.5.1.15 #define DMS_PM_SHUT_DOWN 0x05 /* Shut Down */

9.5.1.16 #define DMS_SET_REG_IND_DISABLE 0

9.5.1.17 #define DMS_SET_REG_IND_ENABLE 1

9.5.1.18 #define DMS_SET_REG_IND_NO_CHANGE 2

9.5.1.19 #define DMS_SET_REPORT_DISABLE 0

9.5.1.20 #define DMS_SET_REPORT_ENABLE 1

9.5.1.21 #define DMS_SLQSFWINFO_APPVERSION_SZ 85

9.5.1.22 #define DMS_SLQSFWINFO_BOOTVERSION_SZ 85

9.5.1.23 #define DMS_SLQSFWINFO_CARRIER_SZ 20

9.5.1.24 #define DMS_SLQSFWINFO_CUR_CARR_NAME 17

9.5.1.25 #define DMS_SLQSFWINFO_CUR_CARR_REV 13

9.5.1.26 #define DMS_SLQSFWINFO_MODELID_SZ 20

9.5.1.27 #define DMS_SLQSFWINFO_PACKAGEID_SZ 85

9.5.1.28 #define DMS_SLQSFWINFO_PRIVERSION_SZ 16

9.5.1.29 #define DMS_SLQSFWINFO_SKU_SZ 15

9.5.1.30 #define DMS_SWI_SET_IND_DISABLE 0

9.5.1.31 #define DMS_SWI_SET_IND_ENABLE 1

9.5.1.32 #define DMS_UINT8_MAX_STRING_SZ 255

9.5.1.33 #define DMS_VALID_FSN_LEN 14

9.5.1.34 #define ERI_DATA_MAX_SIZE 1024

9.5.1.35 #define MAX_BUILD_ID_LEN 255

9.5.1.36 #define MEID_MAX_SIZE 8

9.5.1.37 #define SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160

9.5.1.38 #define SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20

9.5.1.39 #define SPC_SIZE 6

9.5.1.40 #define UNIQUE_ID_LEN 16

9.5.2 Function Documentation

9.5.2.1 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

9.5.2.2 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.

9.5.2.3 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

9.5.2.4 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

9.5.2.5 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 EM74xx/MC74xx, please use pack_dms_GetCustFeatures-V2() instead for EM74xx/MC74xx.

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

9.5.2.6 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

9.5.2.7 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

9.5.2.8 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

9.5.2.9 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

9.5.2.10 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

9.5.2.11 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

9.5.2.12 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

9.5.2.13 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

9.5.2.14 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

9.5.2.15 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

9.5.2.16 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

9.5.2.17 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 all EM74xx 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

9.5.2.18 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

9.5.2.19 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

9.5.2.20 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

9.5.2.21 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

9.5.2.22 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

9.5.2.23 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

9.5.2.24 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

9.5.2.25 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

9.5.2.26 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

9.5.2.27 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

9.5.2.28 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

9.5.2.29 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

9.5.2.30 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 EM74xx/MC74xx, please use pack_dms_SetCustFeatures-V2() instead for EM74xx/MC74xx.

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

9.5.2.31 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

9.5.2.32 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

9.5.2.33 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

9.5.2.34 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

9.5.2.35 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

9.5.2.36 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

9.5.2.37 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

9.5.2.38 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

9.5.2.39 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

9.5.2.40 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 EM/MC74xx onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.5.2.41 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

9.5.2.42 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

9.5.2.43 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

9.5.2.44 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

9.5.2.45 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

9.5.2.46 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

9.5.2.47 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

9.5.2.48 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

9.5.2.49 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

9.5.2.50 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

9.5.2.51 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

9.5.2.52 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

9.5.2.53 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

9.5.2.54 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

9.5.2.55 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

9.5.2.56 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

9.5.2.57 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 all EM74xx firmware versions. Please use API unpack_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

9.5.2.58 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

9.5.2.59 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

9.5.2.60 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

9.5.2.61 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

9.5.2.62 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 all EM74xx 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

9.5.2.63 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

9.5.2.64 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

9.5.2.65 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

9.5.2.66 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

9.5.2.67 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

9.5.2.68 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

9.5.2.69 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

9.5.2.70 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

9.5.2.71 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.

9.5.2.72 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

9.5.2.73 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

9.5.2.74 int unpack_dms_GetCustFeature (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCustFeature_t * *pOutput*)

Get Custom Feature unpack. This API is deprecated for EM74xx/MC74xx, please use unpack_dms_GetCustFeaturesV2() instead for EM74xx/MC74xx.

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

9.5.2.75 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

9.5.2.76 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

9.5.2.77 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

9.5.2.78 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

9.5.2.79 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

9.5.2.80 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

9.5.2.81 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

9.5.2.82 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

9.5.2.83 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

9.5.2.84 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

9.5.2.85 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

9.5.2.86 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 all EM74xx 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

9.5.2.87 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

9.5.2.88 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

9.5.2.89 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

9.5.2.90 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

9.5.2.91 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

9.5.2.92 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

9.5.2.93 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

9.5.2.94 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

9.5.2.95 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

9.5.2.96 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

9.5.2.97 int unpack_dms_ResetToFactoryDefaults (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactory-Defaults_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

9.5.2.98 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

9.5.2.99 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

9.5.2.100 int unpack_dms_SetCustFeature (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetCustFeature_t * *pOutput*)

Set Custom Feature unpack. This API is deprecated for EM74xx/MC74xx, please use unpack_dms_SetCustFeaturesV2() instead for EM74xx/MC74xx.

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

9.5.2.101 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

9.5.2.102 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

9.5.2.103 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

9.5.2.104 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

9.5.2.105 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

9.5.2.106 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

9.5.2.107 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

9.5.2.108 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

9.5.2.109 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

9.5.2.110 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 EM/MC74xx onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.5.2.111 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

9.5.2.112 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 EM/MC74xx onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.5.2.113 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

9.5.2.114 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

9.5.2.115 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

9.5.2.116 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

9.5.2.117 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

9.5.2.118 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

9.5.2.119 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

9.5.2.120 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

9.5.2.121 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

9.5.2.122 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

9.5.2.123 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

9.5.2.124 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

9.5.2.125 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

9.5.2.126 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

9.5.2.127 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

9.5.2.128 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

9.5.2.129 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 all EM74xx firmware versions. Please use API unpack_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

9.5.2.130 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

9.5.2.131 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

9.5.2.132 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

9.5.2.133 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

9.5.2.134 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

9.5.2.135 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

9.5.2.136 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

9.5.2.137 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

9.5.2.138 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

9.5.2.139 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

9.5.2.140 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

9.5.2.141 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

9.5.2.142 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

9.6 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)

9.6.1 Macro Definition Documentation

9.6.1.1 #define FMS_FW_PRI_BUILD_MATCH_LEN 11

9.6.1.2 #define FMS_GOBI_LISTENTRIES_MAX 2

9.6.1.3 #define FMS_GOBI_MBN_BUILD_ID_STR_LEN LITE_TOTAL_IMAGE_ID_STRING_SIZE

9.6.1.4 #define FMS_GOBI_MBN_IMG_ID_SIZE 16

9.6.1.5 #define FMS_GOBI_MBN_IMG_ID_STR_LEN LITE_TOTAL_IMAGE_ID_STRING_SIZE

9.6.1.6 #define FMS_IMAGE_ID_BUILD_ID_LEN 32

9.6.1.7 #define FMS_IMAGE_ID_IMG_ID_LEN 16

9.6.1.8 #define FMS_IMAGE_ID_MAX_ENTRIES 2

9.6.1.9 #define FMS_IMAGE_ID_PRI_IMGTYPE 0x01

9.6.1.10 #define FMS_MAX_IMAGE_ID_ELEMENT 50

9.6.1.11 #define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255

9.6.1.12 #define LITE_TOTAL_IMAGE_ID_STRING_SIZE 100

9.6.2 Function Documentation

9.6.2.1 uint32_t GetValidFwPriCombinations (FMSImageList * *pStoredImageList*, uint32_t * *pValidCombinationSize*, CarrierImage_t * *pValidCombinations*)

This API distills valid Firmware/PRI combinations from GetStoredImages result

Parameters

in	*pStoredImage-List*	• image list returned from GetStoredImages • See FMSImageList
in,out	*pValid-CombinationSize*	• number of combination passed in and returned
out	*pValid-Combinations*	• 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

9.6.2.2 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

9.6.2.3 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

9.6.2.4 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

9.6.2.5 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

9.6.2.6 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

9.6.2.7 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

9.7 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)

9.7.1 Macro Definition Documentation

9.7.1.1 #define MAX_NAME_LEN 255

9.7.2 Typedef Documentation

9.7.2.1 typedef unpack_result_t unpack_ims_SLQSImsConfigIndicationRegister_t

9.7.3 Function Documentation

9.7.3.1 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, MC74xx and EM74xx

9.7.3.2 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, MC74xx and EM74xx

9.7.3.3 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, MC74xx and EM74xx

9.7.3.4 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, MC74xx and EM74xx

9.7.3.5 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, MC74xx and EM74xx

9.7.3.6 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, MC74xx and EM74xx

```
- This function is used by a device to register/deregister for different
  QMI IMS indications.
- The device'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.
```

9.7.3.7 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, MC74xx and EM74xx

9.7.3.8 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, MC74xx and EM74xx

9.7.3.9 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, MC74xx and EM74xx

9.7.3.10 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, MC74xx and EM74xx

9.7.3.11 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, MC74xx and EM74xx

9.7.3.12 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

9.7.3.13 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

9.7.3.14 int unpack_ims_SLQSGetIMSVoIPConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSGetIMSVoIP-Config_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

9.7.3.15 int unpack_ims_SLQSGetRegMgrConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_ims_SLQSGetRegMgr-Config_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

9.7.3.16 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

9.7.3.17 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

9.7.3.18 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

9.7.3.19 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

9.7.3.20 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

9.7.3.21 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

9.7.3.22 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

9.7.3.23 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

9.7.3.24 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

9.7.3.25 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

9.7.3.26 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

9.7.3.27 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

9.8 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)

9.8.1 Macro Definition Documentation

9.8.1.1 #define MAX_ERROR_CODE_LEN 256

9.8.2 Typedef Documentation

9.8.2.1 typedef unpack_result_t unpack_imsa_SLQSRegisterIMSAIndication_t

9.8.3 Function Documentation

9.8.3.1 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

9.8.3.2 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

9.8.3.3 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

9.8.3.4 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

9.8.3.5 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

9.8.3.6 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

9.8.3.7 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

9.8.3.8 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

9.8.3.9 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

9.8.3.10 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

9.9 lite-fw.h File Reference

Data Structures

- struct _litefw_FirmwareInfo_
- struct _litefw_FirmwarePartNo_

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 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 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,
  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_Models {
  eModel_Unknown =-1,
  eModel_9X15 =0,
  eModel_WP9X15 =1,
  eModel_9X30 =2,
  eModel_9x07 =3,
  eModel_9x06 =4 }
- 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 ()

9.9.1 Detailed Description

Filename: lite-fw.h

Purpose: Global definitions used inside the SDK

Copyright: © 2016 Sierra Wireless Inc., all rights reserved

9.9.2 Macro Definition Documentation

9.9.2.1 #define FIRMWARE_INFO_STRING_SIZE 20

9.9.2.2 #define IMG_MASK_CLEAR (0x000)

9.9.2.3 #define IMG_MASK_GENERIC (0xFFFF)

9.9.2.4 #define IMG_MASK_MDM (1<<0)

9.9.2.5 #define IMG_MASK_PRI (1<<1)

9.9.2.6 #define libSDP_BuildImagesPreferenceRequest litefw_BuildImagesPreferenceRequest

9.9.2.7 #define libSDP_CalculateImageMask litefw_CalculateImageMask

9.9.2.8 #define LIBSDP_CARRIER_PACKAGE_SKU LITEFW_CARRIER_PACKAGE_SKU

9.9.2.9 #define libSDP_CheckValidFirmwareInfo litefw_CheckValidFirmwareInfo

9.9.2.10 #define libSDP_DownloadFW litefw_DownloadFW

9.9.2.11 #define libSDP_ExtractFirmwareParametersByPath litefw_ExtractFirmwareParametersByPath

9.9.2.12 #define libSDP_getFileType litefw_getFileType

9.9.2.13 #define libSDP_GetModelFamily litefw_GetModelFamily

9.9.2.14 #define libSDP_GetVersion litefw_GetVersion

9.9.2.15 #define libsdp_set_log_func litefw_set_log_func

9.9.2.16 #define libsdp_SetReadBlockSize litefw_SetReadBlockSize

9.9.2.17 #define LIBSDP_SKU_STRING_LENGTH LITEFW_SKU_STRING_LENGTH

9.9.2.18 #define LITEFW_CARRIER_PACKAGE_SKU "9999999"

9.9.2.19 #define LITEFW_SKU_STRING_LENGTH 7

9.9.2.20 #define NULL_TERMINATOR_CHAR_SIZE 1

9.9.3 Typedef Documentation

9.9.3.1 typedef litefw_FirmwareInfo libSDP_FirmwareInfo

9.9.3.2 typedef litefwlogger libsdplogger

9.9.3.3 typedef struct _litefw_FirmwareInfo_ litefw_FirmwareInfo

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

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.

9.9.3.4 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

9.9.3.5 typedef void(∗ litefwlogger)(uint8_t lvl, const char ∗buff)

This Custom Log prototype.

Parameters

in	*lvl*	• Log level.
in	*buff*	• Log String.

Returns

none

9.9.4 Enumeration Type Documentation

9.9.4.1 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

9.9.4.2 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

eSDP_FWDWL_ERR_END

9.9.4.3 enum litefw_Models

Enumerator

eModel_Unknown

eModel_9X15

eModel_WP9X15

eModel_9X30

eModel_9x07

eModel_9x06

9.9.4.4 enum litefw_QDL_FLOW_CONTROLs

Enumerator

eQDL_HW_FLOW_Unknown

eQDL_HW_FLOW_INIT

eQDL_HW_FLOW_ENABLE

eQDL_HW_FLOW_DISABLE

9.9.4.5 enum litefw_QDL_MODEs

Enumerator

eQDL_MODE_Unknown

eQDL_MODE_INIT

eQDL_MODE_TTYUSB

eQDL_MODE_UART0

9.9.5 Function Documentation

9.9.5.1 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

9.9.5.2 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

9.9.5.3 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

9.9.5.4 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

9.9.5.5 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

9.9.5.6 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

9.9.5.7 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

9.9.5.8 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

9.9.5.9 int litefw_GetQTLDownloadMode ( )

This API Get QDL port donwload mode.

Returns

eQDL_MODE_Unknown on Error, litefw_QDL_MODEs value otherwise

9.9.5.10 int litefw_GetQTLHWFlowControl ( )

This API Get QDL port Hardware flow control.

Returns

eQDL_HW_FLOW_Unknown on Error, litefw_QDL_FLOW_CONTROLs value otherwise

9.9.5.11 char* litefw_GetVersion ( )

This API Get Lib SDP Version.

Returns

Version String

9.9.5.12 void litefw_logsenable (int *log_en*)

This function enable/disable lite firmware debug logs.

Parameters

in	*log_en*	• Logs enable/disable flag.

9.9.5.13 int litefw_set_log_func (litefwlogger * *func*)

This API Set Custom Log function.

Parameters

in	*func*	• See litefwlogger for more information.

Returns

none

9.9.5.14 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

9.9.5.15 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

9.9.5.16 void litefw_SetReadBlockSize (unsigned long *lBlockSize*)

This API Set Read BlockSize.

Parameters

in	*lBlockSize*	• Firmware Read Block Size.

Returns

none

9.9.5.17 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

9.9.5.18 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

9.10 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

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

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)

9.10.1 Macro Definition Documentation

9.10.1.1 #define LOC_UINT8_MAX_STRING_SZ 255

9.10.1.2 #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.

9.10.1.3 #define LOCEVENTMASKENGINESTATE 0x00000080

The control point must enable this mask to receive engine state report event indications.

9.10.1.4 #define LOCEVENTMASKFIXSESSIONSTATE 0x00000100

The control point must enable this mask to receive fix session status report event indications.

9.10.1.5 #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.

9.10.1.6 #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.

9.10.1.7 #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.

9.10.1.8 #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).

9.10.1.9 #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.

9.10.1.10 #define LOCEVENTMASKINJECTPOSITIONREQ 0x00000040

The control point must enable this mask to receive position injection request event indications.

9.10.1.11 #define LOCEVENTMASKINJECTPREDICTEDORBITSREQ 0x00000020

The control point must enable this mask to receive predicted orbits request event indications.

9.10.1.12 #define LOCEVENTMASKINJECTTIMEREQ 0x00000010

The control point must enable this mask to receive time injection request event indications.

9.10.1.13 #define LOCEVENTMASKINJECTWIFIAPDATAREQ 0x00200000

The control point must enable this mask to receive Wi-Fi Access Point (AP) data inject request event indications.

9.10.1.14 #define LOCEVENTMASKINVALIDVALUE 0xFFFFFFFF

Invalid Event Mask

9.10.1.15 #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.

9.10.1.16 #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.

9.10.1.17 #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.

9.10.1.18 #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.

9.10.1.19 #define LOCEVENTMASKNINOTIFYVERIFYREQ 0x00000008

The control point must enable this mask to receive NI Notify/Verify request event indications.

9.10.1.20 #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.

9.10.1.21 #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.

9.10.1.22 #define LOCEVENTMASKPOSITIONREPORT 0x00000001

The control point must enable this mask to receive position report event indications.

9.10.1.23 #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.).

9.10.1.24 #define LOCEVENTMASKSETSPISTREAMINGREPORT 0x00001000

The control point must enable this mask to receive Stationary Position Indicator (SPI) streaming report indications.

9.10.1.25 #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.

9.10.1.26 #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.).

9.10.1.27 #define LOCEVENTMASKWIFIREQ 0x00000200

The control point must enable this mask to receive Wi-Fi position request event indications.

9.10.1.28 #define MAX_LOC_NMEA_STR_LEN 201

9.10.1.29 #define MAX_SENSOR_DATA_LEN 64

9.10.1.30 #define MAX_TEMP_DATA_LEN 64

9.10.2 Typedef Documentation

9.10.2.1 typedef unpack_loc_Delete_Assist_Data_t unpack_loc_DeleteAssistData_t

9.10.2.2 typedef unpack_result_t unpack_loc_SLQSLOCGetServer_t

9.10.2.3 typedef unpack_result_t unpack_loc_SLQSLOCInjectPosition_t

9.10.2.4 typedef unpack_result_t unpack_loc_SLQSLOCInjectSensorData_t

9.10.2.5 typedef unpack_result_t unpack_loc_SLQSLOCInjectUTCTime_t

9.10.2.6 typedef unpack_result_t unpack_loc_SLQSLOCSetCradleMountConfig_t

9.10.3 Enumeration Type Documentation

9.10.3.1 anonymous enum

Enumerator

eQMI_LOC_SESS_STATUS_SUCCESS

eQMI_LOC_SESS_STATUS_IN_PROGRESS

eQMI_LOC_SESS_STATUS_FAILURE

eQMI_LOC_SESS_STATUS_TIMEOUT

9.10.4 Function Documentation

9.10.4.1 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

9.10.4.2 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

9.10.4.3 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

9.10.4.4 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

9.10.4.5 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

9.10.4.6 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

9.10.4.7 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

9.10.4.8 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

9.10.4.9 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

9.10.4.10 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

9.10.4.11 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

9.10.4.12 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

9.10.4.13 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

9.10.4.14 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

9.10.4.15 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

9.10.4.16 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

9.10.4.17 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

9.10.4.18 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

9.10.4.19 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

9.10.4.20 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

9.10.4.21 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

9.10.4.22 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

9.10.4.23 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

9.10.4.24 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

9.10.4.25 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

9.10.4.26 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

9.10.4.27 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

9.10.4.28 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

9.10.4.29 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

9.10.4.30 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

9.10.4.31 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

9.10.4.32 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

9.10.4.33 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

9.10.4.34 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

9.10.4.35 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

9.10.4.36 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

9.10.4.37 int unpack_loc_SLQSLOCInjectPosition (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCInjectPosition_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

9.10.4.38 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

9.10.4.39 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

9.10.4.40 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

9.10.4.41 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

9.10.4.42 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

9.10.4.43 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

9.11 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 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_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 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 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_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_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 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 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 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

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

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_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

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_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 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_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 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_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 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 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)

9.11.1 Macro Definition Documentation

9.11.1.1 #define LITE_NW_SCAN_LTE_OP_MODE_MAX_LENGTH 255

9.11.1.2 #define NAS_IMSI_M_S1_LENGTH 7

9.11.1.3 #define NAS_IMSI_M_S2_LENGTH 3

9.11.1.4 #define NAS_MAX_DESCRIPTION_LENGTH 255

9.11.1.5 #define NAS_MAX_NUM_NETWORKS 30

9.11.1.6 #define NAS_MAX_PHY_CA_AGG_SCELL_ARRRY_SIZE 255

9.11.1.7 #define NAS_MAX_SCC_RX_INFO_INSTANCES 255

9.11.1.8 #define NAS_MAX_SLQS_NAS_PCI_INFO_LENGTH 255

9.11.1.9 #define NAS_MAX_SLQS_NAS_PCI_INFO_PLMN_LENGTH 255

9.11.1.10 #define NAS_MCC_MNC_INST_LENGTH 255

9.11.1.11 #define NAS_NAM_NAME_LENGTH 12

9.11.1.12 #define NAS_OTA_MESSAGE_MAX_BUF_SIZE 2048

9.11.1.13 #define NAS_PLMN_LENGTH 3

9.11.1.14 #define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST 255

9.11.2 Typedef Documentation

9.11.2.1 typedef nas_BandPrefInfoTlv nas_LTEBandPrefExtTlv

Contain the LTE Band Preference Extended.

- See nas_BandPrefInfoTlv for more information.

9.11.2.2 typedef nas_BandPrefInfoTlv nas_NR5gBandPrefTlv

Contain the NR5G Band Preference.

- See nas_BandPrefInfoTlv for more information.

9.11.2.3 typedef unpack_result_t unpack_nas_InitiateDomainAttach_t

9.11.2.4 typedef unpack_result_t unpack_nas_SetACCOLC_t

9.11.2.5 typedef unpack_result_t unpack_nas_SetCDMANetworkParameters_t

9.11.2.6 typedef unpack_result_t unpack_nas_SetLURejectCallback_t

9.11.2.7 typedef unpack_result_t unpack_nas_SetRFInfoCallback_t

9.11.2.8 typedef unpack_result_t unpack_nas_SLQSConfigSigInfo_t

9.11.2.9 typedef unpack_result_t unpack_nas_SLQSInitiateNetworkRegistration_t

9.11.2.10 typedef unpack_result_t unpack_nas_SLQSNasConfigSigInfo2_t

9.11.2.11 typedef unpack_result_t unpack_nas_SLQSNasIndicationRegisterExt_t

9.11.2.12 typedef unpack_result_t unpack_nas_SLQSNASSeteDRXParams_t

9.11.2.13 typedef unpack_result_t unpack_nas_SLQSNasSwiIndicationRegister_t

9.11.2.14 typedef unpack_result_t unpack_nas_SLQSNASSwiSetChannelLock_t

9.11.2.15 typedef unpack_result_t unpack_nas_SLQSSetBandPreference_t

9.11.2.16 typedef unpack_result_t unpack_nas_SLQSSetSignalStrengthsCallback_t

9.11.2.17 typedef unpack_result_t unpack_nas_SLQSSetSysSelectionPref_t

9.11.2.18 typedef unpack_result_t unpack_nas_SLQSSetSysSelectionPrefExt_t

9.11.2.19 typedef unpack_result_t unpack_nas_SLQSSwiPSDetach_t

9.11.3 Enumeration Type Documentation

9.11.3.1 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

9.11.3.2 enum LITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND

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

9.11.3.3 enum LITEQMI_NAS_LTE_CPHY_CA_BW_NRB

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

9.11.3.4 enum LITEQMI_NAS_LTE_CPHY_SCELL_STATE

Enumerator

eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED

eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED

eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED

9.11.3.5 enum NAS_LTE_CPHY_CA_BW_NRB_LITE

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

9.11.3.6 enum NAS_LTE_CPHY_SCELL_STATE_LITE

Enumerator

eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE

eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE

eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE

9.11.4 Function Documentation

9.11.4.1 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

9.11.4.2 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

9.11.4.3 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

9.11.4.4 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

9.11.4.5 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

9.11.4.6 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

9.11.4.7 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

9.11.4.8 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

9.11.4.9 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

9.11.4.10 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 all EM74xx 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

9.11.4.11 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

9.11.4.12 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

9.11.4.13 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 and WP76xx.

9.11.4.14 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

9.11.4.15 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

9.11.4.16 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

9.11.4.17 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 all EM74xx firmware versions. Please use pack_nas_SLQSSetSysSelectionPref() 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

9.11.4.18 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

9.11.4.19 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

9.11.4.20 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

9.11.4.21 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

9.11.4.22 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

9.11.4.23 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

9.11.4.24 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

9.11.4.25 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

9.11.4.26 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 all EM74xx firmware versions. Please use pack_nas_SLQS-NasGetSigInfo() 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

9.11.4.27 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

9.11.4.28 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

9.11.4.29 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

9.11.4.30 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 all EM74xx 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

9.11.4.31 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

9.11.4.32 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 prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.11.4.33 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

9.11.4.34 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

9.11.4.35 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

9.11.4.36 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

9.11.4.37 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

9.11.4.38 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

9.11.4.39 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

9.11.4.40 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

9.11.4.41 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

9.11.4.42 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

9.11.4.43 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

9.11.4.44 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

9.11.4.45 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

9.11.4.46 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

9.11.4.47 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

9.11.4.48 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

9.11.4.49 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

9.11.4.50 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

9.11.4.51 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

9.11.4.52 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

9.11.4.53 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

9.11.4.54 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

9.11.4.55 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

9.11.4.56 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

9.11.4.57 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

9.11.4.58 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

9.11.4.59 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

9.11.4.60 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

9.11.4.61 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

9.11.4.62 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

9.11.4.63 int unpack_nas_GetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetNetwork-Preference_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

9.11.4.64 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

9.11.4.65 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 all EM74xx 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

9.11.4.66 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

9.11.4.67 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 all EM74xx 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

9.11.4.68 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

9.11.4.69 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

9.11.4.70 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

9.11.4.71 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

9.11.4.72 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

9.11.4.73 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

9.11.4.74 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

9.11.4.75 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

9.11.4.76 int unpack_nas_SetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetNetwork-Preference_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

9.11.4.77 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

9.11.4.78 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

9.11.4.79 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

9.11.4.80 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

9.11.4.81 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

9.11.4.82 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

9.11.4.83 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

9.11.4.84 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

9.11.4.85 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

9.11.4.86 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

9.11.4.87 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 SWI9X15C_05_xx_xx_xx and all EM74xx 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

9.11.4.88 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

9.11.4.89 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

9.11.4.90 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

9.11.4.91 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 EM74xx 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

9.11.4.92 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

9.11.4.93 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

9.11.4.94 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

9.11.4.95 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

9.11.4.96 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

9.11.4.97 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

9.11.4.98 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

9.11.4.99 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

9.11.4.100 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

9.11.4.101 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

9.11.4.102 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

9.11.4.103 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

9.11.4.104 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

9.11.4.105 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

9.11.4.106 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

9.11.4.107 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

9.11.4.108 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

9.11.4.109 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

9.11.4.110 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

9.11.4.111 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

9.11.4.112 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

9.11.4.113 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

9.11.4.114 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

9.11.4.115 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

9.11.4.116 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

9.11.4.117 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

9.11.4.118 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

9.11.4.119 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

9.11.4.120 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

9.11.4.121 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

9.11.4.122 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

9.11.4.123 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

9.11.4.124 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

9.11.4.125 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

9.11.4.126 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

9.11.4.127 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

9.11.4.128 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

9.11.4.129 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

9.11.4.130 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

9.11.4.131 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

9.11.4.132 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

9.12 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)

9.12.1 Typedef Documentation

9.12.1.1 typedef unpack_result_t unpack_pds_ForceXTRADownload_t

9.12.1.2 typedef unpack_result_t unpack_pds_PDSInjectTimeReference_t

9.12.1.3 typedef unpack_result_t unpack_pds_ResetPDSData_t

9.12.1.4 typedef unpack_result_t unpack_pds_SetEventReportCallback_t

9.12.1.5 typedef unpack_result_t unpack_pds_SetPDSDefaults_t

9.12.1.6 typedef unpack_result_t unpack_pds_SetPDSState_t

9.12.1.7 typedef unpack_result_t unpack_pds_SetPortAutomaticTracking_t

9.12.1.8 typedef unpack_result_t unpack_pds_SetServiceAutomaticTracking_t

9.12.1.9 typedef unpack_result_t unpack_pds_SetXTRAAutomaticDownload_t

9.12.1.10 typedef unpack_result_t unpack_pds_SetXTRANetwork_t

9.12.1.11 typedef unpack_result_t unpack_pds_SLQSPDSDeterminePosition_t

9.12.1.12 typedef unpack_result_t unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t

9.12.1.13 typedef unpack_result_t unpack_pds_SLQSPDSInjectPositionData_t

9.12.1.14 typedef unpack_result_t unpack_pds_SLQSSetAGPSConfig_t

9.12.1.15 typedef unpack_result_t unpack_pds_SLQSSetPositionMethodState_t

9.12.1.16 typedef unpack_result_t unpack_pds_StartPDSTrackingSessionExt_t

9.12.1.17 typedef unpack_result_t unpack_pds_StopPDSTrackingSession_t

9.12.2 Function Documentation

9.12.2.1 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

9.12.2.2 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

9.12.2.3 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

9.12.2.4 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

9.12.2.5 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

9.12.2.6 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

9.12.2.7 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

9.12.2.8 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

9.12.2.9 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

9.12.2.10 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

9.12.2.11 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

9.12.2.12 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

9.12.2.13 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

9.12.2.14 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

9.12.2.15 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

9.12.2.16 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

9.12.2.17 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

9.12.2.18 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

9.12.2.19 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

9.12.2.20 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

9.12.2.21 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

9.12.2.22 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

9.12.2.23 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

9.12.2.24 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

9.12.2.25 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

9.12.2.26 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

9.12.2.27 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

9.12.2.28 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

9.12.2.29 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

9.12.2.30 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

9.12.2.31 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

9.12.2.32 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

9.12.2.33 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

9.12.2.34 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

9.12.2.35 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

9.12.2.36 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

9.12.2.37 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

9.12.2.38 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

9.12.2.39 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

9.12.2.40 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

9.12.2.41 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

9.12.2.42 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

9.12.2.43 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

9.12.2.44 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

9.12.2.45 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

9.12.2.46 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

9.12.2.47 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

9.12.2.48 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

9.12.2.49 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

9.12.2.50 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

9.12.2.51 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

9.12.2.52 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

9.12.2.53 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

9.12.2.54 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

9.13 qaGobiApiTableBandClasses.h File Reference

Network Access Service API Band Classes table.

9.13.1 Detailed Description

Network Access Service API Band Classes table.

9.13.2 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

9.13.2.1 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 Sierra Wireless, Inc. all rights reserved

9.14 qaGobiApiTableCallControlReturnReasons.h File Reference

Call Control Return Reasons table.

9.14.1 Detailed Description

Call Control Return Reasons table.

9.14.2 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 Sierra Wireless, Inc. all rights reserved

9.15 qaGobiApiTableCallEndReasons.h File Reference

Wireless Data Service Call End Reasons.

9.15.1 Detailed Description

Wireless Data Service Call End Reasons.

9.15.2 Call end reason codes (Code - Reason)

9.15.2.1 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

9.15.2.2 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

9.15.2.3 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

9.15.2.4 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

9.15.2.5 call end reason type

- 1 - Mobile IP
- 2 - Internal
- 3 - Call Manager defined
- 6 - 3GPP specification defined
- 7 - PPP
- 8 - EHRPD
- 9 - IPv6

9.15.2.6 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

9.15.2.7 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

9.15.2.8 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

9.15.2.9 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

9.15.2.10 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

9.15.2.11 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

9.15.2.12 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: © 2011-2013 Sierra Wireless, Inc. all rights reserved

9.16 qaGobiApiTableCarrierCodes.h File Reference

Carrier Codes table.

9.16.1 Detailed Description

Carrier Codes table.

9.16.2 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: © 2011-2014 Sierra Wireless, Inc. all rights reserved

9.17 qaGobiApiTableCodingScheme.h File Reference

Data Coding Scheme.

Macros

- #define __GOBI_API_CODING_SCHEME_H__

9.17.1 Detailed Description

Data Coding Scheme.

9.17.2 S1

9.17.2.1 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

9.17.3 Coding Group Bits 7..4(0001)

9.17.3.1 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.

9.17.4 Coding Group Bits 7..4(0010)

9.17.4.1 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

9.17.5 Coding Group Bits 7..4(0011)

9.17.5.1 use of bits 3..0

- 0000..1111 Reserved for other languages using the GSM 7 bit
  default alphabet, with unspecified handling at the MS

9.17.6 Coding Group Bits 7..4(01xx)

9.17.6.1 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

9.17.7 Coding Group Bits 7..4(1001)

9.17.7.1 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

9.17.8 Coding Group Bits 7..4(1010..1101)

9.17.8.1 Reserved coding groups

9.17.9 Coding Group Bits 7..4(1110)

9.17.9.1 Defined by the WAP Forum

9.17.10 Coding Group Bits 7..4 (1111)

9.17.10.1 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 Sierra Wireless, Inc. all rights reserved

9.17.11 Macro Definition Documentation

9.17.11.1 #define __GOBI_API_CODING_SCHEME_H__

9.18 qaGobiApiTableGpsCapabilityCodes.h File Reference

Position Determination Service API GPS Capability Codes.

9.18.1 Detailed Description

Position Determination Service API GPS Capability Codes.

9.18.2 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 Sierra Wireless, Inc. all rights reserved

9.19 qaGobiApiTablePowerModes.h File Reference

Device Management Service API Power Modes table.

9.19.1 Detailed Description

Device Management Service API Power Modes table.

9.19.2 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 Sierra Wireless, Inc. all rights reserved

9.20 qaGobiApiTableRadioInterfaces.h File Reference

Network Access Service API Radio Interfaces table.

9.20.1 Detailed Description

Network Access Service API Radio Interfaces table.

9.20.2 Radio interface

9.20.2.1 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 Sierra Wireless, Inc. all rights reserved

9.21 qaGobiApiTableRegionCodes.h File Reference

Region Codes table.

9.21.1 Detailed Description

Region Codes table.

9.21.2 Region Codes (Code - Region)

- 0 - North America
- 1 - Latin America
- 2 - Europe
- 3 - Asia
- 4 - Australia
- 5 - Global
- 0xFFFFFFFF - Unknown

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.22 qaGobiApiTableServiceOptions.h File Reference

Voice Service Options.

9.22.1 Detailed Description

Voice Service Options.

9.22.2 Service Option codes (Code - Reason)

9.22.2.1 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 Sierra Wireless, Inc. all rights reserved

9.23 qaGobiApiTableSupServiceInfoClasses.h File Reference

Voice Supplementary Service Information Classes.

9.23.1 Detailed Description

Voice Supplementary Service Information Classes.

9.23.2 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 Sierra Wireless, Inc. all rights reserved

9.24 qaGobiApiTableSwiAudio.h File Reference

Swi Audio related tables.

9.24.1 Detailed Description

Swi Audio related tables.

9.24.2 ACDB Device (Device ID - description)

- 0 - Vehicle HF
- 1 - Handset
- 2 - TTY
- 3 - USB
- 4 - NA

9.24.3 Physical Interface (Device ID - description - Interface parameters)

- 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 Sierra Wireless, Inc. all rights reserved

9.25 qaGobiApiTableSwiOMADMSessionStatus.h File Reference

Session Status table.

9.25.1 Detailed Description

Session Status table.

9.25.2 OMA DM Session Status (Session Status - Meaning - Usage)

- 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 Sierra Wireless, Inc. all rights reserved

9.26 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference

Update Complete Status table.

9.26.1 Detailed Description

Update Complete Status table.

9.26.2 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 Sierra Wireless, Inc. all rights reserved

9.27 qaGobiApiTableVoiceCallEndReasons.h File Reference

Voice Service Call and supplementary services end reasons.

9.27.1 Detailed Description

Voice Service Call and supplementary services end reasons.

9.27.2 Voice Call and supplementary services end reason codes (Code - Reason)

9.27.2.1 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

9.27.2.2 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

9.27.2.3 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

9.27.2.4 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

9.27.2.5 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

9.27.2.6 reject causes

- 226 - Timer T303 is expired
- 227 - CNM MM release is pending

9.27.2.7 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

9.27.2.8 reject causes

- 237 - Service option is not supported

9.27.2.9 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 Sierra Wireless, Inc. all rights reserved

9.28 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 }

9.28.1 Enumeration Type Documentation

9.28.1.1 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

eQCWWAN_ERR_SWIIM_FW_SECBOOT_INVALID_CERT_CHAIN

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

9.28.1.2 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

9.29 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)

9.29.1 Macro Definition Documentation

9.29.1.1 #define LITEQMI_MAX_QOS_FILTERS 25

9.29.1.2 #define LITEQMI_MAX_QOS_FLOW_PER_APN_STATS 10

9.29.1.3 #define LITEQMI_MAX_QOS_FLOWS 8

9.29.2 Function Documentation

9.29.2.1 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

9.29.2.2 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

9.29.2.3 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

9.29.2.4 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

9.29.2.5 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

9.29.2.6 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

9.29.2.7 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

9.29.2.8 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

9.29.2.9 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

9.29.2.10 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

9.29.2.11 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

9.29.2.12 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

9.29.2.13 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

9.29.2.14 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

9.30 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)

9.30.1 Macro Definition Documentation

9.30.1.1 #define __LITEQMI_RMS_H__

9.30.2 Function Documentation

9.30.2.1 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

9.30.2.2 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

9.30.2.3 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

9.30.2.4 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

9.31 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)

9.31.1 Typedef Documentation

9.31.1.1 typedef unpack_result_t unpack_sar_SLQSSetRfSarState_t

9.31.2 Function Documentation

9.31.2.1 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

9.31.2.2 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

9.31.2.3 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

9.31.2.4 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

9.32 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)

9.32.1 Macro Definition Documentation

9.32.1.1 #define MAX_CDMA_ENC_MO_TXT_MSG_SIZE 255

9.32.1.2 #define MAX_MS_TRANSFER_ROUTE_MSG 256

9.32.1.3 #define MAX_MSC_ADDRESS_SIZE 256

9.32.1.4 #define MAX_MSE_TWS_MSG 1254

9.32.1.5 #define MAX_SMS_LIST_SIZE 255

9.32.1.6 #define MAX_SMS_MESSAGE_SIZE 2048

9.32.1.7 #define SMS_CONFIG_LEN 0x05

9.32.1.8 #define SMS_MAX_SMS_ROUTES 0x0A

9.32.1.9 #define SMS_NUM_OF_SET 0xFF

9.32.1.10 #define SMSC_TYPE_LEN 0x03

9.32.2 Enumeration Type Documentation

9.32.2.1 enum eqmiCbkSetStatus

Enumerator

LITEQMI_QMI_CBK_PARAM_RESET

LITEQMI_QMI_CBK_PARAM_SET

LITEQMI_QMI_CBK_PARAM_NOCHANGE

9.32.3 Function Documentation

9.32.3.1 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

9.32.3.2 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

9.32.3.3 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

9.32.3.4 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

9.32.3.5 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

9.32.3.6 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

9.32.3.7 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

9.32.3.8 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

9.32.3.9 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

9.32.3.10 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

9.32.3.11 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

9.32.3.12 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

9.32.3.13 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

9.32.3.14 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

9.32.3.15 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

9.32.3.16 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

9.32.3.17 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

9.32.3.18 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

9.32.3.19 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

9.32.3.20 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

9.32.3.21 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

9.32.3.22 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

9.32.3.23 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

9.32.3.24 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

9.32.3.25 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

9.32.3.26 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

9.32.3.27 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

9.32.3.28 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

9.32.3.29 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

9.32.3.30 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

9.32.3.31 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

9.32.3.32 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

9.32.3.33 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

9.32.3.34 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

9.32.3.35 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

9.32.3.36 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

9.32.3.37 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

9.32.3.38 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

9.32.3.39 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

9.32.3.40 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

9.32.3.41 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

9.32.3.42 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

9.32.3.43 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

9.32.3.44 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

9.32.3.45 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

9.32.3.46 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

9.32.3.47 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

9.32.3.48 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

9.32.3.49 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

9.32.3.50 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

9.32.3.51 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

9.32.3.52 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

9.33 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)

9.33.1 Macro Definition Documentation

9.33.1.1 #define SWIAUDIO_MAX_LEN_IFACE_TABLE 255

9.33.2 Typedef Documentation

9.33.2.1 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioAVCFG_t

9.33.2.2 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioLPBK_t

9.33.2.3 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioNVDef_t

9.33.2.4 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioProfile_t

9.33.2.5 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAudioVolume_t

9.33.2.6 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MAVMute_t

9.33.2.7 typedef unpack_result_t unpack_swiaudio_SLQSSetM2MSpkrGain_t

9.33.3 Function Documentation

9.33.3.1 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

9.33.3.2 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

9.33.3.3 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

9.33.3.4 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

9.33.3.5 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

9.33.3.6 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

9.33.3.7 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

9.33.3.8 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

9.33.3.9 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

9.33.3.10 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

9.33.3.11 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

9.33.3.12 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

9.33.3.13 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

9.33.3.14 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

9.33.3.15 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

9.33.3.16 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

9.33.3.17 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

9.33.3.18 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

9.33.3.19 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

9.33.3.20 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

9.33.3.21 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

9.33.3.22 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

9.34 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

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)

9.34.1 Macro Definition Documentation

9.34.1.1 #define LITEQMI_MAX_GET_SETTINGS_AVMS_APN_STRING_LENGTH 49

9.34.1.2 #define LITEQMI_MAX_GET_SETTINGS_AVMS_PWD_STRING_LENGTH 29

9.34.1.3 #define LITEQMI_MAX_GET_SETTINGS_AVMS_UNAME_STRING_LENGTH 29

9.34.1.4 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.34.1.5 #define MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER 8

9.34.1.6 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH 1024

9.34.1.7 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH 128

9.34.1.8 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH 128

9.34.1.9 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH 200

9.34.1.10 #define MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH 49

9.34.1.11 #define MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS 8

9.34.1.12 #define MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 29

9.34.1.13 #define MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 29

9.34.2 Function Documentation

9.34.2.1 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

9.34.2.2 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

9.34.2.3 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

9.34.2.4 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

9.34.2.5 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

9.34.2.6 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

9.34.2.7 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

9.34.2.8 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

9.34.2.9 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

9.34.2.10 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

9.34.2.11 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

9.34.2.12 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

9.34.2.13 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

9.34.2.14 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

9.34.2.15 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

9.34.2.16 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

9.34.2.17 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

9.34.2.18 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

9.34.2.19 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

9.34.2.20 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

9.34.2.21 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

9.35 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

9.35.1 Detailed Description

SWI data types.

9.35.2 Macro Definition Documentation

9.35.2.1 #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.

9.35.2.2 #define QMI_NO_LTE_FW_SUPPORT 0

9.35.2.3 #define QMI_TLV_PLACEHOLDER 0x8F

9.35.2.4 #define SWI_API

9.35.2.5 #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.

9.35.3 Typedef Documentation

9.35.3.1 typedef int BOOL

9.35.3.2 typedef unsigned char BYTE

9.35.3.3 typedef char CHAR

9.35.3.4 typedef float FLOAT

9.35.3.5 typedef signed int INT32

9.35.3.6 typedef signed char INT8

9.35.3.7 typedef const char* LPCSTR

9.35.3.8 typedef unsigned short qmuint16

9.35.3.9 typedef unsigned long qmulong

9.35.3.10 typedef signed short SHORT

9.35.3.11 typedef unsigned int ULONG

9.35.3.12 typedef unsigned long long ULONGLONG

9.35.3.13 typedef unsigned short USHORT

9.35.3.14 typedef unsigned short WORD

9.36 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

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)

9.36.1 Typedef Documentation

9.36.1.1 typedef unpack_result_t unpack_swidms_SLQSSwiDmsSetUsbNetNum_t

9.36.2 Function Documentation

9.36.2.1 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

9.36.2.2 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

9.36.2.3 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

9.36.2.4 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

9.36.2.5 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

9.36.2.6 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

9.36.2.7 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

9.36.2.8 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

9.36.2.9 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

9.36.2.10 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

9.36.2.11 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

9.36.2.12 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

9.37 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)

9.37.1 Typedef Documentation

9.37.1.1 typedef unpack_result_t unpack_swiloc_SwiLocSetAutoStart_t

9.37.2 Function Documentation

9.37.2.1 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

9.37.2.2 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

9.37.2.3 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

9.37.2.4 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

9.38 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)

9.38.1 Macro Definition Documentation

9.38.1.1 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.38.2 Typedef Documentation

9.38.2.1 typedef unpack_result_t unpack_swioma_SLQSOMADMAlertCallback_t

9.38.2.2 typedef unpack_result_t unpack_swioma_SLQSOMADMCancelSession_t

9.38.2.3 typedef unpack_result_t unpack_swioma_SLQSOMADMSendSelection_t

9.38.2.4 typedef unpack_result_t unpack_swioma_SLQSOMADMSetSettings_t

9.38.3 Function Documentation

9.38.3.1 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

9.38.3.2 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

9.38.3.3 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

9.38.3.4 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

9.38.3.5 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

9.38.3.6 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

9.38.3.7 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

9.38.3.8 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

9.38.3.9 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

9.38.3.10 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

9.38.3.11 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

9.38.3.12 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

9.38.3.13 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

9.38.3.14 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

9.38.3.15 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

9.39 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

Macros

- #define LITEQMI_MAX_SWIOMA_STR_LEN 255

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)

9.39.1 Macro Definition Documentation

9.39.1.1 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.39.2 Typedef Documentation

9.39.2.1 typedef unpack_result_t unpack_swioma_SLQSOMADMCancelSessionExt_t

9.39.2.2 typedef unpack_result_t unpack_swioma_SLQSOMADMSendSelectionExt_t

9.39.2.3 typedef unpack_result_t unpack_swioma_SLQSOMADMSetSettingsExt_t

9.39.2.4 typedef unpack_result_t unpack_swioma_SLQSOMADMStartSessionExt_t

9.39.3 Function Documentation

9.39.3.1 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

9.39.3.2 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

9.39.3.3 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_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

9.39.3.4 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

9.39.3.5 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

9.39.3.6 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

9.39.3.7 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

9.39.3.8 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

9.40 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}}

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)

9.40.1 Macro Definition Documentation

9.40.1.1 #define SWI_UINT256_BIT_OFFSET(*val*) ((val) % SWI_UINT256_BITS_PER_WORD)

9.40.1.2 #define SWI_UINT256_BITS_PER_WORD 16

9.40.1.3 #define SWI_UINT256_INT_VALUE {{0}}

9.40.1.4 #define SWI_UINT256_WORD_COUNT 16

9.40.1.5 #define SWI_UINT256_WORD_OFFSET(*val*) ((val) / SWI_UINT256_BITS_PER_WORD)

9.40.2 Function Documentation

9.40.2.1 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

9.40.2.2 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

9.40.2.3 void swi_uint256_print_mask (swi_uint256_t *mask*)

Print the log mask to syslog. Use only for debugging

Parameters

in	*mask*	Bit mask

9.40.2.4 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

9.41 SWIWWANCMAPI.h File Reference

9.42 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)

9.42.1 Macro Definition Documentation

9.42.1.1 #define MAX_MITIGATION_DEV_ID_LEN 255

9.42.1.2 #define MAX_MITIGATION_DEV_LIST_LEN 255

9.42.1.3 #define TMD_MAX_DEV_LIST 255

9.42.2 Function Documentation

9.42.2.1 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

9.42.2.2 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

9.42.2.3 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

9.42.2.4 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

9.42.2.5 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

9.42.2.6 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

9.42.2.7 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

9.42.2.8 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

9.42.2.9 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

9.43 uim.h File Reference

Data Structures

- struct uim_appStatus
- struct uim_slotInfo
- struct uim_cardStatus
- struct uim_hotSwapStatus
- struct unpack_uim_GetCardStatus_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 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 unpack_uim_SLQSUIMGetSlotsStatus_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

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

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

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 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 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 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)

9.43.1 Macro Definition Documentation

9.43.1.1 #define MAX_DESCRIPTION_LENGTH 255

9.43.1.2 #define MAX_ICCID_LENGTH 255

9.43.1.3 #define MAX_NO_OF_APPLICATIONS 10

9.43.1.4 #define MAX_NO_OF_SLOTS 5

9.43.1.5 #define MAX_SLOTS_STATUS 255

9.43.1.6 #define UIM_MAX_ACTIVE_PERS_FEATURES 12

9.43.1.7 #define UIM_MAX_CONTENT_LENGTH 1024

9.43.1.8 #define UIM_MAX_DESCRIPTION_LENGTH 255

9.43.1.9 #define UIM_MAX_NO_OF_APPLICATIONS 10

9.43.1.10 #define UIM_MAX_NO_OF_SLOTS 5

9.43.1.11 #define UIM_UINT8_MAX_STRING_SZ 255

9.43.2 Typedef Documentation

9.43.2.1 typedef unpack_result_t unpack_uim_SLQSUIMPowerDown_t

9.43.2.2 typedef unpack_result_t unpack_uim_SLQSUIMPowerUp_t

9.43.2.3 typedef unpack_result_t unpack_uim_SLQSUIMRefreshComplete_t

9.43.2.4 typedef unpack_result_t unpack_uim_SLQSUIMRefreshOK_t

9.43.2.5 typedef unpack_result_t unpack_uim_SLQSUIMRefreshRegister_t

9.43.2.6 typedef unpack_result_t unpack_uim_SLQSUIMReset_t

9.43.2.7 typedef unpack_result_t unpack_uim_SLQSUIMSwitchSlot_t

9.43.3 Function Documentation

9.43.3.1 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

9.43.3.2 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

9.43.3.3 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

9.43.3.4 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

9.43.3.5 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

9.43.3.6 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

9.43.3.7 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

9.43.3.8 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

9.43.3.9 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

9.43.3.10 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

9.43.3.11 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

9.43.3.12 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

9.43.3.13 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

9.43.3.14 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

9.43.3.15 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

9.43.3.16 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

9.43.3.17 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

9.43.3.18 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

9.43.3.19 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

9.43.3.20 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

9.43.3.21 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

9.43.3.22 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

9.43.3.23 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

9.43.3.24 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

9.43.3.25 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

9.43.3.26 int unpack_uim_SLQSUIMAuthenticate (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMAuthenticate_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

9.43.3.27 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

9.43.3.28 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

9.43.3.29 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

9.43.3.30 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

9.43.3.31 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

9.43.3.32 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

9.43.3.33 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

9.43.3.34 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

9.43.3.35 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

9.43.3.36 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

9.43.3.37 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

9.43.3.38 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

9.43.3.39 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

9.43.3.40 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

9.43.3.41 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

9.43.3.42 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

9.43.3.43 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

9.44 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

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)

9.44.1 Detailed Description

9.44.2 S1

- 3GPP Mobile Radio Interface Layer 3 Specification : Core Network Protocols; Stage 3 (Release 7)
- 3GPP TS 24.008 V7.0.0 (2005-06)

9.44.3 Macro Definition Documentation

9.44.3.1 #define BARRING_PASSWORD_LENGTH 4

9.44.3.2 #define MAX_VOICE_CALL_NO_LEN 81

9.44.3.3 #define MAX_VOICE_DESCRIPTION_LENGTH 255

9.44.3.4 #define MAXVOICEUSSDLENGTH 182

9.44.3.5 #define VOICE_MAX_NO_OF_CALLS 20

9.44.4 Typedef Documentation

9.44.4.1 typedef unpack_result_t unpack_voice_AnswerUSSD_t

9.44.4.2 typedef unpack_result_t unpack_voice_CancelUSSD_t

9.44.4.3 typedef unpack_result_t unpack_voice_OriginateUSSD_t

9.44.4.4 typedef unpack_result_t unpack_voice_SLQSVoiceALSSelectLine_t

9.44.4.5 typedef unpack_result_t unpack_voice_SLQSVoiceALSSetLineSwitching_t

9.44.4.6 typedef unpack_result_t unpack_voice_SLQSVoiceBindSubscription_t

9.44.4.7 typedef unpack_result_t unpack_voice_SLQSVoiceIndicationRegister_t

9.44.4.8 typedef unpack_result_t unpack_voice_SLQSVoiceOrigUSSDNoWait_t

9.44.4.9 typedef unpack_result_t unpack_voice_SLQSVoiceSetPreferredPrivacy_t

9.44.5 Enumeration Type Documentation

9.44.5.1 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

9.44.6 Function Documentation

9.44.6.1 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

9.44.6.2 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

9.44.6.3 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

9.44.6.4 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

9.44.6.5 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

9.44.6.6 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

9.44.6.7 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

9.44.6.8 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

9.44.6.9 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

9.44.6.10 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

9.44.6.11 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

9.44.6.12 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

9.44.6.13 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

9.44.6.14 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

9.44.6.15 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

9.44.6.16 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

9.44.6.17 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

9.44.6.18 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

9.44.6.19 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

9.44.6.20 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

9.44.6.21 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

9.44.6.22 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

9.44.6.23 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

9.44.6.24 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

9.44.6.25 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

9.44.6.26 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

9.44.6.27 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

9.44.6.28 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

9.44.6.29 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

9.44.6.30 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

9.44.6.31 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

9.44.6.32 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

9.44.6.33 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

9.44.6.34 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

9.44.6.35 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

9.44.6.36 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

9.44.6.37 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

9.44.6.38 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

9.44.6.39 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

9.44.6.40 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

9.44.6.41 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

9.44.6.42 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

9.44.6.43 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

9.44.6.44 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

9.44.6.45 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

9.44.6.46 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

9.44.6.47 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

9.44.6.48 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

9.44.6.49 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

9.44.6.50 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

9.44.6.51 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

9.44.6.52 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

9.44.6.53 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

9.44.6.54 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

9.44.6.55 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

9.44.6.56 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

9.44.6.57 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

9.44.6.58 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

9.44.6.59 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

9.44.6.60 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

9.44.6.61 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

9.44.6.62 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

9.44.6.63 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

9.44.6.64 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

9.44.6.65 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

9.44.6.66 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

9.44.6.67 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

9.44.6.68 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

9.44.6.69 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

9.44.6.70 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

9.44.6.71 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

9.44.6.72 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

9.45 wds.h File Reference

Data Structures

- struct LibPackQosClassID
- struct LibPackTFTIDParams
- struct LibPackGPRSRequestedQoS
- struct LibPackUMTSQoS
- struct LibPackUMTSReqQoSSigInd
- 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 unpack_wds_GetSessionState_t
- struct pack_wds_GetDefaultProfile_t
- struct unpack_wds_GetDefaultProfile_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_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

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

Typedefs

- typedef unpack_result_t unpack_wds_SLQSStopDataSession_t
- typedef unpack_result_t unpack_wds_SLQSSetWdsEventCallback_t
- typedef union unpackWdsProfileParam UnpackQmiProfileInfo
- 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 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

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_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_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_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)

9.45.1 Detailed Description

9.45.2 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

9.45.3 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

9.45.4 Macro Definition Documentation

9.45.4.1 #define BYT_STAT_STAT_MASK 0X000000C0

9.45.4.2 #define IPV6_ADDRESS_ARRAY_SIZE 8

9.45.4.3 #define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24

9.45.4.4 #define PACK_WDS_IPV4 4

9.45.4.5 #define PACK_WDS_IPV6 6

9.45.4.6 #define WDS_DHCP_MAX_NUM_OPTIONS 30

9.45.4.7 #define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /∗ current max size of raw message in SDK process is 2048 ∗/

9.45.4.8 #define WDS_PROFILE_3GPP 0

9.45.4.9 #define WDS_PROFILE_3GPP2 1

9.45.4.10 #define WDS_TFTID_SOURCE_IP_SIZE 8

9.45.5 Typedef Documentation

9.45.5.1 typedef unpack_result_t unpack_wds_DHCPv4ClientLeaseChange_t

9.45.5.2 typedef unpack_result_t unpack_wds_SetAutoconnect_t

9.45.5.3 typedef unpack_result_t unpack_wds_SetDefaultProfile_t

9.45.5.4 typedef unpack_result_t unpack_wds_SetDefaultProfileNum_t

9.45.5.5 typedef unpack_result_t unpack_wds_SetMobileIP_t

9.45.5.6 typedef unpack_result_t unpack_wds_SetMobileIPParameters_t

9.45.5.7 typedef unpack_result_t unpack_wds_SetMuxID_t

9.45.5.8 typedef unpack_result_t unpack_wds_SLQSResetPacketStatics_t

9.45.5.9 typedef unpack_result_t unpack_wds_SLQSSet3GPPConfigItem_t

9.45.5.10 typedef unpack_result_t unpack_wds_SLQSSetWdsEventCallback_t

9.45.5.11 typedef unpack_result_t unpack_wds_SLQSSSetDHCPv4ClientConfig_t

9.45.5.12 typedef unpack_result_t unpack_wds_SLQSSSetLoopback_t

9.45.5.13 typedef unpack_result_t unpack_wds_SLQSStopDataSession_t

9.45.5.14 typedef unpack_result_t unpack_wds_SLQSWdsGoActive_t

9.45.5.15 typedef unpack_result_t unpack_wds_SLQSWdsGoDormant_t

9.45.5.16 typedef unpack_result_t unpack_wds_SLQSWdsSetEventReport_t

9.45.5.17 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

9.45.6 Enumeration Type Documentation

9.45.6.1 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

9.45.7 Function Documentation

9.45.7.1 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

9.45.7.2 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

9.45.7.3 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

9.45.7.4 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

9.45.7.5 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

9.45.7.6 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

9.45.7.7 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

9.45.7.8 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

9.45.7.9 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

9.45.7.10 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

9.45.7.11 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

9.45.7.12 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

9.45.7.13 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

9.45.7.14 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

9.45.7.15 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

9.45.7.16 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

9.45.7.17 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

9.45.7.18 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

9.45.7.19 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

9.45.7.20 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

9.45.7.21 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

9.45.7.22 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

9.45.7.23 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

9.45.7.24 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

9.45.7.25 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

9.45.7.26 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

9.45.7.27 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

9.45.7.28 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

9.45.7.29 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

9.45.7.30 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

9.45.7.31 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

9.45.7.32 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

9.45.7.33 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

9.45.7.34 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

9.45.7.35 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

9.45.7.36 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

9.45.7.37 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

9.45.7.38 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

9.45.7.39 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

9.45.7.40 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

9.45.7.41 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

9.45.7.42 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

9.45.7.43 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

9.45.7.44 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

9.45.7.45 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

9.45.7.46 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

9.45.7.47 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

9.45.7.48 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

9.45.7.49 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

9.45.7.50 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

9.45.7.51 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

9.45.7.52 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

9.45.7.53 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

9.45.7.54 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

9.45.7.55 int unpack_wds_GetDefaultProfileNum (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDefaultProfileNum_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

9.45.7.56 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

9.45.7.57 int unpack_wds_GetLastMobileIPError (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetLastMobileIPError_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

9.45.7.58 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

9.45.7.59 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

9.45.7.60 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

9.45.7.61 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

9.45.7.62 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

9.45.7.63 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

9.45.7.64 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

9.45.7.65 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

9.45.7.66 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

9.45.7.67 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

9.45.7.68 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

9.45.7.69 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

9.45.7.70 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

9.45.7.71 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

9.45.7.72 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

9.45.7.73 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

9.45.7.74 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

9.45.7.75 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

9.45.7.76 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

9.45.7.77 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

9.45.7.78 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

9.45.7.79 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

9.45.7.80 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

9.45.7.81 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

9.45.7.82 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

9.45.7.83 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

9.45.7.84 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

9.45.7.85 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

9.45.7.86 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

9.45.7.87 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

9.45.7.88 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

9.45.7.89 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

9.45.7.90 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

9.45.7.91 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

9.45.7.92 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

9.45.7.93 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

9.45.7.94 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

9.45.7.95 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

9.45.7.96 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

9.45.7.97 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

9.45.7.98 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

9.45.7.99 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, 945
_litefw_FirmwareInfo_, 33
    szCarrier_str, 33
    szCarrierPriversion_str, 34
    szFwversion_str, 34
    szModelid_str, 34
    szPackageid_str, 34
    szSku_str, 34
_litefw_FirmwarePartNo_, 34
    szPartno_str, 34

AAASPI
    unpack_wds_GetMobileIPProfile_t, 935
AAAState
    unpack_wds_GetMobileIPProfile_t, 935
ACT_CODE_MAX_SIZE
    dms.h, 1076
ALS
    voice_getAllCallInformation, 1012
AMSSString
    unpack_dms_GetFirmwareRevision_t, 597
    unpack_dms_GetFirmwareRevisions_t, 598
AMTlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 664
AOATlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 664
AOPTlv
    NASQmiCbkNasSystemSelPrefInd, 314
APNName
    unpack_wds_SLQSGetRuntimeSettings_t, 955
AWMTlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 664
AWOATlv
    unpack_ims_SLQSVoIPCfgCallBack_ind_t, 664
accelTemp
    pack_loc_SLQSLOCInjectSensorData_t, 365
acceleroData
    pack_loc_SLQSLOCInjectSensorData_t, 365
acceleroTimeSrc
    pack_loc_SLQSLOCInjectSensorData_t, 365
AccessMac
    nas_protocolSubtypeElement, 248
accolc
    pack_nas_SetACCOLC_t, 371
accuracy
    pack_pds_SetPDSDefaults_t, 411
ackIndicator
    sMSTransferRouteMTMessageInfo, 543
acqOrdeLen
    nas_acqOrderPref, 144
    nas_AcqOrderPrefTlv, 144
AcqOrderLen
    NASAcqOrderPrefTlv, 298
AcqOrderPref
    NASAcqOrderPrefTlv, 298
acroamsetting
    pack_wds_SetAutoconnect_t, 490
acsetting
    pack_wds_SetAutoconnect_t, 490
actCode
    pack_dms_ActivateAutomatic_t, 327
ActSetCnt
    nas_NetworkStat1x, 226
ActSetPilotPN
    nas_ActPilotPNElement, 145
ActSetPilotPNStrength
    nas_ActPilotPNElement, 145
action
    pack_nas_InitiateDomainAttach_t, 370
activated_ind
    sms_qaQmi3GPP2BroadcastCfgInfo, 532
    sms_qaQmi3GPPBroadcastCfgInfo, 532
activationState
    pack_dms_SetActivationStatusCallback_t, 328
activationStatus
    dms_ActivationStatusTlv, 53
ActivationStatusTlv
    unpack_dms_SetEventReport_ind_t, 613
activeBand
    nas_RfBandInfoExtFormatElements, 257
activeBandClass
    nas_RFBandInfoElements, 256
    nas_RFInfoTlv, 262
    RFBandInfoElements, 519
activeChannel
    nas_RFBandInfoElements, 256
    nas_RfBandInfoExtFormatElements, 257
    nas_RFInfoTlv, 262
    RFBandInfoElements, 519
activeInd
    sms_messageWaitingInfoContent, 530
ActiveStatus
    voice_CLIPResp, 1002
    voice_CLIRResp, 1003
    voice_CNAPResp, 1004
    voice_COLPResp, 1004
    voice_COLRResp, 1005
ActiveTechPref

unpack_nas_GetNetworkPreference_t, 710
ActiveTimer
unpack_dms_PSMCfgChange_ind_t, 610
activeTimer
dms_PSMActiveTimerTlv, 55
ActiveTimerInd
dms_PSMActiveTimerIndTlv, 54
addr
unpack_qos_IPv4Addr_t, 795
unpack_qos_IPv6Addr_t, 796
address
loc_IPv4Info, 133
loc_IPv6Info, 134
loc_urlAddr, 141
unpack_wds_GetMobileIPProfile_t, 935
addressSize
unpack_sms_GetSMSCAddress_t, 820
aid
uim_refreshevent, 568
uim_sessionInformation, 570
uim_UIMSessionInformation, 574
aidLength
appStats, 38
uim_appStatus, 553
uim_refreshevent, 568
uim_sessionInformation, 570
uim_UIMSessionInformation, 574
aidVal
appStats, 38
uim_appStatus, 553
aidingIndicatorMask
loc_sensorDataUsage, 139
airTimerValue
voice_airTimer, 980
alertPitch
voice_signalInfo, 1024
alertingPattern
voice_arrAlertingPattern, 983
AlertingType
voice_arrAlertingType, 984
alertmsg
unpack_omaDmConfigTlv_t, 777
alertmsglength
unpack_omaDmConfigTlv_t, 777
allCallsAlphaIDInfoArr
voice_arrAlphaID, 984
AllCallsUUSInfo
voice_arrUUSInfo, 990
alphaDcs
voice_alphaIDInfo, 982
AlphaID
cat_AlPhaIdentifierTlv, 46
AlphaIDInfo
voice_allCallsAlphaIDInfo, 980
alphaIDLen
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
AlphaIDLength
cat_AlPhaIdentifierTlv, 46
alphaLen
voice_alphaIDInfo, 982
alphaText
voice_alphaIDInfo, 982
altSrcInfo_t, 34
coverage, 35
linkage, 35
source, 35
Altitude
unpack_pds_SLQSGetGPSStateInfo_t, 793
altitudeAssumed
unpack_loc_GnssSvInfo_Ind_t, 686
altitudeSrcInfo
pack_loc_SLQSLOCInjectPosition_t, 363
altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 363
altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 363
ambr_dl
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
ambr_dl_ext
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
ambr_dl_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
ambr_ul
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
ambr_ul_ext
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
ambr_ul_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
amrMode
ims_AMRModeInfo, 72
amrOctAlgn
ims_AMROctAlgnInfo, 72
amrWBEnable
ims_EnabAMRWBInfo, 74
amrWBMode
ims_AMRWBModeInfo, 73
amrWBOctAlgn
ims_AMRWBOctAlgnInfo, 74
amssSize
unpack_dms_GetFirmwareRevision_t, 597
unpack_dms_GetFirmwareRevisions_t, 598
apdoxypages.c, 1053
apnId
pack_qos_SLQSQosSwiReadApnExtraParams_t, 421
pack_qos_SLQSQosSwiReadDataStats_t, 421
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
unpack_qos_SLQSQosSwiReadDataStats_t, 803

apnName
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 969
apnname
unpack_wds_GetDefaultProfile_t, 930
apnsize
unpack_wds_GetDefaultProfile_t, 930
appNameLength
loc_LocApplicationInfo, 135
appProviderLength
loc_LocApplicationInfo, 135
appState
appStats, 38
uim_appStatus, 553
appStats, 35
aidLength, 38
aidVal, 38
appState, 38
appType, 38
persoFeature, 38
persoRetries, 38
persoState, 38
persoUnblockRetries, 38
pin1Retries, 38
pin1State, 38
pin2Retries, 38
pin2State, 38
puk1Retries, 38
puk2Retries, 38
univPin, 38
AppStatus
slotInf, 524
uim_slotInfo, 573
appType
appStats, 38
uim_appStatus, 553
appVersionLength
loc_LocApplicationInfo, 135
appVersionValid
loc_LocApplicationInfo, 135
Application
unpack_nas_GetCDMANetworkParameters_t, 705
appversion_str
unpack_dms_GetFirmwareInfo_t, 595
arfcn
nas_GERANInfo, 178
nas_gsmCellInfo, 180
arrCallInfomation
unpack_voice_allCallStatusCallback_ind_t, 880
arrfileInfo
uim_refreshevent, 568
uim_registerRefresh, 569
audio.h, 1053
pack_audio_SLQSGetAudioPathConfig, 1054
pack_audio_SLQSGetAudioProfile, 1054
pack_audio_SLQSGetAudioVolTLBConfig, 1055
pack_audio_SLQSSetAudioPathConfig, 1055
pack_audio_SLQSSetAudioProfile, 1056
pack_audio_SLQSSetAudioVolTLBConfig, 1056
unpack_audio_SLQSGetAudioPathConfig, 1057
unpack_audio_SLQSGetAudioProfile, 1057
unpack_audio_SLQSGetAudioVolTLBConfig, 1057
unpack_audio_SLQSSetAudioPathConfig, 1058
unpack_audio_SLQSSetAudioProfile, 1058
unpack_audio_SLQSSetAudioVolTLBConfig, 1058
audio_RXAGCList, 38
pRXAIG, 39
pRXComprSlope, 39
pRXComprThres, 39
pRXExpSlope, 39
pRXExpThres, 39
pRXStaticGain, 39
audio_RXAVCList, 39
pAVRXAVCHeadroom, 40
pAVRXAVCSens, 40
audio_RXPCMIIRFltr, 40
pFlag, 41
pStage0Val, 41
pStage1Val, 41
pStage2Val, 41
pStage3Val, 41
pStage4Val, 41
pStageCnt, 41
audio_TXAGCList, 41
pTXAIG, 42
pTXComprSlope, 42
pTXComprThres, 42
pTXExpSlope, 42
pTXExpThres, 42
pTXStaticGain, 42
audio_TXPCMIIRFltr, 42
pFlag, 44
pStage0Val, 44
pStage1Val, 44
pStage2Val, 44
pStage3Val, 44
pStage4Val, 44
pStageCnt, 44
auth
unpack_wds_GetDefaultProfile_t, 930
authData
pack_uim_SLQSUIMAuthenticate_t, 458
AuthProt
nas_protocolSubtypeElement, 248
Authentication
unpack_wds_SLQSGetRuntimeSettings_t, 955
authentication
pack_wds_SetDefaultProfile_t, 491
AutoConnect
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 442
unpack_swiavms_SLQSAVMSGetSettings_t, 845
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
AutoReboot
pack_swiavms_SLQSAVMSSetSettings_t, 441

unpack_swiavms_SLQSAVMSGetSettings_t, 845
Autosdm
unpack_swioma_SLQSOMADMGetSettings_t, 862
avgPeriod
nas_LTESigRptCfg, 214
nas_LTESigRptConfig, 215
azimuth
loc_satelliteInfo, 138

bAPNLength
PackSwiAvmsSetSettingsAPNInfo, 513
PackSwiAVMSSettingsAPNInfo, 514
bAuto
pack_pds_SetPortAutomaticTracking_t, 412
pack_pds_SetServiceAutomaticTracking_t, 412
bBinaryType
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
bEnable
pack_dms_UIMSetPINProtection_t, 343
pack_nas_SLQSSetSignalStrengthsCallback_t, 392
bEnabled
pack_pds_SetXTRAAutomaticDownload_t, 413
bForceDownload
pack_fms_SetImagesPreference_t, 346
bICCID
slot_t, 522
bICCIDLength
slot_t, 522
bLogicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 464
slot_t, 522
bNotification
UnpackSwiAvmsEventReportNotification, 976
bNumberOfPhySlots
unpack_uim_SetUimSlotStatusChangeCallback_-ind_t, 871
BOOL
SwiDataTypes.h, 1375
BPTlv
NASQmiCbkNasSystemSelPrefInd, 314
bPWDLength
PackSwiAvmsSetSettingsAPNInfo, 513
PackSwiAVMSSettingsAPNInfo, 514
bPackageID
UnpackSwiAvmsEventReportPackageID, 977
bRegStatus
UnpackSwiAvmsEventReportRegStatus, 977
bResetStatistics
rmTrasnferStaticsReq, 520
bSerity
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
bState
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
UnpackSwiAvmsEventReportConfig, 972
bType
UnpackSwiAvmsEventReportDataSessionStatus, 974
UnpackSwiAvmsEventReportSessionType, 978
bUnameLength
PackSwiAvmsSetSettingsAPNInfo, 513
PackSwiAVMSSettingsAPNInfo, 515
bUserInputRequest
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
UnpackSwiAvmsEventReportConfig, 972
UnpackSwiAvmsEventReportConnectionRequest, 973
BYT_STAT_STAT_MASK
wds.h, 1462
BYTE
SwiDataTypes.h, 1375
band
nas_LTEInfo, 200
NASPhyCaAggScellArray, 308
band1900
nas_gsmCellInfo, 180
band_pref
NASBandPreferenceTlv, 298
BandCapability
unpack_dms_GetBandCapability_t, 584
bandCapability
unpack_dms_SLQSGetBandCapability_t, 627
BandPref
nas_BandPrefTlv, 149
bandPref
pack_nas_SLQSSetBandPreference_t, 391
bandwidth
nas_LTEInfo, 200
nas_RfBandwidthInfoElements, 259
baseId
nas_CDMAInfo, 152
nas_CDMASysInfo, 156
baseLat
nas_CDMAInfo, 152
nas_CDMASysInfo, 156
baseLong
nas_CDMAInfo, 152
nas_CDMASysInfo, 156
BasestationID
unpack_nas_SLQSGetServingSystem_t, 732
BasestationLatitude
unpack_nas_SLQSGetServingSystem_t, 732
BasestationLongitude
unpack_nas_SLQSGetServingSystem_t, 732
batchPerSec
loc_accelAcceptReady, 124
loc_accelTempAcceptReady, 125
loc_gyroAcceptReady, 131
loc_gyroTempAcceptReady, 131
BearerID
unpack_qos_QosFlowInfo_t, 798
bearerID

unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 958
bearerId
unpack_QosFlowStat_t, 816
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 969
bits_129_192
nas_BandPrefInfoTlv, 147
nas_lteBandPrefExt, 194
nas_nr5gBandPref, 230
bits_193_256
nas_BandPrefInfoTlv, 147
nas_lteBandPrefExt, 194
nas_nr5gBandPref, 230
bits_1_64
nas_BandPrefInfoTlv, 147
nas_lteBandPrefExt, 194
nas_nr5gBandPref, 230
bits_65_128
nas_BandPrefInfoTlv, 147
nas_lteBandPrefExt, 194
nas_nr5gBandPref, 230
bootSize
unpack_dms_GetFirmwareRevisions_t, 598
BootString
unpack_dms_GetFirmwareRevisions_t, 598
bootversion_str
unpack_dms_GetFirmwareInfo_t, 595
Broadcast
unpack_nas_GetCDMANetworkParameters_t, 705
broadcastActivate
pack_sms_SLQSSetSmsBroadcastActivation_t, 430
broadcastConfig
sms_qaQmi3GPPBroadcastCfgInfo, 532
bsInfoValid
nas_CDMASysInfo, 156
bsPRev
nas_CDMASysInfo, 156
bsPRevValid
nas_CDMASysInfo, 156
bsic
nas_GERANInfo, 178
bsicId
nas_gsmCellInfo, 180
bucketSz
unpack_qos_tokenBucket_t, 815
buildID
FMSImageIdElement, 68
image_info_t, 71
buildIDLen
image_info_t, 71
buildIDLength
FMSImageIdElement, 68
buildId
FMSImageElement, 67
buildIdLength
FMSImageElement, 67
BurstDTMFInfo
pack_voice_SLQSVoiceBurstDTMF_t, 470
ByteLoopbackMode
unpack_wds_SLQSSGetLoopback_t, 966
ByteLoopbackMultiplier
unpack_wds_SLQSSGetLoopback_t, 966

CCETlv
unpack_cat_SetCatEventCallback_ind_t, 580
CDMA_P_Rev
unpack_nas_SLQSGetServingSystem_t, 732
CDMABroadcastConfig
sms_qaQmi3GPP2BroadcastCfgInfo, 532
CDMAECIOThreshListLen
nas_CDMAECIOThresh, 151
CDMARSSIThreshListLen
nas_CDMARSSIThresh, 153
CDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 756
CDMASystemInfoExt
unpack_nas_SLQSGetServingSystem_t, 732
CHAR
SwiDataTypes.h, 1375
CK_MAX_SIZE
dms.h, 1076
CQIValueCW0
unpack_nas_SLQSSwiGetLteCQI_t, 770
CQIValueCW1
unpack_nas_SLQSSwiGetLteCQI_t, 770
CRTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 944
CSDomain
unpack_nas_GetServingNetwork_t, 713
CSTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 944
CUGIndex
voice_CUGInfo, 1007
CallBarStatus
unpack_nas_SLQSGetServingSystem_t, 732
callDuration
unpack_wds_GetSessionDuration_t, 940
CallEndReason
wds_LastMdmCallEndRsnTlv, 1041
callEndReason
unpack_wds_SLQSGetDUNCallInfo_t, 951
voice_arrCallEndReason, 985
CallFWExtInfo
voice_getCallFWExtInfo, 1014
CallFWInfo
voice_getCallFWInfo, 1014
callID
pack_voice_SLQSVoiceGetCallInfo_t, 474
pack_voice_SLQSVoiceStopContDTMF_t, 485
unpack_voice_OTASPStatusCallback_ind_t, 883
unpack_voice_SLQSVoiceStopContDTMF_t, 915
unpack_voice_SUPSNotificationCallback_ind_t, 920

unpack_voice_VoiceInfoRecCallback_ind_t, 923
unpack_voice_voicePrivacyChangeCallback_ind_t, 924
voice_allCallsAlphaIDInfo, 980
voice_allCallsDiagInfo, 981
voice_allCallsUUSInfo, 981
voice_arrAlertingPattern, 983
voice_arrAlertingType, 984
voice_arrCallEndReason, 985
voice_arrSvcOption, 989
voice_callInfo, 999
voice_DTMFInfo, 1009
voice_getAllCallRmtPtyName, 1013
voice_getAllCallRmtPtyNum, 1013
voice_peerNumberInfo, 1017
callNumber
pack_voice_SLQSVoiceDialCall_t, 471
callState
voice_callInfo, 999
callType
voice_callInfo, 999
CalledPartyNum
voice_arrCalledPartyNum, 985
callerID
voice_callerIDInfo, 994
voice_connectNumInfo, 1006
callerIDLen
voice_callerIDInfo, 994
voice_connectNumInfo, 1006
callerName
voice_remotePartyName, 1022
Callinfo
voice_getAllCallInformation, 1012
campedCiotLteOpMode
nas_LteCiotOpModeTlv, 196
cardState
slotInf, 524
uim_slotInfo, 573
carrier
unpack_dms_SLQSSwiGetFirmwareCurr_t, 633
carrier_str
unpack_dms_GetFirmwareInfo_t, 595
CarrierImage_t, 44
m_FwBuildId, 45
m_FwImageId, 45
m_PriBuildId, 45
m_PriImageId, 45
m_nCarrierId, 45
m_nFolderId, 45
m_nStorage, 45
cat.h, 1059
pack_cat_CATSendEnvelopeCommand, 1060
pack_cat_CATSendTerminalResponse, 1060
pack_cat_SetCATEventCallback, 1060
unpack_cat_CATSendEnvelopeCommand, 1061
unpack_cat_CATSendEnvelopeCommand_t, 1060
unpack_cat_CATSendTerminalResponse, 1061
unpack_cat_CATSendTerminalResponse_t, 1060
unpack_cat_SetCATEventCallback, 1062
unpack_cat_SetCatEventCallback_ind, 1062
cat_AlPhaIdentifierTlv, 45
AlphaID, 46
AlphaIDLength, 46
ReferenceID, 46
cat_EndProactiveSessionTlv, 47
EndProactiveSession, 47
cat_EventIDDataTlv, 48
Data, 48
DataLength, 48
ReferenceID, 48
cat_EventListTlv, 48
SetupEventList, 48
cat_RefreshTlv, 49
RefreshMode, 49
RefreshStage, 49
cat_commonEventTlv, 46
CatEvent, 46
EventID, 46
EventLength, 46
TlvPresent, 46
cat_currentCatEvent, 46
CatAlphaIdtfr, 47
CatEndPS, 47
CatEvIDData, 47
CatEventLst, 47
CatRefresh, 47
CatAlphaIdtfr
cat_currentCatEvent, 47
CatEndPS
cat_currentCatEvent, 47
CatEvIDData
cat_currentCatEvent, 47
CatEvent
cat_commonEventTlv, 46
CatEventLst
cat_currentCatEvent, 47
CatRefresh
cat_currentCatEvent, 47
causeCode
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t, 837
ccsSupported
nas_CDMASysInfo, 156
ccsSupportedValid
nas_CDMASysInfo, 156
cdmaSSInfo, 49
ecio, 49
rssi, 49
cdmaSysIdValid
nas_CDMASysInfo, 156
cell_resel_priority
nas_infoInterFreq, 192
cellBroadcastCap
nas_AddSysInfo, 146
CellID
unpack_nas_SLQSGetServingSystem_t, 732

cellID
nas_GERANInfo, 178
nas_QmisNasSlqsNasPCICellInfo, 252
nas_UMTSInfo, 288
cellId
nas_GSMSysInfo, 184
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 234
nas_WCDMASysInfo, 296
cellIdValid
nas_gsmCellInfo, 180
nas_GSMSysInfo, 184
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 234
nas_WCDMASysInfo, 297
cellInterFreqParams
nas_infoInterFreq, 192
cellIsTDD
nas_umtsLTENbrCell, 290
CellParams
nas_LTEInfoIntrafreq, 203
cellReselPriority
nas_lteGsmCellInfo, 198
nas_LTEInfoIntrafreq, 203
nas_lteWcdmaCellInfo, 220
cells_len
nas_infoInterFreq, 192
nas_lteGsmCellInfo, 198
cellsLen
nas_LTEInfoIntrafreq, 203
nas_lteWcdmaCellInfo, 220
CfgValue
pack_swidms_SLQSSwiDmsSetUsbComp_t, 445
swidms_intfaceCfgTlv, 546
chaddr
wds_DHCPv4HWConfig, 1035
wdsDhcpv4HwConfig, 1049
chaddrLen
wds_DHCPv4HWConfig, 1035
wdsDhcpv4HwConfig, 1049
changePIN
pack_uim_ChangePin_t, 455
ChannelRate
wds_ChannelRateTlv, 1028
channelRate
unpack_wds_SLQSGetDUNCallInfo_t, 951
Chipset
nas_DeviceConfigDetail, 169
CiotAOPTlv
NASQmiCbkNasSystemSelPrefInd, 314
CiotAcqOrderLen
NASCiotAcqOrderPrefTlv, 299
ciotAcqOrderLen
nas_ciotAcqOrderPref, 159
nas_CiotAcqOrderPrefTlv, 160
CiotAcqOrderPref
NASCiotAcqOrderPrefTlv, 299
CiotLteOpModePref
NASCiotLteOpModePrefTlv, 299
ciotLteOpModePref
nas_CiotLteOpModePrefTlv, 160
CiotOpMPTlv
NASQmiCbkNasSystemSelPrefInd, 314
ckLen
uim_depersonalizationInformation, 559
ckVal
uim_depersonalizationInformation, 559
clear
pack_dms_SLQSSwiGetCrashInfo_t, 336
cmdID
pack_cat_CATSendEnvelopeCommand_t, 325
codingScheme
nas_PLMNNetworkNameData, 246
voice_remotePartyName, 1022
common.h
eAUDIO, 1066
eCAT, 1066
eCTL, 1065
eDMS, 1065
eIMS, 1066
eIMSA, 1066
eIND, 1066
eLOC, 1066
eLOG_DEBUG, 1065
eLOG_FATAL, 1065
eLOG_INFO, 1065
eLOG_WARN, 1065
eNAS, 1065
ePDS, 1066
eQOS, 1066
eREQ, 1066
eRMS, 1066
eRSP, 1066
eSAR, 1066
eSMS, 1066
eSWIAUDIO, 1066
eSWIDMS, 1066
eSWILOC, 1066
eSWIM2MCMD, 1066
eSWIM2MCMD_AVC2, 1066
eSWIOMA, 1066
eSWIOMAEXT, 1066
eTIMEOUT_10_S, 1066
eTIMEOUT_20_S, 1066
eTIMEOUT_2_S, 1066
eTIMEOUT_300_S, 1066
eTIMEOUT_30_S, 1066
eTIMEOUT_5_S, 1066
eTIMEOUT_60_S, 1066
eTIMEOUT_8_S, 1066
eTIMEOUT_DEFAULT, 1066
eTMD, 1066
eUIM, 1066
eVOICE, 1066
eWDS, 1065
common.h, 1062

eLOG_LEVEL, 1065
eQMI_SVC, 1065
eTimeout, 1066
fill_pack_ctx, 1067
fill_sdu_hdr, 1067
get_version, 1067
glog, 1068
gloglvl, 1068
helper_get_error_code, 1067
helper_get_error_reason, 1067
helper_get_req_str, 1067
helper_get_resp_ctx, 1067
helper_get_xid, 1067
helper_isBootLoader_DebugEnabled, 1067
helper_set_log_func, 1068
helper_set_log_lvl, 1068
liteqmi_GetVersion, 1068
liteqmi_helper_decode7bitAsciiEncString, 1068
liteqmi_log, 1068
logger, 1065
MINREQBKLEN, 1064
MSGID_AND_LEN, 1064
MSGID_DONT_CARE, 1064
msgtype, 1066
SDU_HDR_LEN, 1065
UNUSEDPARAM, 1065
unpack_result_code_only, 1068
commonInfo
unpack_nas_SLQSNasSwiModemStatus_t, 764
ConcSvcInfo
unpack_nas_SLQSGetServingSystem_t, 733
confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 367
conn_status
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 958
ConnectedPartyNum
voice_arrConnectPartyNum, 987
connectionStatus, 50
MDMCallDuration, 50
MDMConnStatus, 50
unpack_wds_GetSessionState_t, 940
unpack_wds_SLQSGetDUNCallInfo_t, 951
content
uim_authenticateResult, 554
uim_readResult, 566
contentLen
uim_authenticateResult, 554
uim_readResult, 566
context
uim_authenticationData, 555
contextId
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 510
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
contextType
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 510
ControlMac
nas_protocolSubtypeElement, 248
count0
nas_RankIndicatorTlv, 254
count1
nas_RankIndicatorTlv, 254
countryInitials
nas_PLMNNetworkNameData, 246
coverage
altSrcInfo_t, 35
cphy_ca_dl_bandwidth
NASPhyCaAggScellArray, 308
cphy_scell_info_list_len
NASPhyCaAggScellArray, 308
cpich_ecno
nas_wcdmaCellInfo, 292
cpich_rscp
nas_wcdmaCellInfo, 292
cradleMountConfigStatus
unpack_loc_CradleMountCallback_Ind_t, 679
crashAction
pack_dms_SetCrashAction_t, 329
crashData
crashInformation, 52
crashId
crashInformation, 52
crashInfo
crashInfoParams, 51
crashInfoParam
unpack_dms_SLQSSwiGetCrashInfo_t, 631
crashInfoParams, 50
crashInfo, 51
crashStatus, 51
crashInformation, 51
crashData, 52
crashId, 52
crashString, 52
crashStrlen, 52
gcdumpString, 52
gcdumpStrlen, 52
numCrashes, 52
crashStatus
crashInfoParams, 51
crashString
crashInformation, 52
crashStrlen
crashInformation, 52
csAttachState
nas_servSystem, 272
NASServingSystemInfo, 317
csBarStatus
nas_CallBarringSysInfo, 149
nas_callBarStatus, 150
cscfPortName
ims_CSCFPortNameInfo, 74
csgId

nas_CsgId, 163
cur_carr_name
unpack_dms_GetFirmwareInfo_t, 596
cur_carr_rev
unpack_dms_GetFirmwareInfo_t, 596
curDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 948
curProfile
pack_wds_SLQSModifyProfile_t, 502
UnPackGetProfileSettingOut, 970
CurrChanRxRate
dunchannelRate, 65
wds_channelRate, 1028
CurrChanTxRate
dunchannelRate, 65
wds_channelRate, 1028
currDBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 963
currNWInfo
unpack_wds_SLQSSetWdsEventCallback_ind_t, 963
currNetworkInfo, 52
NetworkType, 52
RATMask, 52
SOMask, 52
unpack_wds_SLQSGetCurrDataSystemStat_t, 946
current_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 948
current_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 948
CurrentCfgType
swidms_intfaceCfgTlv, 546
currentChannelRXRate
unpack_wds_GetConnectionRate_t, 927
currentChannelTXRate
unpack_wds_GetConnectionRate_t, 927
currentDataBearer
pack_wds_SLQSSetWdsEventCallback_t, 505
currentNetwork
qmiWDSDataBearerTechnology, 517
CurrentPLMN
unpack_nas_SLQSGetServingSystem_t, 733
CurrentmitigationLvl
unpack_tmd_SLQSTmdGetMitigationLvl_t, 865
cust_attr
DMScustSettingInfo, 62
cust_id
DMScustSettingInfo, 62
DMSgetCustomInput, 64
pack_dms_GetCustFeaturesV2_t, 327
pack_dms_SetCustFeaturesV2_t, 332
cust_value
DMScustSettingInfo, 62
pack_dms_SetCustFeaturesV2_t, 332
custSetting
DMScustSettingList, 63
CustomSCP
unpack_nas_GetCDMANetworkParameters_t, 705
CwtMute
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 839
unpack_swiaudio_SLQSGetM2MAVMute_t, 841
cycleLength
nas_EdrxCycleLength, 172

DBTTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 944
dBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 963
dBTechExtRatValue
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
dBTechExtSoMask
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
dBTechnology
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
dBtechExtAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 963
dBtechnologyExt
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
DHCPRelayEnabled
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
DHCPv4LeaseOptTlv
unpack_wds_DHCPv4ClientLease_ind_t, 925
DHCPv4LeaseStateTlv
unpack_wds_DHCPv4ClientLease_ind_t, 925
dLatitude
unpack_pds_SetEventReport_Ind_t, 787
dLongitude
unpack_pds_SetEventReport_Ind_t, 787
DMS_IMGDETAILS_LEN
dms.h, 1076
DMS_PM_FACTORY
dms.h, 1076
DMS_PM_LOW
dms.h, 1076
DMS_PM_OFFLINE
dms.h, 1076
DMS_PM_ONLINE
dms.h, 1076
DMS_PM_RESET
dms.h, 1076
DMS_PM_SHUT_DOWN
dms.h, 1076
DMS_VALID_FSN_LEN

dms.h, 1077
DMScustSettingInfo, 62
cust_attr, 62
cust_id, 62
cust_value, 62
id_length, 62
value_length, 63
DMScustSettingList, 63
custSetting, 63
list_type, 63
num_instances, 63
DMSgetCustomFeatureV2, 63
pCustSettingInfo, 64
pCustSettingList, 64
pGetCustomInput, 64
DMSgetCustomInput, 64
cust_id, 64
list_type, 64
DRCCover
nas_DRCParams, 170
DRCValue
nas_DRCParams, 170
DSTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 944
DTMFEvent
voice_DTMFInfo, 1009
DTMFInformation
unpack_voice_DTMFEventCallback_ind_t, 881
DTMFInterdigitInterval
voice_DTMFLengths, 1010
DTMFPulseWidth
voice_DTMFLengths, 1010
DTMFdigit
pack_voice_SLQSVoiceStartContDTMF_t, 485
DTMInd
unpack_nas_SLQSGetServingSystem_t, 733
Data
cat_EventIDDataTlv, 48
data
sMSCAddressInfo, 538
sMSEtwsMessageInfo, 539
sMSTransferRouteMTMessageInfo, 543
uim_authenticationData, 555
data_buf
NASOTAMessageTlv, 305
data_len
NASOTAMessageTlv, 306
dataBearer
pack_wds_SLQSSetWdsEventCallback_t, 505
dataBearerMask
unpack_wds_SLQSGetDataBearerTechnology_t, 948
DataBearerTech
wds_DataBearTechTlv, 1031
dataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 951
dataBearerTechExt
pack_wds_SLQSSetWdsEventCallback_t, 505
dataCapabilities
nas_dataSrvCapabilities, 165
dataCapabilitiesLen
nas_dataSrvCapabilities, 165
DataCaps
unpack_nas_GetServingNetwork_t, 713
unpack_nas_GetServingNetworkCapabilities_t, 715
dataCaps
unpack_nas_SetDataCapabilitiesCallback_ind_t, 718
DataCapsLen
unpack_nas_GetServingNetwork_t, 713
unpack_nas_GetServingNetworkCapabilities_t, 715
dataCapsSize
unpack_nas_SetDataCapabilitiesCallback_ind_t, 718
dataLen
pack_cat_CATSendEnvelopeCommand_t, 325
pack_cat_CATSendTerminalResponse_t, 326
uim_authenticationData, 555
DataLength
cat_EventIDDataTlv, 48
DataRate
unpack_qos_swiQosFlow_t, 813
dataRateMax
unpack_qos_dataRate_t, 795
dataServiceCaCapability
unpack_dms_GetDeviceCapabilities_t, 590
DataServiceCapability
unpack_dms_GetDeviceCap_t, 589
DataSrvCapabilities
unpack_nas_SLQSGetServingSystem_t, 733
dataSysStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 963
dataSystemStatus
pack_wds_SLQSSetWdsEventCallback_t, 505
Date
unpack_swioma_SLQSOMADMGetSessionInfo_t, 860
DateLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 860
day
nas_timeInfo, 285
nas_UniversalTime, 291
dayLtSavingAdj
nas_timeInfo, 285
dayOfWeek
nas_timeInfo, 285
nas_UniversalTime, 291
daylightSavings
nas_qaQmi3Gpp2TimeZone, 249
dedicatedBand
nas_RfDedicatedBandInfoElements, 261

defaultPDNEnabled
unpack_wds_SLQSGet3GPPConfigItem_t, 945
DefaultRoamInd
unpack_nas_SLQSGetServingSystem_t, 733
delayClass
LibPackGPRSRequestedQoS, 93
wds_GPRSQoS, 1039
deliveryErrSDU
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1048
depersonilisationInfo
pack_uim_SLQSUIMDepersonalization_t, 459
description
unpack_omaDmFotaTlv_t, 779
descriptionlength
unpack_omaDmFotaTlv_t, 779
Desription
nas_QmiNas3GppNetworkInfo, 250
destPortRangeEnd
LibPackTFTIDParams, 119
destPortRangeStart
LibPackTFTIDParams, 119
DetailedSvcInfo
unpack_nas_SLQSGetServingSystem_t, 733
DevCrashState
unpack_dms_GetCrashAction_t, 584
Device
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 435
DeviceError
qmTlvResult, 517
deviceID
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t, 866
deviceIdLen
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t, 866
DeviceResult
qmTlvResult, 517
DiagInfo
voice_allCallsDiagInfo, 981
voice_arrDiagInfo, 987
diagInfoLen
voice_diagInfo, 1008
diagnosticInfo
voice_diagInfo, 1008
digitBuff
voice_DTMFInfo, 1009
digitCnt
voice_burstDTMFInfo, 991
voice_DTMFInfo, 1009
dirNum
nas_dirNum, 169
dirNumLen
nas_dirNum, 169
direction
voice_callInfo, 999
DisableIMSI
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
dispType
voice_extDispRecInfo, 1011
displayCondition
nas_serviceProviderName, 270
dl_bw_value
nas_PhyCaAggPcellInfo, 238
nas_PhyCaAggScellDlBw, 238
nas_PhyCaAggScellInfo, 242
NASPhyCaAggPcellInfo, 306
NASPhyCaAggScellDlBw, 308
NASPhyCaAggScellInfo, 310
dms.h, 1068
ACT_CODE_MAX_SIZE, 1076
CK_MAX_SIZE, 1076
DMS_IMGDETAILS_LEN, 1076
DMS_PM_FACTORY, 1076
DMS_PM_LOW, 1076
DMS_PM_OFFLINE, 1076
DMS_PM_ONLINE, 1076
DMS_PM_RESET, 1076
DMS_PM_SHUT_DOWN, 1076
DMS_VALID_FSN_LEN, 1077
ERI_DATA_MAX_SIZE, 1077
MAX_BUILD_ID_LEN, 1077
MEID_MAX_SIZE, 1077
pack_dms_ActivateAutomatic, 1077
pack_dms_GetActivationState, 1078
pack_dms_GetBandCapability, 1078
pack_dms_GetCrashAction, 1078
pack_dms_GetCustFeature, 1079
pack_dms_GetCustFeaturesV2, 1079
pack_dms_GetDeviceCap, 1079
pack_dms_GetDeviceCapabilities, 1080
pack_dms_GetDeviceHardwareRev, 1080
pack_dms_GetDeviceMfr, 1081
pack_dms_GetDeviceSerialNumbers, 1081
pack_dms_GetFSN, 1082
pack_dms_GetFirmwareInfo, 1081
pack_dms_GetFirmwareRevision, 1082
pack_dms_GetFirmwareRevisions, 1082
pack_dms_GetHardwareRevision, 1083
pack_dms_GetIMSI, 1083
pack_dms_GetManufacturer, 1083
pack_dms_GetModelID, 1084
pack_dms_GetNetworkTime, 1084
pack_dms_GetOfflineReason, 1084
pack_dms_GetPRLVersion, 1085
pack_dms_GetPower, 1085
pack_dms_GetSerialNumbers, 1085
pack_dms_GetUSBComp, 1086
pack_dms_GetVoiceNumber, 1086
pack_dms_ResetToFactoryDefaults, 1086
pack_dms_SLQSDmsSwiGetPCInfo, 1090
pack_dms_SLQSDmsSwiGetResetInfo, 1091
pack_dms_SLQSDmsSwiGetUimSelection, 1091
pack_dms_SLQSDmsSwiIndicationRegister, 1091
pack_dms_SLQSGetBandCapability, 1092

pack_dms_SLQSGetERIFile, 1092
pack_dms_SLQSGetPowerSaveModeConfig, 1092
pack_dms_SLQSSetPowerSaveModeConfig, 1093
pack_dms_SLQSSwiClearDyingGaspStatistics, 1093
pack_dms_SLQSSwiGetCrashInfo, 1094
pack_dms_SLQSSwiGetDyingGaspCfg, 1094
pack_dms_SLQSSwiGetDyingGaspStatistics, 1094
pack_dms_SLQSSwiGetFirmwareCurr, 1095
pack_dms_SLQSSwiGetFwUpdateStatus, 1095
pack_dms_SLQSSwiGetHostDevInfo, 1095
pack_dms_SLQSSwiGetOSInfo, 1096
pack_dms_SLQSSwiGetSerialNoExt, 1096
pack_dms_SLQSSwiSetDyingGaspCfg, 1096
pack_dms_SLQSSwiSetHostDevInfo, 1097
pack_dms_SLQSSwiSetOSInfo, 1097
pack_dms_SLQSUIMGetState, 1097
pack_dms_SetActivationStatusCallback, 1087
pack_dms_SetCrashAction, 1087
pack_dms_SetCustFeature, 1088
pack_dms_SetCustFeaturesV2, 1088
pack_dms_SetEventReport, 1088
pack_dms_SetFirmwarePreference, 1089
pack_dms_SetIndicationRegister, 1089
pack_dms_SetPower, 1089
pack_dms_SetUSBComp, 1090
pack_dms_SwiSetEventReport, 1098
pack_dms_SwiUimSelect, 1098
pack_dms_UIMChangePIN, 1098
pack_dms_UIMGetControlKeyStatus, 1099
pack_dms_UIMGetICCID, 1099
pack_dms_UIMGetPINStatus, 1100
pack_dms_UIMSetControlKeyProtection, 1100
pack_dms_UIMSetPINProtection, 1100
pack_dms_UIMUnblockControlKey, 1101
pack_dms_UIMUnblockPIN, 1101
pack_dms_UIMVerifyPIN, 1101
pack_dms_ValidateSPC, 1102
SPC_SIZE, 1077
UNIQUE_ID_LEN, 1077
unpack_dms_ActivateAutomatic, 1102
unpack_dms_GetActivationState, 1103
unpack_dms_GetBandCapability, 1103
unpack_dms_GetCrashAction, 1103
unpack_dms_GetCustFeature, 1104
unpack_dms_GetCustFeaturesV2, 1104
unpack_dms_GetDeviceCap, 1104
unpack_dms_GetDeviceCapabilities, 1105
unpack_dms_GetDeviceHardwareRev, 1105
unpack_dms_GetDeviceMfr, 1105
unpack_dms_GetDeviceSerialNumbers, 1106
unpack_dms_GetFSN, 1107
unpack_dms_GetFirmwareInfo, 1106
unpack_dms_GetFirmwareRevision, 1106
unpack_dms_GetFirmwareRevisions, 1107
unpack_dms_GetHardwareRevision, 1107
unpack_dms_GetIMSI, 1108
unpack_dms_GetManufacturer, 1108
unpack_dms_GetModelID, 1108
unpack_dms_GetNetworkTime, 1109
unpack_dms_GetOfflineReason, 1109
unpack_dms_GetPRLVersion, 1110
unpack_dms_GetPower, 1109
unpack_dms_GetSerialNumbers, 1110
unpack_dms_GetUSBComp, 1110
unpack_dms_GetVoiceNumber, 1111
unpack_dms_PSMCfgChange_ind, 1111
unpack_dms_ResetToFactoryDefaults, 1111
unpack_dms_SLQSDmsSwiGetPCInfo, 1115
unpack_dms_SLQSDmsSwiGetResetInfo, 1116
unpack_dms_SLQSDmsSwiGetResetInfo_Ind, 1116
unpack_dms_SLQSDmsSwiGetUimSelection, 1116
unpack_dms_SLQSDmsSwiIndicationRegister, 1117
unpack_dms_SLQSGetBandCapability, 1117
unpack_dms_SLQSGetERIFile, 1117
unpack_dms_SLQSGetPowerSaveModeConfig, 1118
unpack_dms_SLQSSetPowerSaveModeConfig, 1118
unpack_dms_SLQSSwiClearDyingGaspStatistics, 1119
unpack_dms_SLQSSwiGetCrashInfo, 1119
unpack_dms_SLQSSwiGetDyingGaspCfg, 1119
unpack_dms_SLQSSwiGetDyingGaspStatistics, 1120
unpack_dms_SLQSSwiGetFirmwareCurr, 1120
unpack_dms_SLQSSwiGetFwUpdateStatus, 1120
unpack_dms_SLQSSwiGetHostDevInfo, 1121
unpack_dms_SLQSSwiGetOSInfo, 1121
unpack_dms_SLQSSwiGetSerialNoExt, 1121
unpack_dms_SLQSSwiSetDyingGaspCfg, 1122
unpack_dms_SLQSSwiSetHostDevInfo, 1122
unpack_dms_SLQSSwiSetOSInfo, 1122
unpack_dms_SLQSUIMGetState, 1123
unpack_dms_SetActivationStatusCallback, 1112
unpack_dms_SetCrashAction, 1112
unpack_dms_SetCustFeature, 1112
unpack_dms_SetCustFeaturesV2, 1113
unpack_dms_SetEventReport, 1113
unpack_dms_SetEventReport_ind, 1113
unpack_dms_SetFirmwarePreference, 1114
unpack_dms_SetIndicationRegister, 1114
unpack_dms_SetPower, 1115
unpack_dms_SetUSBComp, 1115
unpack_dms_SwiEventReportCallBack_ind, 1123
unpack_dms_SwiSetEventReport, 1123
unpack_dms_SwiUimSelect, 1124
unpack_dms_UIMChangePIN, 1124
unpack_dms_UIMGetControlKeyStatus, 1124
unpack_dms_UIMGetICCID, 1125
unpack_dms_UIMGetPINStatus, 1125
unpack_dms_UIMSetControlKeyProtection, 1125

unpack_dms_UIMSetPINProtection, 1126
unpack_dms_UIMUnblockControlKey, 1126
unpack_dms_UIMUnblockPIN, 1126
unpack_dms_UIMVerifyPIN, 1127
unpack_dms_ValidateSPC, 1127
dms_ActivationStatusTlv, 52
activationStatus, 53
TlvPresent, 53
dms_OperatingModeTlv, 53
operatingMode, 54
TlvPresent, 54
dms_PSMActiveTimerIndTlv, 54
ActiveTimerInd, 54
TlvPresent, 54
dms_PSMActiveTimerTlv, 54
activeTimer, 55
TlvPresent, 55
dms_PSMDurationDueToOOSTlv, 55
durationDueToOOS, 55
TlvPresent, 55
dms_PSMDurationThresholdTlv, 55
durationThreshold, 56
TlvPresent, 56
dms_PSMEarlyWakeupTimeTlv, 56
earlyWakeupTime, 56
TlvPresent, 56
dms_PSMEnableStateIndTlv, 56
EnableStateInd, 57
TlvPresent, 57
dms_PSMEnableStateTlv, 57
enableState, 57
TlvPresent, 57
dms_PSMPeriodicUpdateTimerIndTlv, 57
PeriodicUpdateTimerInd, 58
TlvPresent, 58
dms_PSMPeriodicUpdateTimerTlv, 58
periodicUpdateTimer, 58
TlvPresent, 58
dms_PSMRandomizationWindowTlv, 58
randomizationWindow, 59
TlvPresent, 59
dms_TemperatureTlv, 59
TempStat, 60
Temperature, 60
TlvPresent, 60
dms_UimAutoSwitchActSlotTlv, 60
TlvPresent, 60
uimAutoSwitchActSlot, 60
dms_UimStatusTlv, 60
event, 61
intf, 61
TlvPresent, 61
dms_VoltageTlv, 61
TlvPresent, 62
VoltStat, 62
Voltage, 62
domain
wds_DomainNameList, 1038
domainLen
wds_Domain, 1037
DomainList
unpack_wds_SLQSGetRuntimeSettings_t, 955
domainName
wds_Domain, 1037
DormancyStat
wds_DormStatTlv, 1038
dormancyStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
dormancyState
unpack_wds_GetDormancyState_t, 932
dormancyStatus
pack_wds_SLQSSetWdsEventCallback_t, 505
unpack_wds_SLQSGetDUNCallInfo_t, 951
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
downLink
voice_NSSAudioCtrl, 1016
dtmSupp
nas_GSMSysInfo, 184
dtmSuppValid
nas_GSMSysInfo, 184
dunchannelRate, 64
CurrChanRxRate, 65
CurrChanTxRate, 65
MaxChanRxRate, 65
MaxChanTxRate, 65
Duration
pack_nas_SetNetworkPreference_t, 374
unpack_nas_GetNetworkPreference_t, 710
durationDueToOOS
dms_PSMDurationDueToOOSTlv, 55
durationThreshold
dms_PSMDurationThresholdTlv, 56

eAUDIO
common.h, 1066
eCAT
common.h, 1066
eCTL
common.h, 1065
eDMS
common.h, 1065
eFIREHOSE_ERR_SECBOOT_INVALID_CERT_CHAIN
lite-fw.h, 1154
eFW_TYPE_CWE
lite-fw.h, 1153
eFW_TYPE_CWE_NVU
lite-fw.h, 1153
eFW_TYPE_INVALID
lite-fw.h, 1153
eFW_TYPE_MBN
lite-fw.h, 1153
eFW_TYPE_MBN_GOBI
lite-fw.h, 1153
eFW_TYPE_NVU

lite-fw.h, 1153
eFW_TYPE_SPK
lite-fw.h, 1153
eIMS
common.h, 1066
eIMSA
common.h, 1066
eIND
common.h, 1066
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_1
nas.h, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_10
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_11
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_12
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_125
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_126
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_127
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_13
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_14
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_17
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_18
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_19
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_2
nas.h, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_20
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_21
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_23
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_24
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_25
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_250
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_26
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_27
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_28
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_29
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_3
nas.h, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_30
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_31
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_32
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_33
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_34
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_35
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_36
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_37
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_38
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_39
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_4
nas.h, 1195

eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_40
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_41
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_42
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_43
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_46
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_47
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_48
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_5
nas.h, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_6
nas.h, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_66
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_7
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_71
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_8
nas.h, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_9
nas.h, 1196
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED
nas.h, 1197
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
nas.h, 1197
eLOC
common.h, 1066
eLOG_DEBUG
common.h, 1065
eLOG_FATAL
common.h, 1065
eLOG_INFO
common.h, 1065
eLOG_WARN
common.h, 1065
eModel_9X15
lite-fw.h, 1154
eModel_9X30
lite-fw.h, 1154
eModel_9x06
lite-fw.h, 1154
eModel_9x07
lite-fw.h, 1154
eModel_Unknown
lite-fw.h, 1154
eModel_WP9X15
lite-fw.h, 1154
eNAS
common.h, 1065
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100
nas.h, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15
nas.h, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25
nas.h, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50
nas.h, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6
nas.h, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75
nas.h, 1197
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE
nas.h, 1197
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE
nas.h, 1197
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE
nas.h, 1197
ePDS
common.h, 1066
eQCWWAN_ERR_API_MUTEX_TIMEOUT
qmerrno.h, 1310
eQCWWAN_ERR_BUFFER_SZ
qmerrno.h, 1309
eQCWWAN_ERR_CANCEL_OP

qmerrno.h, 1310
eQCWWAN_ERR_DRIVER
qmerrno.h, 1310
eQCWWAN_ERR_ENUM_BEGIN
qmerrno.h, 1309
eQCWWAN_ERR_ENUM_END
qmerrno.h, 1310
eQCWWAN_ERR_FILE_COPY
qmerrno.h, 1309
eQCWWAN_ERR_FILE_OPEN
qmerrno.h, 1309
eQCWWAN_ERR_GENERAL
qmerrno.h, 1309
eQCWWAN_ERR_INTERNAL
qmerrno.h, 1309
eQCWWAN_ERR_INVALID_ARG
qmerrno.h, 1309
eQCWWAN_ERR_INVALID_DEVID
qmerrno.h, 1309
eQCWWAN_ERR_INVALID_FILE
qmerrno.h, 1309
eQCWWAN_ERR_INVALID_QMI_RSP
qmerrno.h, 1309
eQCWWAN_ERR_INVALID_XID
qmerrno.h, 1310
eQCWWAN_ERR_MALFORMED_QMI_RSP
qmerrno.h, 1309
eQCWWAN_ERR_MEMORY
qmerrno.h, 1309
eQCWWAN_ERR_MULTIPLE_DEVICES
qmerrno.h, 1310
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED
qmerrno.h, 1310
eQCWWAN_ERR_NO_CANCELABLE_OP
qmerrno.h, 1310
eQCWWAN_ERR_NO_CONNECTION
qmerrno.h, 1309
eQCWWAN_ERR_NO_DEVICE
qmerrno.h, 1309
eQCWWAN_ERR_NO_SIGNAL
qmerrno.h, 1310
eQCWWAN_ERR_NONE
qmerrno.h, 1309
eQCWWAN_ERR_NULL_TLV
qmerrno.h, 1313
eQCWWAN_ERR_OFFLINE
qmerrno.h, 1310
eQCWWAN_ERR_PDU_GENERATION
qmerrno.h, 1310
eQCWWAN_ERR_QMI_ABORTED
qmerrno.h, 1310
eQCWWAN_ERR_QMI_ACCESS_DENIED
qmerrno.h, 1312
eQCWWAN_ERR_QMI_ACK_NOT_SENT
qmerrno.h, 1312
eQCWWAN_ERR_QMI_ARG_TOO_LONG
qmerrno.h, 1310
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED
qmerrno.h, 1311
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK
qmerrno.h, 1311
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED
qmerrno.h, 1312
eQCWWAN_ERR_QMI_CALL_FAILED
qmerrno.h, 1310
eQCWWAN_ERR_QMI_CARD_BUSY_RSP
qmerrno.h, 1313
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED
qmerrno.h, 1312
eQCWWAN_ERR_QMI_CAT_END
qmerrno.h, 1313
eQCWWAN_ERR_QMI_CAT_START
qmerrno.h, 1313
eQCWWAN_ERR_QMI_CAUSE_CODE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED
qmerrno.h, 1310
eQCWWAN_ERR_QMI_CONNECT
qmerrno.h, 1309
eQCWWAN_ERR_QMI_DEVICE_IN_USE
qmerrno.h, 1310
eQCWWAN_ERR_QMI_DEVICE_NOT_READY
qmerrno.h, 1311
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL
qmerrno.h, 1311
eQCWWAN_ERR_QMI_DISABLED
qmerrno.h, 1311
eQCWWAN_ERR_QMI_ENCODING
qmerrno.h, 1311
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE
qmerrno.h, 1313
eQCWWAN_ERR_QMI_EVENT_REG_FAILED
qmerrno.h, 1313
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL
qmerrno.h, 1312
eQCWWAN_ERR_QMI_FDN_RESTRICT
qmerrno.h, 1312
eQCWWAN_ERR_QMI_FLOW_SUSPENDED
qmerrno.h, 1311
eQCWWAN_ERR_QMI_GENERAL
qmerrno.h, 1311
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED
qmerrno.h, 1312
eQCWWAN_ERR_QMI_IFACE
qmerrno.h, 1309
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE
qmerrno.h, 1312
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INCORRECT_PIN
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE
qmerrno.h, 1312
eQCWWAN_ERR_QMI_INJECT_TIMEOUT

qmerrno.h, 1312
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INTERNAL
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_ARG
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD
qmerrno.h, 1313
eQCWWAN_ERR_QMI_INVALID_HANDLE
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_ID
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_INDEX
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_OPERATION
qmerrno.h, 1312
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_PINID
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_PROFILE
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_QMI_CMD
qmerrno.h, 1312
eQCWWAN_ERR_QMI_INVALID_QOS_ID
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_TECH_PREF
qmerrno.h, 1310
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP
qmerrno.h, 1313
eQCWWAN_ERR_QMI_INVALID_TRANSITION
qmerrno.h, 1311
eQCWWAN_ERR_QMI_INVALID_TX_ID
qmerrno.h, 1310
eQCWWAN_ERR_QMI_MALFORMED_MSG
qmerrno.h, 1310
eQCWWAN_ERR_QMI_MAX
qmerrno.h, 1312
eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT
qmerrno.h, 1311
eQCWWAN_ERR_QMI_MISSING_ARG
qmerrno.h, 1310
eQCWWAN_ERR_QMI_MSG_BLOCKED
qmerrno.h, 1312
eQCWWAN_ERR_QMI_NETWORK_ABORTED
qmerrno.h, 1312
eQCWWAN_ERR_QMI_NETWORK_NOT_READY
qmerrno.h, 1311
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_NO_EFFECT
qmerrno.h, 1310
eQCWWAN_ERR_QMI_NO_ENTRY
qmerrno.h, 1311
eQCWWAN_ERR_QMI_NO_FREE_PROFILE
qmerrno.h, 1310
eQCWWAN_ERR_QMI_NO_MEMORY
qmerrno.h, 1310
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND
qmerrno.h, 1310
eQCWWAN_ERR_QMI_NO_RADIO
qmerrno.h, 1312
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION
qmerrno.h, 1312
eQCWWAN_ERR_QMI_NO_THRESHOLDS
qmerrno.h, 1310
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_NOT_PROVISIONED
qmerrno.h, 1310
eQCWWAN_ERR_QMI_NOT_SUPPORTED
qmerrno.h, 1312
eQCWWAN_ERR_QMI_OFFSET
qmerrno.h, 1310
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED
qmerrno.h, 1310
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED
qmerrno.h, 1310
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE
qmerrno.h, 1312
eQCWWAN_ERR_QMI_OUT_OF_CALL

qmerrno.h, 1310
eQCWWAN_ERR_QMI_PIN_BLOCKED
qmerrno.h, 1311
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED
qmerrno.h, 1311
eQCWWAN_ERR_QMI_POLICY_MISMATCH
qmerrno.h, 1312
eQCWWAN_ERR_QMI_REQ
qmerrno.h, 1309
eQCWWAN_ERR_QMI_REQ_SCH
qmerrno.h, 1309
eQCWWAN_ERR_QMI_REQ_TO
qmerrno.h, 1309
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED
qmerrno.h, 1311
eQCWWAN_ERR_QMI_RSP
qmerrno.h, 1309
eQCWWAN_ERR_QMI_RSP_TO
qmerrno.h, 1309
eQCWWAN_ERR_QMI_SEGMENT_ORDER
qmerrno.h, 1312
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG
qmerrno.h, 1312
eQCWWAN_ERR_QMI_SESSION_INACTIVE
qmerrno.h, 1311
eQCWWAN_ERR_QMI_SESSION_INVALID
qmerrno.h, 1311
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP
qmerrno.h, 1311
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND
qmerrno.h, 1312
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED
qmerrno.h, 1311
eQCWWAN_ERR_QMI_SMSC_ADDR
qmerrno.h, 1312
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE
qmerrno.h, 1312
eQCWWAN_ERR_QMI_TPDU_TYPE
qmerrno.h, 1312
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION
qmerrno.h, 1310
eQCWWAN_ERR_QMI_UNKNOWN
qmerrno.h, 1311
eQCWWAN_ERR_QMI_WIDTH
qmerrno.h, 1313
eQCWWAN_ERR_RESET
qmerrno.h, 1310
eQCWWAN_ERR_SWICM_AM_VERS_ERROR
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_END
qmerrno.h, 1313
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_SOCKET_IN_USE
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_START
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_TIMEOUT
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_V4DWN_V6DWN
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_V4DWN_V6UP
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_V4UP_V6DWN
qmerrno.h, 1312
eQCWWAN_ERR_SWICM_V4UP_V6UP
qmerrno.h, 1312
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED
qmerrno.h, 1313
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND
qmerrno.h, 1313
eQCWWAN_ERR_SWIDCS_END
qmerrno.h, 1313
eQCWWAN_ERR_SWIDCS_FILEIO_ERR
qmerrno.h, 1313
eQCWWAN_ERR_SWIDCS_IOCTL_ERR
qmerrno.h, 1313
eQCWWAN_ERR_SWIDCS_START
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_END
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE
qmerrno.h, 1313

eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_SECBOOT_IMAGE_NOT_SIGNED
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_SECBOOT_INVALID_CERT_CHAIN
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_INVALID_PATH
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_OPENING_DIR
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_OPENING_FILE
qmerrno.h, 1313
eQCWWAN_ERR_SWIIM_START
qmerrno.h, 1313
eQCWWAN_ERR_SWISM_END
qmerrno.h, 1313
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND
qmerrno.h, 1313
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED
qmerrno.h, 1313
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG
qmerrno.h, 1313
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED
qmerrno.h, 1313
eQCWWAN_ERR_SWISMS_START
qmerrno.h, 1313
eQDL_HW_FLOW_DISABLE
lite-fw.h, 1154
eQDL_HW_FLOW_ENABLE
lite-fw.h, 1154
eQDL_HW_FLOW_INIT
lite-fw.h, 1154
eQDL_HW_FLOW_Unknown
lite-fw.h, 1154
eQDL_MODE_INIT
lite-fw.h, 1154
eQDL_MODE_TTYUSB
lite-fw.h, 1154
eQDL_MODE_UART0
lite-fw.h, 1154
eQDL_MODE_Unknown
lite-fw.h, 1154
eQMI_LOC_SESS_STATUS_FAILURE
loc.h, 1166
eQMI_LOC_SESS_STATUS_IN_PROGRESS
loc.h, 1166
eQMI_LOC_SESS_STATUS_SUCCESS
loc.h, 1166
eQMI_LOC_SESS_STATUS_TIMEOUT
loc.h, 1166
eQOS
common.h, 1066
eREQ
common.h, 1066
eRMS
common.h, 1066
eRSP
common.h, 1066
eSAR
common.h, 1066
eSDP_FWDWL_ERR_END
lite-fw.h, 1154
eSDP_FWDWL_ERR_FAIL
lite-fw.h, 1154
eSDP_FWDWL_ERR_FW_UPGRADE
lite-fw.h, 1153
eSDP_FWDWL_ERR_FW_VERSION_FAIL
lite-fw.h, 1154
eSDP_FWDWL_ERR_GENERAL
lite-fw.h, 1153
eSDP_FWDWL_ERR_INVALID_DEV
lite-fw.h, 1153
eSDP_FWDWL_ERR_INVALID_PATH
lite-fw.h, 1154
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED
lite-fw.h, 1153
eSDP_FWDWL_ERR_PATH_TOO_LONG
lite-fw.h, 1153
eSDP_FWDWL_ERR_PRI_FAIL
lite-fw.h, 1154
eSDP_FWDWL_ERR_SDK
lite-fw.h, 1153
eSDP_FWDWL_ERR_SDP_TIMEOUT
lite-fw.h, 1154
eSDP_FWDWL_ERR_SET_CBK
lite-fw.h, 1153
eSDP_FWDWL_ERR_TIMEOUT
lite-fw.h, 1154
eSDP_FWDWL_SUCCESS
lite-fw.h, 1153
eSMS
common.h, 1066
eSWIAUDIO
common.h, 1066
eSWIDMS

common.h, 1066
eSWILOC
common.h, 1066
eSWIM2MCMD
common.h, 1066
eSWIM2MCMD_AVC2
common.h, 1066
eSWIOMA
common.h, 1066
eSWIOMAEXT
common.h, 1066
eTIMEOUT_10_S
common.h, 1066
eTIMEOUT_20_S
common.h, 1066
eTIMEOUT_2_S
common.h, 1066
eTIMEOUT_300_S
common.h, 1066
eTIMEOUT_30_S
common.h, 1066
eTIMEOUT_5_S
common.h, 1066
eTIMEOUT_60_S
common.h, 1066
eTIMEOUT_8_S
common.h, 1066
eTIMEOUT_DEFAULT
common.h, 1066
eTMD
common.h, 1066
eUIM
common.h, 1066
eVOICE
common.h, 1066
eWDS
common.h, 1065
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_END
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_FAIL
qmerrno.h, 1314
eWDS_ERR_PROFILE_REG_RESULT_LIST_END
qmerrno.h, 1314
EAWTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
ECIOThresListLen
nas_ECIOThresh, 171
ECTCallState
voice_ECTNum, 1011
eLOG_LEVEL
common.h, 1065
EMTlv
NASQmiCbkNasSystemSelPrefInd, 314
eQCWWANError
qmerrno.h, 1309
eQMI_SVC
common.h, 1065
ERI_DATA_MAX_SIZE
dms.h, 1077
ESATlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
ESAWTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
ESNString
unpack_dms_GetDeviceSerialNumbers_t, 594
ETWSPLMNInfo
eTWSPLMNInfoTlv, 66

eTWSPLMNInfoTlv, 66
ETWSPLMNInfo, 66
TlvPresent, 66
ETWSPLMNTlv
unpack_sms_SetNewSMSCallback_ind_t, 823
ETWSTlv
unpack_sms_SetNewSMSCallback_ind_t, 823
eTimeout
common.h, 1066
eValid
LibPackTFTIDParams, 119
EarMute
pack_audio_SLQSSetAudioProfile_t, 323
pack_swiaudio_SLQSSetM2MAVMute_t, 438
unpack_audio_SLQSGetAudioProfile_t, 578
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 839
unpack_swiaudio_SLQSGetM2MAVMute_t, 841
earfcn
nas_infoInterFreq, 192
nas_LTEInfoIntrafreq, 203
nas_ltePCI, 211
nas_umtsLTENbrCell, 290
earfcn0
nas_lteEARFCN, 197
earfcn1
nas_lteEARFCN, 197
earlyWakeupTime
dms_PSMEarlyWakeupTimeTlv, 56
ecio
cdmaSSInfo, 49
hdrSSInfo, 70
nas_ecioListElement, 170
nas_rxInfo, 267
nas_UMTSInfo, 288
tdscdmaSigInfoExt, 548
ecioDelta
nas_SLQSSignalStrengthsIndReq, 274
ecioInfo
nas_SLQSSignalStrengthsInformation, 276
ecioList
unpack_nas_SLQSGetSignalStrength_t, 735
ecioListLen
unpack_nas_SLQSGetSignalStrength_t, 735
ecioThresholdList
nas_SLQSSignalStrengthsIndReq, 274
ecioThresholdListLen
nas_SLQSSignalStrengthsIndReq, 274
edrxCiotLteModeTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 748
edrxCycleLengthTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 748
edrxEnableTypeTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 748
edrxEnabled
nas_EdrxEnableType, 173
edrxPagingTimeWindowTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 748
edrxPtw
nas_EdrxPagingTimeWindow, 173
edrxRatType
nas_EdrxRatType, 174
edrxRatTypeTlv
unpack_nas_SLQSNasEdrxChangeInfoCallBack_-Ind_t, 748
egprsSupp
nas_GSMSysInfo, 184
egprsSuppValid
nas_GSMSysInfo, 184
ehrpdMTUSize
swidms_ehrpdMTUSizeTlv, 544
elevation
loc_satelliteInfo, 138
EmerMode
nas_EmerModeTlv, 175
NASEmergencyModeTlv, 300
emmConnState
nas_LTEInfo, 200
emmState
nas_LTEInfo, 200
emmSubState
nas_LTEInfo, 200
Enable
pack_swiaudio_SLQSSetM2MAudioLPBK_t, 435
enable
pack_pds_SetPDSState_t, 411
pack_qos_SLQSSetQosEventCallback_t, 422
EnableState
unpack_dms_PSMCfgChange_ind_t, 610
enableState
dms_PSMEnableStateTlv, 57
EnableStateInd
dms_PSMEnableStateIndTlv, 57
enabled
pack_rms_SetSMSWake_t, 422
unpack_rms_GetSMSWake_t, 817
unpack_wds_GetMobileIPProfile_t, 935
EnabledStatus
unpack_pds_SetPdsState_Ind_t, 788
EncryptProt
nas_protocolSubtypeElement, 248
EncryptedPIN1
pack_uim_ChangePin_t, 455
pack_uim_SetPinProtection_t, 457
pack_uim_UnblockPin_t, 465
EndPointType
qos_BindDataPortPeripheralEndPointID_t, 518
EndProactiveSession
cat_EndProactiveSessionTlv, 47
EngineState
unpack_pds_SLQSGetGPSStateInfo_t, 793
engineState

unpack_loc_EngineState_Ind_t, 682
eqmiCbkSetStatus
sms.h, 1332
eriData
eriDataparams, 66
eriDataLen
eriDataparams, 66
eriDataparams, 65
eriData, 66
eriDataLen, 66
eriFile
unpack_dms_SLQSGetERIFile_t, 628
error
unpack_wds_GetLastMobileIPError_t, 932
errorClass
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
ErrorCode
imsa_IMSRegStatusErrorCodeInfo, 85
ErrorCodeData
imsa_RatHandoverStatusInfo, 87
ErrorCodeLen
imsa_RatHandoverStatusInfo, 87
errorMask
unpack_cat_SetCATEventCallback_t, 581
errorRate
nas_errorRateListElement, 176
errorRateInfo
nas_SLQSSignalStrengthsInformation, 276
errorRateList
unpack_nas_SLQSGetSignalStrength_t, 735
errorRateListLen
unpack_nas_SLQSGetSignalStrength_t, 735
errorState
slotInf, 524
uim_slotInfo, 573
esn
unpack_dms_GetSerialNumbers_t, 606
esnSize
unpack_dms_GetDeviceSerialNumbers_t, 594
EspSpi
unpack_qos_swiQosFilter_t, 809
EtwsMessageInfo
sMSEtwsMessageTlv, 540
event
dms_UimStatusTlv, 61
unpack_qos_SLQSSetQosPriEventCallback_ind_-t, 805
unpack_qos_SLQSSetQosStatusCallback_ind_t, 807
event_Index
unpack_cat_SetCatEventCallback_ind_t, 580
EventID
cat_commonEventTlv, 46
EventLength
cat_commonEventTlv, 46
eventMask
pack_cat_SetCATEventCallback_t, 326
pack_uim_SLQSUIMEventRegister_t, 459
unpack_uim_SLQSUIMEventRegister_t, 873
eventRegister
pack_loc_EventRegister_t, 357
eventType
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 858
evrcCapability
voice_prefVoiceSO, 1020
executingImage
FMSImageIDEntries, 68
exponent
unpack_qos_pktErrRate_t, 797
extBit
voice_calledPartySubAdd, 993
extDispInfo
voice_extDispRecInfo, 1011
extDispInfoLen
voice_extDispRecInfo, 1011
ExtErrorCode
PackCreateProfileOut, 511
extPowerState
pack_loc_SetExtPowerState_t, 357
extendedErrorCode
unpack_wds_SLQSDeleteProfile_t, 942

FLOAT
SwiDataTypes.h, 1375
FMSImageElement, 66
buildId, 67
buildIdLength, 67
imageId, 67
imageType, 67
FMSImageIDEntries, 68
executingImage, 68
imageIDElement, 68
imageIDSize, 68
imageType, 69
maxImages, 69
FMSImageIdElement, 67
buildID, 68
buildIDLength, 68
failureCount, 68
imageID, 68
storageIndex, 68
FMSImageList, 69
imageIDEntries, 69
listSize, 69
FMSPrefImageList, 69
listEntries, 70
listSize, 70
FOTAUpdate
pack_swioma_SLQSOMADMSetSettings_t, 451
pack_swioma_SLQSOMADMSetSettingsExt_t, 452
unpack_swioma_SLQSOMADMGetSettings_t, 862
FOTAdownload
pack_swioma_SLQSOMADMSetSettings_t, 451

pack_swioma_SLQSOMADMSetSettingsExt_t, 452
unpack_swioma_SLQSOMADMGetSettings_t, 862
FUMOEnable
pack_swioma_SLQSOMADMSetSettingsExt_t, 452
facility
pack_dms_UIMGetControlKeyStatus_t, 341
pack_dms_UIMSetControlKeyProtection_t, 342
pack_dms_UIMUnblockControlKey_t, 343
facilityCk
pack_dms_UIMSetControlKeyProtection_t, 342
pack_dms_UIMUnblockControlKey_t, 343
facilityState
pack_dms_UIMSetControlKeyProtection_t, 342
unpack_dms_UIMGetControlKeyStatus_t, 643
FailErrCode
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 669
failureCount
FMSImageIdElement, 68
family
pack_wds_GetDefaultProfileNum_t, 486
pack_wds_SetDefaultProfileNum_t, 492
feature
uim_depersonalizationInformation, 559
uim_personalizationStatus, 565
fileID
uim_fileAttributes, 563
uim_fileInfo, 564
fileIndex
pack_uim_ReadTransparent_t, 456
pack_uim_SLQSUIMGetFileAttributes_t, 460
fileSize
uim_fileAttributes, 563
fileType
uim_fileAttributes, 563
fill_pack_ctx
common.h, 1067
fill_sdu_hdr
common.h, 1067
filterId
LibPackTFTIDParams, 120
fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 447
unpack_swiloc_SwiLocGetAutoStart_t, 857
fix_rate_reported
unpack_swiloc_SwiLocGetAutoStart_t, 857
fix_type
pack_swiloc_SwiLocSetAutoStart_t, 447
unpack_swiloc_SwiLocGetAutoStart_t, 857
fix_type_reported
unpack_swiloc_SwiLocGetAutoStart_t, 857
fixAccuracy
pack_pds_StartPDSTrackingSessionExt_t, 419
fixCount
pack_pds_StartPDSTrackingSessionExt_t, 419
fixInterval
pack_pds_StartPDSTrackingSessionExt_t, 419
fixTimeout
pack_pds_StartPDSTrackingSessionExt_t, 419
flags
sensorData_t, 521
flowLabel
LibPackTFTIDParams, 120
fms.h, 1128
GetValidFwPriCombinations, 1129
pack_fms_GetImagesPreference, 1129
pack_fms_GetStoredImages, 1130
pack_fms_SetImagesPreference, 1130
unpack_fms_GetImagesPreference, 1130
unpack_fms_GetStoredImages, 1130
unpack_fms_SetImagesPreference, 1131
Forbidden
nas_QmiNas3GppNetworkInfo, 250
forbiddenNwInstLen
nas_ForbiddenNetworks3GPP, 177
forceFlag
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 414
ForceRev0
unpack_nas_GetCDMANetworkParameters_t, 705
format
sMSTransferRouteMTMessageInfo, 543
ForwardMac
nas_protocolSubtypeElement, 248
fqdnAddr
wds_PCSCFFQDNAddress, 1042
fqdnLen
wds_PCSCFFQDNAddress, 1042
freeSlots
sms_maxStorageSizeResp, 530
freq
nas_PhyCaAggPcellInfo, 238
nas_PhyCaAggScellIndType, 239
nas_PhyCaAggScellInfo, 242
nas_QmisNasSlqsNasPCICellInfo, 252
NASPhyCaAggPcellInfo, 306
NASPhyCaAggScellArray, 308
NASPhyCaAggScellIndType, 309
NASPhyCaAggScellInfo, 310
freqsLen
nas_LTEInfoInterfreq, 201
nas_LTEInfoNeighboringGSM, 204
nas_LTEInfoNeighboringWCDMA, 205
fromServiceId
sms_BroadcastConfig, 525
function
pack_swiloc_SwiLocSetAutoStart_t, 447
unpack_swiloc_SwiLocGetAutoStart_t, 857
function_reported
unpack_swiloc_SwiLocGetAutoStart_t, 857
FwAutoCheck
unpack_swioma_SLQSOMADMGetSettings_t, 862
FwAutoSDM
unpack_swiavms_SLQSAVMSGetSettings_t, 845

unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
FwAvailability
unpack_swioma_SLQSOMADMStartSession_t, 863
FwPromptUpdate
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
FwPromptdownload
unpack_swiavms_SLQSAVMSGetSettings_t, 845
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
fwdloadsize
unpack_omaDmFotaTlv_t, 779
fwloadComplete
unpack_omaDmFotaTlv_t, 779
fwvers
unpack_dms_SLQSSwiGetFirmwareCurr_t, 633

gDlBitRate
LibPackQosClassID, 118
GPRSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 955
GPSLPM
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
GPSSel
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
GSMRSSIThreshListLen
nas_GSMRSSIThresh, 181
GSMSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 756
gUlBitRate
LibPackQosClassID, 118
GWAOPTlv
NASQmiCbkNasSystemSelPrefInd, 314
GWAcqOrderPref
nas_GWAcqOrderPrefTlv, 185
NASGWAcqOrderPrefTlv, 301
GWAddressV4
unpack_wds_SLQSGetRuntimeSettings_t, 955
gcdumpString
crashInformation, 52
gcdumpStrlen
crashInformation, 52
Generator
pack_audio_SLQSGetAudioProfile_t, 319
pack_audio_SLQSGetAudioVolTLBConfig_t, 320
pack_audio_SLQSSetAudioProfile_t, 323
pack_audio_SLQSSetAudioVolTLBConfig_t, 324
pack_swiaudio_SLQSGetM2MAudioVolume_t, 433
pack_swiaudio_SLQSSetM2MAudioVolume_t, 437
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 839
geoSysIdx
nas_AddCDMASysInfo, 146
nas_AddSysInfo, 146
geranArfcn
nas_geranInstInfo, 179
geranBsicBcc
nas_geranInstInfo, 179
geranBsicNcc
nas_geranInstInfo, 179
geranInst
nas_UMTSInfo, 288
GeranInstInfo
nas_UMTSInfo, 288
geranRssi
nas_geranInstInfo, 179
get_version
common.h, 1067
getAllCallInfo
voice_arrCallInfo, 986
GetAllCallRmtPtyName
voice_arrRemotePartyName, 988
GetCustomFeatureV2
unpack_dms_GetCustFeaturesV2_t, 588
GetValidFwPriCombinations
fms.h, 1129
glo_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
glo_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
glo_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
glo_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
GlobalCellID
nas_QmisNasSlqsNasPCICellInfo, 252
globalCellId
nas_LTEInfoIntrafreq, 203
glog
common.h, 1068
gloglvl
common.h, 1068
gnssSvId
loc_satelliteInfo, 138
gnssSvUsedList
loc_svUsedforFix, 141
gnssSvUsedList_len
loc_svUsedforFix, 141
Gpp2TimeZone
unpack_nas_SLQSGetServingSystem_t, 733
GppNetworkDSTAdjustment
unpack_nas_SLQSGetServingSystem_t, 733
GppTimeZone
unpack_nas_SLQSGetServingSystem_t, 733
gps_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
gps_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
gps_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
gps_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793

GpsEnable
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
gpsTimeOfWeekMs
loc_gpsTime, 130
gpsWeek
loc_gpsTime, 130
grntDownlinkBitrate
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1048
grntUplinkBitrate
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
gsmAmrStat
voice_curAMRConfig, 1008
GsmCellInfo
nas_lteGsmCellInfo, 198
gsmUmtsDl
pack_nas_SLQSNasSwiIndicationRegister_t, 389
gsmUmtsUl
pack_nas_SLQSNasSwiIndicationRegister_t, 389
guaranteedRate
unpack_qos_dataRate_t, 795
gwAddressV6
wds_IPV6GWAddressInfo, 1041
gwV6PrefixLen
wds_IPV6GWAddressInfo, 1041
gyroData
pack_loc_SLQSLOCInjectSensorData_t, 365
gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 365
gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 366

HASPI
unpack_wds_GetMobileIPProfile_t, 935
HAState
unpack_wds_GetMobileIPProfile_t, 935
HDOP
loc_precisionDilution, 136
HDRECIOThreshListLen
nas_HDRECIOThresh, 186
HDRIOThreshListLen
nas_HDRIOThresh, 186
HDRRSSIThreshListLen
nas_HDRRSSIThresh, 187
HDRSINRThresListLen
nas_HDRSINRThresh, 188
HDRSINRThreshListLen
nas_HDRSINRThreshold, 189
HDRSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 756
HWVersion
nas_DeviceConfigDetail, 169
HardwareControlledMode
unpack_dms_GetPower_t, 604
has_LpmFlag
unpack_dms_SLQSDmsSwiGetPCInfo_t, 618
has_PdpStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
has_PersistentLpm
unpack_dms_SLQSDmsSwiGetPCInfo_t, 618
has_PositionDataNMEA
unpack_pds_SetEventReport_Ind_t, 787
has_PowerOffMode
unpack_dms_SLQSDmsSwiGetPCInfo_t, 618
has_RatHandoverStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
has_RegStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
has_ServiceStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
has_SessionStatus
unpack_pds_SetEventReport_Ind_t, 787
has_Wdisable
unpack_dms_SLQSDmsSwiGetPCInfo_t, 618
has_accelTemp
pack_loc_SLQSLOCInjectSensorData_t, 366
has_acceleroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 366
has_accleroData
pack_loc_SLQSLOCInjectSensorData_t, 366
has_altitudeSrcInfo
pack_loc_SLQSLOCInjectPosition_t, 363
has_altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 363
has_altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 363
has_confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 367
has_dLatitude
unpack_pds_SetEventReport_Ind_t, 787
has_dLongitude
unpack_pds_SetEventReport_Ind_t, 787
has_gyroData
pack_loc_SLQSLOCInjectSensorData_t, 366
has_gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 366
has_gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 366
has_horConfidence
pack_loc_SLQSLOCInjectPosition_t, 363
has_horReliability
pack_loc_SLQSLOCInjectPosition_t, 363
has_horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 363
has_latitude
pack_loc_SLQSLOCInjectPosition_t, 363
has_longitude
pack_loc_SLQSLOCInjectPosition_t, 363
has_opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 366
has_posSrc
unpack_pds_SetEventReport_Ind_t, 787
has_positionSrc
pack_loc_SLQSLOCInjectPosition_t, 363
has_rawHorConfidence

pack_loc_SLQSLOCInjectPosition_t, 363
has_rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 363
has_serverAddrTypeMask
pack_loc_SLQSLOCGetServer_t, 359
has_timestampAge
pack_loc_SLQSLOCInjectPosition_t, 363
has_timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 363
has_vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 363
has_vertRelicability
pack_loc_SLQSLOCInjectPosition_t, 363
has_vertUnc
pack_loc_SLQSLOCInjectPosition_t, 364
hdrActiveProt
nas_HDRSysInfo, 191
hdrActiveProtValid
nas_HDRSysInfo, 191
hdrHybrid
nas_detailSvcInfo, 168
HdrPersonality
unpack_nas_SLQSGetServingSystem_t, 733
hdrPersonality
nas_HDRSysInfo, 191
NASServingSystemInfo, 317
hdrPersonalityValid
nas_HDRSysInfo, 191
hdrSSInfo, 70
ecio, 70
io, 70
rssi, 71
sinr, 71
hdrSrvStatus
nas_detailSvcInfo, 168
healthStatus
loc_satelliteInfo, 138
helper_get_error_code
common.h, 1067
helper_get_error_reason
common.h, 1067
helper_get_req_str
common.h, 1067
helper_get_resp_ctx
common.h, 1067
helper_get_xid
common.h, 1067
helper_isBootLoader_DebugEnabled
common.h, 1067
helper_set_log_func
common.h, 1068
helper_set_log_lvl
common.h, 1068
homeOrigVoiceSO
voice_prefVoiceSO, 1020
homePageVoiceSO
voice_prefVoiceSO, 1020
horConfidence
pack_loc_SLQSLOCInjectPosition_t, 364
horReliability
pack_loc_SLQSLOCInjectPosition_t, 364
horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 364
HorizontalUncertainty
unpack_pds_SLQSGetGPSStateInfo_t, 793
hostID
pack_dms_SLQSSwiSetHostDevInfo_t, 338
unpack_dms_SLQSSwiGetHostDevInfo_t, 636
hotSwap
uim_hotSwapStatus, 564
hotSwapLength
uim_hotSwapStatus, 564
hour
nas_timeInfo, 285
nas_UniversalTime, 291
hrpdMTUSize
swidms_hrpdMTUSizeTlv, 545
hsCallStatus
nas_WCDMASysInfo, 297
hsCallStatusValid
nas_WCDMASysInfo, 297
hsInd
nas_WCDMASysInfo, 297
hsIndValid
nas_WCDMASysInfo, 297
hwType
wds_DHCPv4HWConfig, 1035
wdsDhcpv4HwConfig, 1049
hwVer
unpack_dms_GetHardwareRevision_t, 599

IDTlv
unpack_ims_SLQSUserCfgCallBack_ind_t, 663
iFaceTab
swiaudio_PCMparams, 544
iFaceTabLen
swiaudio_PCMparams, 544
iLTEbandValue
nas_PhyCaAggPcellInfo, 238
nas_PhyCaAggScellInfo, 242
NASPhyCaAggPcellInfo, 306
NASPhyCaAggScellInfo, 311
IMCNflag
unpack_wds_SLQSGetRuntimeSettings_t, 955
IMEIString
unpack_dms_GetDeviceSerialNumbers_t, 594
IMG_MASK_CLEAR
lite-fw.h, 1151
IMG_MASK_GENERIC
lite-fw.h, 1151
IMG_MASK_MDM
lite-fw.h, 1151
IMG_MASK_PRI
lite-fw.h, 1151
IMSInfo
sMSOnIMSTlv, 542
IMSRegistration

unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 670
IMSRegistrationError
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 670
IMSTlv
unpack_sms_SetNewSMSCallback_ind_t, 823
INT32
SwiDataTypes.h, 1375
INT8
SwiDataTypes.h, 1375
IOThresListLen
nas_IOThresh, 193
IPAddress
nas_DataStatusDetail, 166
IPAddressV6
ipv6AddressInfo, 92
wds_IPV6AddressInfo, 1040
IPFamSupport
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
IPFamilyPreference
pack_wds_SLQSSetIPFamilyPreference_t, 504
unpack_wds_SLQSGetRuntimeSettings_t, 955
IPSECSPI
LibPackTFTIDParams, 120
IPV6AddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 955
IPV6GWAddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 955
IPV6PrefixLen
ipv6AddressInfo, 92
wds_IPV6AddressInfo, 1040
IPv4
unpack_wds_SLQSGetRuntimeSettings_t, 955
IPv4Addr
loc_IPv4Config, 132
wds_IPv4AdTlv, 1040
IPv4AddrTlv
unpack_wds_DHCPv4ClientLease_ind_t, 925
IPv4DstAddr
unpack_qos_swiQosFilter_t, 809
IPv4Port
loc_IPv4Config, 132
IPv4SrcAddr
unpack_qos_swiQosFilter_t, 809
IPv4Tos
unpack_qos_swiQosFilter_t, 809
IPv6Addr
loc_IPv6Config, 133
IPv6DstAddr
unpack_qos_swiQosFilter_t, 809
IPv6Label
unpack_qos_swiQosFilter_t, 809
IPv6Port
loc_IPv6Config, 133
IPv6SrcAddr
unpack_qos_swiQosFilter_t, 809
IPv6TrafCls
unpack_qos_swiQosFilter_t, 809
Id
unpack_qos_swiQosFilter_t, 809
id
loc_BdsSV, 125
loc_SV, 140
nas_CSGID, 163
pack_dms_UIMChangePIN_t, 340
pack_dms_UIMSetPINProtection_t, 343
pack_dms_UIMUnblockPIN_t, 344
pack_dms_UIMVerifyPIN_t, 344
unpack_qos_QosFlowInfoState_t, 799
unpack_qos_SLQSSetQosStatusCallback_ind_t, 807
id_length
DMScustSettingInfo, 62
IdleState
nas_protocolSubtypeElement, 248
IfaceID
qos_BindDataPortPeripheralEndPointID_t, 518
image_info_t, 71
buildID, 71
buildIDLen, 71
imageType, 71
uniqueID, 71
imageID
FMSImageIdElement, 68
imageIDElement
FMSImageIDEntries, 68
imageIDEntries
FMSImageList, 69
imageIDSize
FMSImageIDEntries, 68
imageId
FMSImageElement, 67
imageList
unpack_fms_GetStoredImages_t, 649
ImageListSize
unpack_fms_GetImagesPreference_t, 648
imageListSize
pack_fms_SetImagesPreference_t, 346
imageType
FMSImageElement, 67
FMSImageIDEntries, 69
image_info_t, 71
ImageTypes
unpack_fms_SetImagesPreference_t, 649
ImageTypesSize
unpack_fms_SetImagesPreference_t, 649
imagelistSize
unpack_fms_GetStoredImages_t, 649
imei_no
unpack_dms_GetSerialNumbers_t, 606
imeiSize
unpack_dms_GetDeviceSerialNumbers_t, 594
imeiSvnSize
unpack_dms_GetDeviceSerialNumbers_t, 594

ImeiSvnString
unpack_dms_GetDeviceSerialNumbers_t, 594
imeisv_svn
unpack_dms_GetSerialNumbers_t, 606
imgType
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 635
ims.h, 1131
MAX_NAME_LEN, 1133
pack_ims_SLQSGetIMSSMSConfig, 1133
pack_ims_SLQSGetIMSUserConfig, 1133
pack_ims_SLQSGetIMSVoIPConfig, 1134
pack_ims_SLQSGetRegMgrConfig, 1134
pack_ims_SLQSGetSIPConfig, 1135
pack_ims_SLQSImsConfigIndicationRegister, 1135
pack_ims_SLQSSetIMSSMSConfig, 1136
pack_ims_SLQSSetIMSUserConfig, 1136
pack_ims_SLQSSetIMSVoIPConfig, 1137
pack_ims_SLQSSetRegMgrConfig, 1137
pack_ims_SLQSSetSIPConfig, 1137
unpack_ims_SLQSGetIMSSMSConfig, 1138
unpack_ims_SLQSGetIMSUserConfig, 1138
unpack_ims_SLQSGetIMSVoIPConfig, 1139
unpack_ims_SLQSGetRegMgrConfig, 1139
unpack_ims_SLQSGetSIPConfig, 1139
unpack_ims_SLQSImsConfigIndicationRegister, 1140
unpack_ims_SLQSImsConfigIndicationRegister_t, 1133
unpack_ims_SLQSRegMgrCfgCallBack_ind, 1140
unpack_ims_SLQSSIPCfgCallBack_ind, 1142
unpack_ims_SLQSSMSCfgCallBack_ind, 1142
unpack_ims_SLQSSetIMSSMSConfig, 1140
unpack_ims_SLQSSetIMSUserConfig, 1141
unpack_ims_SLQSSetIMSVoIPConfig, 1141
unpack_ims_SLQSSetRegMgrConfig, 1141
unpack_ims_SLQSSetSIPConfig, 1142
unpack_ims_SLQSUserCfgCallBack_ind, 1143
unpack_ims_SLQSVoIPCfgCallBack_ind, 1143
ims_AMRModeInfo, 71
amrMode, 72
TlvPresent, 72
ims_AMROctAlgnInfo, 72
amrOctAlgn, 72
TlvPresent, 72
ims_AMRWBModeInfo, 73
amrWBMode, 73
TlvPresent, 73
ims_AMRWBOctAlgnInfo, 73
amrWBOctAlgn, 74
TlvPresent, 74
ims_CSCFPortNameInfo, 74
cscfPortName, 74
TlvPresent, 74
ims_EnabAMRWBInfo, 74
amrWBEnable, 74
TlvPresent, 74
ims_EnabSCRAMRInfo, 75
scrAmrEnable, 75
TlvPresent, 75
ims_EnabSCRAMRWBInfo, 75
scrAmrWBEnable, 75
TlvPresent, 75
ims_IMSDomainInfo, 76
imsDomainName, 76
TlvPresent, 76
ims_IMSTestModeInfo, 76
imsTestMode, 76
TlvPresent, 76
ims_MinSessExpInfo, 76
minSessExp, 77
TlvPresent, 77
ims_PCSCFPortInfo, 77
priCSCFPort, 77
TlvPresent, 77
ims_PhCtxtURIInfo, 77
PhCtxtURI, 78
TlvPresent, 78
ims_RTPRTCPInactTmrDurInfo, 79
InactTmr, 79
TlvPresent, 79
ims_RngBkTmrInfo, 78
RingBkTmr, 78
TlvPresent, 78
ims_RngTmrInfo, 78
RingTmr, 79
TlvPresent, 79
ims_SIPPortInfo, 80
SIPLocalPort, 81
TlvPresent, 81
ims_SIPRegnTmrInfo, 81
TlvPresent, 81
tmrSIPRegn, 81
ims_SMSFmtInfo, 81
smsFormat, 82
TlvPresent, 82
ims_SMSoIPNwInfo, 82
smsoIPNW, 82
TlvPresent, 82
ims_SessDurInfo, 79
sessExp, 80
TlvPresent, 80
ims_SigCompEnInfo, 80
SigCompEn, 80
TlvPresent, 80
ims_SubscrTmrInfo, 82
subscrTmr, 83
TlvPresent, 83
ims_TmrT1Info, 83
TlvPresent, 83
tmrT1, 83
ims_TmrT2Info, 83
TlvPresent, 84
tmrT2, 84
ims_TmrTfInfo, 84
TlvPresent, 84

tmrTf, 84
imsDomainName
ims_IMSDomainInfo, 76
ImsFailErrCode
imsa_IMSFailErrCodeTlv, 85
ImsRegErrCode
unpack_imsa_SLQSGetIMSARegStatus_t, 666
imsRegState
nas_CommInfo, 162
ImsRegStatus
imsa_NewIMSRegStatusInfo, 86
unpack_imsa_SLQSGetIMSARegStatus_t, 666
ImsRegistered
imsa_IMSRegStatusInfo, 86
imsTestMode
ims_IMSTestModeInfo, 76
imsa.h, 1143
MAX_ERROR_CODE_LEN, 1145
pack_imsa_SLQSGetIMSARegStatus, 1145
pack_imsa_SLQSGetIMSAServiceStatus, 1145
pack_imsa_SLQSRegisterIMSAIndication, 1146
unpack_imsa_SLQSGetIMSARegStatus, 1146
unpack_imsa_SLQSGetIMSAServiceStatus, 1147
unpack_imsa_SLQSImsaPdpStatusCallBack_ind, 1147
unpack_imsa_SLQSImsaRatStatusCallBack_ind, 1147
unpack_imsa_SLQSImsaRegStatusCallBack_ind, 1148
unpack_imsa_SLQSImsaSvcStatusCallBack_ind, 1148
unpack_imsa_SLQSRegisterIMSAIndication, 1149
unpack_imsa_SLQSRegisterIMSAIndication_t, 1145
imsa_IMSFailErrCodeTlv, 84
ImsFailErrCode, 85
TlvPresent, 85
imsa_IMSRegStatusErrorCodeInfo, 85
ErrorCode, 85
TlvPresent, 85
imsa_IMSRegStatusInfo, 85
ImsRegistered, 86
TlvPresent, 86
imsa_NewIMSRegStatusInfo, 86
ImsRegStatus, 86
TlvPresent, 86
imsa_RatHandoverStatusInfo, 86
ErrorCodeData, 87
ErrorCodeLen, 87
RatHandoverStatus, 87
SourceRAT, 87
TargetRAT, 87
TlvPresent, 87
imsa_SmsRatInfo, 87
SmsRatVal, 88
TlvPresent, 88
imsa_SmsSvcStatusInfo, 88
SmsSvcStatus, 88
TlvPresent, 88
imsa_UtRatInfo, 88
TlvPresent, 89
UtRatVal, 89
imsa_UtSvcStatusInfo, 89
TlvPresent, 89
UtSvcStatus, 89
imsa_VoipRatInfo, 90
TlvPresent, 90
VoipRatVal, 90
imsa_VoipSvcStatusInfo, 90
TlvPresent, 90
VoipSvcStatus, 90
imsa_VtRatInfo, 91
TlvPresent, 91
VtRatVal, 91
imsa_VtSvcStatusInfo, 91
TlvPresent, 92
VtSvcStatus, 92
imsi
unpack_dms_GetIMSI_t, 600
imsi_11_12
nas_CDMASysInfoExt, 157
imsiM1112
nas_minBasedIMSI, 221
imsiMS1
nas_minBasedIMSI, 221
imsiMS2
nas_minBasedIMSI, 221
imsiT1112
nas_trueIMSI, 286
imsiTS1
nas_trueIMSI, 286
imsiTS2
nas_trueIMSI, 286
imsiTaddrNum
nas_trueIMSI, 286
InUse
nas_QmiNas3GppNetworkInfo, 250
InactTmr
ims_RTPRTCPInactTmrDurInfo, 79
includes_pcs_digit
nas_QmisNasPcsDigit, 252
index
pack_wds_GetMobileIPProfile_t, 488
pack_wds_SetDefaultProfileNum_t, 492
pack_wds_SetMobileIPProfile_t, 495
unpack_qos_swiQosFilter_t, 809
unpack_qos_swiQosFlow_t, 813
unpack_wds_GetDefaultProfileNum_t, 931
index1xPri
uim_cardStatus, 557
index1xSec
uim_cardStatus, 557
indexGwPri
uim_cardStatus, 557
indexGwSec
uim_cardStatus, 557

Info
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 765
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 766
InfoInterfreq
nas_LTEInfoInterfreq, 201
injectEnable
loc_accelAcceptReady, 124
loc_accelTempAcceptReady, 125
loc_gyroAcceptReady, 131
loc_gyroTempAcceptReady, 131
injectSensorDataStatus
unpack_loc_InjectSensorDataCallback_Ind_t, 688
injectTimeSyncStatus
unpack_loc_InjectTimeSyncDataCallback_Ind_t, 689
insNmrCellInfo
nas_GERANInfo, 178
instancesSize
unpack_nas_GetRFInfo_t, 711
interval
pack_pds_SetPDSDefaults_t, 411
pack_pds_SetXTRAAutomaticDownload_t, 413
pack_wds_SLQSSetWdsEventCallback_t, 505
intf
dms_UimStatusTlv, 61
Io
unpack_nas_SLQSGetSignalStrength_t, 735
io
hdrSSInfo, 70
nas_SLQSSignalStrengthsInformation, 276
ioDelta
nas_SLQSSignalStrengthsIndReq, 275
Iono_valid
unpack_pds_SLQSGetGPSStateInfo_t, 793
ipAddress
pack_wds_SetDefaultProfile_t, 492
ipFamily
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 959
ipVersion
LibPackTFTIDParams, 120
ipaddr
unpack_wds_GetDefaultProfile_t, 930
ipaddrv6
unpack_wds_GetDefaultProfile_t, 931
ipv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
ipv4GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
ipv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
ipv6AddressInfo, 92
IPAddressV6, 92
IPV6PrefixLen, 92
ipv6GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
is856SysId
nas_HDRSysInfo, 191
is856SysIdValid
nas_HDRSysInfo, 191
is_DataRate_Available
unpack_qos_swiQosFlow_t, 813
is_EspSpi_Available
unpack_qos_swiQosFilter_t, 809
is_IPv4DstAddr_Available
unpack_qos_swiQosFilter_t, 809
is_IPv4SrcAddr_Available
unpack_qos_swiQosFilter_t, 809
is_IPv4Tos_Available
unpack_qos_swiQosFilter_t, 809
is_IPv6DstAddr_Available
unpack_qos_swiQosFilter_t, 809
is_IPv6Label_Available
unpack_qos_swiQosFilter_t, 809
is_IPv6SrcAddr_Available
unpack_qos_swiQosFilter_t, 809
is_IPv6TrafCls_Available
unpack_qos_swiQosFilter_t, 809
is_Id_Available
unpack_qos_swiQosFilter_t, 809
is_Jitter_Available
unpack_qos_swiQosFlow_t, 813
is_Latency_Available
unpack_qos_swiQosFlow_t, 813
is_LteBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 627
is_LteQci_Available
unpack_qos_swiQosFlow_t, 813
is_MaxAllowedPktSz_Available
unpack_qos_swiQosFlow_t, 813
is_MinPolicedPktSz_Available
unpack_qos_swiQosFlow_t, 813
is_NxtHdrProto_Available
unpack_qos_swiQosFilter_t, 810
is_PktErrRate_Available
unpack_qos_swiQosFlow_t, 813
is_Precedence_Available
unpack_qos_swiQosFilter_t, 810
is_ProfileId3GPP2_Available
unpack_qos_swiQosFlow_t, 813
is_RxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 798
is_TCPDstPort_Available
unpack_qos_swiQosFilter_t, 810
is_TCPSrcPort_Available
unpack_qos_swiQosFilter_t, 810
is_TdsBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 627
is_TokenBucket_Available
unpack_qos_swiQosFlow_t, 813

is_TrafficClass_Available
unpack_qos_swiQosFlow_t, 813
is_TranDstPort_Available
unpack_qos_swiQosFilter_t, 810
is_TranSrcPort_Available
unpack_qos_swiQosFilter_t, 810
is_TxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 798
is_UDPDstPort_Available
unpack_qos_swiQosFilter_t, 810
is_UDPSrcPort_Available
unpack_qos_swiQosFilter_t, 810
is_val_3GPP2Pri_Available
unpack_qos_swiQosFlow_t, 813
is_val_3GPPImCn_Available
unpack_qos_swiQosFlow_t, 813
is_val_3GPPResResidualBER_Available
unpack_qos_swiQosFlow_t, 813
is_val_3GPPSigInd_Available
unpack_qos_swiQosFlow_t, 814
is_val_3GPPTraHdlPri_Available
unpack_qos_swiQosFlow_t, 814
isEmpty
voice_getAllCallInformation, 1012
isInTraffic
nas_txInfo, 286
isModByCC
voice_SUPSInfo, 1025
isNewFlow
unpack_qos_QosFlowInfoState_t, 799
isPrefDataPath
nas_GSMSrvStatusInfo, 182
nas_NR5GSerStatTlv, 231
nas_SrvStatusInfo, 278
isRadioTuned
nas_rxInfo, 267
nas_RxSigInfo, 268
isSysForbidden
nas_detailSvcInfo, 168
nas_sysInfoCommon, 280
isSysForbiddenValid
nas_sysInfoCommon, 280
isSysPrlMatch
nas_CDMASysInfo, 156
nas_HDRSysInfo, 191
isSysPrlMatchValid
nas_CDMASysInfo, 156
nas_HDRSysInfo, 191
IsVoiceEnabled
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
Item
pack_audio_SLQSGetAudioPathConfig_t, 319
pack_audio_SLQSGetAudioVolTLBConfig_t, 320
pack_audio_SLQSSetAudioVolTLBConfig_t, 324

Jitter
unpack_qos_swiQosFlow_t, 814

KeyExchange
nas_protocolSubtypeElement, 248

LITE_SYS_SRV_DOMAIN_CAMPED
nas.h, 1195
LITE_SYS_SRV_DOMAIN_CS_ONLY
nas.h, 1195
LITE_SYS_SRV_DOMAIN_CS_PS
nas.h, 1195
LITE_SYS_SRV_DOMAIN_NO_SRV
nas.h, 1195
LITE_SYS_SRV_DOMAIN_PS_ONLY
nas.h, 1195
LITE_VOICE_SUPS_SRV_CLASS_DATA
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITAS-
YNC
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSY-
NC
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_FAX
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_NONE
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_PADACCESS
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_SMS
voice.h, 1429
LITE_VOICE_SUPS_SRV_CLASS_VOICE
voice.h, 1429
LITEQMI_QMI_CBK_PARAM_NOCHANGE
sms.h, 1332
LITEQMI_QMI_CBK_PARAM_RESET
sms.h, 1332
LITEQMI_QMI_CBK_PARAM_SET
sms.h, 1332
LBPTlv
NASQmiCbkNasSystemSelPrefInd, 314
LMCERTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t,
944
LOCEVENTMASKGNSSSVINFO
loc.h, 1164
LOCEVENTMASKNMEA
loc.h, 1165
LOCEVENTMASKWIFIREQ
loc.h, 1166
LPCSTR
SwiDataTypes.h, 1375
LTEAttachProfile
unpack_wds_SLQSGet3GPPConfigItem_t, 945
LTEAttachProfileList
unpack_wds_SLQSGet3GPPConfigItem_t, 946
LTEAttachProfileListLen
pack_wds_SLQSSet3GPPConfigItem_t, 503
unpack_wds_SLQSGet3GPPConfigItem_t, 946

LTEBandPref
nas_LTEBandPrefTlv, 195
NASLTEBandPreferenceTlv, 302
LTECphyCaInfo
unpack_nas_SlqsGetLTECphyCAInfo_t, 724
LTERSRPThreshListLen
nas_LTERSRPThresh, 212
LTERSRQThreshListLen
nas_LTERSRQThresh, 213
LTERSSIThreshListLen
nas_LTERSSIThresh, 213
LTESNRThresListLen
nas_LTESNRThresh, 216
LTESNRThreshListLen
nas_LTESNRThreshold, 217
LTESSInfo
unpack_nas_SLQSNasGetSigInfo_t, 756
LTEWCDMACellInfo
nas_LTEInfoNeighboringWCDMA, 205
Lac
unpack_nas_SLQSGetServingSystem_t, 733
lac
nas_GERANInfo, 178
nas_GSMSysInfo, 184
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 234
nas_UMTSInfo, 288
nas_WCDMASysInfo, 297
lac1
nas_OperatorPLMNData, 236
lac2
nas_OperatorPLMNData, 236
lacValid
nas_GSMSysInfo, 184
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 234
nas_WCDMASysInfo, 297
language
sms_CDMABroadcastConfig, 526
lastCallDataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 951
lastCallDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 949
lastCallRXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 951
lastCallTXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 951
LastErrCode
nas_DataStatusDetail, 166
Latency
unpack_qos_swiQosFlow_t, 814
Latitude
unpack_pds_SLQSGetGPSStateInfo_t, 793
latitude
pack_loc_SLQSLOCInjectPosition_t, 364
leapSeconds
nas_qaQmi3Gpp2TimeZone, 249
leaseState
wds_DHCPLeaseStateTlv, 1033
len
loc_BdsSVInfo, 126
loc_SVInfo, 140
unpack_nas_GetSignalStrengths_t, 715
length
sMSCAddressInfo, 538
sMSEtwsMessageInfo, 539
sMSTransferRouteMTMessageInfo, 543
uim_readTransparentInfo, 566
Level
pack_swiaudio_SLQSSetM2MAudioVolume_t, 437
unpack_swiaudio_SLQSGetM2MAudioVolume_t, 840
LibPackGPRSRequestedQoS, 92
delayClass, 93
meanThroughputClass, 93
peakThroughputClass, 93
precedenceClass, 93
reliabilityClass, 93
LibPackQosClassID, 117
gDlBitRate, 118
gUlBitRate, 118
maxDlBitRate, 118
maxUlBitRate, 118
QCI, 118
LibPackTFTIDParams, 118
destPortRangeEnd, 119
destPortRangeStart, 119
eValid, 119
filterId, 120
flowLabel, 120
IPSECSPI, 120
ipVersion, 120
nextHeader, 120
pSourceIP, 120
sourceIPMask, 120
srcPortRangeEnd, 120
srcPortRangeStart, 120
tosMask, 120
LibPackUMTSQoS, 120
deliveryErrSDU, 122
grntDownlinkBitrate, 122
grntUplinkBitrate, 122
maxDownlinkBitrate, 122
maxSDUSize, 122
maxUplinkBitrate, 122
qosDeliveryOrder, 122
resBerRatio, 122
sduErrorRatio, 122
trafficClass, 122
trafficPriority, 122
transferDelay, 122
LibPackUMTSReqQoSSigInd, 123
SigInd, 123
UMTSReqQoS, 123
LibPackprofile_3GPP, 106

pAPNClass, 110
pAPNDisabledFlag, 110
pAPNName, 110
pAPNnameSize, 111
pAddrAllocPref, 110
pAuthenticationPref, 111
pGPRSMinimumQoS, 111
pGPRSRequestedQos, 111
pIPv4AddrPref, 111
pIPv6AddPref, 111
pImCnFlag, 111
pPDNInactivTimeout, 111
pPDPtype, 111
pPassword, 111
pPasswordSize, 111
pPcscfAddrUsingDhcp, 111
pPcscfAddrUsingPCO, 111
pPdpAccessConFlag, 111
pPdpContext, 111
pPdpDataCompType, 111
pPdpHdrCompType, 111
pPriDNSIPv4AddPref, 111
pPriDNSIPv6addpref, 111
pPrimaryID, 111
pProfilename, 111
pProfilenameSize, 111
pQosClassID, 111
pSecDNSIPv4AddPref, 111
pSecDNSIPv6addpref, 111
pSecondaryFlag, 111
pSupportEmergencyCalls, 111
pTFTID1Params, 111
pTFTID2Params, 112
pUMTSMinQoS, 112
pUMTSMinQosSigInd, 112
pUMTSReqQoS, 112
pUMTSReqQoSSigInd, 112
pUsername, 112
pUsernameSize, 112
LibPackprofile_3GPP2, 112
pAPNClass3GPP2, 116
pAPNEnabled3GPP2, 116
pAllowLinger, 116
pApnString, 116
pApnStringSize, 116
pAppPriority, 116
pAppType, 116
pAuthPassword, 116
pAuthPassword_tSize, 116
pAuthProtocol, 116
pAuthRetryCount, 116
pAuthTimeout, 116
pDataMode, 116
pDataRate, 116
pIpcpAckTimeout, 116
pIpcpCreqRetryCount, 117
pIsPcscfAddressNedded, 117
pLcpAckTimeout, 117
pLcpCreqRetryCount, 117
pNegoDnsSrvrPref, 117
pPDNInactivTimeout3GPP2, 117
pPdnType, 117
pPppSessCloseTimer1x, 117
pPppSessCloseTimerDO, 117
pPriV6DnsAddress, 117
pPrimaryV4DnsAddress, 117
pRATType, 117
pSecV6DnsAddress, 117
pSecondaryV4DnsAddress, 117
pUserId, 117
pUserIdSize, 117
libSDP_BuildImagesPreferenceRequest
lite-fw.h, 1151
libSDP_CalculateImageMask
lite-fw.h, 1151
libSDP_CheckValidFirmwareInfo
lite-fw.h, 1151
libSDP_DownloadFW
lite-fw.h, 1152
libSDP_ExtractFirmwareParametersByPath
lite-fw.h, 1152
libSDP_FirmwareInfo
lite-fw.h, 1152
libSDP_GetModelFamily
lite-fw.h, 1152
libSDP_GetVersion
lite-fw.h, 1152
libSDP_getFileType
lite-fw.h, 1152
LibpackProfile3GPP, 93
pAPNClass, 99
pAPNDisabledFlag, 99
pAPNName, 99
pAPNnameSize, 99
pAddrAllocPref, 99
pAuthenticationPref, 99
pGPRSMinimumQoS, 99
pGPRSRequestedQos, 99
pIPv4AddrPref, 99
pIPv6AddPref, 99
pImCnFlag, 99
pPDNInactivTimeout, 99
pPDPtype, 99
pPassword, 99
pPasswordSize, 99
pPcscfAddrUsingDhcp, 99
pPcscfAddrUsingPCO, 99
pPdpAccessConFlag, 99
pPdpContext, 99
pPdpDataCompType, 99
pPdpHdrCompType, 99
pPriDNSIPv4AddPref, 99
pPriDNSIPv6addpref, 99
pPrimaryID, 99
pProfilename, 99
pProfilenameSize, 99

pQosClassID, 99
pSecDNSIPv4AddPref, 99
pSecDNSIPv6addpref, 100
pSecondaryFlag, 100
pSupportEmergencyCalls, 100
pTFTID1Params, 100
pTFTID2Params, 100
pUMTSMinQoS, 100
pUMTSMinQosSigInd, 100
pUMTSReqQoS, 100
pUMTSReqQoSSigInd, 100
pUsername, 100
pUsernameSize, 100
LibpackProfile3GPP2, 100
pAPNClass3GPP2, 105
pAPNEnabled3GPP2, 105
pAllowLinger, 105
pApnString, 105
pApnStringSize, 105
pAppPriority, 105
pAppType, 105
pAuthPassword, 105
pAuthPasswordSize, 105
pAuthProtocol, 105
pAuthRetryCount, 105
pAuthTimeout, 105
pDataMode, 105
pDataRate, 105
pIpcpAckTimeout, 105
pIpcpCreqRetryCount, 105
pIsPcscfAddressNedded, 105
pLcpAckTimeout, 105
pLcpCreqRetryCount, 105
pNegoDnsSrvrPref, 105
pPDNInactivTimeout3GPP2, 105
pPdnType, 105
pPppSessCloseTimer1x, 105
pPppSessCloseTimerDO, 106
pPriV6DnsAddress, 106
pPrimaryV4DnsAddress, 106
pRATType, 106
pSecV6DnsAddress, 106
pSecondaryV4DnsAddress, 106
pUserId, 106
pUserIdSize, 106
libsdp_SetReadBlockSize
lite-fw.h, 1152
libsdp_set_log_func
lite-fw.h, 1152
libsdplogger
lite-fw.h, 1152
lineValue
pack_voice_SLQSVoiceALSSelectLine_t, 468
linkage
altSrcInfo_t, 35
list_type
DMScustSettingList, 63
DMSgetCustomInput, 64
pack_dms_GetCustFeaturesV2_t, 327
listEntries
FMSPrefImageList, 70
listSize
FMSImageList, 69
FMSPrefImageList, 70
lite-fw.h
eFIREHOSE_ERR_SECBOOT_INVALID_CERT_-CHAIN, 1154
eFW_TYPE_CWE, 1153
eFW_TYPE_CWE_NVU, 1153
eFW_TYPE_INVALID, 1153
eFW_TYPE_MBN, 1153
eFW_TYPE_MBN_GOBI, 1153
eFW_TYPE_NVU, 1153
eFW_TYPE_SPK, 1153
eModel_9X15, 1154
eModel_9X30, 1154
eModel_9x06, 1154
eModel_9x07, 1154
eModel_Unknown, 1154
eModel_WP9X15, 1154
eQDL_HW_FLOW_DISABLE, 1154
eQDL_HW_FLOW_ENABLE, 1154
eQDL_HW_FLOW_INIT, 1154
eQDL_HW_FLOW_Unknown, 1154
eQDL_MODE_INIT, 1154
eQDL_MODE_TTYUSB, 1154
eQDL_MODE_UART0, 1154
eQDL_MODE_Unknown, 1154
eSDP_FWDWL_ERR_END, 1154
eSDP_FWDWL_ERR_FAIL, 1154
eSDP_FWDWL_ERR_FW_UPGRADE, 1153
eSDP_FWDWL_ERR_FW_VERSION_FAIL, 1154
eSDP_FWDWL_ERR_GENERAL, 1153
eSDP_FWDWL_ERR_INVALID_DEV, 1153
eSDP_FWDWL_ERR_INVALID_PATH, 1154
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED, 1153
eSDP_FWDWL_ERR_PATH_TOO_LONG, 1153
eSDP_FWDWL_ERR_PRI_FAIL, 1154
eSDP_FWDWL_ERR_SDK, 1153
eSDP_FWDWL_ERR_SDP_TIMEOUT, 1154
eSDP_FWDWL_ERR_SET_CBK, 1153
eSDP_FWDWL_ERR_TIMEOUT, 1154
eSDP_FWDWL_SUCCESS, 1153
lite-fw.h, 1149
IMG_MASK_CLEAR, 1151
IMG_MASK_GENERIC, 1151
IMG_MASK_MDM, 1151
IMG_MASK_PRI, 1151
libSDP_BuildImagesPreferenceRequest, 1151
libSDP_CalculateImageMask, 1151
libSDP_CheckValidFirmwareInfo, 1151
libSDP_DownloadFW, 1152
libSDP_ExtractFirmwareParametersByPath, 1152
libSDP_FirmwareInfo, 1152
libSDP_GetModelFamily, 1152

libSDP_GetVersion, 1152
libSDP_getFileType, 1152
libsdp_SetReadBlockSize, 1152
libsdp_set_log_func, 1152
libsdplogger, 1152
litefw_BuildImagesPreferenceRequest, 1154
litefw_CalculateImageMask, 1155
litefw_CheckValidFirmwareInfo, 1155
litefw_DownloadFW, 1155
litefw_ExtractFirmwareParametersByPath, 1156
litefw_ExtractFirmwarePartNoByPath, 1156
litefw_FirmwareInfo, 1152
litefw_FirmwarePartNo, 1152
litefw_Fw_Type, 1153
litefw_GetModelFamily, 1157
litefw_GetQTLDownloadMode, 1157
litefw_GetQTLHWFlowControl, 1157
litefw_GetVersion, 1158
litefw_Models, 1154
litefw_QDL_FLOW_CONTROLs, 1154
litefw_QDL_MODEs, 1154
litefw_SetQTLDownloadMode, 1158
litefw_SetQTLHWFlowControl, 1159
litefw_SetReadBlockSize, 1159
litefw_fwdwl_error_codes, 1153
litefw_getFileType, 1157
litefw_logsenable, 1158
litefw_set_log_func, 1158
litefw_switch_9x07_to_downloadmode, 1159
litefw_switch_to_BootHoldMode, 1159
litefwlogger, 1153
liteQmiDataBearerMasks
wds.h, 1463
liteServiceClassInformation
voice.h, 1428
litefw_BuildImagesPreferenceRequest
lite-fw.h, 1154
litefw_CalculateImageMask
lite-fw.h, 1155
litefw_CheckValidFirmwareInfo
lite-fw.h, 1155
litefw_DownloadFW
lite-fw.h, 1155
litefw_ExtractFirmwareParametersByPath
lite-fw.h, 1156
litefw_ExtractFirmwarePartNoByPath
lite-fw.h, 1156
litefw_FirmwareInfo
lite-fw.h, 1152
litefw_FirmwarePartNo
lite-fw.h, 1152
litefw_Fw_Type
lite-fw.h, 1153
litefw_GetModelFamily
lite-fw.h, 1157
litefw_GetQTLDownloadMode
lite-fw.h, 1157
litefw_GetQTLHWFlowControl
lite-fw.h, 1157
litefw_GetVersion
lite-fw.h, 1158
litefw_Models
lite-fw.h, 1154
litefw_QDL_FLOW_CONTROLs
lite-fw.h, 1154
litefw_QDL_MODEs
lite-fw.h, 1154
litefw_SetQTLDownloadMode
lite-fw.h, 1158
litefw_SetQTLHWFlowControl
lite-fw.h, 1159
litefw_SetReadBlockSize
lite-fw.h, 1159
litefw_fwdwl_error_codes
lite-fw.h, 1153
litefw_getFileType
lite-fw.h, 1157
litefw_logsenable
lite-fw.h, 1158
litefw_set_log_func
lite-fw.h, 1158
litefw_switch_9x07_to_downloadmode
lite-fw.h, 1159
litefw_switch_to_BootHoldMode
lite-fw.h, 1159
litefwlogger
lite-fw.h, 1153
liteqmi_GetVersion
common.h, 1068
liteqmi_helper_decode7bitAsciiEncString
common.h, 1068
liteqmi_log
common.h, 1068
loc.h
eQMI_LOC_SESS_STATUS_FAILURE, 1166
eQMI_LOC_SESS_STATUS_IN_PROGRESS, 1166
eQMI_LOC_SESS_STATUS_SUCCESS, 1166
eQMI_LOC_SESS_STATUS_TIMEOUT, 1166
loc.h, 1160
LOCEVENTMASKNMEA, 1165
LOCEVENTMASKWIFIREQ, 1166
MAX_TEMP_DATA_LEN, 1166
pack_loc_DeleteAssistData, 1166
pack_loc_EventRegister, 1167
pack_loc_SLQSLOCGetBestAvailPos, 1168
pack_loc_SLQSLOCGetServer, 1168
pack_loc_SLQSLOCInjectPosition, 1169
pack_loc_SLQSLOCInjectSensorData, 1169
pack_loc_SLQSLOCInjectUTCTime, 1169
pack_loc_SLQSLOCSetCradleMountConfig, 1170
pack_loc_SLQSLOCSetServer, 1170
pack_loc_SetExtPowerState, 1167
pack_loc_SetOperationMode, 1167
pack_loc_Start, 1171
pack_loc_Stop, 1171

unpack_loc_BestAvailPos_Ind, 1171
unpack_loc_CradleMountCallback_Ind, 1172
unpack_loc_DeleteAssistData, 1172
unpack_loc_DeleteAssistData_Ind, 1173
unpack_loc_DeleteAssistData_t, 1166
unpack_loc_EngineState_Ind, 1173
unpack_loc_EventNMEA_Ind, 1173
unpack_loc_EventRegister, 1174
unpack_loc_EventTimeSyncCallback_Ind, 1174
unpack_loc_GetServer_Ind, 1174
unpack_loc_GnssSvInfo_Ind, 1175
unpack_loc_InjectPositionCallback_Ind, 1175
unpack_loc_InjectSensorDataCallback_Ind, 1175
unpack_loc_InjectTimeSyncDataCallback_Ind, 1176
unpack_loc_InjectUTCTimeCallback_Ind, 1176
unpack_loc_PositionRpt_Ind, 1176
unpack_loc_SLQSLOCGetBestAvailPos, 1179
unpack_loc_SLQSLOCGetServer, 1179
unpack_loc_SLQSLOCGetServer_t, 1166
unpack_loc_SLQSLOCInjectPosition, 1179
unpack_loc_SLQSLOCInjectPosition_t, 1166
unpack_loc_SLQSLOCInjectSensorData, 1180
unpack_loc_SLQSLOCInjectSensorData_t, 1166
unpack_loc_SLQSLOCInjectUTCTime, 1180
unpack_loc_SLQSLOCInjectUTCTime_t, 1166
unpack_loc_SLQSLOCSetCradleMountConfig, 1180
unpack_loc_SLQSLOCSetCradleMountConfig_t, 1166
unpack_loc_SLQSLOCSetServer, 1181
unpack_loc_SensorStreamingCallback_Ind, 1177
unpack_loc_SetExtPowerConfig_Ind, 1177
unpack_loc_SetExtPowerState, 1177
unpack_loc_SetOperationMode, 1178
unpack_loc_SetOperationMode_Ind, 1178
unpack_loc_SetServer_Ind, 1178
unpack_loc_Start, 1181
unpack_loc_Stop, 1181
loc_BdsSV, 125
id, 125
mask, 125
loc_BdsSVInfo, 125
len, 126
pSV, 126
loc_CellDb, 126
mask, 126
loc_ClkInfo, 126
mask, 127
loc_GnssData, 128
mask, 129
loc_IPv4Config, 132
IPv4Addr, 132
IPv4Port, 132
loc_IPv4Info, 132
address, 133
port, 133
TlvPresent, 133
loc_IPv6Config, 133
IPv6Addr, 133
IPv6Port, 133
loc_IPv6Info, 133
address, 134
port, 134
TlvPresent, 134
loc_LocApplicationInfo, 134
appNameLength, 135
appProviderLength, 135
appVersionLength, 135
appVersionValid, 135
pAppName, 135
pAppProvider, 135
pAppVersion, 135
loc_SV, 139
id, 140
mask, 140
system, 140
loc_SVInfo, 140
len, 140
pSV, 140
loc_URLAddrInfo, 141
urlAddr, 142
loc_accelAcceptReady, 123
batchPerSec, 124
injectEnable, 124
samplesPerBatch, 124
loc_accelTempAcceptReady, 124
batchPerSec, 125
injectEnable, 125
samplesPerBatch, 125
loc_gpsTime, 129
gpsTimeOfWeekMs, 130
gpsWeek, 130
loc_gyroAcceptReady, 130
batchPerSec, 131
injectEnable, 131
samplesPerBatch, 131
loc_gyroTempAcceptReady, 131
batchPerSec, 131
injectEnable, 131
samplesPerBatch, 131
loc_precisionDilution, 135
HDOP, 136
PDOP, 136
VDOP, 136
loc_satelliteInfo, 136
azimuth, 138
elevation, 138
gnssSvId, 138
healthStatus, 138
snr, 138
svInfoMask, 138
svListLen, 138
svStatus, 138
system, 138
validMask, 138

loc_sensorDataUsage, 138
aidingIndicatorMask, 139
usageMask, 139
loc_svUsedforFix, 140
gnssSvUsedList, 141
gnssSvUsedList_len, 141
loc_urlAddr, 141
address, 141
TlvPresent, 141
localTimeOffset
nas_qaQmi3Gpp2TimeZone, 249
logString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 635
logger
common.h, 1065
longName
nas_PLMNNetworkNameData, 246
unpack_nas_SLQSGetPLMNName_t, 728
longNameCI
unpack_nas_SLQSGetPLMNName_t, 728
longNameEn
unpack_nas_SLQSGetPLMNName_t, 728
longNameLen
nas_PLMNNetworkNameData, 246
unpack_nas_SLQSGetPLMNName_t, 728
longNameSB
unpack_nas_SLQSGetPLMNName_t, 728
longNameSpareBits
nas_PLMNNetworkNameData, 246
Longitude
unpack_pds_SLQSGetGPSStateInfo_t, 793
longitude
pack_loc_SLQSLOCInjectPosition_t, 364
loopbackMode
pack_wds_SLQSSSetLoopback_t, 506
loopbackMultiplier
pack_wds_SLQSSSetLoopback_t, 506
LpmFlag
unpack_dms_SLQSDmsSwiGetPCInfo_t, 619
LteBandCapability
unpack_dms_SLQSGetBandCapability_t, 627
lteEmmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 389
lteEmmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 389
lteEsmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 389
lteEsmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 389
LteGsmCellInfo
nas_LTEInfoNeighboringGSM, 204
LteM1BandPref
NASLteM1BandPrefTlv, 302
lteM1BandPref
nas_LteM1BandPrefTlv, 206
LteNB1BandPref
NASLteNB1BandPrefTlv, 303
lteNb1BandPref
nas_LteNb1BandPrefTlv, 208
lteOpMode
nas_EdrxCiotLteMode, 172
nas_LteOpMode, 210
nas_lteOpModeTlv, 211
lteOpModeLen
nas_lteOpModeTlv, 211
LteQci
unpack_qos_swiQosFlow_t, 814
lteRsrpDelta
nas_SLQSSignalStrengthsIndReq, 275
lteRsrpinfo
nas_SLQSSignalStrengthsInformation, 276
lteSSInfo, 142
rsrp, 142
rsrq, 143
rssi, 143
snr, 143
lteSnrDelta
nas_SLQSSignalStrengthsIndReq, 275
lteSnrinfo
nas_SLQSSignalStrengthsInformation, 276
ltersrp
unpack_nas_SLQSGetSignalStrength_t, 736
ltesnr
unpack_nas_SLQSGetSignalStrength_t, 736
lvl
unpack_tmd_SLQSTmdMitigationLvlRptCallback_-
ind_t, 866

M1BandPTlv
NASQmiCbkNasSystemSelPrefInd, 314
m_FwBuildId
CarrierImage_t, 45
m_FwImageId
CarrierImage_t, 45
m_PriBuildId
CarrierImage_t, 45
m_PriImageId
CarrierImage_t, 45
m_nCarrierId
CarrierImage_t, 45
m_nFolderId
CarrierImage_t, 45
m_nStorage
CarrierImage_t, 45
MACIndex
nas_NetworkStatEVDO, 228
MAX_BUILD_ID_LEN
dms.h, 1077
MAX_ERROR_CODE_LEN
imsa.h, 1145
MAX_ICCID_LENGTH
uim.h, 1406
MAX_MSE_TWS_MSG
sms.h, 1331
MAX_NAME_LEN
ims.h, 1133
MAX_NO_OF_SLOTS

uim.h, 1406
MAX_SENSOR_DATA_LEN
loc.h, 1166
MAX_SLOTS_STATUS
uim.h, 1406
MAX_SMS_LIST_SIZE
sms.h, 1331
MAX_TEMP_DATA_LEN
loc.h, 1166
MAXVOICEUSSDLENGTH
voice.h, 1428
MCC
nas_CDMASysInfo, 156
nas_CDMASysInfoExt, 157
nas_currentPLMN, 164
nas_ForbiddenNetworks3GPP, 177
nas_GSMSysInfo, 184
nas_lteOpModeTlv, 211
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 234
nas_QmiNas3GppNetworkInfo, 250
nas_QmiNas3GppNetworkRAT, 251
nas_QmisNasPcsDigit, 252
nas_WCDMASysInfo, 297
unpack_nas_GetServingNetwork_t, 713
MDMCallDuration
connectionStatus, 50
MDMConnStatus
connectionStatus, 50
wds_ConnStatusTlv, 1029
MEID_MAX_SIZE
dms.h, 1077
MEIDString
unpack_dms_GetDeviceSerialNumbers_t, 594
MIN
unpack_dms_GetVoiceNumber_t, 609
MINREQBKLEN
common.h, 1064
MMTlv
unpack_sms_SetNewSMSCallback_ind_t, 823
MNC
nas_CDMASysInfo, 156
nas_currentPLMN, 164
nas_ForbiddenNetworks3GPP, 177
nas_GSMSysInfo, 184
nas_lteOpModeTlv, 211
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 234
nas_QmiNas3GppNetworkInfo, 250
nas_QmiNas3GppNetworkRAT, 251
nas_QmisNasPcsDigit, 252
nas_WCDMASysInfo, 297
unpack_nas_GetServingNetwork_t, 713
MPTlv
NASQmiCbkNasSystemSelPrefInd, 314
MSETlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
MSGID_AND_LEN
common.h, 1064
MSGID_DONT_CARE
common.h, 1064
MTMessageInfo
newMTMessageTlv, 318
MTUSize
pack_swidms_SLQSSwiDmsSetMTU_t, 444
MTUSize3gpp
swidms_mtuSize3gppTlv, 547
manString
pack_dms_SLQSSwiSetHostDevInfo_t, 338
unpack_dms_SLQSSwiGetHostDevInfo_t, 636
manufacturer
unpack_dms_GetManufacturer_t, 600
Mask
pack_wds_SLQSGetDUNCallInfo_t, 499
mask
loc_BdsSV, 125
loc_CellDb, 126
loc_ClkInfo, 127
loc_GnssData, 129
loc_SV, 140
unpack_qos_IPv6TrafCls_t, 796
unpack_qos_Tos_t, 815
max
PackSwiAVMSSettingsPeriodsInfo, 516
max_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 948
max_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 948
max_dist
pack_swiloc_SwiLocSetAutoStart_t, 447
unpack_swiloc_SwiLocGetAutoStart_t, 857
max_dist_reported
unpack_swiloc_SwiLocGetAutoStart_t, 857
max_time
pack_swiloc_SwiLocSetAutoStart_t, 447
unpack_swiloc_SwiLocGetAutoStart_t, 857
max_time_reported
unpack_swiloc_SwiLocGetAutoStart_t, 857
MaxAllowedPktSz
unpack_qos_swiQosFlow_t, 814
MaxChanRxRate
dunchannelRate, 65
MaxChanTxRate
dunchannelRate, 65
maxChannelRXRate
unpack_wds_GetConnectionRate_t, 927
maxChannelTXRate
unpack_wds_GetConnectionRate_t, 927
maxDlBitRate
LibPackQosClassID, 118
maxDownlinkBitrate
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
maxImages

FMSImageIDEntries, 69
maxMitigationLevel
tmd_mitigationDevList, 550
MaxRXChannelRate
unpack_dms_GetDeviceCap_t, 589
maxRxChannelRate
unpack_dms_GetDeviceCapabilities_t, 591
maxSDUSize
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
maxStorageSize
sms_maxStorageSizeResp, 530
MaxTXChannelRate
unpack_dms_GetDeviceCap_t, 589
maxTxChannelRate
unpack_dms_GetDeviceCapabilities_t, 591
maxUlBitRate
LibPackQosClassID, 118
maxUplinkBitrate
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
mcc
nas_CSGID, 163
nas_MNRInfo, 222
nas_netSelectionPref, 224
nas_OperatorPLMNData, 236
nas_PlmnID, 244
pack_nas_SLQSGetPLMNName_t, 376
unpack_nas_GetHomeNetwork_t, 709
mccM
nas_minBasedIMSI, 221
mccT
nas_trueIMSI, 286
mdmCallDurationActive
unpack_wds_SLQSGetDUNCallInfo_t, 952
meanThroughputClass
LibPackGPRSRequestedQoS, 93
wds_GPRSQoS, 1039
meid
unpack_dms_GetSerialNumbers_t, 606
meidSize
unpack_dms_GetDeviceSerialNumbers_t, 594
meidString
unpack_dms_SLQSSwiGetSerialNoExt_t, 638
message
unpack_sms_SLQSGetSMS_t, 826
message_type
NASOTAMessageTlv, 306
messageClass
sms_routeEntry, 533
messageFailureCode
unpack_sms_SendSMS_t, 821
messageFormat
pack_sms_SaveSMS_t, 424
pack_sms_SendSMS_t, 424
sms_sendAsyncsmsParams, 535
unpack_sms_SLQSGetSMS_t, 826
messageID
unpack_sms_SendSMS_t, 821
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
messageIndex
pack_sms_SLQSGetSMS_t, 427
pack_sms_SLQSModifySMSStatus_t, 429
qmiSmsMessageList, 516
sMSMTMessageInfo, 541
messageList
unpack_sms_SLQSGetSMSList_t, 828
messageListSize
unpack_sms_SLQSGetSMSList_t, 828
messageMode
sMSMessageModeInfo, 541
unpack_sms_SLQSWmsMemoryFullCallBack_ind-_t, 838
MessageModeInfo
messageModeTlv, 143
messageModeTlv, 143
MessageModeInfo, 143
TlvPresent, 143
messageSize
pack_sms_SaveSMS_t, 424
pack_sms_SendSMS_t, 424
sms_sendAsyncsmsParams, 535
unpack_sms_SLQSGetSMS_t, 826
messageTag
pack_sms_SLQSModifySMSStatus_t, 429
qmiSmsMessageList, 516
unpack_sms_SLQSGetSMS_t, 826
messageType
sms_routeEntry, 533
MicMute
pack_audio_SLQSSetAudioProfile_t, 323
pack_swiaudio_SLQSSetM2MAVMute_t, 438
unpack_audio_SLQSGetAudioProfile_t, 579
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 839
unpack_swiaudio_SLQSGetM2MAVMute_t, 841
min
PackSwiAVMSSettingsPeriodsInfo, 516
MinPolicedPktSz
unpack_qos_swiQosFlow_t, 814
minSessExp
ims_MinSessExpInfo, 77
minSize
unpack_dms_GetVoiceNumber_t, 609
minute
nas_timeInfo, 285
nas_UniversalTime, 291
mipMode
unpack_wds_GetMobileIP_t, 933
mipStatus
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
mipstatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964

mitigationDevID
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 453
pack_tmd_SLQSTmdGetMitigationLvl_t, 454
pack_tmd_SLQSTmdRegNotMitigationLvl_t, 454
mitigationDevIDLen
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 453
pack_tmd_SLQSTmdGetMitigationLvl_t, 454
pack_tmd_SLQSTmdRegNotMitigationLvl_t, 454
mitigationDevId
tmd_mitigationDevList, 550
mitigationDevIdLen
tmd_mitigationDevList, 550
MitigationDevList
unpack_tmd_SLQSTmdGetMitigationDevList_t, 864
MitigationDevListLen
unpack_tmd_SLQSTmdGetMitigationDevList_t, 864
mnc
nas_CSGID, 163
nas_MNRInfo, 222
nas_netSelectionPref, 224
nas_OperatorPLMNData, 236
nas_PlmnID, 244
pack_nas_SLQSGetPLMNName_t, 376
unpack_nas_GetHomeNetwork_t, 709
mncPcsDigits
nas_CSGID, 163
mobileCountryCode
sMSEtwsPlmnInfo, 540
mobileIP
pack_wds_SLQSSetWdsEventCallback_t, 505
mobileNetworkCode
sMSEtwsPlmnInfo, 540
mode
pack_dms_SetEventReport_t, 332
pack_dms_SetPower_t, 334
pack_loc_SetOperationMode_t, 358
pack_sms_SLQSGetSmsBroadcastConfig_t, 427
pack_sms_SLQSSetSmsBroadcastActivation_t, 430
pack_sms_SLQSSetSmsBroadcastConfig_t, 431
pack_wds_SetMobileIP_t, 493
uim_refreshevent, 568
voice_callInfo, 999
ModePref
nas_ModePrefTlv, 222
NASModePreferenceTlv, 304
modelString
pack_dms_SLQSSwiSetHostDevInfo_t, 338
unpack_dms_SLQSSwiGetHostDevInfo_t, 636
modelid
unpack_dms_GetModelID_t, 601
modelid_str
unpack_dms_GetFirmwareInfo_t, 596
modemMode
nas_CommInfo, 162
modemindex
pack_fms_SetImagesPreference_t, 346
month
nas_timeInfo, 285
nas_UniversalTime, 291
msgCount
sms_messageWaitingInfoContent, 530
msgDelFailureCause
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
msgDelFailureType
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
msgProtocol
sms_msgProtocolResp, 531
msgType
sms_messageWaitingInfoContent, 530
msgWaitInfo
sms_getMsgWaitingInfo, 527
unpack_sms_SLQSWmsMessageWaitingCall-Back_ind_t, 838
msgid
pack_qmi_t, 420
unpack_qmi_t, 794
msgtype
common.h, 1066
Mtu
unpack_wds_SLQSGetRuntimeSettings_t, 955
MultDisc
nas_protocolSubtypeElement, 248
multiplier
unpack_qos_pktErrRate_t, 797
MuxID
qos_BindDataPortMuxID_t, 518

NAI
unpack_wds_GetMobileIPProfile_t, 935
NAS_NAM_NAME_LENGTH
nas.h, 1194
NAS_PLMN_LENGTH
nas.h, 1194
NASAcqOrderPrefTlv, 298
AcqOrderLen, 298
AcqOrderPref, 298
TlvPresent, 298
NASBandPreferenceTlv, 298
band_pref, 298
TlvPresent, 298
NASCiotAcqOrderPrefTlv, 299
CiotAcqOrderLen, 299
CiotAcqOrderPref, 299
TlvPresent, 299
NASCiotLteOpModePrefTlv, 299
CiotLteOpModePref, 299
TlvPresent, 299
NASEmergencyModeTlv, 299
EmerMode, 300
TlvPresent, 300

NASGWAcqOrderPrefTlv, 301
GWAcqOrderPref, 301
TlvPresent, 301
NASLTEBandPreferenceTlv, 301
LTEBandPref, 302
TlvPresent, 302
NASLteM1BandPrefTlv, 302
LteM1BandPref, 302
TlvPresent, 302
NASLteNB1BandPrefTlv, 303
LteNB1BandPref, 303
TlvPresent, 303
NASLteNasReleaseInfoTlv, 302
nas_major, 303
nas_minor, 303
nas_release, 303
TlvPresent, 303
NASModePreferenceTlv, 303
ModePref, 304
TlvPresent, 304
NASNetSelPreferenceTlv, 304
NetSelPref, 304
TlvPresent, 304
NASNr5gBandPrefTlv, 304
Nr5gBandPrefbits1_64, 305
Nr5gBandPrefbits_129_192, 305
Nr5gBandPrefbits_193_256, 305
Nr5gBandPrefbits_65_128, 305
TlvPresent, 305
NASOTAMessageTlv, 305
data_buf, 305
data_len, 306
message_type, 306
TlvPresent, 306
NASPRLPreferenceTlv, 311
PRLPref, 311
TlvPresent, 311
NASPhyCaAggPcellInfo, 306
dl_bw_value, 306
freq, 306
iLTEbandValue, 306
pci, 306
TlvPresent, 306
NASPhyCaAggScellArray, 307
band, 308
cphy_ca_dl_bandwidth, 308
cphy_scell_info_list_len, 308
freq, 308
pci, 308
scell_idx, 308
scell_state, 308
TlvPresent, 308
NASPhyCaAggScellDlBw, 308
dl_bw_value, 308
TlvPresent, 308
NASPhyCaAggScellIndType, 309
freq, 309
pci, 310
scell_state, 310
TlvPresent, 310
NASPhyCaAggScellIndex, 308
scell_idx, 309
TlvPresent, 309
NASPhyCaAggScellInfo, 310
dl_bw_value, 310
freq, 310
iLTEbandValue, 311
pci, 311
scell_state, 311
TlvPresent, 311
NASQmiCbkNasSwiOTAMessageInd, 311
nasRelInfoTlv, 312
otaMsgTlv, 312
timeTlv, 312
NASQmiCbkNasSystemSelPrefInd, 312
AOPTlv, 314
BPTlv, 314
CiotAOPTlv, 314
CiotOpMPTlv, 314
EMTlv, 314
GWAOPTlv, 314
LBPTlv, 314
M1BandPTlv, 314
MPTlv, 314
NB1BandPTlv, 314
NR5gBandPTlv, 314
NSPTlv, 314
PRLPTlv, 314
RPTlv, 314
RatDMTlv, 314
SDPTlv, 314
NASRatDisabledMaskTlv, 314
RatDisabledMask, 315
TlvPresent, 315
NASRoamPreferenceTlv, 315
RoamPref, 315
TlvPresent, 315
NASServDomainPrefTlv, 315
SrvDomainPref, 315
TlvPresent, 315
NASServingSystemInfo, 315
csAttachState, 317
hdrPersonality, 317
psAttachState, 317
radioInterfaceList, 317
radioInterfaceNo, 317
registrationState, 317
selectedNetwork, 317
NASTimeInfoTlv, 317
time, 317
TlvPresent, 317
NB1BandPTlv
NASQmiCbkNasSystemSelPrefInd, 314
NMEAData
unpack_loc_EventNMEA_Ind_t, 682
NR5gBandPTlv

NASQmiCbkNasSystemSelPrefInd, 314
NSPTlv
NASQmiCbkNasSystemSelPrefInd, 314
nUsbNetNum
pack_swidms_SLQSSwiDmsSetUsbNetNum_t, 445
NWQoSStatus
unpack_qos_SLQSQosGetNetworkStatus_t, 800
NWRegStat
unpack_sms_SLQSNWRegInfoCallback_ind_t, 830
naiSize
unpack_wds_GetMobileIPProfile_t, 935
namID
pack_nas_SLQSNasGet3GPP2Subscription_t, 382
voice_airTimer, 980
voice_prefVoiceSO, 1020
voice_roamTimer, 1024
namName
nas_namName, 223
namNameLen
nas_namName, 223
Name
unpack_nas_GetServingNetwork_t, 713
name
unpack_nas_GetHomeNetwork_t, 709
unpack_wds_GetDefaultProfile_t, 931
nameLen
voice_remotePartyName, 1022
namePI
voice_remotePartyName, 1022
nameSize
unpack_nas_GetHomeNetwork3GPP2_t, 708
unpack_nas_GetServingNetwork_t, 714
nameString
pack_dms_SLQSSwiSetOSInfo_t, 338
unpack_dms_SLQSSwiGetOSInfo_t, 637
namelength
unpack_omaDmFotaTlv_t, 779
namesize
unpack_wds_GetDefaultProfile_t, 931
nas.h
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_1, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_10, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_11, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_12, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_125, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_126, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_127, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_13, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_14, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_17, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_18, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_19, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_2, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_20, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_21, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_23, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_24, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_25, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_250, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_26, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_27, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_28, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_29, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_3, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_30, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_31, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_32, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_33, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_34, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_35, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_36, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_37, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_38, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_39, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_4, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_40, 1196

eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_41, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_42, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_43, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_46, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_47, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_48, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_5, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_6, 1195
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_66, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_7, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_71, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_8, 1196
eLITEQMI_LTE_BAND_CLASS_E_UTRA_OPERATING_BAND_9, 1196
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6, 1197
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75, 1197
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED, 1197
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED, 1197
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6, 1197
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75, 1197
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE, 1197
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE, 1197
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE, 1197
LITE_SYS_SRV_DOMAIN_CAMPED, 1195
LITE_SYS_SRV_DOMAIN_CS_ONLY, 1195
LITE_SYS_SRV_DOMAIN_CS_PS, 1195
LITE_SYS_SRV_DOMAIN_NO_SRV, 1195
LITE_SYS_SRV_DOMAIN_PS_ONLY, 1195
nas.h, 1182
NAS_PLMN_LENGTH, 1194
nas_LTEBandPrefExtTlv, 1194
nas_NR5gBandPrefTlv, 1194
pack_nas_GetACCOLC, 1197
pack_nas_GetANAAAAuthenticationStatus, 1198
pack_nas_GetCDMANetworkParameters, 1198
pack_nas_GetHomeNetwork, 1198
pack_nas_GetHomeNetwork3GPP2, 1199
pack_nas_GetNetworkPreference, 1199
pack_nas_GetRFInfo, 1199
pack_nas_GetServingNetwork, 1199
pack_nas_GetServingNetworkCapabilities, 1200
pack_nas_GetSignalStrengths, 1200
pack_nas_InitiateDomainAttach, 1200
pack_nas_PerformNetworkScan, 1201
pack_nas_PerformNetworkScanPCI, 1201
pack_nas_SLQSConfigSigInfo, 1203
pack_nas_SLQSGetErrorRate, 1204
pack_nas_SLQSGetNetworkTime, 1204
pack_nas_SLQSGetOperatorNameData, 1205
pack_nas_SLQSGetPLMNName, 1205
pack_nas_SLQSGetServingSystem, 1205
pack_nas_SLQSGetSignalStrength, 1206
pack_nas_SLQSGetSysInfo, 1206
pack_nas_SLQSGetSysSelectionPref, 1206
pack_nas_SLQSGetSysSelectionPrefExt, 1207
pack_nas_SLQSInitiateNetworkRegistration, 1207
pack_nas_SLQSNASGetForbiddenNetworks, 1209
pack_nas_SLQSNASGeteDRXParams, 1209
pack_nas_SLQSNASGeteDRXParamsExt, 1209
pack_nas_SLQSNASSeteDRXParams, 1211
pack_nas_SLQSNASSwiGetChannelLock, 1212
pack_nas_SLQSNASSwiSetChannelLock, 1213
pack_nas_SLQSNasConfigSigInfo2, 1207
pack_nas_SLQSNasGet3GPP2Subscription, 1208
pack_nas_SLQSNasGetCellLocationInfo, 1208
pack_nas_SLQSNasGetHDRColorCode, 1210
pack_nas_SLQSNasGetRFInfo, 1210
pack_nas_SLQSNasGetSigInfo, 1210
pack_nas_SLQSNasGetTxRxInfo, 1211
pack_nas_SLQSNasIndicationRegisterExt, 1211
pack_nas_SLQSNasSwiIndicationRegister, 1212
pack_nas_SLQSNasSwiModemStatus, 1213
pack_nas_SLQSSetBandPreference, 1213
pack_nas_SLQSSetSignalStrengthsCallback, 1214
pack_nas_SLQSSetSysSelectionPref, 1214
pack_nas_SLQSSetSysSelectionPrefExt, 1214
pack_nas_SLQSSwiGetHDRPersonality, 1215
pack_nas_SLQSSwiGetHDRProtSubtype, 1215
pack_nas_SLQSSwiGetHRPDStats, 1215
pack_nas_SLQSSwiGetLteCQI, 1216

pack_nas_SLQSSwiGetLteSccRxInfo, 1216
pack_nas_SLQSSwiNetworkDebug, 1216
pack_nas_SLQSSwiPSDetach, 1217
pack_nas_SetACCOLC, 1202
pack_nas_SetCDMANetworkParameters, 1202
pack_nas_SetLURejectCallback, 1202
pack_nas_SetNetworkPreference, 1203
pack_nas_SetRFInfoCallback, 1203
pack_nas_SlqsGetLTECphyCAInfo, 1204
unpack_nas_GetACCOLC, 1217
unpack_nas_GetANAAAAuthenticationStatus, 1217
unpack_nas_GetCDMANetworkParameters, 1218
unpack_nas_GetHomeNetwork, 1218
unpack_nas_GetHomeNetwork3GPP2, 1219
unpack_nas_GetNetworkPreference, 1219
unpack_nas_GetRFInfo, 1219
unpack_nas_GetServingNetwork, 1220
unpack_nas_GetServingNetworkCapabilities, 1220
unpack_nas_GetSignalStrengths, 1220
unpack_nas_InitiateDomainAttach, 1221
unpack_nas_InitiateDomainAttach_t, 1194
unpack_nas_PerformNetworkScan, 1221
unpack_nas_SLQSConfigSigInfo, 1225
unpack_nas_SLQSConfigSigInfo_t, 1195
unpack_nas_SLQSGetErrorRate, 1225
unpack_nas_SLQSGetNetworkTime, 1226
unpack_nas_SLQSGetOperatorNameData, 1226
unpack_nas_SLQSGetPLMNName, 1227
unpack_nas_SLQSGetServingSystem, 1227
unpack_nas_SLQSGetSignalStrength, 1227
unpack_nas_SLQSGetSysInfo, 1228
unpack_nas_SLQSGetSysSelectionPref, 1228
unpack_nas_SLQSGetSysSelectionPrefExt, 1229
unpack_nas_SLQSInitiateNetworkRegistration, 1229
unpack_nas_SLQSInitiateNetworkRegistration_t, 1195
unpack_nas_SLQSNASGetForbiddenNetworks, 1232
unpack_nas_SLQSNASGeteDRXParams, 1231
unpack_nas_SLQSNASGeteDRXParamsExt, 1231
unpack_nas_SLQSNASSeteDRXParams, 1235
unpack_nas_SLQSNASSeteDRXParams_t, 1195
unpack_nas_SLQSNASSwiGetChannelLock, 1235
unpack_nas_SLQSNASSwiSetChannelLock, 1237
unpack_nas_SLQSNASSwiSetChannelLock_t, 1195
unpack_nas_SLQSNasConfigSigInfo2, 1229
unpack_nas_SLQSNasConfigSigInfo2_t, 1195
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind, 1230
unpack_nas_SLQSNasGet3GPP2Subscription, 1230
unpack_nas_SLQSNasGetCellLocationInfo, 1230
unpack_nas_SLQSNasGetHDRColorCode, 1232
unpack_nas_SLQSNasGetRFInfo, 1232
unpack_nas_SLQSNasGetSigInfo, 1233
unpack_nas_SLQSNasGetTxRxInfo, 1233
unpack_nas_SLQSNasIndicationRegisterExt, 1233
unpack_nas_SLQSNasIndicationRegisterExt_t, 1195
unpack_nas_SLQSNasNetworkRejectCallback_Ind, 1234
unpack_nas_SLQSNasNetworkTimeCallBack_ind, 1234
unpack_nas_SLQSNasSigInfoCallback_ind, 1235
unpack_nas_SLQSNasSwiIndicationRegister, 1236
unpack_nas_SLQSNasSwiIndicationRegister_t, 1195
unpack_nas_SLQSNasSwiModemStatus, 1236
unpack_nas_SLQSNasSwiOTAMessageCallback_ind, 1236
unpack_nas_SLQSNasSysInfoCallback_ind, 1237
unpack_nas_SLQSNasTimerCallback_ind, 1237
unpack_nas_SLQSSetBandPreference, 1238
unpack_nas_SLQSSetBandPreference_t, 1195
unpack_nas_SLQSSetSignalStrengthsCallback, 1238
unpack_nas_SLQSSetSignalStrengthsCallback_t, 1195
unpack_nas_SLQSSetSysSelectionPref, 1238
unpack_nas_SLQSSetSysSelectionPref_t, 1195
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind, 1239
unpack_nas_SLQSSetSysSelectionPrefExt, 1239
unpack_nas_SLQSSetSysSelectionPrefExt_t, 1195
unpack_nas_SLQSSwiGetHDRPersonality, 1239
unpack_nas_SLQSSwiGetHDRProtSubtype, 1240
unpack_nas_SLQSSwiGetHRPDStats, 1240
unpack_nas_SLQSSwiGetLteCQI, 1240
unpack_nas_SLQSSwiGetLteSccRxInfo, 1241
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind, 1241
unpack_nas_SLQSSwiNetworkDebug, 1241
unpack_nas_SLQSSwiPSDetach, 1242
unpack_nas_SLQSSwiPSDetach_t, 1195
unpack_nas_SLQSSwiRandIndicatorCallback_Ind, 1242
unpack_nas_SetACCOLC, 1221
unpack_nas_SetACCOLC_t, 1195
unpack_nas_SetCDMANetworkParameters, 1222
unpack_nas_SetCDMANetworkParameters_t, 1195
unpack_nas_SetDataCapabilitiesCallback_ind, 1222
unpack_nas_SetEventReportInd, 1222
unpack_nas_SetLURejectCallback, 1223
unpack_nas_SetLURejectCallback_t, 1195
unpack_nas_SetNasLTECphyCaIndCallback_ind, 1223
unpack_nas_SetNetworkPreference, 1224

unpack_nas_SetRFInfoCallback, 1224
unpack_nas_SetRFInfoCallback_t, 1195
unpack_nas_SetRoamingIndicatorCallback_ind, 1224
unpack_nas_SetServingSystemCallback_ind, 1225
unpack_nas_SlqsGetLTECphyCAInfo, 1226
unpack_valid_nas_GetCDMANetworkParameters, 1242
unpack_valid_nas_SLQSGetServingSystem, 1243
unpack_valid_nas_SLQSGetSignalStrength, 1244
unpack_valid_nas_SLQSNasGetSigInfo, 1244
nas_AcqOrderPrefTlv, 144
acqOrdeLen, 144
pAcqOrder, 144
TlvPresent, 145
nas_ActPilotPNElement, 145
ActSetPilotPN, 145
ActSetPilotPNStrength, 145
nas_AddCDMASysInfo, 145
geoSysIdx, 146
regPrd, 146
nas_AddSysInfo, 146
cellBroadcastCap, 146
geoSysIdx, 146
nas_BandPrefInfoTlv, 146
bits_129_192, 147
bits_193_256, 147
bits_1_64, 147
bits_65_128, 147
TlvPresent, 147
nas_BandPrefTlv, 147
BandPref, 149
TlvPresent, 149
nas_CDMAChannel, 150
priChA, 151
priChB, 151
secChA, 151
secChB, 151
nas_CDMAECIOThresh, 151
CDMAECIOThreshListLen, 151
pCDMAECIOThreshList, 151
nas_CDMAInfo, 152
baseId, 152
baseLat, 152
baseLong, 152
nid, 152
refpn, 153
sid, 153
nas_CDMARSSIThresh, 153
CDMARSSIThreshListLen, 153
pCDMARSSIThreshList, 153
nas_CDMASysInfo, 153
baseId, 156
baseLat, 156
baseLong, 156
bsInfoValid, 156
bsPRev, 156
bsPRevValid, 156
ccsSupported, 156
ccsSupportedValid, 156
cdmaSysIdValid, 156
isSysPrlMatch, 156
isSysPrlMatchValid, 156
MCC, 156
MNC, 156
networkID, 156
networkIdValid, 156
pRevInUse, 157
pRevInUseValid, 157
packetZone, 157
packetZoneValid, 157
sysInfoCDMA, 157
systemID, 157
nas_CDMASysInfoExt, 157
imsi_11_12, 157
MCC, 157
nas_CSGID, 162
id, 163
mcc, 163
mnc, 163
mncPcsDigits, 163
rat, 163
nas_CallBarringSysInfo, 149
csBarStatus, 149
psBarStatus, 149
nas_CiotAcqOrderPrefTlv, 159
ciotAcqOrderLen, 160
pCiotAcqOrder, 160
TlvPresent, 160
nas_CiotLteOpModePrefTlv, 160
ciotLteOpModePref, 160
TlvPresent, 160
nas_CommInfo, 160
imsRegState, 162
modemMode, 162
psState, 162
systemMode, 162
temperature, 162
nas_CsgId, 163
csgId, 163
TlvPresent, 163
nas_DRCParams, 169
DRCCover, 170
DRCValue, 170
nas_DataStatusDetail, 165
IPAddress, 166
LastErrCode, 166
nas_DeviceConfigDetail, 168
Chipset, 169
HWVersion, 169
QLIC, 169
Technology, 169
nas_ECIOThresh, 171
ECIOThresListLen, 171
pECIOThresList, 171

nas_EdrxCiotLteMode, 171
lteOpMode, 172
TlvPresent, 172
nas_EdrxCycleLength, 172
cycleLength, 172
TlvPresent, 172
nas_EdrxEnableType, 172
edrxEnabled, 173
TlvPresent, 173
nas_EdrxPagingTimeWindow, 173
edrxPtw, 173
TlvPresent, 173
nas_EdrxRatType, 173
edrxRatType, 174
TlvPresent, 174
nas_EmerModeTlv, 174
EmerMode, 175
TlvPresent, 175
nas_ForbiddenNetworks3GPP, 176
forbiddenNwInstLen, 177
MCC, 177
MNC, 177
TlvPresent, 177
nas_GERANInfo, 177
arfcn, 178
bsic, 178
cellID, 178
insNmrCellInfo, 178
lac, 178
nmrInst, 178
plmn, 178
rxLev, 178
timingAdvance, 178
nas_GSMRSSIThresh, 180
GSMRSSIThreshListLen, 181
pGSMRSSIThreshList, 181
nas_GSMSrvStatusInfo, 181
isPrefDataPath, 182
srvStatus, 182
trueSrvStatus, 182
nas_GSMSysInfo, 182
cellId, 184
cellIdValid, 184
dtmSupp, 184
dtmSuppValid, 184
egprsSupp, 184
egprsSuppValid, 184
lac, 184
lacValid, 184
MCC, 184
MNC, 184
networkIdValid, 184
regRejectInfoValid, 184
rejCause, 184
rejectSrvDomain, 185
sysInfoGSM, 185
nas_GWAcqOrderPrefTlv, 185
GWAcqOrderPref, 185
TlvPresent, 185
nas_HDRECIOThresh, 185
HDRECIOThreshListLen, 186
pHDRECIOThreshList, 186
nas_HDRIOThresh, 186
HDRIOThreshListLen, 186
pHDRIOThreshList, 186
nas_HDRPersonality_Ind_Data, 186
pCurrentPersonality, 187
pPersonalityListLength, 187
pProtocolSubtypeElement, 187
nas_HDRRSSIThresh, 187
HDRRSSIThreshListLen, 187
pHDRRSSIThreshList, 187
nas_HDRSINRThresh, 188
HDRSINRThresListLen, 188
pHDRSINRThresList, 188
nas_HDRSINRThreshold, 188
HDRSINRThreshListLen, 189
pHDRSINRThreshList, 189
nas_HDRSysInfo, 189
hdrActiveProt, 191
hdrActiveProtValid, 191
hdrPersonality, 191
hdrPersonalityValid, 191
is856SysId, 191
is856SysIdValid, 191
isSysPrlMatch, 191
isSysPrlMatchValid, 191
sysInfoHDR, 191
nas_IOThresh, 193
IOThresListLen, 193
pIOThresList, 193
nas_LTEBandPrefExtTlv
nas.h, 1194
nas_LTEBandPrefTlv, 194
LTEBandPref, 195
TlvPresent, 196
nas_LTEInfo, 198
band, 200
bandwidth, 200
emmConnState, 200
emmState, 200
emmSubState, 200
RXChan, 200
TXChan, 200
nas_LTEInfoInterfreq, 201
freqsLen, 201
InfoInterfreq, 201
ueInIdle, 201
nas_LTEInfoIntrafreq, 201
CellParams, 203
cellReselPriority, 203
cellsLen, 203
earfcn, 203
globalCellId, 203
plmn, 203
sIntraSearch, 203

sNonIntraSearch, 203
servingCellId, 203
tac, 203
threshServingLow, 203
ueInIdle, 203
nas_LTEInfoNeighboringGSM, 203
freqsLen, 204
LteGsmCellInfo, 204
ueInIdle, 204
nas_LTEInfoNeighboringWCDMA, 204
freqsLen, 205
ueInIdle, 205
nas_LTEOperationMode, 209
pLTEOperationMode, 209
TlvPresent, 209
nas_LTERSRPThresh, 212
LTERSRPThreshListLen, 212
pLTERSRPThreshList, 212
nas_LTERSRQThresh, 212
LTERSRQThreshListLen, 213
pLTERSRQThreshList, 213
nas_LTERSSIThresh, 213
LTERSSIThreshListLen, 213
pLTERSSIThreshList, 213
nas_LTESNRThresh, 216
LTESNRThresListLen, 216
pLTESNRThresList, 216
nas_LTESNRThreshold, 216
LTESNRThreshListLen, 217
pLTESNRThreshList, 217
nas_LTESigRptCfg, 214
avgPeriod, 214
rptRate, 214
nas_LTESigRptConfig, 214
avgPeriod, 215
rptRate, 215
nas_LTESysInfo, 217
cellId, 219
cellIdValid, 219
lac, 219
lacValid, 219
MCC, 219
MNC, 219
networkIdValid, 219
regRejectInfoValid, 219
rejCause, 219
rejectSrvDomain, 219
sysInfoLTE, 219
tac, 219
tacValid, 219
nas_LteCiotOpModeTlv, 196
campedCiotLteOpMode, 196
TlvPresent, 196
nas_LteM1BandPrefTlv, 205
lteM1BandPref, 206
TlvPresent, 207
nas_LteNb1BandPrefTlv, 207
lteNb1BandPref, 208
TlvPresent, 209
nas_LteOpMode, 209
lteOpMode, 210
TlvPresent, 210
nas_MNRInfo, 221
mcc, 222
mnc, 222
rat, 222
nas_ModePrefTlv, 222
ModePref, 222
TlvPresent, 222
nas_NR5GCellStatusTlv, 230
nr5gCellStatus, 230
TlvPresent, 230
nas_NR5GSerStatTlv, 231
isPrefDataPath, 231
srvStatus, 231
TlvPresent, 231
trueSrvStatus, 231
nas_NR5GSystemInfoTlv, 232
cellId, 234
cellIdValid, 234
lac, 234
lacValid, 234
MCC, 234
MNC, 234
nwIdValid, 234
regRejectInfoValid, 235
rejCause, 235
rejectSrvDomain, 235
roamStatus, 235
roamStatusValid, 235
srvCapValid, 235
srvDomain, 235
srvDomainValid, 235
srvcapability, 235
sysForbidden, 235
sysForbiddenValid, 235
tac, 235
tacValid, 235
TlvPresent, 235
nas_NR5gBandPrefTlv
nas.h, 1194
nas_NetSelPrefTlv, 224
NetSelPref, 224
TlvPresent, 224
nas_NetworkStat1x, 224
ActSetCnt, 226
NeighborSetCnt, 226
pActPilotPNElements, 226
pNeighborSetPilotPN, 226
RX_EC_IO, 226
RX_PWR, 226
SO, 226
State, 226
TX_PWR, 226
nas_NetworkStatEVDO, 226
MACIndex, 228

PER, 228
pSectorID, 228
PilotEnergy, 228
RX_PWR, 228
SNR, 228
SectorIDLen, 228
State, 228
nas_OperatorPLMNData, 235
lac1, 236
lac2, 236
mcc, 236
mnc, 236
PLMNRecID, 236
nas_PLMNNetworkName, 244
numInstance, 244
PLMNNetName, 244
nas_PLMNNetworkNameData, 244
codingScheme, 246
countryInitials, 246
longName, 246
longNameLen, 246
longNameSpareBits, 246
shortName, 246
shortNameLen, 246
shortNameSpareBits, 246
nas_PRLPrefTlv, 246
PRLPref, 247
TlvPresent, 247
nas_PhyCaAggPcellInfo, 237
dl_bw_value, 238
freq, 238
iLTEbandValue, 238
pci, 238
TlvPresent, 238
nas_PhyCaAggScellDlBw, 238
dl_bw_value, 238
TlvPresent, 238
nas_PhyCaAggScellIndType, 239
freq, 239
pci, 239
scell_state, 239
TlvPresent, 239
nas_PhyCaAggScellIndex, 238
scell_idx, 239
TlvPresent, 239
nas_PhyCaAggScellInfo, 240
dl_bw_value, 242
freq, 242
iLTEbandValue, 242
pci, 242
scell_state, 242
TlvPresent, 242
nas_PilotSetData, 242
NumPilots, 242
pPilotSetInfo, 242
nas_PilotSetParams, 243
PilotPN, 243
PilotStrength, 243
PilotType, 243
nas_PlmnID, 243
mcc, 244
mnc, 244
pcsDigit, 244
TlvPresent, 244
nas_QmiNas3GppNetworkInfo, 249
Desription, 250
Forbidden, 250
InUse, 250
MCC, 250
MNC, 250
Preferred, 250
Roaming, 250
nas_QmiNas3GppNetworkRAT, 250
MCC, 251
MNC, 251
RAT, 251
nas_QmisNasPcsDigit, 251
includes_pcs_digit, 252
MCC, 252
MNC, 252
nas_QmisNasSlqsNasPCICellInfo, 252
cellID, 252
freq, 252
GlobalCellID, 252
nasQmisNasPcsDigit, 252
PlmnLen, 252
nas_QmisNasSlqsNasPCIInfo, 253
nasQmisNasSlqsNasPCICellInfo, 253
PCICellInfoLen, 253
rsrp, 253
rsrpRx0, 253
rsrpRx1, 253
rsrq, 254
rsrqRx0, 254
rsrqRx1, 254
nas_RFBandInfoElements, 255
activeBandClass, 256
activeChannel, 256
radioInterface, 256
nas_RFInfoTlv, 261
activeBandClass, 262
activeChannel, 262
radioInterface, 262
radioInterfaceSize, 262
TlvPresent, 262
nas_RSRPThresh, 263
pRSRPThresList, 264
RSRPThresListLen, 264
nas_RSRQThresh, 264
pRSRQThresList, 265
RSRQThresListLen, 265
nas_RSSIThresh, 265
pRSSIThresList, 266
RSSIThresListLen, 266
nas_RankIndicatorTlv, 254
count0, 254

count1, 254
TlvPresent, 254
nas_RatDisabledMaskTlv, 254
ratDisabledMask, 255
TlvPresent, 255
nas_RejectReasonTlv, 255
rejectCause, 255
serviceDomain, 255
TlvPresent, 255
nas_RfBandInfoExtFormat, 256
pInstancesSize, 257
pRfBandInfoExtFormatParam, 257
TlvPresent, 257
nas_RfBandInfoExtFormatElements, 257
activeBand, 257
activeChannel, 257
radioInterface, 257
nas_RfBandInfoList, 258
pInstanceSize, 258
pRFBandInfoParam, 258
nas_RfBandwidthInfo, 258
pInstancesSize, 259
pRfBandwidthInfoParam, 259
TlvPresent, 259
nas_RfBandwidthInfoElements, 259
bandwidth, 259
radioInterface, 259
nas_RfDedicatedBandInfo, 260
pInstancesSize, 260
pRfDedicatedBandInfoParam, 260
TlvPresent, 260
nas_RfDedicatedBandInfoElements, 260
dedicatedBand, 261
radioInterface, 261
nas_RoamPrefTlv, 263
RoamPref, 263
TlvPresent, 263
nas_RxSigInfo, 267
isRadioTuned, 268
rsrp, 268
rxChainIndex, 268
rxPower, 268
nas_SLQSSignalStrengthsIndReq, 273
ecioDelta, 274
ecioThresholdList, 274
ecioThresholdListLen, 274
ioDelta, 275
lteRsrpDelta, 275
lteSnrDelta, 275
rsrqDelta, 275
rxSignalStrengthDelta, 275
sinrDelta, 275
sinrThresholdList, 275
sinrThresholdListLen, 275
nas_SLQSSignalStrengthsInformation, 275
ecioInfo, 276
errorRateInfo, 276
io, 276
lteRsrpinfo, 276
lteSnrinfo, 276
rsrqInfo, 276
rxSignalStrengthInfo, 276
sinr, 276
nas_SLQSSignalStrengthsTlv, 276
sSLQSSignalStrengthsInfo, 277
TlvPresent, 277
nas_SccRxInfo, 269
numInstances, 270
rsrq, 270
sigInfo, 270
snr, 270
TlvPresent, 270
nas_SignalStrengthTlv, 273
radioInterface, 273
signalStrength, 273
TlvPresent, 273
nas_SrvDomainPrefTlv, 277
SrvDomainPref, 277
TlvPresent, 277
nas_SrvStatusInfo, 277
isPrefDataPath, 278
srvStatus, 278
nas_TDSCDMAECIOThresh, 281
nas_TDSCDMARSCPThresh, 281
nas_TDSCDMARSSIThresh, 282
nas_TDSCDMASINRCONFThresh, 282
nas_TDSCDMASINRThresh, 283
nas_UMTSInfo, 286
cellID, 288
ecio, 288
geranInst, 288
GeranInstInfo, 288
lac, 288
plmn, 288
psc, 288
rscp, 288
UMTSInstInfo, 288
uarfcn, 288
umtsInst, 288
nas_UMTSinstInfo, 288
umtsEcio, 289
umtsPsc, 289
umtsRscp, 289
umtsUarfcn, 289
nas_UniversalTime, 290
day, 291
dayOfWeek, 291
hour, 291
minute, 291
month, 291
second, 291
year, 291
nas_WCDMAECIOThresh, 292
nas_WCDMAInfoLTENeighborCell, 292
umtsLTENbrCellLen, 293
wcdmaRRCState, 293

nas_WCDMARSSIThresh, 293
nas_WCDMASysInfo, 294
cellId, 296
cellIdValid, 297
hsCallStatus, 297
hsCallStatusValid, 297
hsInd, 297
hsIndValid, 297
lac, 297
lacValid, 297
MCC, 297
MNC, 297
networkIdValid, 297
psc, 297
pscValid, 297
regRejectInfoValid, 297
rejCause, 297
rejectSrvDomain, 297
sysInfoWCDMA, 297
nas_acqOrderPref, 143
acqOrdeLen, 144
pAcqOrder, 144
nas_callBarStatus, 149
csBarStatus, 150
psBarStatus, 150
nas_cellParams, 157
pci, 158
rsrp, 158
rsrq, 158
rssi, 158
srxlev, 158
nas_ciotAcqOrderPref, 158
ciotAcqOrderLen, 159
pCiotAcqOrder, 159
nas_currentPLMN, 163
MCC, 164
MNC, 164
netDescr, 164
netDescrLength, 164
nas_dataSrvCapabilities, 164
dataCapabilities, 165
dataCapabilitiesLen, 165
nas_detailSvcInfo, 167
hdrHybrid, 168
hdrSrvStatus, 168
isSysForbidden, 168
srvCapability, 168
srvStatus, 168
nas_dirNum, 169
dirNum, 169
dirNumLen, 169
nas_ecioListElement, 170
ecio, 170
radioIf, 171
nas_errorRateListElement, 175
errorRate, 176
radioIf, 176
nas_geranInstInfo, 178
geranArfcn, 179
geranBsicBcc, 179
geranBsicNcc, 179
geranRssi, 179
nas_gsmCellInfo, 179
arfcn, 180
band1900, 180
bsicId, 180
cellIdValid, 180
rssi, 180
srxlev, 180
nas_homeSIDNID, 191
numInstances, 191
SidNid, 191
nas_infoInterFreq, 191
cell_resel_priority, 192
cellInterFreqParams, 192
cells_len, 192
earfcn, 192
threshXHigh, 192
threshXLow, 193
nas_lteBandPrefExt, 193
bits_129_192, 194
bits_193_256, 194
bits_1_64, 194
bits_65_128, 194
nas_lteEARFCN, 196
earfcn0, 197
earfcn1, 197
status, 197
nas_lteGsmCellInfo, 197
cellReselPriority, 198
cells_len, 198
GsmCellInfo, 198
nccPermitted, 198
threshGsmHigh, 198
threshGsmLow, 198
nas_lteOpModeTlv, 210
lteOpMode, 211
lteOpModeLen, 211
MCC, 211
MNC, 211
TlvPresent, 211
nas_ltePCI, 211
earfcn, 211
pci, 211
status, 211
nas_lteRsrpinformation, 211
rsrplevel, 212
nas_lteSnrinformation, 215
snrlevel, 216
nas_lteWcdmaCellInfo, 219
cellReselPriority, 220
cellsLen, 220
threshXhigh, 220
threshXlow, 220
uarfcn, 220
WCDMACellInfo, 220

nas_major
NASLteNasReleaseInfoTlv, 303
nas_minBasedIMSI, 220
imsiM1112, 221
imsiMS1, 221
imsiMS2, 221
mccM, 221
nas_minor
NASLteNasReleaseInfoTlv, 303
nas_namName, 222
namName, 223
namNameLen, 223
nas_netSelectionPref, 223
mcc, 224
mnc, 224
netReg, 224
nas_nmrCellInfo, 228
nmrArfcn, 229
nmrBsic, 229
nmrCellID, 229
nmrLac, 229
nmrPlmn, 229
nmrRxLev, 229
nas_nr5gBandPref, 229
bits_129_192, 230
bits_193_256, 230
bits_1_64, 230
bits_65_128, 230
nas_operatorNameString, 235
PLMNName, 235
nas_operatorPLMNList, 236
numInstance, 237
PLMNData, 237
nas_protocolSubtypeElement, 247
AccessMac, 248
AuthProt, 248
ControlMac, 248
EncryptProt, 248
ForwardMac, 248
IdleState, 248
KeyExchange, 248
MultDisc, 248
PhysicalLayer, 248
ReverseMac, 248
SecProt, 248
VirtStream, 248
nas_qaQmi3Gpp2TimeZone, 248
daylightSavings, 249
leapSeconds, 249
localTimeOffset, 249
nas_release
NASLteNasReleaseInfoTlv, 303
nas_roamIndList, 262
numInstances, 262
radioInterface, 263
roamIndicator, 263
nas_rsrqInformation, 264
radioIf, 264
rsrq, 264
nas_rxInfo, 266
ecio, 267
isRadioTuned, 267
phase, 267
rscp, 267
rsrp, 267
rxPower, 267
nas_rxSignalStrengthListElement, 268
radioIf, 269
rxSignalStrength, 269
nas_servSystem, 271
csAttachState, 272
numRadioInterfaces, 272
psAttachState, 272
radioInterface, 272
regState, 272
selNetwork, 272
nas_serviceProviderName, 270
displayCondition, 270
spn, 270
spnLength, 270
nas_sidNid, 272
nid, 272
sid, 272
nas_sysInfoCommon, 278
isSysForbidden, 280
isSysForbiddenValid, 280
roamStatus, 280
roamStatusValid, 280
srvCapability, 280
srvCapabilityValid, 280
srvDomain, 280
srvDomainValid, 280
nas_timeInfo, 283
day, 285
dayLtSavingAdj, 285
dayOfWeek, 285
hour, 285
minute, 285
month, 285
radioInterface, 285
second, 285
timeZone, 285
TlvPresent, 285
year, 285
nas_trueIMSI, 285
imsiT1112, 286
imsiTS1, 286
imsiTS2, 286
imsiTaddrNum, 286
mccT, 286
nas_txInfo, 286
isInTraffic, 286
txPower, 286
nas_umtsLTENbrCell, 289
cellIsTDD, 290
earfcn, 290

pci, 290
rsrp, 290
rsrq, 290
srxlev, 290
nas_wcdmaCellInfo, 291
cpich_ecno, 292
cpich_rscp, 292
psc, 292
srxlev, 292
nas_wcdmaUARFCN, 297
status, 298
uarfcn, 298
NasGetLTECphyCaInfo, 300
PhyCaAggPcellInfo, 301
PhyCaAggScellArray, 301
PhyCaAggScellDlBw, 301
PhyCaAggScellIndType, 301
PhyCaAggScellIndex, 301
PhyCaAggScellInfo, 301
nasQmisNasPcsDigit
nas_QmisNasSlqsNasPCICellInfo, 252
nasQmisNasSlqsNasPCICellInfo
nas_QmisNasSlqsNasPCIInfo, 253
nasRelInfoTlv
NASQmiCbkNasSwiOTAMessageInd, 312
nccPermitted
nas_lteGsmCellInfo, 198
NeighborSetCnt
nas_NetworkStat1x, 226
netDescr
nas_currentPLMN, 164
netDescrLength
nas_currentPLMN, 164
netInfoLen
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
netReg
nas_netSelectionPref, 224
NetSelPref
nas_NetSelPrefTlv, 224
NASNetSelPreferenceTlv, 304
NetworkID
unpack_nas_SLQSGetServingSystem_t, 733
networkID
nas_CDMASysInfo, 156
networkIdValid
nas_CDMASysInfo, 156
nas_GSMSysInfo, 184
nas_LTESysInfo, 219
nas_WCDMASysInfo, 297
networkInfo
voice_USSDNotificationNetworkInfo, 1026
networkInfoLen
unpack_wds_SLQSGetCurrDataSystemStat_t, 946
NetworkType
currNetworkInfo, 52
wds_currNetworkInfo, 1030
NewIMSRegistration
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 670
NewImsRegStatus
unpack_imsa_SLQSGetIMSARegStatus_t, 666
NewMMTlv
unpack_sms_SetNewSMSCallback_ind_t, 823
newMTMessageTlv, 318
MTMessageInfo, 318
TlvPresent, 318
newPINLen
uim_unblockUIMPIN, 574
newPINVal
uim_unblockUIMPIN, 574
newPasswd
pack_voice_SLQSVoiceSetCallBarringPassword_t, 480
newPasswdAgain
pack_voice_SLQSVoiceSetCallBarringPassword_t, 480
newPin
pack_dms_UIMUnblockPIN_t, 344
newPwd
voice_newPwdData, 1016
newPwdAgain
voice_newPwdData, 1016
newValue
pack_dms_UIMChangePIN_t, 340
nextHeader
LibPackTFTIDParams, 120
nid
nas_CDMAInfo, 152
nas_sidNid, 272
unpack_nas_GetHomeNetwork_t, 709
nmrArfcn
nas_nmrCellInfo, 229
nmrBsic
nas_nmrCellInfo, 229
nmrCellID
nas_nmrCellInfo, 229
nmrInst
nas_GERANInfo, 178
nmrLac
nas_nmrCellInfo, 229
nmrPlmn
nas_nmrCellInfo, 229
nmrRxLev
nas_nmrCellInfo, 229
noReplyTimer
voice_callFWExtInfo, 996
voice_callFWInfo, 997
notifType
unpack_voice_SUPSNotificationCallback_ind_t, 920
notification
unpack_omaDmNotificationsTlv_t, 780
notification_Type

unpack_voice_USSDNotificationCallback_ind_t, 920
notificationType
sMSEtwsMessageInfo, 539
notused
unpack_dms_SetCrashAction_t, 612
Nr5gBandPrefbits1_64
NASNr5gBandPrefTlv, 305
Nr5gBandPrefbits_129_192
NASNr5gBandPrefTlv, 305
Nr5gBandPrefbits_193_256
NASNr5gBandPrefTlv, 305
Nr5gBandPrefbits_65_128
NASNr5gBandPrefTlv, 305
nr5gCellStatus
nas_NR5GCellStatusTlv, 230
num_instances
DMScustSettingList, 63
sms_qaQmi3GPP2BroadcastCfgInfo, 532
sms_qaQmi3GPPBroadcastCfgInfo, 532
numApp
slotInf, 524
uim_slotInfo, 573
numCrashes
crashInformation, 52
numEntries
unpack_dms_SLQSSwiGetFirmwareCurr_t, 633
numFeatures
uim_personalizationStatus, 565
numFiles
uim_registerRefresh, 569
NumFlows
unpack_qos_SLQSSetQosEventCallback_ind_t, 804
numInstance
nas_operatorPLMNList, 237
nas_PLMNNetworkName, 244
numInstances
nas_homeSIDNID, 191
nas_roamIndList, 262
nas_SccRxInfo, 270
sms_getMsgWaitingInfo, 527
unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t, 838
voice_arrAlertingPattern, 983
voice_arrAlertingType, 984
voice_arrAlphaID, 984
voice_arrCalledPartyNum, 985
voice_arrCallEndReason, 986
voice_arrCallInfo, 986
voice_arrConnectPartyNum, 987
voice_arrDiagInfo, 987
voice_arrRedirPartyNum, 988
voice_arrRemotePartyName, 988
voice_arrRemotePartyNum, 989
voice_arrSvcOption, 989
voice_arrUUSInfo, 990
voice_getCallFWExtInfo, 1014
voice_getCallFWInfo, 1014
wds_DomainNameList, 1038
wds_PCSCFFQDNAddressList, 1042
wds_PCSCFIPv4ServerAddressList, 1043
numLen
voice_calledPartyInfo, 992
voice_callFWExtInfo, 996
voice_callFWInfo, 998
voice_callingPartyInfo, 1001
voice_peerNumberInfo, 1018
voice_redirNumInfo, 1021
voice_remotePartyNum, 1023
numOfFiles
uim_refreshevent, 568
numOfRoutes
sms_setRoutesReq, 537
numOpt
wds_DHCPLeaseOptTlv, 1032
wds_DHCPv4OptionList, 1036
wdsDhcpv4OptionList, 1050
numPI
voice_peerNumberInfo, 1018
NumPilots
nas_PilotSetData, 242
numPlan
voice_calledPartyInfo, 992
voice_callFWExtInfo, 997
voice_callingPartyInfo, 1001
voice_connectNumInfo, 1006
voice_peerNumberInfo, 1018
voice_redirNumInfo, 1021
numPresInd
voice_connectNumInfo, 1006
numQosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 803
numRadioInterfaces
nas_servSystem, 272
NumRxFilters
unpack_qos_QosFlowInfo_t, 798
numSI
voice_peerNumberInfo, 1018
numSlot
uim_cardStatus, 557
NumSupUSBComps
unpack_dms_GetUSBComp_t, 608
NumTxFilters
unpack_qos_QosFlowInfo_t, 799
numType
voice_calledPartyInfo, 992
voice_callFWExtInfo, 997
voice_callingPartyInfo, 1001
voice_connectNumInfo, 1007
voice_peerNumberInfo, 1018
voice_redirNumInfo, 1021
number
voice_calledPartyInfo, 992
voice_callFWExtInfo, 996
voice_callFWInfo, 997

voice_callingPartyInfo, 1001
voice_ECTNum, 1011
voice_peerNumberInfo, 1018
voice_redirNumInfo, 1021
numberPlan
voice_callFwdTypeAndPlan, 995
numberType
voice_callFwdTypeAndPlan, 995
nwIdValid
nas_NR5GSystemInfoTlv, 234
NxtHdrProto
unpack_qos_swiQosFilter_t, 810

OKtoRefresh
pack_uim_SLQSUIMRefreshOK_t, 463
OMADMEnable
pack_swioma_SLQSOMADMSetSettingsExt_t, 452
OMADMEnabled
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
unpack_swioma_SLQSOMADMGetSettings_t, 863
OMADMLogEnable
pack_swioma_SLQSOMADMSetSettingsExt_t, 452
OTASPStatus
unpack_voice_OTASPStatusCallback_ind_t, 883
oddEvenInd
voice_calledPartySubAdd, 993
OfflineReason
unpack_dms_GetPower_t, 604
offset
uim_readTransparentInfo, 566
oldPINLen
uim_changeUIMPIN, 558
oldPINVal
uim_changeUIMPIN, 558
oldPasswd
pack_voice_SLQSVoiceSetCallBarringPassword_t, 480
oldValue
pack_dms_UIMChangePIN_t, 340
opMode
unpack_dms_SLQSDmsSwiGetPCInfo_t, 619
opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 366
operatingMode
dms_OperatingModeTlv, 54
OperatingModeTlv
unpack_dms_SetEventReport_ind_t, 613
operation
pack_pds_SetPDSDefaults_t, 411
uim_depersonalizationInformation, 559
OperationMode
unpack_dms_GetPower_t, 604
optCode
wds_DHCPOpt, 1034
wds_DHCPv4Option, 1036
wdsDhcpv4Option, 1050
optList
wds_DHCPLeaseOptTlv, 1032
optListData
wds_DHCPLeaseOptTlv, 1033
optVal
wds_DHCPv4Option, 1036
wdsDhcpv4Option, 1050
optValLen
wds_DHCPOpt, 1034
wds_DHCPv4Option, 1036
wdsDhcpv4Option, 1050
otaMsgTlv
NASQmiCbkNasSwiOTAMessageInd, 312

p1Status
unpack_dms_UIMGetPINStatus_t, 645
p1UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 645
p1VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 645
p2Status
unpack_dms_UIMGetPINStatus_t, 645
p2UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 645
p2VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 645
p3GPP2TimeInfo
unpack_nas_SLQSGetNetworkTime_t, 724
p3GPPTimeInfo
unpack_nas_SLQSGetNetworkTime_t, 724
p3GppNetworkInfoInstances
unpack_nas_PerformNetworkScan_t, 717
p3GppNetworkInstanceSize
unpack_nas_PerformNetworkScan_t, 717
p3gppRelease
pack_wds_SLQSSet3GPPConfigItem_t, 503
pAAASPI
pack_wds_SetMobileIPProfile_t, 496
PACK_WDS_IPV4
wds.h, 1462
PACK_WDS_IPV6
wds.h, 1462
pAMRStatus
pack_voice_SLQSVoiceGetConfig_t, 476
pAPN
PackSwiAVMSSettingsAPNInfo, 515
pAPNClass
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 110
pAPNClass3GPP2
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAPNDisabledFlag
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 110
pAPNEnabled3GPP2
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116

pAPNInfo
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
pAPNName
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 110
pAPNnameSize
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pAVRXAVCHeadroom
audio_RXAVCList, 40
pAVRXAVCSens
audio_RXAVCList, 40
pAccelAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 697
pAccelSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 688
pAccelTempAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 697
pAccelTempSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 688
pAccoIc
unpack_nas_GetACCOLC_t, 702
pAccuracy
unpack_pds_GetPDSDefaults_t, 781
pAcqOrder
nas_acqOrderPref, 144
nas_AcqOrderPrefTlv, 144
pAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPrefExt_t, 746
pActPilotPNElements
nas_NetworkStat1x, 226
pActiveTimer
pack_dms_SLQSSetPowerSaveModeConfig_t, 336
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
pAddCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 775
pAddGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pAddHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pAddLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pAddWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pAddrAllocPref
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 110
pAddress
pack_wds_SetMobileIPProfile_t, 496
pAirTimer
pack_voice_SLQSVoiceGetConfig_t, 476
pAirTimerCnt
unpack_voice_SLQSVoiceGetConfig_t, 908
pAirTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 482
pAirTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 912
pAlertType
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pAlertingPattern
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pAllowLinger
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAlphaID
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
pAlphaIDInfo
unpack_voice_SLQSOriginateUSSD_t, 885
unpack_voice_SLQSVoiceDialCall_t, 887
unpack_voice_SLQSVoiceGetCallBarring_t, 893
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 894
unpack_voice_SLQSVoiceGetCallInfo_t, 897
unpack_voice_SLQSVoiceGetCallWaiting_t, 899
unpack_voice_SLQSVoiceGetCLIP_t, 900
unpack_voice_SLQSVoiceGetCLIR_t, 902
unpack_voice_SLQSVoiceGetCNAP_t, 903
unpack_voice_SLQSVoiceGetCOLP_t, 905
unpack_voice_SLQSVoiceGetCOLR_t, 906
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 911
unpack_voice_SLQSVoiceSetSUPSService_t, 914
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 917
pAltitudeAssumed
unpack_loc_PositionRpt_Ind_t, 695
pAltitudeWrtEllipsoid
pack_pds_SLQSPDSInjectPositionData_t, 416
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pAltitudeWrtMeanSeaLevel
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pAltitudeWrtSealevel
pack_pds_SLQSPDSInjectPositionData_t, 416
pAmrMode
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 653
pAmrOctetAligned
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 653
pAmrWBMode

pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 653
pAmrWBOctetAligned
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pAmrWbEnable
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 653
pApnString
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pApnStringSize
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pApnname
pack_wds_SetDefaultProfile_t, 492
pAppName
loc_LocApplicationInfo, 135
pAppPriority
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAppProvider
loc_LocApplicationInfo, 135
pAppSubType
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 768
pAppType
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAppVersion
loc_LocApplicationInfo, 135
pApplication
pack_nas_SetCDMANetworkParameters_t, 373
pApplicationInfo
pack_loc_Start_t, 369
pArrAlertingPattern
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrAlertingType
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrAlphaID
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrCallEndReason
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrCallInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrCalledPartyNum
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrConnectPartyNum
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrDiagInfo
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrRedirPartyNum
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrRemotePartyName
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrRemotePartyNum
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrSvcOption
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pArrUUSInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
pAuth
pack_wds_SLQSStartDataSession_t, 507
pAuthPassword
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAuthPassword_tSize
LibPackprofile_3GPP2, 116
pAuthPasswordSize
LibpackProfile3GPP2, 105
pAuthProtocol
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAuthRetryCount
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAuthStatus
unpack_nas_GetANAAAAuthenticationStatus_t, 703
pAuthTimeout
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pAuthenticateResult
unpack_uim_SLQSUIMAuthenticate_t, 871
pAuthenticationPref
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pAutoAnsStatus
unpack_voice_SLQSVoiceSetConfig_t, 912
pAutoAnswer
pack_voice_SLQSVoiceGetConfig_t, 476
pack_voice_SLQSVoiceSetConfig_t, 482
pAutoAnswerStat
unpack_voice_SLQSVoiceGetConfig_t, 908
pAutoReboot
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
pAutoSelection
unpack_uim_SLQSUIMGetConfiguration_t, 874
pAutosdm
pack_swioma_SLQSOMADMSetSettings_t, 451
pBandPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPref_t, 744

unpack_nas_SLQSGetSysSelectionPrefExt_t, 746
pBdsSVInfo
pack_loc_Delete_Assist_Data_t, 355
pBinaryUpdateSessionInfo
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 849
pBroadcast
pack_nas_SetCDMANetworkParameters_t, 373
pBroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 431
unpack_sms_SLQSGetSmsBroadcastConfig_t, 827
pBurstDTMFLengths
pack_voice_SLQSVoiceBurstDTMF_t, 470
pCCResType
unpack_voice_SLQSVoiceGetCallBarring_t, 893
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 894
unpack_voice_SLQSVoiceGetCallWaiting_t, 899
unpack_voice_SLQSVoiceGetCLIP_t, 900
unpack_voice_SLQSVoiceGetCLIR_t, 902
unpack_voice_SLQSVoiceGetCNAP_t, 903
unpack_voice_SLQSVoiceGetCOLP_t, 905
unpack_voice_SLQSVoiceGetCOLR_t, 906
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 911
pCCResultType
unpack_voice_SLQSVoiceDialCall_t, 887
unpack_voice_SLQSVoiceSetSUPSService_t, 914
pCCSUPSType
unpack_voice_SLQSVoiceDialCall_t, 887
unpack_voice_SLQSVoiceGetCallBarring_t, 893
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 894
unpack_voice_SLQSVoiceGetCallWaiting_t, 899
unpack_voice_SLQSVoiceGetCLIP_t, 900
unpack_voice_SLQSVoiceGetCLIR_t, 902
unpack_voice_SLQSVoiceGetCNAP_t, 903
unpack_voice_SLQSVoiceGetCOLP_t, 905
unpack_voice_SLQSVoiceGetCOLR_t, 906
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 911
unpack_voice_SLQSVoiceSetSUPSService_t, 914
pCCSuppsType
unpack_voice_SLQSOriginateUSSD_t, 885
pCDMABroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 431
unpack_sms_SLQSGetSmsBroadcastConfig_t, 827
pCDMAChannel
unpack_nas_SLQSNasGet3GPP2Subscription_t, 749
pCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 380
pCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 380
pCDMAECIOThreshList
nas_CDMAECIOThresh, 151
pCDMAFrameErrRate
unpack_nas_SLQSGetErrorRate_t, 723
pCDMAInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 380
pCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pCDMARSSIThreshList
nas_CDMARSSIThresh, 153
pCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 762
pCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
PCICellInfoLen
nas_QmisNasSlqsNasPCIInfo, 253
pCLIPResp
unpack_voice_SLQSVoiceGetCLIP_t, 900
pCLIPstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pCLIRCause
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pCLIRResp
unpack_voice_SLQSVoiceGetCLIR_t, 902
pCLIRType
pack_voice_SLQSVoiceDialCall_t, 471
pCLIRstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pCNAPResp
unpack_voice_SLQSVoiceGetCNAP_t, 903
pCNAPstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pCOLPResp
unpack_voice_SLQSVoiceGetCOLP_t, 905
pCOLPstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pCOLRResp
unpack_voice_SLQSVoiceGetCOLR_t, 906
pCOLRstatus
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
PCSCFAddrPCO
unpack_wds_SLQSGetRuntimeSettings_t, 955
PCSCFFQDNAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 955
pCSCFPortName
pack_ims_SLQSSetRegMgrConfig_t, 352
pCSCFPortNameLen

pack_ims_SLQSSetRegMgrConfig_t, 352
pCSGID
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
PCTlv
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 657
pCUGIndex
unpack_voice_SUPSNotificationCallback_ind_t, 920
pCUGInfo
pack_voice_SLQSVoiceDialCall_t, 471
PCURTlv
unpack_ims_SLQSSMSCfgCallBack_ind_t, 662
pCallBarPasswd
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 917
pCallBarringPasswd
pack_voice_SLQSVoiceSetSUPSService_t, 484
pCallFWNum
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pCallFWTimerVal
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pCallForwardingNumber
pack_voice_SLQSVoiceSetSUPSService_t, 484
pCallFwdInfo
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 917
pCallFwdTypeAndPlan
pack_voice_SLQSVoiceSetSUPSService_t, 484
pCallID
pack_voice_SLQSVoiceManageCalls_t, 478
pack_voice_SLQSVoiceSendFlash_t, 479
pack_voice_SLQSVoiceStartContDTMF_t, 485
unpack_voice_SLQSVoiceBurstDTMF_t, 886
unpack_voice_SLQSVoiceDialCall_t, 887
unpack_voice_SLQSVoiceGetCallBarring_t, 893
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 894
unpack_voice_SLQSVoiceGetCallWaiting_t, 899
unpack_voice_SLQSVoiceGetCLIP_t, 900
unpack_voice_SLQSVoiceGetCLIR_t, 902
unpack_voice_SLQSVoiceGetCNAP_t, 903
unpack_voice_SLQSVoiceGetCOLP_t, 905
unpack_voice_SLQSVoiceGetCOLR_t, 906
unpack_voice_SLQSVoiceSendFlash_t, 909
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 911
unpack_voice_SLQSVoiceSetSUPSService_t, 914
unpack_voice_SLQSVoiceStartContDTMF_t, 914
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
voice_burstDTMFInfo, 991
pCallId
pack_voice_SLQSVoiceAnswerCall_t, 469
pack_voice_SLQSVoiceEndCall_t, 472
unpack_voice_SLQSOriginateUSSD_t, 885
unpack_voice_SLQSVoiceAnswerCall_t, 886
unpack_voice_SLQSVoiceEndCall_t, 888
pCallInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pCallPartySubAdd
pack_voice_SLQSVoiceDialCall_t, 471
pCallType
pack_voice_SLQSVoiceDialCall_t, 471
pCallWaitInd
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pCalledPartyInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pCallerIDInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pCallerNameInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pCallingPartyInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pCampedCiotLteOpMode
unpack_nas_SLQSGetSysInfo_t, 739
pCardResult
unpack_uim_ReadTransparent_t, 869
unpack_uim_SLQSUIMAuthenticate_t, 872
unpack_uim_SLQSUIMGetFileAttributes_t, 874
pCardStatus
unpack_uim_GetCardStatus_t, 868
unpack_uim_SLQSUIMSetStatusChangeCall-Back_ind_t, 877
pCcResultType
unpack_voice_SLQSOriginateUSSD_t, 885
pCellDataMask
pack_pds_ResetPDSData_t, 409
pCellDb
pack_loc_Delete_Assist_Data_t, 355
pChangeDuration
pack_nas_SLQSInitiateNetworkRegistration_t, 377
pChgDuration
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
pCiotAcqOrder
nas_ciotAcqOrderPref, 159
nas_CiotAcqOrderPrefTlv, 160
pCiotAcqOrderPref
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPrefExt_t, 746
pCiotLteOpMode
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
pCiotLteOpModePref
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPrefExt_t, 746
pCiotOpModePref
pack_nas_PerformNetworkScanPCI_t, 371
pClientPerformOperationFlag
pack_swiavms_SLQSAVMSSendSelection_t, 439
pClkInfo
pack_loc_Delete_Assist_Data_t, 355
pCodecSTGain
pack_audio_SLQSSetAudioPathConfig_t, 322

unpack_audio_SLQSGetAudioPathConfig_t, 577
pColorCode
unpack_nas_SLQSNasGetHDRColorCode_t, 754
pConfig
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 849
pConfigAltitudeAssumed
pack_loc_Start_t, 369
pConfigurationMask
pack_uim_SLQSUIMGetConfiguration_t, 459
pConnectNumInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 897
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pConnectionRequest
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
pConnectionRetryTimers
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
pCreateProfileOut
unpack_wds_SLQSCreateProfile_t, 942
pCsgId
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 760
pCurAMRConfig
unpack_voice_SLQSVoiceGetConfig_t, 908
pCurPrefVoiceSO
unpack_voice_SLQSVoiceGetConfig_t, 908
pCurProfile
pack_wds_SLQSCreateProfile_t, 497
pCurVoiceDomainPref
unpack_voice_SLQSVoiceGetConfig_t, 908
pCurVoicePrivacyPref
unpack_voice_SLQSVoiceGetConfig_t, 908
pCurrChannelRateInd
pack_wds_SLQSWdsSetEventReport_t, 509
pCurrDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 509
pCurrImgInfo
unpack_dms_SLQSSwiGetFirmwareCurr_t, 633
pCurrPrefDataSysInd
pack_wds_SLQSWdsSetEventReport_t, 509
pCurrTTYMode
unpack_voice_SLQSVoiceGetConfig_t, 908
pCurrentPersonality
nas_HDRPersonality_Ind_Data, 187
unpack_nas_SLQSSwiGetHDRPersonality_t, 767
pCurrentPrsnlty
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 768
pCustSettingInfo
DMSgetCustomFeatureV2, 64
pCustSettingList
DMSgetCustomFeatureV2, 64
pCustomSCP
pack_nas_SetCDMANetworkParameters_t, 373
pCwtMute
pack_swiaudio_SLQSSetM2MAudioProfile_t, 436
pack_swiaudio_SLQSSetM2MAVMute_t, 438
pCycleLen
pack_nas_SLQSNASSeteDRXParams_t, 388
unpack_nas_SLQSNASGeteDRXParams_t, 752
unpack_nas_SLQSNASGeteDRXParamsExt_t, 753
pDDTMInd
pack_nas_SLQSNasIndicationRegisterExt_t, 386
PDOP
loc_precisionDilution, 136
PDPType
unpack_wds_SLQSGetRuntimeSettings_t, 955
pDRCParams
unpack_nas_SLQSSwiGetHRPDStats_t, 769
pDTMFTXGain
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pData
pack_cat_CATSendEnvelopeCommand_t, 325
pack_cat_CATSendTerminalResponse_t, 326
pDataBearer
unpack_wds_GetDataBearerTechnology_t, 928
pDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 509
pDataCallStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 509
pDataMode
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pDataRate
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pDataSessionStatus
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
pDataSrc
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pDataStatusDetail
unpack_nas_SLQSSwiNetworkDebug_t, 772
pDataSystemStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 509
pDayltSavAdj
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 761
pDefaultPDNEnabled
pack_wds_SLQSSet3GPPConfigItem_t, 503
pDeferTime
pack_swiavms_SLQSAVMSSendSelection_t, 439
pack_swioma_SLQSOMADMSendSelection_t, 449
pDestSMSContent
pack_dms_SLQSSwiSetDyingGaspCfg_t, 337
packgetDyingGaspCfg, 511

pDestSMSNum
pack_dms_SLQSSwiSetDyingGaspCfg_t, 337
packgetDyingGaspCfg, 511
pDetachAction
pack_nas_SLQSSwiPSDetach_t, 408
pDeviceConfigDetail
unpack_nas_SLQSSwiNetworkDebug_t, 772
pDiagInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pDigitBuff
voice_burstDTMFInfo, 991
pDirNum
unpack_nas_SLQSNasGet3GPP2Subscription_t, 749
pDispInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pDormancyStatusInd
pack_wds_SLQSWdsSetEventReport_t, 510
pDualStandByPrefInd
pack_nas_SLQSNasIndicationRegisterExt_t, 386
pDuration
unpack_pds_GetXTRAValidity_t, 785
pDurationDueToOOS
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
pDurationThreshold
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
pECIOThresList
nas_ECIOThresh, 171
pECIOThresh
pack_nas_SLQSConfigSigInfo_t, 375
pECMode
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pECTNum
unpack_voice_SUPSNotificationCallback_ind_t, 920
PER
nas_NetworkStatEVDO, 228
pEVDOPageMonPerChangeInd
pack_wds_SLQSWdsSetEventReport_t, 510
pEarMute
pack_swiaudio_SLQSSetM2MAudioProfile_t, 436
pEarlyWakeupTime
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
pEdrxChangeInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 386
pEdrxCiotLteMode
pack_nas_SLQSNASSeteDRXParams_t, 388
pEdrxEnable
pack_nas_SLQSNASSeteDRXParams_t, 388
unpack_nas_SLQSNASGeteDRXParams_t, 752
unpack_nas_SLQSNASGeteDRXParamsExt_t, 753
pEdrxRAT
pack_nas_SLQSNASGeteDRXParamsExt_t, 383
unpack_nas_SLQSNASGeteDRXParamsExt_t, 753
pEdrxRatType
pack_nas_SLQSNASSeteDRXParams_t, 388
pEhrpdMTUSize
unpack_swidms_SLQSSwiDmsGetMTU_t, 853
pEmerMode
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pEmergencyCategory
pack_voice_SLQSVoiceDialCall_t, 471
pEnableNotification
pack_wds_DHCPv4ClientLeaseChange_t, 486
pEnabled
pack_wds_SetMobileIPProfile_t, 496
pEnabledStatus
unpack_pds_GetPDSState_t, 782
pEncryptData
pack_uim_ReadTransparent_t, 456
pEncryptedData
unpack_uim_ReadTransparent_t, 869
pEncryptedPIN1
pack_uim_VerifyPin_t, 466
unpack_uim_ChangePin_t, 867
unpack_uim_SetPinProtection_t, 870
unpack_uim_UnblockPin_t, 877
unpack_uim_VerifyPin_t, 878
pErrorRateInd
pack_nas_SLQSNasIndicationRegisterExt_t, 386
pExtDispInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pExtDispRecInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pExtErrCode
UnPackGetProfileSettingOut, 970
pExtErrorCode
unpack_wds_SLQSModifyProfile_t, 956
pFailCause
unpack_voice_SLQSVoiceGetCallBarring_t, 893
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 894
unpack_voice_SLQSVoiceGetCallWaiting_t, 899
unpack_voice_SLQSVoiceGetCLIP_t, 901
unpack_voice_SLQSVoiceGetCLIR_t, 902
unpack_voice_SLQSVoiceGetCNAP_t, 903
unpack_voice_SLQSVoiceGetCOLP_t, 905
unpack_voice_SLQSVoiceGetCOLR_t, 906
unpack_voice_SLQSVoiceManageCalls_t, 909
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 911
unpack_voice_SLQSVoiceSetSUPSService_t, 914
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pFailureReason
unpack_wds_SLQSStartDataSession_t, 966
pFileAttributes

unpack_uim_SLQSUIMGetFileAttributes_t, 874
pFixId
unpack_loc_PositionRpt_Ind_t, 695
pFlag
audio_RXPCMIIRFltr, 41
audio_TXPCMIIRFltr, 44
pFlashPayLd
pack_voice_SLQSVoiceSendFlash_t, 479
pFlashType
pack_voice_SLQSVoiceSendFlash_t, 479
pFollowOnDC
sms_sendAsyncsmsParams, 535
pForbiddenNetworks3GPP
unpack_nas_SLQSNASGetForbiddenNetworks_t, 754
pForceOnDC
sms_sendAsyncsmsParams, 535
pForceRev0
pack_nas_SetCDMANetworkParameters_t, 373
pFwAutoCheck
pack_swioma_SLQSOMADMSetSettings_t, 451
pFwAutoSDM
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
pGERANInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pGPRSMinimumQoS
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pGPRSRequestedQos
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pGPSDataMask
pack_pds_ResetPDSData_t, 410
pGPSWeek
unpack_pds_GetXTRAValidity_t, 785
pGPSWeekOffset
unpack_pds_GetXTRAValidity_t, 785
pGSMBER
unpack_nas_SLQSGetErrorRate_t, 723
pGSMCallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pGSMCipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pGSMRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pGSMRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pGSMRSSIThreshList
nas_GSMRSSIThresh, 181
pGSMSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 762
pGSMSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pGWAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pGenerator
pack_swiaudio_SLQSGetM2MAudioProfile_t, 433
pack_swiaudio_SLQSSetM2MAudioProfile_t, 436
pGetCallFWExtInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 895
pGetCallFWInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 895
pGetCustomInput
DMSgetCustomFeatureV2, 64
pGetDyingGaspCfg
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 632
pGetDyingGaspStatistics
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 632
pGetIndicationRegInfo
unpack_sms_SLQSGetIndicationRegister_t, 825
pGetMsgWaitingInfoResp
unpack_sms_SLQSGetMessageWaiting_t, 825
pGetTransLayerInfo
unpack_sms_SLQSGetTransLayerInfo_t, 828
pGetTransNWRegInfo
unpack_sms_SLQSGetTransNWRegInfo_t, 829
pGnssData
pack_loc_Delete_Assist_Data_t, 355
pGpsTime
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pGyroAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 697
pGyroSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 688
pGyroTempAcceptReady
unpack_loc_SensorStreamingCallback_Ind_t, 697
pGyroTempSamplesAccepted
unpack_loc_InjectSensorDataCallback_Ind_t, 688
pHA2002bis
pack_wds_SetMobileIPParameters_t, 494
pHAAuthenticator
pack_wds_SetMobileIPParameters_t, 494
pHASPI
pack_wds_SetMobileIPProfile_t, 496
pHDRECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pHDRECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pHDRECIOThreshList
nas_HDRECIOThresh, 186
pHDRIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 381

pHDRIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pHDRIOThreshList
nas_HDRIOThresh, 186
pHDRNewUATIAssInd
pack_nas_SLQSNasIndicationRegisterExt_t, 386
pHDRPackErrRate
unpack_nas_SLQSGetErrorRate_t, 723
pHDRPersInd
unpack_nas_SLQSSwiHDRPersonalityCallback_-Ind_t, 771
pHDRRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pHDRRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pHDRRSSIThreshList
nas_HDRRSSIThresh, 187
pHDRSINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pHDRSINRThresList
nas_HDRSINRThresh, 188
pHDRSINRThresh
pack_nas_SLQSConfigSigInfo_t, 375
pack_nas_SLQSNasConfigSigInfo2_t, 381
pHDRSINRThreshList
nas_HDRSINRThreshold, 189
pHDRSessionCloseInd
pack_nas_SLQSNasIndicationRegisterExt_t, 386
pHDRSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 762
pHDRSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pHTTPStatus
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
pHaltSubscription
unpack_uim_SLQSUIMGetConfiguration_t, 874
pHeading
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pHeadingUnc
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pHomeSIDNID
unpack_nas_SLQSNasGet3GPP2Subscription_t, 749
pHorCirConf
unpack_loc_BestAvailPos_Ind_t, 678
pHorConfidence
unpack_loc_PositionRpt_Ind_t, 695
pHorEllpConf
unpack_loc_BestAvailPos_Ind_t, 678
pHorReliability
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pHorUncCircular
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pHorUncEllipseOrientAzimuth
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pHorUncEllipseSemiMajor
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pHorUncEllipseSemiMinor
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pHorizontalAccuracyLvl
pack_loc_Start_t, 369
pHorizontalConfidence
pack_pds_SLQSPDSInjectPositionData_t, 416
pHorizontalUncCircular
pack_pds_SLQSPDSInjectPositionData_t, 416
pHotSwapStatus
unpack_uim_GetCardStatus_t, 868
pHrpdMTUSize
unpack_swidms_SLQSSwiDmsGetMTU_t, 853
pHwConfig
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 506
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 965
PI
voice_calledPartyInfo, 992
voice_callerIDInfo, 994
voice_callFWExtInfo, 997
voice_callingPartyInfo, 1001
voice_redirNumInfo, 1021
PIFACEId
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 435
pIMSDomain
pack_ims_SLQSSetIMSUserConfig_t, 349
unpack_ims_SLQSGetIMSUserConfig_t, 651
pIMSDomainLen
pack_ims_SLQSSetIMSUserConfig_t, 349
unpack_ims_SLQSGetIMSUserConfig_t, 651
pIMSTestMode
pack_ims_SLQSSetRegMgrConfig_t, 352
unpack_ims_SLQSGetRegMgrConfig_t, 655
pIOThresList
nas_IOThresh, 193
pIOThresh
pack_nas_SLQSConfigSigInfo_t, 375
pIPv4AddrInfo
unpack_loc_GetServer_Ind_t, 685
pIPv4AddrPref
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pIPv4Config
pack_loc_SLQSLOCSetServer_t, 368
pIPv6AddPref
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111

pIPv6AddrInfo
unpack_loc_GetServer_Ind_t, 685
pIPv6Config
pack_loc_SLQSLOCSetServer_t, 368
pIgnoreHotSwapSwitch
pack_uim_SLQSUIMPowerUp_t, 461
pImCnFlag
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pImageList
pack_fms_SetImagesPreference_t, 346
unpack_fms_GetImagesPreference_t, 648
pIndicationToken
pack_uim_ChangePin_t, 455
pack_uim_ReadTransparent_t, 456
pack_uim_SetPinProtection_t, 457
pack_uim_SLQSUIMAuthenticate_t, 458
pack_uim_SLQSUIMGetFileAttributes_t, 460
pack_uim_UnblockPin_t, 465
pack_uim_VerifyPin_t, 466
unpack_uim_ChangePin_t, 867
unpack_uim_ReadTransparent_t, 869
unpack_uim_SetPinProtection_t, 870
unpack_uim_SLQSUIMAuthenticate_t, 872
unpack_uim_SLQSUIMGetFileAttributes_t, 874
unpack_uim_UnblockPin_t, 878
unpack_uim_VerifyPin_t, 878
pInfo
pack_voice_AnswerUSSD_t, 467
pack_voice_OriginateUSSD_t, 467
pInstanceSize
nas_RfBandInfoList, 258
pInstancesSize
nas_RfBandInfoExtFormat, 257
nas_RfBandwidthInfo, 259
nas_RfDedicatedBandInfo, 260
pInterfaceCfg
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 854
pIntermediateReportState
pack_loc_Start_t, 369
pInterval
unpack_pds_GetPDSDefaults_t, 781
unpack_pds_GetXTRAAutomaticDownload_t, 784
pIpcpAckTimeout
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 116
pIpcpCreqRetryCount
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pIsPcscfAddressNedded
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pKeyReferenceID
pack_uim_ChangePin_t, 455
pack_uim_SetPinProtection_t, 457
pack_uim_UnblockPin_t, 465
pack_uim_VerifyPin_t, 466
PLMNData
nas_operatorPLMNList, 237
PLMNName
nas_operatorNameString, 235
PLMNNetName
nas_PLMNNetworkName, 244
PLMNRecID
nas_OperatorPLMNData, 236
pLTEAttachProfile
pack_wds_SLQSSet3GPPConfigItem_t, 503
pLTEAttachProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 503
pLTEBandPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pLTEBandPrefExt
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pLTECphyCa
pack_nas_SLQSNasIndicationRegisterExt_t, 386
pLTEInfo
unpack_nas_SLQSNasSwiModemStatus_t, 764
pLTEInfoInterfreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pLTEInfoIntrafreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pLTEInfoNeighboringGSM
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pLTEInfoNeighboringWCDMA
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pLTEOperationMode
nas_LTEOperationMode, 209
unpack_nas_SLQSNasGetRFInfo_t, 755
pLTERSRPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTERSRPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTERSRPThreshList
nas_LTERSRPThresh, 212
pLTERSRQDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTERSRQThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTERSRQThreshList
nas_LTERSRQThresh, 213
pLTERSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTERSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTERSSIThreshList
nas_LTERSSIThresh, 213
pLTESNRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTESNRThresList
nas_LTESNRThresh, 216
pLTESNRThresh

pack_nas_SLQSConfigSigInfo_t, 375
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTESNRThreshList
nas_LTESNRThreshold, 217
pLTESigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 762
pLTESigRptCfg
pack_nas_SLQSConfigSigInfo_t, 375
pLTESigRptConfig
pack_nas_SLQSNasConfigSigInfo2_t, 381
pLTESrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pLTEVoiceSupportSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pLatitude
pack_pds_SLQSPDSInjectPositionData_t, 416
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pLcpAckTimeout
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pLcpCreqRetryCount
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pLeapSeconds
unpack_loc_PositionRpt_Ind_t, 695
pLineCtrlInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pLinktimer
pack_sms_SendSMS_t, 424
sms_sendAsyncsmsParams, 535
pLongitude
pack_pds_SLQSPDSInjectPositionData_t, 416
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pLteCiotOpModeTlv
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pLteEARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 390
unpack_nas_SLQSNASSwiGetChannelLock_t, 763
pLteM1BandPref
pack_nas_PerformNetworkScanPCI_t, 371
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pLteNB1BandPref
pack_nas_PerformNetworkScanPCI_t, 371
pLteNb1BandPref
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pLteOpMode
pack_nas_SLQSNASGeteDRXParamsExt_t, 383
unpack_nas_SLQSNASGeteDRXParamsExt_t, 753
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t, 760
pLteOpModeTlv
unpack_nas_PerformNetworkScan_t, 717
pLtePCI
pack_nas_SLQSNASSwiSetChannelLock_t, 390
unpack_nas_SLQSNASSwiGetChannelLock_t, 763
pMCC
unpack_nas_GetHomeNetwork3GPP2_t, 708
pMICGainSelect
unpack_audio_SLQSGetAudioPathConfig_t, 577
pMIPStatusInd
pack_wds_SLQSWdsSetEventReport_t, 510
pMNAAA
pack_wds_SetMobileIPProfile_t, 496
pMNC
unpack_nas_GetHomeNetwork3GPP2_t, 708
pMNCIncPCSDigStat
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
pMNHA
pack_wds_SetMobileIPProfile_t, 496
pMNRInfo
pack_nas_SLQSInitiateNetworkRegistration_t, 377
pMTUSize3gpp
unpack_swidms_SLQSSwiDmsGetMTU_t, 853
pMagneticDeviation
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 695
pManagedRoamingInd
pack_nas_SLQSNasIndicationRegisterExt_t, 386
pMaxStorageSizeReq
pack_sms_SLQSSmsGetMaxStorageSize_t, 432
pMaxStorageSizeResp
unpack_sms_SLQSSmsGetMaxStorageSize_t, 833
pMessage
pack_sms_SaveSMS_t, 424
pack_sms_SendSMS_t, 425
sms_sendAsyncsmsParams, 535
pMessageIndex
pack_sms_SLQSDeleteSMS_t, 426
unpack_sms_SaveSMS_t, 821
pMessageMode
pack_sms_SLQSDeleteSMS_t, 426
pack_sms_SLQSGetSMS_t, 427
pack_sms_SLQSGetSMSList_t, 428
pack_sms_SLQSModifySMSStatus_t, 429
sms_maxStorageSizeReq, 529
pMessageProtocol
unpack_sms_SLQSSmsGetMessageProtocol_t, 833
pMessageTag
pack_sms_SLQSDeleteSMS_t, 426
pMicMute

pack_swiaudio_SLQSSetM2MAudioProfile_t, 436
pMinBasedIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 749
pMinIntervalTime
pack_loc_Start_t, 369
pMinSessionExpiryTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pMncPcsDigitStatus
pack_nas_SLQSInitiateNetworkRegistration_t, 377
pMncPcsStatus
pack_nas_SLQSGetPLMNName_t, 376
pMode
pack_wds_SetMobileIPParameters_t, 494
pModePref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pMuxID
pack_qos_BindDataPort_t, 421
pNAI
pack_wds_SetMobileIPProfile_t, 496
pNAMNameInfo
unpack_nas_SLQSNasGet3GPP2Subscription_t, 749
pNID
unpack_nas_GetHomeNetwork3GPP2_t, 708
pNITZInformation
unpack_nas_SLQSGetOperatorNameData_t, 725
pNR5GCellStatus
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pNR5GSerStatTlv
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pNR5GSystemInfoTlv
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pNSEnable
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pNSSAudioCtrl
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pNSSRelease
unpack_voice_VoiceInfoRecCallback_ind_t, 923
PNTlv
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 657
pNamID
pack_voice_SLQSVoiceGetConfig_t, 476
pName
pack_wds_SetDefaultProfile_t, 492
unpack_nas_GetHomeNetwork3GPP2_t, 708
pNegoDnsSrvrPref
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pNeighborSetPilotPN
nas_NetworkStat1x, 226
pNetSelPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pNetworkMode
pack_pds_SLQSGetAGPSConfig_t, 414
pack_pds_SLQSSetAGPSConfig_t, 417
pNetworkRejectInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pNetworkStat1x
unpack_nas_SLQSSwiNetworkDebug_t, 772
pNetworkStatEVDO
unpack_nas_SLQSSwiNetworkDebug_t, 772
pNetworkTimeInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pNetworkType
pack_nas_PerformNetworkScanPCI_t, 371
pNewPwdData
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pNotifStore
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
pNotification
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 850
pNotificationStore
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
pNr5gBandPref
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pNumberOfPhySlot
unpack_uim_SLQSUIMGetSlotsStatus_t, 875
pNw2DescDisp
unpack_nas_GetHomeNetwork3GPP2_t, 708
pNw2DescEnc
unpack_nas_GetHomeNetwork3GPP2_t, 708
pNw2DescLen
unpack_nas_GetHomeNetwork3GPP2_t, 708
pNw2MCC
unpack_nas_GetHomeNetwork3GPP2_t, 708
pNw2MNC
unpack_nas_GetHomeNetwork3GPP2_t, 708
pNw2Name
unpack_nas_GetHomeNetwork3GPP2_t, 708
pOTASPStatus
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pObjectVer
unpack_nas_SLQSSwiNetworkDebug_t, 772
pOffLength
unpack_voice_DTMFEventCallback_ind_t, 881
pOnLength
unpack_voice_DTMFEventCallback_ind_t, 881
pOpaqueIdentifier

unpack_loc_InjectSensorDataCallback_Ind_t, 688
pOperation
unpack_pds_GetPDSDefaults_t, 781
pOperatorNameString
unpack_nas_SLQSGetOperatorNameData_t, 725
pOperatorPLMNList
unpack_nas_SLQSGetOperatorNameData_t, 725
pOptList
wds_DHCPv4OptionList, 1036
wdsDhcpv4OptionList, 1051
pOptVal
wds_DHCPOpt, 1034
pPCIInfo
unpack_nas_PerformNetworkScan_t, 717
pPCMParams
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 435
pPCSCFPort
unpack_ims_SLQSGetRegMgrConfig_t, 655
pPCSInstance
unpack_nas_PerformNetworkScan_t, 717
pPCSInstanceSize
unpack_nas_PerformNetworkScan_t, 717
pPDNInactivTimeout
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPDNInactivTimeout3GPP2
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pPDPtype
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPLMNNetworkName
unpack_nas_SLQSGetOperatorNameData_t, 725
pPRLPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 407
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pPWD
PackSwiAVMSSettingsAPNInfo, 515
pPackageID
pack_swiavms_SLQSAVMSSendSelection_t, 439
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 850
pPagingTimeWindow
pack_nas_SLQSNASSeteDRXParams_t, 388
unpack_nas_SLQSNASGeteDRXParams_t, 752
unpack_nas_SLQSNASGeteDRXParamsExt_t, 753
pPass
pack_wds_SLQSStartDataSession_t, 507
pPassword
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pack_wds_SetDefaultProfile_t, 492
pPasswordSize
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPcscfAddrUsingDhcp
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPcscfAddrUsingPCO
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPdnType
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pPdpAccessConFlag
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPdpContext
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPdpDataCompType
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPdpHdrCompType
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPeriodInfo
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
pPeriodicUpdateTimer
pack_dms_SLQSSetPowerSaveModeConfig_t, 336
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
pPeripheralEndPointID
pack_qos_BindDataPort_t, 421
pPeroidsInfo
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
pPersonalityListLength
nas_HDRPersonality_Ind_Data, 187
unpack_nas_SLQSSwiGetHDRPersonality_t, 767
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 768
pPersonalizationStatus
unpack_uim_SLQSUIMGetConfiguration_t, 874
pPhoneCtxtURI
pack_ims_SLQSSetIMSSMSConfig_t, 348
unpack_ims_SLQSGetIMSSMSConfig_t, 650
pPhoneCtxtURILen
pack_ims_SLQSSetIMSSMSConfig_t, 348
unpack_ims_SLQSGetIMSSMSConfig_t, 650
pPilotSetData
unpack_nas_SLQSSwiGetHRPDStats_t, 769
pPilotSetInfo
nas_PilotSetData, 242
pPlmnId
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 760
pPollingTimer
pack_swiavms_SLQSAVMSSetSettings_t, 441

pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
pPositionSource
pack_pds_SLQSPDSInjectPositionData_t, 416
pPppSessCloseTimer1x
LibpackProfile3GPP2, 105
LibPackprofile_3GPP2, 117
pPppSessCloseTimerDO
LibpackProfile3GPP2, 106
LibPackprofile_3GPP2, 117
pPrecisionDilution
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pPrefVoiceDomain
pack_voice_SLQSVoiceSetConfig_t, 482
pPrefVoicePrivacy
pack_voice_SLQSVoiceGetConfig_t, 476
pPrefVoiceSO
pack_voice_SLQSVoiceGetConfig_t, 476
pack_voice_SLQSVoiceSetConfig_t, 482
pPrefVoiceSOStatus
unpack_voice_SLQSVoiceSetConfig_t, 912
pPreference
unpack_pds_GetXTRANetwork_t, 784
pPriCSCFPort
pack_ims_SLQSSetRegMgrConfig_t, 352
pPriCSCFPortName
unpack_ims_SLQSGetRegMgrConfig_t, 655
pPriCSCFPortNameLen
unpack_ims_SLQSGetRegMgrConfig_t, 655
pPriDNSIPv4AddPref
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPriDNSIPv6addpref
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPriV6DnsAddress
LibpackProfile3GPP2, 106
LibPackprofile_3GPP2, 117
pPrimaryHA
pack_wds_SetMobileIPProfile_t, 496
pPrimaryID
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pPrimaryV4DnsAddress
LibpackProfile3GPP2, 106
LibPackprofile_3GPP2, 117
pProfileID
unpack_wds_SLQSCreateProfile_t, 942
pProfileId
pack_wds_SLQSCreateProfile_t, 497
pack_wds_SLQSModifyProfile_t, 502
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 505
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 506
pProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 503
pProfileSettings
unpack_wds_SLQSGetProfileSettings_t, 952
pProfileType
pack_wds_SLQSCreateProfile_t, 497
pack_wds_SLQSModifyProfile_t, 502
pProfilename
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pProfilenameSize
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pProtoSubTypElmnt
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 768
pProtocol
pack_nas_SetCDMANetworkParameters_t, 373
pProtocolSubtypeElement
nas_HDRPersonality_Ind_Data, 187
unpack_nas_SLQSSwiGetHDRPersonality_t, 767
pPsmEnableState
pack_dms_SLQSSetPowerSaveModeConfig_t, 336
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
pQosClassID
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pRAT
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 408
pRATINstance
unpack_nas_PerformNetworkScan_t, 717
pRATInstanceSize
unpack_nas_PerformNetworkScan_t, 717
pRATType
LibpackProfile3GPP2, 106
LibPackprofile_3GPP2, 117
pRFBandInfoParam
nas_RfBandInfoList, 258
PRIString
unpack_dms_GetFirmwareRevision_t, 597
unpack_dms_GetFirmwareRevisions_t, 598
PRLEnable
pack_swioma_SLQSOMADMSetSettingsExt_t, 452
PRLInd
unpack_nas_SLQSGetServingSystem_t, 733
PRLPTlv
NASQmiCbkNasSystemSelPrefInd, 314
PRLPref
nas_PRLPrefTlv, 247
NASPRLPreferenceTlv, 311
pRSRPThresList
nas_RSRPThresh, 264
pRSRPThresh
pack_nas_SLQSConfigSigInfo_t, 375
pRSRQThresList
nas_RSRQThresh, 265
pRSRQThresh

pack_nas_SLQSConfigSigInfo_t, 375
pRSSIThresList
nas_RSSIThresh, 266
pRSSIThresh
pack_nas_SLQSConfigSigInfo_t, 375
pRTPRTCPInactTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pRXAGCList
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pRXAIG
audio_RXAGCList, 39
pRXAVCAGCSwitch
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pRXAVCList
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pRXChain0Info
unpack_nas_SLQSNasGetTxRxInfo_t, 757
pRXChain1Info
unpack_nas_SLQSNasGetTxRxInfo_t, 757
pRXComprSlope
audio_RXAGCList, 39
pRXComprThres
audio_RXAGCList, 39
pRXDroppedCount
unpack_wds_GetPacketStatistics_t, 937
pRXExpSlope
audio_RXAGCList, 39
pRXExpThres
audio_RXAGCList, 39
pRXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 937
pRXOkBytesCount
unpack_wds_GetPacketStatistics_t, 937
pRXPCMIIRFltr
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pRXPacketErrors
unpack_wds_GetPacketStatistics_t, 937
pRXPacketOverflows
unpack_wds_GetPacketStatistics_t, 937
pRXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 937
pRXStaticGain
audio_RXAGCList, 39
pRXTotalBytes
unpack_wds_GetByteTotals_t, 926
pRadioInterface
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 761
pRandomizationWindow
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
pRankIndicatorInd
pack_nas_SLQSNasSwiIndicationRegister_t, 390
pRatDisabledMask
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pReRegPeriod
pack_wds_SetMobileIPParameters_t, 494
pReRegTraffic
pack_wds_SetMobileIPParameters_t, 494
pReadResult
unpack_uim_ReadTransparent_t, 869
pReason
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pReasonMask
unpack_dms_GetOfflineReason_t, 603
pRecurrenceType
pack_loc_Start_t, 369
pRedirNumInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pRefreshEvent
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 876
pRegCallStatInfoEvt
sms_getIndicationReg, 527
sms_setIndicationReg, 536
pRegDTMFEvents
pack_voice_SLQSVoiceIndicationRegister_t, 477
pRegInd
sms_getTransLayerInfo, 528
pRegMgrConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 347
pRegStatus
sms_getTransNWRegInfo, 528
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
pRegTransLayerInfoEvt
sms_getIndicationReg, 527
sms_setIndicationReg, 536
pRegTransNWRegInfoEvt
sms_getIndicationReg, 527
sms_setIndicationReg, 536
pRegVoicePrivacyEvents
pack_voice_SLQSVoiceIndicationRegister_t, 477
pRejectReason
pack_swiavms_SLQSAVMSSendSelection_t, 439
pack_swioma_SLQSOMADMSendSelection_t, 449
pRemainingRetries
unpack_uim_ChangePin_t, 867
unpack_uim_SetPinProtection_t, 870
unpack_uim_SLQSUIMDepersonalization_t, 872
unpack_uim_UnblockPin_t, 878
unpack_uim_VerifyPin_t, 878
pRemotePartyName
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pRemotePartyNum
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pReportChannelRate
pack_wds_SLQSGetDUNCallInfo_t, 499

pReportConnStatus
pack_wds_SLQSGetDUNCallInfo_t, 499
pReportDataBearerTech
pack_wds_SLQSGetDUNCallInfo_t, 499
pReportDormStatus
pack_wds_SLQSGetDUNCallInfo_t, 499
pReqSettings
pack_wds_SLQSGetRuntimeSettings_t, 501
pRequestOptionList
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 506
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 965
pRequestedTag
pack_sms_SLQSGetSMSList_t, 428
pRetryInterval
pack_wds_SetMobileIPParameters_t, 494
pRetryLimit
pack_wds_SetMobileIPParameters_t, 494
pRetryMessage
sms_sendAsyncsmsParams, 535
pRetryMessageId
sms_sendAsyncsmsParams, 535
pRevInUse
nas_CDMASysInfo, 157
pRevInUseValid
nas_CDMASysInfo, 157
pRevTunneling
pack_wds_SetMobileIPProfile_t, 496
pRfBandInfoExtFormat
unpack_nas_SLQSNasGetRFInfo_t, 755
pRfBandInfoExtFormatParam
nas_RfBandInfoExtFormat, 257
pRfBandwidthInfo
unpack_nas_SLQSNasGetRFInfo_t, 755
pRfBandwidthInfoParam
nas_RfBandwidthInfo, 259
pRfDedicatedBandInfo
unpack_nas_SLQSNasGetRFInfo_t, 755
pRfDedicatedBandInfoParam
nas_RfDedicatedBandInfo, 260
pRingBackTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pRingingTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pRoamPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 408
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pRoamTimer
pack_voice_SLQSVoiceGetConfig_t, 476
pRoamTimerCnt
unpack_voice_SLQSVoiceGetConfig_t, 908
pRoamTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 482
pRoamTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 912
pRoaming
pack_nas_SetCDMANetworkParameters_t, 373
pRscp
unpack_nas_SLQSNasSigInfoCallback_ind_t, 762
PSDomain
unpack_nas_GetServingNetwork_t, 714
pSID
unpack_nas_GetHomeNetwork3GPP2_t, 708
pSIODataPort
pack_qos_BindDataPort_t, 421
pSIPConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 347
pSIPLocalPort
pack_ims_SLQSSetSIPConfig_t, 353
unpack_ims_SLQSGetSIPConfig_t, 656
PSMCfgChangeInfo
pack_dms_SetIndicationRegister_t, 333
pSMSAttemptedFlag
packgetDyingGaspStatistics, 512
pSMSCAddress
pack_sms_SetSMSCAddress_t, 425
unpack_sms_GetSMSCAddress_t, 820
pSMSCType
pack_sms_SetSMSCAddress_t, 425
unpack_sms_GetSMSCAddress_t, 820
pSMSConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 347
pSMSFormat
pack_ims_SLQSSetIMSSMSConfig_t, 348
unpack_ims_SLQSGetIMSSMSConfig_t, 650
pSMSOverIPNwInd
pack_ims_SLQSSetIMSSMSConfig_t, 348
unpack_ims_SLQSGetIMSSMSConfig_t, 650
PSMStatus
pack_dms_SetIndicationRegister_t, 333
pSPC
pack_nas_SetCDMANetworkParameters_t, 373
pack_wds_SetMobileIPParameters_t, 494
pSV
loc_BdsSVInfo, 126
loc_SVInfo, 140
pSVInfo
pack_loc_Delete_Assist_Data_t, 355
pSarRFState
unpack_sar_SLQSGetRfSarState_t, 819
pSatelliteInfo
unpack_loc_GnssSvInfo_Ind_t, 686
pScanResult
unpack_nas_PerformNetworkScan_t, 717
pScanType
pack_nas_PerformNetworkScanPCI_t, 371
pSccRxInfo
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 770
pScrAmrEnable
pack_ims_SLQSSetIMSVoIPConfig_t, 351

unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pScrAmrWbEnable
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pSecDNSIPv4AddPref
LibpackProfile3GPP, 99
LibPackprofile_3GPP, 111
pSecDNSIPv6addpref
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 111
pSecV6DnsAddress
LibpackProfile3GPP2, 106
LibPackprofile_3GPP2, 117
pSecondaryFlag
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 111
pSecondaryHA
pack_wds_SetMobileIPProfile_t, 496
pSecondaryV4DnsAddress
LibpackProfile3GPP2, 106
LibPackprofile_3GPP2, 117
pSectorID
nas_NetworkStatEVDO, 228
pSendSmsParams
pack_sms_SLQSSendAsyncSMS_t, 429
pSensorDataUsage
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pServerAddress
pack_pds_SLQSSetAGPSConfig_t, 417
unpack_pds_SLQSGetAGPSConfig_t, 789
pServerPort
pack_pds_SLQSSetAGPSConfig_t, 417
unpack_pds_SLQSGetAGPSConfig_t, 789
pServerURL
pack_pds_SLQSSetAGPSConfig_t, 417
unpack_pds_SLQSGetAGPSConfig_t, 789
pServerURLLength
pack_pds_SLQSSetAGPSConfig_t, 417
unpack_pds_SLQSGetAGPSConfig_t, 789
pServiceClass
pack_voice_SLQSVoiceSetSUPSService_t, 484
pServiceOption
sms_sendAsyncsmsParams, 535
pServingSystemInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pSessionExpiryTimer
pack_ims_SLQSSetIMSVoIPConfig_t, 351
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
pSessionType
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
pSetIndicationRegReq
pack_sms_SLQSSetIndicationRegister_t, 430
pSetRoutesReq
pack_sms_SLQSSmsSetRoutes_t, 432
pSettingResp
unpack_ims_SLQSGetIMSSMSConfig_t, 650
unpack_ims_SLQSGetIMSUserConfig_t, 651
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
unpack_ims_SLQSGetRegMgrConfig_t, 655
unpack_ims_SLQSGetSIPConfig_t, 656
unpack_ims_SLQSSetIMSSMSConfig_t, 658
unpack_ims_SLQSSetIMSUserConfig_t, 658
unpack_ims_SLQSSetIMSVoIPConfig_t, 659
unpack_ims_SLQSSetRegMgrConfig_t, 659
unpack_ims_SLQSSetSIPConfig_t, 660
pSigCompEnabled
pack_ims_SLQSSetSIPConfig_t, 352
unpack_ims_SLQSGetSIPConfig_t, 656
pSigIndReq
pack_nas_SLQSSetSignalStrengthsCallback_t, 392
pSignalInfo
unpack_voice_VoiceInfoRecCallback_ind_t, 923
pSignalStrengthInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pSmsOnIms
sms_sendAsyncsmsParams, 535
pSmsStorage
unpack_sms_SLQSSwiGetSMSStorage_t, 834
pSourceIP
LibPackTFTIDParams, 120
pSpeedHorizontal
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pSpeedUnc
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pSpeedVertical
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pSpeedVerticalUnc
unpack_loc_BestAvailPos_Ind_t, 678
pSrvDomainPref
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 408
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSGetSysSelectionPrefExt_t, 747
pSrvOpt
unpack_voice_SLQSVoiceGetCallInfo_t, 897
pSrvRegRestriction
pack_nas_SLQSSetSysSelectionPref_t, 397
pack_nas_SLQSSetSysSelectionPrefExt_t, 408
pSrvcProviderName
unpack_nas_SLQSGetOperatorNameData_t, 725
pStage0Val
audio_RXPCMIIRFltr, 41
audio_TXPCMIIRFltr, 44
pStage1Val
audio_RXPCMIIRFltr, 41
audio_TXPCMIIRFltr, 44
pStage2Val
audio_RXPCMIIRFltr, 41
audio_TXPCMIIRFltr, 44
pStage3Val

audio_RXPCMIIRFltr, 41
audio_TXPCMIIRFltr, 44
pStage4Val
audio_RXPCMIIRFltr, 41
audio_TXPCMIIRFltr, 44
pStageCnt
audio_RXPCMIIRFltr, 41
audio_TXPCMIIRFltr, 44
pStatMask
pack_wds_GetPacketStatistics_t, 488
pSubscribeTimer
pack_ims_SLQSSetSIPConfig_t, 353
unpack_ims_SLQSGetSIPConfig_t, 656
pSubscriptionInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pSupportEmergencyCalls
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 111
pSupportedBitmasks
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 854
pSuppressSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pSuppsNotifEvents
pack_voice_SLQSVoiceIndicationRegister_t, 477
pSvUsedforFix
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pSvcClass
pack_voice_SLQSVoiceGetCallBarring_t, 473
pack_voice_SLQSVoiceGetCallForwardingStatus-_t, 474
pack_voice_SLQSVoiceGetCallWaiting_t, 475
unpack_voice_SLQSVoiceGetCallBarring_t, 893
unpack_voice_SLQSVoiceGetCallWaiting_t, 899
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pSvcType
pack_voice_SLQSVoiceDialCall_t, 472
pSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pSysInfoNoChange
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pSystemSelectionInd
pack_nas_SLQSNasIndicationRegisterExt_t, 387
pTDSCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMAECIOThreshList
nas_TDSCDMAECIOThresh, 281
pTDSCDMARSCPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMARSCPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMARSCPThreshList
nas_TDSCDMARSCPThresh, 281
pTDSCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMARSSIThreshList
nas_TDSCDMARSSIThresh, 282
pTDSCDMASINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMASINRThresh
pack_nas_SLQSNasConfigSigInfo2_t, 381
pTDSCDMASINRThreshList
nas_TDSCDMASINRThresh, 283
pTDSCDMASigInfoExt
unpack_nas_SLQSNasSigInfoCallback_ind_t, 762
pTFTID1Params
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 111
pTFTID2Params
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 112
pTTYConfigStatus
unpack_voice_SLQSVoiceSetConfig_t, 912
pTTYMode
pack_voice_SLQSVoiceGetConfig_t, 476
pack_voice_SLQSVoiceSetConfig_t, 482
pTXAGCList
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pTXAIG
audio_TXAGCList, 42
pTXAVCSwitch
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pTXComprSlope
audio_TXAGCList, 42
pTXComprThres
audio_TXAGCList, 42
pTXDroppedCount
unpack_wds_GetPacketStatistics_t, 937
pTXExpSlope
audio_TXAGCList, 42
pTXExpThres
audio_TXAGCList, 42
pTXGain
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pTXInfo
unpack_nas_SLQSNasGetTxRxInfo_t, 757
pTXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 937
pTXOkBytesCount
unpack_wds_GetPacketStatistics_t, 937
pTXPCMIIRFltr
pack_audio_SLQSSetAudioPathConfig_t, 322
unpack_audio_SLQSGetAudioPathConfig_t, 577
pTXPacketErrors
unpack_wds_GetPacketStatistics_t, 937
pTXPacketOverflows
unpack_wds_GetPacketStatistics_t, 937

pTXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 937
pTXStaticGain
audio_TXAGCList, 42
pTXTotalBytes
unpack_wds_GetByteTotals_t, 926
pTdscdmaBandPref
pack_nas_SLQSSetSysSelectionPref_t, 398
pack_nas_SLQSSetSysSelectionPrefExt_t, 408
pTech
pack_wds_SLQSStartDataSession_t, 507
pTechnologyMask
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pTempReport
pack_dms_SwiSetEventReport_t, 339
pTimeSrc
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pTimeStamp
pack_pds_SLQSPDSInjectPositionData_t, 416
packgetDyingGaspStatistics, 512
pTimeType
pack_pds_SLQSPDSInjectPositionData_t, 416
pTimeUnc
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pTimeZone
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 761
pTimeout
unpack_pds_GetPDSDefaults_t, 781
pTimer
pack_nas_SLQSNasSwiIndicationRegister_t, 390
pTimerSIPReg
pack_ims_SLQSSetSIPConfig_t, 353
unpack_ims_SLQSGetSIPConfig_t, 656
pTimerT1
pack_ims_SLQSSetSIPConfig_t, 353
unpack_ims_SLQSGetSIPConfig_t, 656
pTimerT2
pack_ims_SLQSSetSIPConfig_t, 353
unpack_ims_SLQSGetSIPConfig_t, 656
pTimerTf
pack_ims_SLQSSetSIPConfig_t, 353
unpack_ims_SLQSGetSIPConfig_t, 656
pTimerVal
pack_voice_SLQSVoiceSetSUPSService_t, 484
pTimestampUtc
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pTrackingStatus
unpack_pds_GetPDSState_t, 782
pTransLayerInfo
sms_getTransLayerInfo, 528
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 835
pTransferStatInd
pack_wds_SLQSGetDUNCallInfo_t, 499
pack_wds_SLQSWdsSetEventReport_t, 510
pTransferStatusReport
sms_setRoutesReq, 537
pTrueIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 749
pUATI
unpack_nas_SLQSSwiGetHRPDStats_t, 769
pUIMStatusReport
pack_dms_SwiSetEventReport_t, 339
pUMTSCellID
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pUMTSInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pUMTSMinQoS
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 112
pUMTSMinQosSigInd
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 112
pUMTSReqQoS
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 112
pUMTSReqQoSSigInd
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 112
pURL
unpack_loc_GetServer_Ind_t, 685
pURLAddr
pack_loc_SLQSLOCSetServer_t, 368
pUSSDInfo
unpack_voice_SLQSOriginateUSSD_t, 885
pUSSInfo
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
pUUSInFo
pack_voice_SLQSVoiceDialCall_t, 472
pUUSInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 898
pUimAutoSwitchActSlot
unpack_dms_SLQSDmsSwiGetUimSelection_t, 622
pUimSlotsStatus
unpack_uim_SLQSUIMGetSlotsStatus_t, 875
pUname
PackSwiAVMSSettingsAPNInfo, 515
pUsbMTUSize
unpack_swidms_SLQSSwiDmsGetMTU_t, 853
pUser
pack_wds_SLQSStartDataSession_t, 507
pUserConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 347
pUserData
sms_sendAsyncsmsParams, 535
pUserId
LibpackProfile3GPP2, 106

LibPackprofile_3GPP2, 117
pUserIdSize
LibpackProfile3GPP2, 106
LibPackprofile_3GPP2, 117
pUsername
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 112
pack_wds_SetDefaultProfile_t, 492
pUsernameSize
LibpackProfile3GPP, 100
LibPackprofile_3GPP, 112
pVerboseFailReasonType
unpack_wds_SLQSStartDataSession_t, 967
pVerboseFailureReason
unpack_wds_SLQSStartDataSession_t, 967
pVertConfidence
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_PositionRpt_Ind_t, 696
pVertReliability
unpack_loc_BestAvailPos_Ind_t, 679
unpack_loc_PositionRpt_Ind_t, 696
pVertUnc
unpack_loc_BestAvailPos_Ind_t, 679
unpack_loc_PositionRpt_Ind_t, 696
pVerticalConfidence
pack_pds_SLQSPDSInjectPositionData_t, 416
pVerticalUnc
pack_pds_SLQSPDSInjectPositionData_t, 416
pVoIPConfigEvents
pack_ims_SLQSImsConfigIndicationRegister_t, 347
pVoiceDomainPref
pack_voice_SLQSVoiceGetConfig_t, 476
pVoiceDomainPrefStatus
unpack_voice_SLQSVoiceSetConfig_t, 912
pVoicePrivacy
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
unpack_voice_SLQSVoiceGetCallInfo_t, 898
pVoltReport
pack_dms_SwiSetEventReport_t, 339
pVolume
pack_swiaudio_SLQSSetM2MAudioProfile_t, 436
pWAMSParaChanged
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
pWCDMABER
unpack_nas_SLQSGetErrorRate_t, 723
pWCDMACallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pWCDMACipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pWCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 382
pWCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 382
pWCDMAECIOThreshList
nas_WCDMAECIOThresh, 292
pWCDMAInfoLTENeighborCell
unpack_nas_SLQSNasGetCellLocationInfo_t, 751
pWCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 382
pWCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 382
pWCDMARSSIThreshList
nas_WCDMARSSIThresh, 294
pWCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 763
pWCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 740
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 740
unpack_nas_SLQSSysInfoCallback_ind_t, 776
pWcdmaUARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 390
unpack_nas_SLQSNASSwiGetChannelLock_t, 763
pWifiState
pack_pds_SLQSSetPositionMethodState_t, 418
pXid
unpack_loc_BestAvailPos_Ind_t, 679
pXtraDataState
pack_pds_SLQSSetPositionMethodState_t, 418
pXtraTimeState
pack_pds_SLQSSetPositionMethodState_t, 418
pack_audio_SLQSGetAudioPathConfig
audio.h, 1054
pack_audio_SLQSGetAudioPathConfig_t, 318
Item, 319
Profile, 319
pack_audio_SLQSGetAudioProfile
audio.h, 1054
pack_audio_SLQSGetAudioProfile_t, 319
Generator, 319
pack_audio_SLQSGetAudioVolTLBConfig
audio.h, 1055
pack_audio_SLQSGetAudioVolTLBConfig_t, 320
Generator, 320
Item, 320
Profile, 320
Volume, 320
pack_audio_SLQSSetAudioPathConfig
audio.h, 1055
pack_audio_SLQSSetAudioPathConfig_t, 320
pCodecSTGain, 322
pECMode, 322
pNSEnable, 322
pRXAGCList, 322
pRXAVCList, 322
pTXAGCList, 322
pTXAVCSwitch, 322
pTXGain, 322
Profile, 322
pack_audio_SLQSSetAudioProfile

audio.h, 1056
pack_audio_SLQSSetAudioProfile_t, 322
EarMute, 323
Generator, 323
MicMute, 323
Profile, 323
Volume, 323
pack_audio_SLQSSetAudioVolTLBConfig
audio.h, 1056
pack_audio_SLQSSetAudioVolTLBConfig_t, 324
Generator, 324
Item, 324
Profile, 324
VolValue, 324
Volume, 324
pack_cat_CATSendEnvelopeCommand
cat.h, 1060
pack_cat_CATSendEnvelopeCommand_t, 324
cmdID, 325
dataLen, 325
pData, 325
pack_cat_CATSendTerminalResponse
cat.h, 1060
pack_cat_CATSendTerminalResponse_t, 325
dataLen, 326
pData, 326
refID, 326
pack_cat_SetCATEventCallback
cat.h, 1060
pack_cat_SetCATEventCallback_t, 326
eventMask, 326
pack_dms_ActivateAutomatic
dms.h, 1077
pack_dms_ActivateAutomatic_t, 327
actCode, 327
pack_dms_GetActivationState
dms.h, 1078
pack_dms_GetBandCapability
dms.h, 1078
pack_dms_GetCrashAction
dms.h, 1078
pack_dms_GetCustFeature
dms.h, 1079
pack_dms_GetCustFeaturesV2
dms.h, 1079
pack_dms_GetCustFeaturesV2_t, 327
cust_id, 327
list_type, 327
Tlvresult, 327
pack_dms_GetDeviceCap
dms.h, 1079
pack_dms_GetDeviceCapabilities
dms.h, 1080
pack_dms_GetDeviceHardwareRev
dms.h, 1080
pack_dms_GetDeviceMfr
dms.h, 1081
pack_dms_GetDeviceSerialNumbers
dms.h, 1081
pack_dms_GetFSN
dms.h, 1082
pack_dms_GetFirmwareInfo
dms.h, 1081
pack_dms_GetFirmwareRevision
dms.h, 1082
pack_dms_GetFirmwareRevisions
dms.h, 1082
pack_dms_GetHardwareRevision
dms.h, 1083
pack_dms_GetIMSI
dms.h, 1083
pack_dms_GetManufacturer
dms.h, 1083
pack_dms_GetModelID
dms.h, 1084
pack_dms_GetNetworkTime
dms.h, 1084
pack_dms_GetOfflineReason
dms.h, 1084
pack_dms_GetPRLVersion
dms.h, 1085
pack_dms_GetPower
dms.h, 1085
pack_dms_GetSerialNumbers
dms.h, 1085
pack_dms_GetUSBComp
dms.h, 1086
pack_dms_GetVoiceNumber
dms.h, 1086
pack_dms_ResetToFactoryDefaults
dms.h, 1086
pack_dms_ResetToFactoryDefaults_t, 328
spc, 328
pack_dms_SLQSDmsSwiGetPCInfo
dms.h, 1090
pack_dms_SLQSDmsSwiGetResetInfo
dms.h, 1091
pack_dms_SLQSDmsSwiGetUimSelection
dms.h, 1091
pack_dms_SLQSDmsSwiIndicationRegister
dms.h, 1091
pack_dms_SLQSDmsSwiIndicationRegister_t, 335
resetInfoInd, 335
pack_dms_SLQSGetBandCapability
dms.h, 1092
pack_dms_SLQSGetERIFile
dms.h, 1092
pack_dms_SLQSGetPowerSaveModeConfig
dms.h, 1092
pack_dms_SLQSSetPowerSaveModeConfig
dms.h, 1093
pack_dms_SLQSSetPowerSaveModeConfig_t, 335
pActiveTimer, 336
pPeriodicUpdateTimer, 336
pPsmEnableState, 336
pack_dms_SLQSSwiClearDyingGaspStatistics

dms.h, 1093
pack_dms_SLQSSwiGetCrashInfo
dms.h, 1094
pack_dms_SLQSSwiGetCrashInfo_t, 336
clear, 336
pack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 1094
pack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 1094
pack_dms_SLQSSwiGetFirmwareCurr
dms.h, 1095
pack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 1095
pack_dms_SLQSSwiGetHostDevInfo
dms.h, 1095
pack_dms_SLQSSwiGetOSInfo
dms.h, 1096
pack_dms_SLQSSwiGetSerialNoExt
dms.h, 1096
pack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 1096
pack_dms_SLQSSwiSetDyingGaspCfg_t, 336
pDestSMSContent, 337
pDestSMSNum, 337
pack_dms_SLQSSwiSetHostDevInfo
dms.h, 1097
pack_dms_SLQSSwiSetHostDevInfo_t, 337
hostID, 338
manString, 338
modelString, 338
plasmaIDString, 338
swVerString, 338
pack_dms_SLQSSwiSetOSInfo
dms.h, 1097
pack_dms_SLQSSwiSetOSInfo_t, 338
nameString, 338
versionString, 338
pack_dms_SLQSUIMGetState
dms.h, 1097
pack_dms_SetActivationStatusCallback
dms.h, 1087
pack_dms_SetActivationStatusCallback_t, 328
activationState, 328
pack_dms_SetCrashAction
dms.h, 1087
pack_dms_SetCrashAction_t, 328
crashAction, 329
pack_dms_SetCustFeature
dms.h, 1088
pack_dms_SetCustFeature_t, 329
DHCPRelayEnabled, 331
DisableIMSI, 331
GPSLPM, 331
GPSSel, 331
GpsEnable, 331
IPFamSupport, 331
IsVoiceEnabled, 331
RMAutoConnect, 331
SMSSupport, 331
pack_dms_SetCustFeaturesV2
dms.h, 1088
pack_dms_SetCustFeaturesV2_t, 331
cust_id, 332
cust_value, 332
Tlvresult, 332
value_length, 332
pack_dms_SetEventReport
dms.h, 1088
pack_dms_SetEventReport_t, 332
mode, 332
pack_dms_SetFirmwarePreference
dms.h, 1089
pack_dms_SetIndicationRegister
dms.h, 1089
pack_dms_SetIndicationRegister_t, 332
PSMCfgChangeInfo, 333
PSMStatus, 333
RptIMSCapability, 333
pack_dms_SetPower
dms.h, 1089
pack_dms_SetPower_t, 333
mode, 334
Tlvresult, 334
pack_dms_SetUSBComp
dms.h, 1090
pack_dms_SetUSBComp_t, 334
Tlvresult, 334
USBComp, 334
pack_dms_SwiSetEventReport
dms.h, 1098
pack_dms_SwiSetEventReport_t, 338
pTempReport, 339
pUIMStatusReport, 339
pVoltReport, 339
pack_dms_SwiUimSelect
dms.h, 1098
pack_dms_SwiUimSelect_t, 339
uim_select, 339
pack_dms_UIMChangePIN
dms.h, 1098
pack_dms_UIMChangePIN_t, 340
id, 340
newValue, 340
oldValue, 340
pack_dms_UIMGetControlKeyStatus
dms.h, 1099
pack_dms_UIMGetControlKeyStatus_t, 340
facility, 341
pack_dms_UIMGetICCID
dms.h, 1099
pack_dms_UIMGetICCID_t, 341
ParamPresenceMask, 341
Tlvresult, 341
pack_dms_UIMGetPINStatus
dms.h, 1100
pack_dms_UIMSetControlKeyProtection

dms.h, 1100
pack_dms_UIMSetControlKeyProtection_t, 341
facility, 342
facilityCk, 342
facilityState, 342
pack_dms_UIMSetPINProtection
dms.h, 1100
pack_dms_UIMSetPINProtection_t, 342
bEnable, 343
id, 343
value, 343
pack_dms_UIMUnblockControlKey
dms.h, 1101
pack_dms_UIMUnblockControlKey_t, 343
facility, 343
facilityCk, 343
pack_dms_UIMUnblockPIN
dms.h, 1101
pack_dms_UIMUnblockPIN_t, 343
id, 344
newPin, 344
pukValue, 344
pack_dms_UIMVerifyPIN
dms.h, 1101
pack_dms_UIMVerifyPIN_t, 344
id, 344
value, 344
pack_dms_ValidateSPC
dms.h, 1102
pack_fms_GetImagesPreference
fms.h, 1129
pack_fms_GetImagesPreference_t, 344
Tlvresult, 345
pack_fms_GetStoredImages
fms.h, 1130
pack_fms_GetStoredImages_t, 345
Tlvresult, 345
pack_fms_SetImagesPreference
fms.h, 1130
pack_fms_SetImagesPreference_t, 345
bForceDownload, 346
imageListSize, 346
modemindex, 346
pImageList, 346
Tlvresult, 346
pack_ims_SLQSGetIMSSMSConfig
ims.h, 1133
pack_ims_SLQSGetIMSUserConfig
ims.h, 1133
pack_ims_SLQSGetIMSVoIPConfig
ims.h, 1134
pack_ims_SLQSGetRegMgrConfig
ims.h, 1134
pack_ims_SLQSGetSIPConfig
ims.h, 1135
pack_ims_SLQSImsConfigIndicationRegister
ims.h, 1135
pack_ims_SLQSImsConfigIndicationRegister_t, 346
pRegMgrConfigEvents, 347
pSIPConfigEvents, 347
pSMSConfigEvents, 347
pUserConfigEvents, 347
pVoIPConfigEvents, 347
pack_ims_SLQSSetIMSSMSConfig
ims.h, 1136
pack_ims_SLQSSetIMSSMSConfig_t, 347
pack_ims_SLQSSetIMSUserConfig
ims.h, 1136
pack_ims_SLQSSetIMSUserConfig_t, 348
pIMSDomain, 349
pack_ims_SLQSSetIMSVoIPConfig
ims.h, 1137
pack_ims_SLQSSetIMSVoIPConfig_t, 349
pAmrMode, 351
pRingingTimer, 351
pack_ims_SLQSSetRegMgrConfig
ims.h, 1137
pack_ims_SLQSSetRegMgrConfig_t, 351
pCSCFPortName, 352
pIMSTestMode, 352
pPriCSCFPort, 352
pack_ims_SLQSSetSIPConfig
ims.h, 1137
pack_ims_SLQSSetSIPConfig_t, 352
pSIPLocalPort, 353
pSigCompEnabled, 352
pSubscribeTimer, 353
pTimerSIPReg, 353
pTimerT1, 353
pTimerT2, 353
pTimerTf, 353
pack_imsa_SLQSGetIMSARegStatus
imsa.h, 1145
pack_imsa_SLQSGetIMSAServiceStatus
imsa.h, 1145
pack_imsa_SLQSRegisterIMSAIndication
imsa.h, 1146
pack_imsa_SLQSRegisterIMSAIndication_t, 353
PdpStatusConfig, 354
RatHandoverStatusConfig, 354
RegStatusConfig, 354
ServiceStatusConfig, 354
pack_loc_Delete_Assist_Data_t, 354
pBdsSVInfo, 355
pCellDb, 355
pClkInfo, 355
pGnssData, 355
pSVInfo, 355
Tlvresult, 355
pack_loc_DeleteAssistData
loc.h, 1166
pack_loc_EventRegister
loc.h, 1167
pack_loc_EventRegister_t, 355
eventRegister, 357
Tlvresult, 357

pack_loc_SLQSLOCGetBestAvailPos
loc.h, 1168
pack_loc_SLQSLOCGetBestAvailPos_t, 358
Tlvresult, 358
xid, 358
pack_loc_SLQSLOCGetServer
loc.h, 1168
pack_loc_SLQSLOCGetServer_t, 359
has_serverAddrTypeMask, 359
serverAddrTypeMask, 359
serverType, 359
pack_loc_SLQSLOCInjectPosition
loc.h, 1169
pack_loc_SLQSLOCInjectPosition_t, 359
altitudeSrcInfo, 363
altitudeWrtEllipsoid, 363
altitudeWrtMeanSeaLevel, 363
has_altitudeSrcInfo, 363
has_altitudeWrtEllipsoid, 363
has_horConfidence, 363
has_horReliability, 363
has_horUncCircular, 363
has_latitude, 363
has_longitude, 363
has_positionSrc, 363
has_rawHorConfidence, 363
has_rawHorUncCircular, 363
has_timestampAge, 363
has_timestampUtc, 363
has_vertConfidence, 363
has_vertRelicability, 363
has_vertUnc, 364
horConfidence, 364
horReliability, 364
horUncCircular, 364
latitude, 364
longitude, 364
positionSrc, 364
rawHorConfidence, 364
rawHorUncCircular, 364
timestampAge, 364
timestampUtc, 364
vertConfidence, 364
vertReliability, 364
vertUnc, 364
pack_loc_SLQSLOCInjectSensorData
loc.h, 1169
pack_loc_SLQSLOCInjectSensorData_t, 364
accelTemp, 365
acceleroData, 365
acceleroTimeSrc, 365
gyroData, 365
gyroTemp, 365
gyroTimeSrc, 366
has_accelTemp, 366
has_acceleroTimeSrc, 366
has_accleroData, 366
has_gyroData, 366
has_gyroTemp, 366
has_gyroTimeSrc, 366
has_opaqueId, 366
opaqueId, 366
pack_loc_SLQSLOCInjectUTCTime
loc.h, 1169
pack_loc_SLQSLOCInjectUTCTime_t, 366
timeMsec, 366
timeUncMsec, 366
pack_loc_SLQSLOCSetCradleMountConfig
loc.h, 1170
pack_loc_SLQSLOCSetCradleMountConfig_t, 366
confidence, 367
has_confidence, 367
state, 367
pack_loc_SLQSLOCSetServer
loc.h, 1170
pack_loc_SLQSLOCSetServer_t, 367
pIPv4Config, 368
pIPv6Config, 368
pURLAddr, 368
serverType, 368
pack_loc_SetExtPowerState
loc.h, 1167
pack_loc_SetExtPowerState_t, 357
extPowerState, 357
Tlvresult, 357
pack_loc_SetOperationMode
loc.h, 1167
pack_loc_SetOperationMode_t, 357
mode, 358
Tlvresult, 358
pack_loc_Start
loc.h, 1171
pack_loc_Start_t, 368
pApplicationInfo, 369
pConfigAltitudeAssumed, 369
pHorizontalAccuracyLvl, 369
pIntermediateReportState, 369
pMinIntervalTime, 369
pRecurrenceType, 369
SessionId, 370
Tlvresult, 370
pack_loc_Stop
loc.h, 1171
pack_loc_Stop_t, 370
SessionId, 370
Tlvresult, 370
pack_nas_GetACCOLC
nas.h, 1197
pack_nas_GetANAAAAuthenticationStatus
nas.h, 1198
pack_nas_GetCDMANetworkParameters
nas.h, 1198
pack_nas_GetHomeNetwork
nas.h, 1198
pack_nas_GetHomeNetwork3GPP2
nas.h, 1199

pack_nas_GetNetworkPreference
nas.h, 1199
pack_nas_GetRFInfo
nas.h, 1199
pack_nas_GetServingNetwork
nas.h, 1199
pack_nas_GetServingNetworkCapabilities
nas.h, 1200
pack_nas_GetSignalStrengths
nas.h, 1200
pack_nas_InitiateDomainAttach
nas.h, 1200
pack_nas_InitiateDomainAttach_t, 370
action, 370
pack_nas_PerformNetworkScan
nas.h, 1201
pack_nas_PerformNetworkScanPCI
nas.h, 1201
pack_nas_PerformNetworkScanPCI_t, 371
pCiotOpModePref, 371
pLteM1BandPref, 371
pLteNB1BandPref, 371
pNetworkType, 371
pScanType, 371
pack_nas_SLQSConfigSigInfo
nas.h, 1203
pack_nas_SLQSConfigSigInfo_t, 374
pECIOThresh, 375
pIOThresh, 375
pLTESNRThresh, 375
pLTESigRptCfg, 375
pRSRPThresh, 375
pRSRQThresh, 375
pRSSIThresh, 375
pack_nas_SLQSGetErrorRate
nas.h, 1204
pack_nas_SLQSGetNetworkTime
nas.h, 1204
pack_nas_SLQSGetOperatorNameData
nas.h, 1205
pack_nas_SLQSGetPLMNName
nas.h, 1205
pack_nas_SLQSGetPLMNName_t, 375
mcc, 376
mnc, 376
pMncPcsStatus, 376
pack_nas_SLQSGetServingSystem
nas.h, 1205
pack_nas_SLQSGetSignalStrength
nas.h, 1206
pack_nas_SLQSGetSysInfo
nas.h, 1206
pack_nas_SLQSGetSysSelectionPref
nas.h, 1206
pack_nas_SLQSGetSysSelectionPrefExt
nas.h, 1207
pack_nas_SLQSInitiateNetworkRegistration
nas.h, 1207
pack_nas_SLQSInitiateNetworkRegistration_t, 376
pChangeDuration, 377
pMNRInfo, 377
pMncPcsDigitStatus, 377
regAction, 377
pack_nas_SLQSNASGetForbiddenNetworks
nas.h, 1209
pack_nas_SLQSNASGeteDRXParams
nas.h, 1209
pack_nas_SLQSNASGeteDRXParamsExt
nas.h, 1209
pack_nas_SLQSNASGeteDRXParamsExt_t, 382
pack_nas_SLQSNASSeteDRXParams
nas.h, 1211
pack_nas_SLQSNASSeteDRXParams_t, 387
pCycleLen, 388
pEdrxEnable, 388
pack_nas_SLQSNASSwiGetChannelLock
nas.h, 1212
pack_nas_SLQSNASSwiSetChannelLock
nas.h, 1213
pack_nas_SLQSNASSwiSetChannelLock_t, 390
pack_nas_SLQSNasConfigSigInfo2
nas.h, 1207
pack_nas_SLQSNasConfigSigInfo2_t, 377
pHDRIODelta, 381
pHDRIOThresh, 381
pLTESigRptConfig, 381
pack_nas_SLQSNasGet3GPP2Subscription
nas.h, 1208
pack_nas_SLQSNasGet3GPP2Subscription_t, 382
namID, 382
pack_nas_SLQSNasGetCellLocationInfo
nas.h, 1208
pack_nas_SLQSNasGetHDRColorCode
nas.h, 1210
pack_nas_SLQSNasGetRFInfo
nas.h, 1210
pack_nas_SLQSNasGetSigInfo
nas.h, 1210
pack_nas_SLQSNasGetTxRxInfo
nas.h, 1211
pack_nas_SLQSNasGetTxRxInfo_t, 383
radio_if, 384
pack_nas_SLQSNasIndicationRegisterExt
nas.h, 1211
pack_nas_SLQSNasIndicationRegisterExt_t, 384
pDDTMInd, 386
pDualStandByPrefInd, 386
pEdrxChangeInfoInd, 386
pErrorRateInd, 386
pHDRSessionCloseInd, 386
pLTECphyCa, 386
pManagedRoamingInd, 386
pNetworkRejectInd, 387
pNetworkTimeInd, 387
pServingSystemInd, 387
pSignalStrengthInd, 387

pSubscriptionInfoInd, 387
pSuppressSysInfoInd, 387
pSysInfoInd, 387
pSystemSelectionInd, 387
pack_nas_SLQSNasSwiIndicationRegister
nas.h, 1212
pack_nas_SLQSNasSwiIndicationRegister_t, 388
gsmUmtsDl, 389
gsmUmtsUl, 389
lteEmmDl, 389
lteEmmUl, 389
lteEsmDl, 389
lteEsmUl, 389
pRankIndicatorInd, 390
pTimer, 390
pack_nas_SLQSNasSwiModemStatus
nas.h, 1213
pack_nas_SLQSSetBandPreference
nas.h, 1213
pack_nas_SLQSSetBandPreference_t, 390
bandPref, 391
pack_nas_SLQSSetSignalStrengthsCallback
nas.h, 1214
pack_nas_SLQSSetSignalStrengthsCallback_t, 392
bEnable, 392
pSigIndReq, 392
pack_nas_SLQSSetSysSelectionPref
nas.h, 1214
pack_nas_SLQSSetSysSelectionPref_t, 392
pAcqOrderPref, 397
pBandPref, 397
pCSGID, 397
pChgDuration, 397
pEmerMode, 397
pGWAcqOrderPref, 397
pLTEBandPref, 397
pModePref, 397
pNetSelPref, 397
pPRLPref, 397
pRAT, 397
pRoamPref, 397
pSrvDomainPref, 397
pSrvRegRestriction, 397
pTdscdmaBandPref, 398
pack_nas_SLQSSetSysSelectionPrefExt
nas.h, 1214
pack_nas_SLQSSetSysSelectionPrefExt_t, 398
pAcqOrderPref, 407
pBandPref, 407
pCSGID, 407
pChgDuration, 407
pCiotAcqOrderPref, 407
pCiotLteOpMode, 407
pCiotLteOpModePref, 407
pEmerMode, 407
pGWAcqOrderPref, 407
pLTEBandPref, 407
pLteM1BandPref, 407
pLteNb1BandPref, 407
pModePref, 407
pNetSelPref, 407
pNr5gBandPref, 407
pPRLPref, 407
pRAT, 408
pRoamPref, 408
pSrvDomainPref, 408
pSrvRegRestriction, 408
pTdscdmaBandPref, 408
pack_nas_SLQSSwiGetHDRPersonality
nas.h, 1215
pack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 1215
pack_nas_SLQSSwiGetHRPDStats
nas.h, 1215
pack_nas_SLQSSwiGetLteCQI
nas.h, 1216
pack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 1216
pack_nas_SLQSSwiNetworkDebug
nas.h, 1216
pack_nas_SLQSSwiPSDetach
nas.h, 1217
pack_nas_SLQSSwiPSDetach_t, 408
pDetachAction, 408
pack_nas_SetACCOLC
nas.h, 1202
pack_nas_SetACCOLC_t, 371
accolc, 371
spc, 371
pack_nas_SetCDMANetworkParameters
nas.h, 1202
pack_nas_SetCDMANetworkParameters_t, 371
pApplication, 373
pBroadcast, 373
pCustomSCP, 373
pForceRev0, 373
pProtocol, 373
pRoaming, 373
pSPC, 373
pack_nas_SetLURejectCallback
nas.h, 1202
pack_nas_SetNetworkPreference
nas.h, 1203
pack_nas_SetNetworkPreference_t, 373
Duration, 374
TechnologyPref, 374
Tlvresult, 374
pack_nas_SetRFInfoCallback
nas.h, 1203
pack_nas_SlqsGetLTECphyCAInfo
nas.h, 1204
pack_pds_ForceXTRADownload
pds.h, 1248
pack_pds_GetPDSDefaults
pds.h, 1248
pack_pds_GetPDSState

pds.h, 1249
pack_pds_GetPortAutomaticTracking
pds.h, 1249
pack_pds_GetServiceAutomaticTracking
pds.h, 1249
pack_pds_GetXTRAAutomaticDownload
pds.h, 1250
pack_pds_GetXTRANetwork
pds.h, 1250
pack_pds_GetXTRAValidity
pds.h, 1250
pack_pds_PDSInjectTimeReference
pds.h, 1251
pack_pds_PDSInjectTimeReference_t, 408
systemDiscontinuities, 409
systemTime, 409
pack_pds_ResetPDSData
pds.h, 1251
pack_pds_ResetPDSData_t, 409
pCellDataMask, 409
pGPSDataMask, 410
pack_pds_SLQSGetAGPSConfig
pds.h, 1254
pack_pds_SLQSGetAGPSConfig_t, 413
pNetworkMode, 414
pack_pds_SLQSGetGPSStateInfo
pds.h, 1254
pack_pds_SLQSPDSDeterminePosition
pds.h, 1255
pack_pds_SLQSPDSInjectAbsoluteTimeReference
pds.h, 1255
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 414
forceFlag, 414
timeBase, 414
timeMsec, 414
timeUncMsec, 415
pack_pds_SLQSPDSInjectPositionData
pds.h, 1255
pack_pds_SLQSPDSInjectPositionData_t, 415
pAltitudeWrtEllipsoid, 416
pAltitudeWrtSealevel, 416
pHorizontalConfidence, 416
pHorizontalUncCircular, 416
pLatitude, 416
pLongitude, 416
pPositionSource, 416
pTimeStamp, 416
pTimeType, 416
pVerticalConfidence, 416
pVerticalUnc, 416
pack_pds_SLQSSetAGPSConfig
pds.h, 1256
pack_pds_SLQSSetAGPSConfig_t, 417
pNetworkMode, 417
pServerAddress, 417
pServerPort, 417
pServerURL, 417
pack_pds_SLQSSetPositionMethodState
pds.h, 1256
pack_pds_SLQSSetPositionMethodState_t, 417
pWifiState, 418
pXtraDataState, 418
pXtraTimeState, 418
pack_pds_SetEventReportCallback
pds.h, 1251
pack_pds_SetEventReportCallback_t, 410
posDataNmea, 410
rptPosData, 410
pack_pds_SetPDSDefaults
pds.h, 1252
pack_pds_SetPDSDefaults_t, 410
accuracy, 411
interval, 411
operation, 411
timeout, 411
pack_pds_SetPDSState
pds.h, 1252
pack_pds_SetPDSState_t, 411
enable, 411
pack_pds_SetPortAutomaticTracking
pds.h, 1252
pack_pds_SetPortAutomaticTracking_t, 411
bAuto, 412
pack_pds_SetServiceAutomaticTracking
pds.h, 1253
pack_pds_SetServiceAutomaticTracking_t, 412
bAuto, 412
pack_pds_SetXTRAAutomaticDownload
pds.h, 1253
pack_pds_SetXTRAAutomaticDownload_t, 412
bEnabled, 413
interval, 413
pack_pds_SetXTRANetwork
pds.h, 1253
pack_pds_SetXTRANetwork_t, 413
preference, 413
pack_pds_StartPDSTrackingSessionExt
pds.h, 1256
pack_pds_StartPDSTrackingSessionExt_t, 418
fixAccuracy, 419
fixCount, 419
fixInterval, 419
fixTimeout, 419
sessionControl, 419
sessionOperation, 419
sessionServerOption, 419
sessionType, 419
pack_pds_StopPDSTrackingSession
pds.h, 1257
pack_qmi_t, 419
msgid, 420
svc, 420
timeout, 420
xid, 420
pack_qos_BindDataPort

qos.h, 1316
pack_qos_BindDataPort_t, 420
pMuxID, 421
pPeripheralEndPointID, 421
pSIODataPort, 421
pack_qos_SLQSQosGetNetworkStatus
qos.h, 1316
pack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 1317
pack_qos_SLQSQosSwiReadApnExtraParams_t, 421
apnId, 421
pack_qos_SLQSQosSwiReadDataStats
qos.h, 1318
pack_qos_SLQSQosSwiReadDataStats_t, 421
apnId, 421
pack_qos_SLQSSetQosEventCallback
qos.h, 1318
pack_qos_SLQSSetQosEventCallback_t, 421
enable, 422
pack_rms_GetSMSWake
rms.h, 1324
pack_rms_SetSMSWake
rms.h, 1325
pack_rms_SetSMSWake_t, 422
enabled, 422
wake_mask, 422
pack_sar_SLQSGetRfSarState
sar.h, 1326
pack_sar_SLQSSetRfSarState
sar.h, 1326
pack_sar_SLQSSetRfSarState_t, 422
RfSarState, 423
pack_sms_GetSMSCAddress
sms.h, 1332
pack_sms_SLQSDeleteSMS
sms.h, 1334
pack_sms_SLQSDeleteSMS_t, 425
pMessageIndex, 426
pMessageMode, 426
pMessageTag, 426
storageType, 426
pack_sms_SLQSGetIndicationRegister
sms.h, 1334
pack_sms_SLQSGetMessageWaiting
sms.h, 1334
pack_sms_SLQSGetSMS
sms.h, 1335
pack_sms_SLQSGetSMS_t, 426
messageIndex, 427
pMessageMode, 427
storageType, 427
pack_sms_SLQSGetSMSList
sms.h, 1335
pack_sms_SLQSGetSMSList_t, 427
pMessageMode, 428
pRequestedTag, 428
storageType, 428
pack_sms_SLQSGetSmsBroadcastConfig
sms.h, 1335
pack_sms_SLQSGetSmsBroadcastConfig_t, 427
mode, 427
pack_sms_SLQSGetTransLayerInfo
sms.h, 1336
pack_sms_SLQSGetTransNWRegInfo
sms.h, 1336
pack_sms_SLQSModifySMSStatus
sms.h, 1336
pack_sms_SLQSModifySMSStatus_t, 428
messageIndex, 429
messageTag, 429
pMessageMode, 429
storageType, 429
pack_sms_SLQSSendAsyncSMS
sms.h, 1337
pack_sms_SLQSSendAsyncSMS_t, 429
pSendSmsParams, 429
pack_sms_SLQSSetIndicationRegister
sms.h, 1337
pack_sms_SLQSSetIndicationRegister_t, 429
pSetIndicationRegReq, 430
pack_sms_SLQSSetSmsBroadcastActivation
sms.h, 1337
pack_sms_SLQSSetSmsBroadcastActivation_t, 430
broadcastActivate, 430
mode, 430
pack_sms_SLQSSetSmsBroadcastConfig
sms.h, 1338
pack_sms_SLQSSetSmsBroadcastConfig_t, 430
mode, 431
pBroadcastConfig, 431
pCDMABroadcastConfig, 431
pack_sms_SLQSSetSmsStorage
sms.h, 1338
pack_sms_SLQSSetSmsStorage_t, 431
smsStorage, 431
pack_sms_SLQSSmsGetMaxStorageSize
sms.h, 1339
pack_sms_SLQSSmsGetMaxStorageSize_t, 432
pMaxStorageSizeReq, 432
pack_sms_SLQSSmsGetMessageProtocol
sms.h, 1339
pack_sms_SLQSSmsSetRoutes
sms.h, 1339
pack_sms_SLQSSmsSetRoutes_t, 432
pSetRoutesReq, 432
pack_sms_SLQSSwiGetSMSStorage
sms.h, 1340
pack_sms_SaveSMS
sms.h, 1332
pack_sms_SaveSMS_t, 423
messageFormat, 424
messageSize, 424
pMessage, 424
storageType, 424
pack_sms_SendSMS
sms.h, 1333

pack_sms_SendSMS_t, 424
messageFormat, 424
messageSize, 424
pLinktimer, 424
pMessage, 425
pack_sms_SetNewSMSCallback
sms.h, 1333
pack_sms_SetNewSMSCallback_t, 425
status, 425
pack_sms_SetSMSCAddress
sms.h, 1333
pack_sms_SetSMSCAddress_t, 425
pSMSCAddress, 425
pSMSCType, 425
pack_swiaudio_SLQSGetM2MAVMute
swiaudio.h, 1353
pack_swiaudio_SLQSGetM2MAVMute_t, 433
Profile, 434
pack_swiaudio_SLQSGetM2MAudioProfile
swiaudio.h, 1352
pack_swiaudio_SLQSGetM2MAudioProfile_t, 432
pGenerator, 433
pack_swiaudio_SLQSGetM2MAudioVolume
swiaudio.h, 1352
pack_swiaudio_SLQSGetM2MAudioVolume_t, 433
Generator, 433
Profile, 433
pack_swiaudio_SLQSGetM2MSpkrGain
swiaudio.h, 1353
pack_swiaudio_SLQSGetM2MSpkrGain_t, 434
Profile, 434
pack_swiaudio_SLQSSetM2MAVMute
swiaudio.h, 1355
pack_swiaudio_SLQSSetM2MAVMute_t, 437
EarMute, 438
MicMute, 438
pCwtMute, 438
Profile, 438
pack_swiaudio_SLQSSetM2MAudioAVCFG
swiaudio.h, 1354
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 434
Device, 435
Profile, 435
pack_swiaudio_SLQSSetM2MAudioLPBK
swiaudio.h, 1354
pack_swiaudio_SLQSSetM2MAudioLPBK_t, 435
Enable, 435
pack_swiaudio_SLQSSetM2MAudioNVDef
swiaudio.h, 1354
pack_swiaudio_SLQSSetM2MAudioProfile
swiaudio.h, 1355
pack_swiaudio_SLQSSetM2MAudioProfile_t, 435
pCwtMute, 436
pEarMute, 436
pGenerator, 436
pMicMute, 436
pVolume, 436
Profile, 436
pack_swiaudio_SLQSSetM2MAudioVolume
swiaudio.h, 1355
pack_swiaudio_SLQSSetM2MAudioVolume_t, 437
Generator, 437
Level, 437
Profile, 437
pack_swiaudio_SLQSSetM2MSpkrGain
swiaudio.h, 1356
pack_swiaudio_SLQSSetM2MSpkrGain_t, 438
Profile, 438
Value, 438
pack_swiavms_SLQSAVMSGetSettings
swiavms.h, 1362
pack_swiavms_SLQSAVMSGetSettings_v2
swiavms.h, 1363
pack_swiavms_SLQSAVMSSendSelection
swiavms.h, 1363
pack_swiavms_SLQSAVMSSendSelection_t, 439
pDeferTime, 439
pPackageID, 439
pRejectReason, 439
selection, 439
pack_swiavms_SLQSAVMSSessionGetInfo
swiavms.h, 1364
pack_swiavms_SLQSAVMSSetSettings
swiavms.h, 1365
pack_swiavms_SLQSAVMSSetSettings_t, 439
AutoConnect, 441
AutoReboot, 441
pAPNInfo, 441
pConnectionRetryTimers, 441
pNotifStore, 441
pPeriodInfo, 441
pPollingTimer, 441
PromptFwDownload, 441
PromptFwUpdate, 441
pack_swiavms_SLQSAVMSSetSettings_v2
swiavms.h, 1366
pack_swiavms_SLQSAVMSSetSettings_v2_t, 441
AutoConnect, 442
pAutoReboot, 443
pNotifStore, 443
pPeriodInfo, 443
pPollingTimer, 443
PromptFwDownload, 443
PromptFwUpdate, 443
pack_swiavms_SLQSAVMSSetSettingsNoAutoReboot-
Field
swiavms.h, 1366
pack_swiavms_SLQSAVMSStartSession
swiavms.h, 1367
pack_swiavms_SLQSAVMSStartSession_t, 443
sessionType, 443
pack_swiavms_SLQSAVMSStopSession
swiavms.h, 1368
pack_swiavms_SLQSAVMSStopSession_t, 443
sessionType, 444
pack_swiavms_SLQSAvmsSetEventReport

swiavms.h, 1365
pack_swidms_SLQSSwiDmsGetMTU
swidms.h, 1376
pack_swidms_SLQSSwiDmsGetUsbComp
swidms.h, 1377
pack_swidms_SLQSSwiDmsGetUsbNetNum
swidms.h, 1377
pack_swidms_SLQSSwiDmsSetMTU
swidms.h, 1377
pack_swidms_SLQSSwiDmsSetMTU_t, 444
MTUSize, 444
pack_swidms_SLQSSwiDmsSetUsbComp
swidms.h, 1378
pack_swidms_SLQSSwiDmsSetUsbComp_t, 444
CfgValue, 445
pack_swidms_SLQSSwiDmsSetUsbNetNum
swidms.h, 1378
pack_swidms_SLQSSwiDmsSetUsbNetNum_t, 445
nUsbNetNum, 445
pack_swiloc_SwiLocGetAutoStart
swiloc.h, 1381
pack_swiloc_SwiLocSetAutoStart
swiloc.h, 1381
pack_swiloc_SwiLocSetAutoStart_t, 446
fix_rate, 447
fix_type, 447
function, 447
max_dist, 447
max_time, 447
set_fix_rate, 447
set_fix_type, 447
set_function, 447
set_max_dist, 447
set_max_time, 447
pack_swioma_SLQSOMADMAlertCallback
swioma.h, 1384
pack_swioma_SLQSOMADMCancelSession
swioma.h, 1384
pack_swioma_SLQSOMADMCancelSession_t, 447
sessionType, 448
pack_swioma_SLQSOMADMCancelSessionExt
swiomaext.h, 1393
pack_swioma_SLQSOMADMCancelSessionExt_t, 448
sessionType, 448
pack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 1385
pack_swioma_SLQSOMADMGetSessionInfo_t, 448
SessionType, 449
pack_swioma_SLQSOMADMGetSettings
swioma.h, 1386
pack_swioma_SLQSOMADMSendSelection
swioma.h, 1386
pack_swioma_SLQSOMADMSendSelection_t, 449
pDeferTime, 449
pRejectReason, 449
selection, 449
pack_swioma_SLQSOMADMSendSelectionExt
swiomaext.h, 1394
pack_swioma_SLQSOMADMSendSelectionExt_t, 449
selection, 450
pack_swioma_SLQSOMADMSetSettings
swioma.h, 1387
pack_swioma_SLQSOMADMSetSettings_t, 450
FOTAdownload, 451
pAutosdm, 451
pFwAutoCheck, 451
pack_swioma_SLQSOMADMSetSettingsExt
swiomaext.h, 1394
pack_swioma_SLQSOMADMSetSettingsExt_t, 451
pack_swioma_SLQSOMADMStartSession
swioma.h, 1388
pack_swioma_SLQSOMADMStartSession_t, 452
sessionType, 452
pack_swioma_SLQSOMADMStartSessionExt
swiomaext.h, 1395
pack_swioma_SLQSOMADMStartSessionExt_t, 452
sessionType, 453
pack_tmd_SLQSTmdDeRegNotMitigationLvl
tmd.h, 1400
pack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 453
mitigationDevID, 453
mitigationDevIDLen, 453
pack_tmd_SLQSTmdGetMitigationDevList
tmd.h, 1400
pack_tmd_SLQSTmdGetMitigationLvl
tmd.h, 1400
pack_tmd_SLQSTmdGetMitigationLvl_t, 453
mitigationDevID, 454
mitigationDevIDLen, 454
pack_tmd_SLQSTmdRegNotMitigationLvl
tmd.h, 1401
pack_tmd_SLQSTmdRegNotMitigationLvl_t, 454
mitigationDevID, 454
mitigationDevIDLen, 454
pack_uim_ChangePin
uim.h, 1407
pack_uim_ChangePin_t, 454
changePIN, 455
EncryptedPIN1, 455
pIndicationToken, 455
pKeyReferenceID, 455
sessionInfo, 455
Tlvresult, 455
pack_uim_GetCardStatus
uim.h, 1407
pack_uim_ReadTransparent
uim.h, 1407
pack_uim_ReadTransparent_t, 455
fileIndex, 456
pEncryptData, 456
pIndicationToken, 456
readTransparent, 456
sessionInfo, 456
Tlvresult, 456
pack_uim_SLQSUIMAuthenticate
uim.h, 1408

pack_uim_SLQSUIMAuthenticate_t, 458
authData, 458
pIndicationToken, 458
sessionInfo, 458
pack_uim_SLQSUIMDepersonalization
uim.h, 1408
pack_uim_SLQSUIMDepersonalization_t, 458
depersonilisationInfo, 459
pack_uim_SLQSUIMEventRegister
uim.h, 1409
pack_uim_SLQSUIMEventRegister_t, 459
eventMask, 459
pack_uim_SLQSUIMGetConfiguration
uim.h, 1409
pack_uim_SLQSUIMGetConfiguration_t, 459
pConfigurationMask, 459
pack_uim_SLQSUIMGetFileAttributes
uim.h, 1410
pack_uim_SLQSUIMGetFileAttributes_t, 460
fileIndex, 460
pIndicationToken, 460
sessionInfo, 460
pack_uim_SLQSUIMGetSlotsStatus
uim.h, 1410
pack_uim_SLQSUIMPowerDown
uim.h, 1410
pack_uim_SLQSUIMPowerDown_t, 460
slot, 461
pack_uim_SLQSUIMPowerUp
uim.h, 1411
pack_uim_SLQSUIMPowerUp_t, 461
pIgnoreHotSwapSwitch, 461
slot, 461
pack_uim_SLQSUIMRefreshComplete
uim.h, 1411
pack_uim_SLQSUIMRefreshComplete_t, 461
refreshComplete, 462
sessionInfo, 462
pack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 1411
pack_uim_SLQSUIMRefreshGetLastEvent_t, 462
sessionInfo, 462
pack_uim_SLQSUIMRefreshOK
uim.h, 1412
pack_uim_SLQSUIMRefreshOK_t, 462
OKtoRefresh, 463
sessionInfo, 463
pack_uim_SLQSUIMRefreshRegister
uim.h, 1412
pack_uim_SLQSUIMRefreshRegister_t, 463
regRefresh, 463
sessionInfo, 463
pack_uim_SLQSUIMReset
uim.h, 1413
pack_uim_SLQSUIMSwitchSlot
uim.h, 1413
pack_uim_SLQSUIMSwitchSlot_t, 463
bLogicalSlot, 464
ulPhysicalSlot, 464
pack_uim_SetPinProtection
uim.h, 1408
pack_uim_SetPinProtection_t, 457
EncryptedPIN1, 457
pIndicationToken, 457
pKeyReferenceID, 457
pinProtection, 457
sessionInfo, 457
Tlvresult, 457
pack_uim_UnblockPin
uim.h, 1413
pack_uim_UnblockPin_t, 464
EncryptedPIN1, 465
pIndicationToken, 465
pKeyReferenceID, 465
pinProtection, 465
sessionInfo, 465
Tlvresult, 465
pack_uim_VerifyPin
uim.h, 1414
pack_uim_VerifyPin_t, 465
pEncryptedPIN1, 466
pIndicationToken, 466
pKeyReferenceID, 466
sessionInfo, 466
Tlvresult, 466
verifyPIN, 466
pack_voice_AnswerUSSD
voice.h, 1429
pack_voice_AnswerUSSD_t, 466
pInfo, 467
pack_voice_CancelUSSD
voice.h, 1429
pack_voice_OriginateUSSD
voice.h, 1430
pack_voice_OriginateUSSD_t, 467
pInfo, 467
pack_voice_SLQSOriginateUSSD
voice.h, 1430
pack_voice_SLQSOriginateUSSD_t, 467
ussDCS, 468
ussData, 468
ussLen, 468
pack_voice_SLQSVoiceALSSelectLine
voice.h, 1430
pack_voice_SLQSVoiceALSSelectLine_t, 468
lineValue, 468
pack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 1431
pack_voice_SLQSVoiceALSSetLineSwitching_t, 468
switchOption, 468
pack_voice_SLQSVoiceAnswerCall
voice.h, 1431
pack_voice_SLQSVoiceAnswerCall_t, 469
pCallId, 469
pack_voice_SLQSVoiceBindSubscription
voice.h, 1431

pack_voice_SLQSVoiceBindSubscription_t, 469
subsType, 469
pack_voice_SLQSVoiceBurstDTMF
voice.h, 1432
pack_voice_SLQSVoiceBurstDTMF_t, 469
pack_voice_SLQSVoiceDialCall
voice.h, 1432
pack_voice_SLQSVoiceDialCall_t, 470
callNumber, 471
pCLIRType, 471
pCUGInfo, 471
pCallPartySubAdd, 471
pCallType, 471
pEmergencyCategory, 471
pSvcType, 472
pUUSInFo, 472
pack_voice_SLQSVoiceEndCall
voice.h, 1432
pack_voice_SLQSVoiceEndCall_t, 472
pCallId, 472
pack_voice_SLQSVoiceGetAllCallInfo
voice.h, 1433
pack_voice_SLQSVoiceGetCLIP
voice.h, 1435
pack_voice_SLQSVoiceGetCLIR
voice.h, 1435
pack_voice_SLQSVoiceGetCNAP
voice.h, 1435
pack_voice_SLQSVoiceGetCOLP
voice.h, 1436
pack_voice_SLQSVoiceGetCOLR
voice.h, 1436
pack_voice_SLQSVoiceGetCallBarring
voice.h, 1433
pack_voice_SLQSVoiceGetCallBarring_t, 472
pSvcClass, 473
reason, 473
pack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 1434
pack_voice_SLQSVoiceGetCallForwardingStatus_t, 473
pSvcClass, 474
Reason, 474
pack_voice_SLQSVoiceGetCallInfo
voice.h, 1434
pack_voice_SLQSVoiceGetCallInfo_t, 474
callID, 474
pack_voice_SLQSVoiceGetCallWaiting
voice.h, 1434
pack_voice_SLQSVoiceGetCallWaiting_t, 474
pSvcClass, 475
pack_voice_SLQSVoiceGetConfig
voice.h, 1436
pack_voice_SLQSVoiceGetConfig_t, 475
pAMRStatus, 476
pAirTimer, 476
pAutoAnswer, 476
pNamID, 476
pPrefVoicePrivacy, 476
pPrefVoiceSO, 476
pRoamTimer, 476
pTTYMode, 476
pVoiceDomainPref, 476
pack_voice_SLQSVoiceIndicationRegister
voice.h, 1437
pack_voice_SLQSVoiceIndicationRegister_t, 477
pRegDTMFEvents, 477
pRegVoicePrivacyEvents, 477
pSuppsNotifEvents, 477
pack_voice_SLQSVoiceManageCalls
voice.h, 1437
pack_voice_SLQSVoiceManageCalls_t, 477
pCallID, 478
SUPSType, 478
pack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 1437
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 478
pack_voice_SLQSVoiceSendFlash
voice.h, 1438
pack_voice_SLQSVoiceSendFlash_t, 479
pCallID, 479
pFlashPayLd, 479
pFlashType, 479
pack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 1438
pack_voice_SLQSVoiceSetCallBarringPassword_t, 479
newPasswd, 480
newPasswdAgain, 480
oldPasswd, 480
Reason, 480
pack_voice_SLQSVoiceSetConfig
voice.h, 1438
pack_voice_SLQSVoiceSetConfig_t, 480
pAirTimerConfig, 482
pAutoAnswer, 482
pPrefVoiceDomain, 482
pPrefVoiceSO, 482
pRoamTimerConfig, 482
pTTYMode, 482
pack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 1439
pack_voice_SLQSVoiceSetPreferredPrivacy_t, 482
privacyPref, 482
pack_voice_SLQSVoiceSetSUPSService
voice.h, 1439
pack_voice_SLQSVoiceSetSUPSService_t, 482
pCallBarringPasswd, 484
pCallForwardingNumber, 484
pServiceClass, 484
pTimerVal, 484
reason, 484
voiceSvc, 484
pack_voice_SLQSVoiceStartContDTMF
voice.h, 1440
pack_voice_SLQSVoiceStartContDTMF_t, 484
DTMFdigit, 485
pCallID, 485

pack_voice_SLQSVoiceStopContDTMF
voice.h, 1440
pack_voice_SLQSVoiceStopContDTMF_t, 485
callID, 485
pack_wds_DHCPv4ClientLeaseChange
wds.h, 1463
pack_wds_DHCPv4ClientLeaseChange_t, 485
pEnableNotification, 486
pack_wds_GetAutoconnect
wds.h, 1464
pack_wds_GetByteTotals
wds.h, 1464
pack_wds_GetConnectionRate
wds.h, 1465
pack_wds_GetDataBearerTechnology
wds.h, 1465
pack_wds_GetDefaultProfile
wds.h, 1465
pack_wds_GetDefaultProfile_t, 486
profiletype, 486
pack_wds_GetDefaultProfileNum
wds.h, 1466
pack_wds_GetDefaultProfileNum_t, 486
family, 486
type, 486
pack_wds_GetDormancyState
wds.h, 1466
pack_wds_GetDormancyState_t, 487
pack_wds_GetLastMobileIPError
wds.h, 1467
pack_wds_GetLastMobileIPError_t, 487
pack_wds_GetMobileIP
wds.h, 1467
pack_wds_GetMobileIP_t, 487
pack_wds_GetMobileIPProfile
wds.h, 1468
pack_wds_GetMobileIPProfile_t, 487
index, 488
pack_wds_GetPacketStatistics
wds.h, 1468
pack_wds_GetPacketStatistics_t, 488
pStatMask, 488
pack_wds_GetPacketStatus
wds.h, 1468
pack_wds_GetPacketStatus_t, 488
statmask, 489
pack_wds_GetSessionDuration
wds.h, 1469
pack_wds_GetSessionDuration_t, 489
pack_wds_GetSessionState
wds.h, 1469
pack_wds_RMSetTransferStatistics
wds.h, 1470
pack_wds_RMSetTransferStatistics_t, 489
RmTrasnferStaticsReq, 489
pack_wds_SLQSCreateProfile
wds.h, 1473
pack_wds_SLQSCreateProfile_t, 496
pCurProfile, 497
pProfileId, 497
pProfileType, 497
pack_wds_SLQSDeleteProfile
wds.h, 1473
pack_wds_SLQSDeleteProfile_t, 497
profileIndex, 497
profileType, 497
pack_wds_SLQSGet3GPPConfigItem
wds.h, 1474
pack_wds_SLQSGetCurrDataSystemStat
wds.h, 1474
pack_wds_SLQSGetCurrDataSystemStat_t, 497
pack_wds_SLQSGetCurrentChannelRate
wds.h, 1475
pack_wds_SLQSGetDUNCallInfo
wds.h, 1476
pack_wds_SLQSGetDUNCallInfo_t, 498
Mask, 499
pReportChannelRate, 499
pReportConnStatus, 499
pReportDataBearerTech, 499
pReportDormStatus, 499
pTransferStatInd, 499
pack_wds_SLQSGetDataBearerTechnology
wds.h, 1475
pack_wds_SLQSGetDataBearerTechnology_t, 498
pack_wds_SLQSGetProfileSettings
wds.h, 1476
pack_wds_SLQSGetProfileSettings_t, 499
ProfileId, 500
ProfileType, 500
pack_wds_SLQSGetRuntimeSettings
wds.h, 1476
pack_wds_SLQSGetRuntimeSettings_t, 500
pReqSettings, 501
pack_wds_SLQSModifyProfile
wds.h, 1477
pack_wds_SLQSModifyProfile_t, 501
curProfile, 502
pProfileId, 502
pProfileType, 502
pack_wds_SLQSResetPacketStatics
wds.h, 1477
pack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 1479
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 505
pProfileId, 505
pack_wds_SLQSSGetLoopback
wds.h, 1479
pack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 1480
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 505
pHwConfig, 506
pProfileId, 506
pack_wds_SLQSSSetLoopback
wds.h, 1480
pack_wds_SLQSSSetLoopback_t, 506

loopbackMode, 506
loopbackMultiplier, 506
pack_wds_SLQSSet3GPPConfigItem
wds.h, 1478
pack_wds_SLQSSet3GPPConfigItem_t, 502
p3gppRelease, 503
pProfileList, 503
pack_wds_SLQSSetIPFamilyPreference
wds.h, 1478
pack_wds_SLQSSetIPFamilyPreference_t, 504
IPFamilyPreference, 504
pack_wds_SLQSSetWdsEventCallback
wds.h, 1478
pack_wds_SLQSSetWdsEventCallback_t, 504
currentDataBearer, 505
dataBearer, 505
dataBearerTechExt, 505
dataSystemStatus, 505
dormancyStatus, 505
interval, 505
mobileIP, 505
transferStats, 505
pack_wds_SLQSStartDataSession
wds.h, 1480
pack_wds_SLQSStartDataSession_t, 506
pAuth, 507
pPass, 507
pTech, 507
pUser, 507
pprofileid3gpp, 507
pprofileid3gpp2, 507
pack_wds_SLQSStopDataSession
wds.h, 1481
pack_wds_SLQSStopDataSession_t, 508
psid, 508
pack_wds_SLQSWdsGoActive
wds.h, 1481
pack_wds_SLQSWdsGoDormant
wds.h, 1482
pack_wds_SLQSWdsSetEventReport
wds.h, 1482
pack_wds_SLQSWdsSetEventReport_t, 508
pCurrChannelRateInd, 509
pCurrDataBearerTechInd, 509
pCurrPrefDataSysInd, 509
pDataBearerTechInd, 509
pDataCallStatusChangeInd, 509
pDataSystemStatusChangeInd, 509
pDormancyStatusInd, 510
pMIPStatusInd, 510
pTransferStatInd, 510
pack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 1483
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 510
contextId, 510
contextType, 510
pack_wds_SetAutoconnect
wds.h, 1470
pack_wds_SetAutoconnect_t, 490
acroamsetting, 490
acsetting, 490
pack_wds_SetDefaultProfile
wds.h, 1470
pack_wds_SetDefaultProfile_t, 490
authentication, 491
ipAddress, 492
pApnname, 492
pName, 492
pPassword, 492
pUsername, 492
pdpType, 492
primaryDNS, 492
profileType, 492
secondaryDNS, 492
pack_wds_SetDefaultProfileNum
wds.h, 1471
pack_wds_SetDefaultProfileNum_t, 492
family, 492
index, 492
type, 492
pack_wds_SetMobileIP
wds.h, 1471
pack_wds_SetMobileIP_t, 493
mode, 493
pack_wds_SetMobileIPParameters
wds.h, 1472
pack_wds_SetMobileIPParameters_t, 493
pHA2002bis, 494
pHAAuthenticator, 494
pMode, 494
pReRegPeriod, 494
pReRegTraffic, 494
pRetryInterval, 494
pRetryLimit, 494
pSPC, 494
pack_wds_SetMobileIPProfile
wds.h, 1472
pack_wds_SetMobileIPProfile_t, 494
index, 495
pAAASPI, 496
pAddress, 496
pEnabled, 496
pHASPI, 496
pMNAAA, 496
pMNHA, 496
pNAI, 496
pPrimaryHA, 496
pRevTunneling, 496
pSecondaryHA, 496
spc, 496
pack_wds_SetMuxID
wds.h, 1473
PackCreateProfileOut, 510
ExtErrorCode, 511
ProfileIndex, 511
ProfileType, 511

PackSwiAVMSSettingsAPNInfo, 514
bAPNLength, 514
bPWDLength, 514
bUnameLength, 515
pAPN, 515
pPWD, 515
pUname, 515
PackSwiAVMSSettingsConnectionRetryTimers, 515
Timers, 515
PackSwiAVMSSettingsPeriodsInfo, 516
max, 516
min, 516
PackSwiAvmsSetSettingsAPNInfo, 512
bAPNLength, 513
bPWDLength, 513
bUnameLength, 513
szAPN, 513
szPWD, 513
szUname, 513
PackSwiAvmsSetSettingsConnectionRetryTimers, 513
Timers, 513
PackSwiAvmsSetSettingsPeriodInfo, 514
ulMax, 514
ulMin, 514
package_name
unpack_omaDmFotaTlv_t, 779
packageid_str
unpack_dms_GetFirmwareInfo_t, 596
packetZone
nas_CDMASysInfo, 157
packetZoneValid
nas_CDMASysInfo, 157
packgetDyingGaspCfg, 511
pDestSMSContent, 511
pDestSMSNum, 511
packgetDyingGaspStatistics, 511
pSMSAttemptedFlag, 512
pTimeStamp, 512
ParamPresenceMask
pack_dms_UIMGetICCID_t, 341
unpack_audio_SLQSGetAudioPathConfig_t, 577
unpack_audio_SLQSGetAudioProfile_t, 579
unpack_audio_SLQSGetAudioVolTLBConfig_t, 579
unpack_audio_SLQSSetAudioVolTLBConfig_t, 580
unpack_cat_SetCatEventCallback_ind_t, 580
unpack_cat_SetCATEventCallback_t, 581
unpack_dms_GetActivationState_t, 582
unpack_dms_GetBandCapability_t, 584
unpack_dms_GetCrashAction_t, 584
unpack_dms_GetCustFeature_t, 587
unpack_dms_GetCustFeaturesV2_t, 588
unpack_dms_GetDeviceCap_t, 589
unpack_dms_GetDeviceCapabilities_t, 591
unpack_dms_GetDeviceHardwareRev_t, 591
unpack_dms_GetDeviceMfr_t, 592
unpack_dms_GetDeviceSerialNumbers_t, 594
unpack_dms_GetFirmwareInfo_t, 596
unpack_dms_GetFirmwareRevision_t, 597
unpack_dms_GetFirmwareRevisions_t, 598
unpack_dms_GetFSN_t, 599
unpack_dms_GetHardwareRevision_t, 599
unpack_dms_GetIMSI_t, 600
unpack_dms_GetManufacturer_t, 600
unpack_dms_GetModelID_t, 601
unpack_dms_GetNetworkTime_t, 602
unpack_dms_GetOfflineReason_t, 603
unpack_dms_GetPower_t, 604
unpack_dms_GetPRLVersion_t, 605
unpack_dms_GetSerialNumbers_t, 606
unpack_dms_GetUSBComp_t, 608
unpack_dms_GetVoiceNumber_t, 609
unpack_dms_PSMCfgChange_ind_t, 610
unpack_dms_ResetToFactoryDefaults_t, 611
unpack_dms_SetActivationStatusCallback_t, 611
unpack_dms_SetCrashAction_t, 612
unpack_dms_SetCustFeature_t, 612
unpack_dms_SetCustFeaturesV2_t, 613
unpack_dms_SetEventReport_ind_t, 614
unpack_dms_SetEventReport_t, 614
unpack_dms_SetFirmwarePreference_t, 615
unpack_dms_SetIndicationRegister_t, 615
unpack_dms_SetPower_t, 616
unpack_dms_SetUSBComp_t, 616
unpack_dms_SLQSDmsSwiGetPCInfo_t, 619
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 620
unpack_dms_SLQSDmsSwiGetResetInfo_t, 621
unpack_dms_SLQSDmsSwiGetUimSelection_t, 622
unpack_dms_SLQSDmsSwiIndicationRegister_t, 623
unpack_dms_SLQSGetBandCapability_t, 627
unpack_dms_SLQSGetERIFile_t, 628
unpack_dms_SLQSGetPowerSaveModeConfig_t, 629
unpack_dms_SLQSSetPowerSaveModeConfig_t, 630
unpack_dms_SLQSSwiClearDyingGaspStatistics-_t, 630
unpack_dms_SLQSSwiGetCrashInfo_t, 631
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 632
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 632
unpack_dms_SLQSSwiGetFirmwareCurr_t, 633
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 635
unpack_dms_SLQSSwiGetHostDevInfo_t, 636
unpack_dms_SLQSSwiGetOSInfo_t, 637
unpack_dms_SLQSSwiGetSerialNoExt_t, 638
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 638
unpack_dms_SLQSSwiSetHostDevInfo_t, 639
unpack_dms_SLQSSwiSetOSInfo_t, 639
unpack_dms_SLQSUIMGetState_t, 640
unpack_dms_SwiEventReportCallBack_ind_t, 641
unpack_dms_SwiSetEventReport_t, 641

unpack_dms_SwiUimSelect_t, 642
unpack_dms_UIMGetControlKeyStatus_t, 643
unpack_dms_UIMGetICCID_t, 643
unpack_dms_UIMGetPINStatus_t, 645
unpack_dms_UIMSetControlKeyProtection_t, 646
unpack_dms_UIMSetPINProtection_t, 647
unpack_dms_UIMUnblockControlKey_t, 647
unpack_fms_GetImagesPreference_t, 648
unpack_fms_GetStoredImages_t, 649
unpack_fms_SetImagesPreference_t, 649
unpack_ims_SLQSGetIMSSMSConfig_t, 650
unpack_ims_SLQSGetIMSUserConfig_t, 651
unpack_ims_SLQSGetIMSVoIPConfig_t, 654
unpack_ims_SLQSGetRegMgrConfig_t, 655
unpack_ims_SLQSGetSIPConfig_t, 656
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 657
unpack_ims_SLQSSetIMSSMSConfig_t, 658
unpack_ims_SLQSSetIMSUserConfig_t, 658
unpack_ims_SLQSSetIMSVoIPConfig_t, 659
unpack_ims_SLQSSetRegMgrConfig_t, 659
unpack_ims_SLQSSetSIPConfig_t, 660
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
unpack_ims_SLQSSMSCfgCallBack_ind_t, 662
unpack_ims_SLQSUserCfgCallBack_ind_t, 663
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
unpack_imsa_SLQSGetIMSARegStatus_t, 666
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 669
unpack_imsa_SLQSImsaRatStatusCallBack_ind_t, 670
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 670
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
unpack_loc_BestAvailPos_Ind_t, 678
unpack_loc_CradleMountCallback_Ind_t, 679
unpack_loc_Delete_Assist_Data_t, 680
unpack_loc_DeleteAssistData_Ind_t, 681
unpack_loc_EngineState_Ind_t, 682
unpack_loc_EventNMEA_Ind_t, 682
unpack_loc_EventRegister_t, 683
unpack_loc_EventTimeSyncCallback_Ind_t, 683
unpack_loc_GetServer_Ind_t, 685
unpack_loc_GnssSvInfo_Ind_t, 686
unpack_loc_InjectPositionCallback_Ind_t, 686
unpack_loc_InjectSensorDataCallback_Ind_t, 688
unpack_loc_InjectTimeSyncDataCallback_Ind_t, 689
unpack_loc_InjectUTCTimeCallback_Ind_t, 689
unpack_loc_PositionRpt_Ind_t, 695
unpack_loc_SensorStreamingCallback_Ind_t, 697
unpack_loc_SetExtPowerConfig_Ind_t, 698
unpack_loc_SetExtPowerState_t, 698
unpack_loc_SetOperationMode_Ind_t, 699
unpack_loc_SetOperationMode_t, 700
unpack_loc_SetServer_Ind_t, 701
unpack_loc_SLQSLOCGetBestAvailPos_t, 701
unpack_loc_Start_t, 701
unpack_loc_Stop_t, 702
unpack_nas_GetACCOLC_t, 703
unpack_nas_GetANAAAAuthenticationStatus_t, 703
unpack_nas_GetCDMANetworkParameters_t, 705
unpack_nas_GetHomeNetwork3GPP2_t, 708
unpack_nas_GetHomeNetwork_t, 709
unpack_nas_GetNetworkPreference_t, 710
unpack_nas_GetRFInfo_t, 711
unpack_nas_GetServingNetwork_t, 714
unpack_nas_GetServingNetworkCapabilities_t, 715
unpack_nas_GetSignalStrengths_t, 715
unpack_nas_PerformNetworkScan_t, 717
unpack_nas_SetDataCapabilitiesCallback_ind_t, 718
unpack_nas_SetEventReportInd_t, 719
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 720
unpack_nas_SetNetworkPreference_t, 720
unpack_nas_SetRoamingIndicatorCallback_ind_t, 721
unpack_nas_SetServingSystemCallback_ind_t, 722
unpack_nas_SLQSGetErrorRate_t, 723
unpack_nas_SlqsGetLTECphyCAInfo_t, 724
unpack_nas_SLQSGetNetworkTime_t, 724
unpack_nas_SLQSGetOperatorNameData_t, 725
unpack_nas_SLQSGetPLMNName_t, 728
unpack_nas_SLQSGetServingSystem_t, 733
unpack_nas_SLQSGetSignalStrength_t, 736
unpack_nas_SLQSGetSysInfo_t, 739
unpack_nas_SLQSGetSysSelectionPref_t, 744
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t, 748
unpack_nas_SLQSNasGet3GPP2Subscription_t, 749
unpack_nas_SLQSNasGetCellLocationInfo_t, 750
unpack_nas_SLQSNASGeteDRXParams_t, 752
unpack_nas_SLQSNASGeteDRXParamsExt_t, 753
unpack_nas_SLQSNASGetForbiddenNetworks_t, 754
unpack_nas_SLQSNasGetHDRColorCode_t, 754
unpack_nas_SLQSNasGetSigInfo_t, 756
unpack_nas_SLQSNasGetTxRxInfo_t, 757
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t, 760
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 761
unpack_nas_SLQSNasSigInfoCallback_ind_t, 762
unpack_nas_SLQSNASSwiGetChannelLock_t, 763
unpack_nas_SLQSNasSwiModemStatus_t, 764
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 765
unpack_nas_SLQSNasTimerCallback_ind_t, 765

unpack_nas_SLQSSetSysSelectionPrefCallBack-_ind_t, 766
unpack_nas_SLQSSwiGetHDRPersonality_t, 767
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 768
unpack_nas_SLQSSwiGetHRPDStats_t, 769
unpack_nas_SLQSSwiGetLteCQI_t, 770
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 770
unpack_nas_SLQSSwiHDRPersonalityCallback_-Ind_t, 771
unpack_nas_SLQSSwiNetworkDebug_t, 772
unpack_nas_SLQSSwiRandIndicatorCallback_-Ind_t, 772
unpack_nas_SLQSSysInfoCallback_ind_t, 776
unpack_pds_GetPDSDefaults_t, 781
unpack_pds_GetPDSState_t, 782
unpack_pds_GetPortAutomaticTracking_t, 782
unpack_pds_GetServiceAutomaticTracking_t, 783
unpack_pds_GetXTRAAutomaticDownload_t, 784
unpack_pds_GetXTRANetwork_t, 784
unpack_pds_GetXTRAValidity_t, 785
unpack_pds_SetEventReport_Ind_t, 787
unpack_pds_SetPdsState_Ind_t, 788
unpack_pds_SLQSGetAGPSConfig_t, 789
unpack_pds_SLQSGetGPSStateInfo_t, 793
unpack_qos_BindDataPort_t, 794
unpack_qos_SLQSQosGetNetworkStatus_t, 800
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 801
unpack_qos_SLQSQosSwiReadDataStats_t, 803
unpack_qos_SLQSSetQosEventCallback_ind_t, 804
unpack_qos_SLQSSetQosEventCallback_t, 804
unpack_qos_SLQSSetQosNWStatusCallback_ind-_t, 805
unpack_qos_SLQSSetQosPriEventCallback_ind_-t, 805
unpack_qos_SLQSSetQosStatusCallback_ind_t, 807
unpack_result_t, 816
unpack_rms_GetSMSWake_t, 817
unpack_rms_SetSMSWake_t, 818
unpack_RMTransferStatistics_ind_t, 818
unpack_sar_SLQSGetRfSarState_t, 819
unpack_sms_GetSMSCAddress_t, 820
unpack_sms_SaveSMS_t, 821
unpack_sms_SendSMS_t, 821
unpack_sms_SetNewSMSCallback_ind_t, 823
unpack_sms_SetNewSMSCallback_t, 823
unpack_sms_SetSMSCAddress_t, 824
unpack_sms_SLQSDeleteSMS_t, 824
unpack_sms_SLQSGetIndicationRegister_t, 825
unpack_sms_SLQSGetMessageWaiting_t, 825
unpack_sms_SLQSGetSMS_t, 826
unpack_sms_SLQSGetSmsBroadcastConfig_t, 827
unpack_sms_SLQSGetSMSList_t, 828
unpack_sms_SLQSGetTransLayerInfo_t, 828
unpack_sms_SLQSGetTransNWRegInfo_t, 829
unpack_sms_SLQSModifySMSStatus_t, 829
unpack_sms_SLQSNWRegInfoCallback_ind_t, 830
unpack_sms_SLQSSendAsyncSMS_t, 830
unpack_sms_SLQSSetIndicationRegister_t, 831
unpack_sms_SLQSSetSmsBroadcastActivation_t, 831
unpack_sms_SLQSSetSmsBroadcastConfig_t, 832
unpack_sms_SLQSSetSmsStorage_t, 832
unpack_sms_SLQSSmsGetMaxStorageSize_t, 832
unpack_sms_SLQSSmsGetMessageProtocol_t, 833
unpack_sms_SLQSSmsSetRoutes_t, 833
unpack_sms_SLQSSwiGetSMSStorage_t, 834
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 835
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
unpack_sms_SLQSWmsMemoryFullCallBack_ind-_t, 838
unpack_sms_SLQSWmsMessageWaitingCall-Back_ind_t, 838
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 840
unpack_swiaudio_SLQSGetM2MAudioVolume_t, 840
unpack_swiaudio_SLQSGetM2MAVMute_t, 841
unpack_swiaudio_SLQSGetM2MSpkrGain_t, 842
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
unpack_swiavms_SLQSAVMSSendSelection_t, 849
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 849
unpack_swiavms_SLQSAvmsSetEventReport_t, 850
unpack_swiavms_SLQSAVMSSetSettings_t, 850
unpack_swiavms_SLQSAVMSSetSettings_v2_t, 851
unpack_swiavms_SLQSAVMSStartSession_t, 851
unpack_swiavms_SLQSAVMSStopSession_t, 852
unpack_swidms_SLQSSwiDmsGetMTU_t, 853
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 853
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t, 854
unpack_swidms_SLQSSwiDmsSetMTU_t, 855
unpack_swidms_SLQSSwiDmsSetUsbComp_t, 855
unpack_swiloc_SwiLocGetAutoStart_t, 857
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 858

unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
unpack_swioma_SLQSOMADMGetSettings_t, 863
unpack_swioma_SLQSOMADMStartSession_t, 863
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 864
unpack_tmd_SLQSTmdGetMitigationDevList_t, 865
unpack_tmd_SLQSTmdGetMitigationLvl_t, 865
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t, 866
unpack_tmd_SLQSTmdRegNotMitigationLvl_t, 867
unpack_uim_ChangePin_t, 867
unpack_uim_GetCardStatus_t, 868
unpack_uim_ReadTransparent_t, 869
unpack_uim_SetPinProtection_t, 870
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 871
unpack_uim_SLQSUIMAuthenticate_t, 871
unpack_uim_SLQSUIMDepersonalization_t, 872
unpack_uim_SLQSUIMEventRegister_t, 873
unpack_uim_SLQSUIMGetConfiguration_t, 873
unpack_uim_SLQSUIMGetFileAttributes_t, 874
unpack_uim_SLQSUIMGetSlotsStatus_t, 875
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 876
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 876
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t, 877
unpack_uim_UnblockPin_t, 877
unpack_uim_VerifyPin_t, 878
unpack_voice_allCallStatusCallback_ind_t, 880
unpack_voice_DTMFEventCallback_ind_t, 881
unpack_voice_OTASPStatusCallback_ind_t, 883
unpack_voice_SLQSOriginateUSSD_t, 885
unpack_voice_SLQSVoiceAnswerCall_t, 886
unpack_voice_SLQSVoiceBurstDTMF_t, 886
unpack_voice_SLQSVoiceDialCall_t, 887
unpack_voice_SLQSVoiceEndCall_t, 888
unpack_voice_SLQSVoiceGetAllCallInfo_t, 891
unpack_voice_SLQSVoiceGetCallBarring_t, 893
unpack_voice_SLQSVoiceGetCallForwardingStatus_t, 894
unpack_voice_SLQSVoiceGetCallInfo_t, 897
unpack_voice_SLQSVoiceGetCallWaiting_t, 899
unpack_voice_SLQSVoiceGetCLIP_t, 900
unpack_voice_SLQSVoiceGetCLIR_t, 902
unpack_voice_SLQSVoiceGetCNAP_t, 903
unpack_voice_SLQSVoiceGetCOLP_t, 905
unpack_voice_SLQSVoiceGetCOLR_t, 906
unpack_voice_SLQSVoiceGetConfig_t, 908
unpack_voice_SLQSVoiceManageCalls_t, 909
unpack_voice_SLQSVoiceSendFlash_t, 909
unpack_voice_SLQSVoiceSetCallBarringPassword_t, 911
unpack_voice_SLQSVoiceSetConfig_t, 912
unpack_voice_SLQSVoiceSetSUPSService_t, 914
unpack_voice_SLQSVoiceStartContDTMF_t, 914
unpack_voice_SLQSVoiceStopContDTMF_t, 915
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 917
unpack_voice_SUPSNotificationCallback_ind_t, 920
unpack_voice_USSDNotificationCallback_ind_t, 920
unpack_voice_VoiceInfoRecCallback_ind_t, 923
unpack_voice_voicePrivacyChangeCallback_ind_t, 924
unpack_wds_DHCPv4ClientLease_ind_t, 925
unpack_wds_GetAutoconnect_t, 925
unpack_wds_GetByteTotals_t, 926
unpack_wds_GetConnectionRate_t, 927
unpack_wds_GetDataBearerTechnology_t, 928
unpack_wds_GetDefaultProfile_t, 931
unpack_wds_GetDefaultProfileNum_t, 931
unpack_wds_GetDormancyState_t, 932
unpack_wds_GetLastMobileIPError_t, 933
unpack_wds_GetMobileIP_t, 933
unpack_wds_GetMobileIPProfile_t, 935
unpack_wds_GetPacketStatistics_t, 937
unpack_wds_GetPacketStatus_t, 939
unpack_wds_GetSessionDuration_t, 940
unpack_wds_GetSessionState_t, 940
unpack_wds_RMSetTransferStatistics_t, 941
unpack_wds_SetMobileIPProfile_t, 941
unpack_wds_SLQSCreateProfile_t, 942
unpack_wds_SLQSDeleteProfile_t, 942
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 944
unpack_wds_SLQSGet3GPPConfigItem_t, 946
unpack_wds_SLQSGetCurrDataSystemStat_t, 946
unpack_wds_SLQSGetCurrentChannelRate_t, 948
unpack_wds_SLQSGetDataBearerTechnology_t, 949
unpack_wds_SLQSGetDUNCallInfo_t, 952
unpack_wds_SLQSGetProfileSettings_t, 952
unpack_wds_SLQSGetRuntimeSettings_t, 955
unpack_wds_SLQSModifyProfile_t, 956
unpack_wds_SLQSSetIPFamilyPreference_t, 957
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 959
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 965
unpack_wds_SLQSSGetLoopback_t, 966
unpack_wds_SLQSStartDataSession_t, 966
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
path
uim_fileInfo, 564

pathLen
uim_fileInfo, 564
pbAuto
unpack_pds_GetPortAutomaticTracking_t, 783
unpack_pds_GetServiceAutomaticTracking_t, 783
pbEnabled
unpack_pds_GetXTRAAutomaticDownload_t, 784
pbPlatform
unpack_dms_GetOfflineReason_t, 603
pci
nas_cellParams, 158
nas_ltePCI, 211
nas_PhyCaAggPcellInfo, 238
nas_PhyCaAggScellIndType, 239
nas_PhyCaAggScellInfo, 242
nas_umtsLTENbrCell, 290
NASPhyCaAggPcellInfo, 306
NASPhyCaAggScellArray, 308
NASPhyCaAggScellIndType, 310
NASPhyCaAggScellInfo, 311
pcsDigit
nas_PlmnID, 244
pcsfFQDNAddress
wds_PCSCFFQDNAddressList, 1042
PdpConnState
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 669
PdpStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
pdpType
pack_wds_SetDefaultProfile_t, 492
pdptype
unpack_wds_GetDefaultProfile_t, 931
pds.h, 1245
pack_pds_ForceXTRADownload, 1248
pack_pds_GetPDSDefaults, 1248
pack_pds_GetPDSState, 1249
pack_pds_GetPortAutomaticTracking, 1249
pack_pds_GetServiceAutomaticTracking, 1249
pack_pds_GetXTRAAutomaticDownload, 1250
pack_pds_GetXTRANetwork, 1250
pack_pds_GetXTRAValidity, 1250
pack_pds_PDSInjectTimeReference, 1251
pack_pds_ResetPDSData, 1251
pack_pds_SLQSGetAGPSConfig, 1254
pack_pds_SLQSGetGPSStateInfo, 1254
pack_pds_SLQSPDSDeterminePosition, 1255
pack_pds_SLQSPDSInjectAbsoluteTimeReference, 1255
pack_pds_SLQSPDSInjectPositionData, 1255
pack_pds_SLQSSetAGPSConfig, 1256
pack_pds_SLQSSetPositionMethodState, 1256
pack_pds_SetEventReportCallback, 1251
pack_pds_SetPDSDefaults, 1252
pack_pds_SetPDSState, 1252
pack_pds_SetPortAutomaticTracking, 1252
pack_pds_SetServiceAutomaticTracking, 1253
pack_pds_SetXTRAAutomaticDownload, 1253
pack_pds_SetXTRANetwork, 1253
pack_pds_StartPDSTrackingSessionExt, 1256
pack_pds_StopPDSTrackingSession, 1257
unpack_pds_ForceXTRADownload, 1257
unpack_pds_ForceXTRADownload_t, 1247
unpack_pds_GetPDSDefaults, 1257
unpack_pds_GetPDSState, 1258
unpack_pds_GetPortAutomaticTracking, 1258
unpack_pds_GetServiceAutomaticTracking, 1258
unpack_pds_GetXTRAAutomaticDownload, 1259
unpack_pds_GetXTRANetwork, 1259
unpack_pds_GetXTRAValidity, 1260
unpack_pds_PDSInjectTimeReference, 1260
unpack_pds_PDSInjectTimeReference_t, 1247
unpack_pds_ResetPDSData, 1260
unpack_pds_ResetPDSData_t, 1247
unpack_pds_SLQSGetAGPSConfig, 1264
unpack_pds_SLQSGetGPSStateInfo, 1264
unpack_pds_SLQSPDSDeterminePosition, 1264
unpack_pds_SLQSPDSDeterminePosition_t, 1248
unpack_pds_SLQSPDSInjectAbsoluteTimeReference, 1265
unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 1248
unpack_pds_SLQSPDSInjectPositionData, 1265
unpack_pds_SLQSPDSInjectPositionData_t, 1248
unpack_pds_SLQSSetAGPSConfig, 1265
unpack_pds_SLQSSetAGPSConfig_t, 1248
unpack_pds_SLQSSetPositionMethodState, 1266
unpack_pds_SLQSSetPositionMethodState_t, 1248
unpack_pds_SetEventReport_Ind, 1261
unpack_pds_SetEventReportCallback, 1261
unpack_pds_SetEventReportCallback_t, 1247
unpack_pds_SetPDSDefaults, 1261
unpack_pds_SetPDSDefaults_t, 1247
unpack_pds_SetPDSState, 1262
unpack_pds_SetPDSState_t, 1247
unpack_pds_SetPdsState_Ind, 1262
unpack_pds_SetPortAutomaticTracking, 1262
unpack_pds_SetPortAutomaticTracking_t, 1247
unpack_pds_SetServiceAutomaticTracking, 1263
unpack_pds_SetServiceAutomaticTracking_t, 1248
unpack_pds_SetXTRAAutomaticDownload, 1263
unpack_pds_SetXTRAAutomaticDownload_t, 1248
unpack_pds_SetXTRANetwork, 1263
unpack_pds_SetXTRANetwork_t, 1248
unpack_pds_StartPDSTrackingSessionExt, 1266
unpack_pds_StartPDSTrackingSessionExt_t, 1248
unpack_pds_StopPDSTrackingSession, 1266
unpack_pds_StopPDSTrackingSession_t, 1248
peakRate
unpack_qos_tokenBucket_t, 815
peakThroughputClass
LibPackGPRSRequestedQoS, 93
wds_GPRSQoS, 1039
PeriodicUpdateTimer
unpack_dms_PSMCfgChange_ind_t, 610

periodicUpdateTimer
dms_PSMPeriodicUpdateTimerTlv, 58
PeriodicUpdateTimerInd
dms_PSMPeriodicUpdateTimerIndTlv, 58
PersistentLpm
unpack_dms_SLQSDmsSwiGetPCInfo_t, 619
PersistentTechPref
unpack_nas_GetNetworkPreference_t, 710
persoFeature
appStats, 38
uim_appStatus, 553
persoRetries
appStats, 38
uim_appStatus, 553
persoState
appStats, 38
uim_appStatus, 553
persoUnblockRetries
appStats, 38
uim_appStatus, 553
pfailureCause
unpack_voice_SLQSOriginateUSSD_t, 885
PhCtxtURI
ims_PhCtxtURIInfo, 78
phase
nas_rxInfo, 267
PhyCaAggPcellInfo
NasGetLTECphyCaInfo, 301
PhyCaAggScellArray
NasGetLTECphyCaInfo, 301
PhyCaAggScellDlBw
NasGetLTECphyCaInfo, 301
PhyCaAggScellIndType
NasGetLTECphyCaInfo, 301
PhyCaAggScellIndex
NasGetLTECphyCaInfo, 301
PhyCaAggScellInfo
NasGetLTECphyCaInfo, 301
PhysicalLayer
nas_protocolSubtypeElement, 248
PilotEnergy
nas_NetworkStatEVDO, 228
PilotPN
nas_PilotSetParams, 243
PilotStrength
nas_PilotSetParams, 243
PilotType
nas_PilotSetParams, 243
pin1Len
uim_encryptedPIN1, 560
pin1Retries
appStats, 38
uim_appStatus, 553
pin1State
appStats, 38
uim_appStatus, 553
pin1Val
uim_encryptedPIN1, 560
pin2Retries
appStats, 38
uim_appStatus, 553
pin2State
appStats, 38
uim_appStatus, 554
pinID
uim_changeUIMPIN, 558
uim_setPINProtection, 571
uim_unblockUIMPIN, 574
uim_verifyUIMPIN, 575
pinLen
uim_changeUIMPIN, 558
uim_verifyUIMPIN, 575
pinLength
uim_setPINProtection, 571
pinOperation
uim_setPINProtection, 571
pinProtection
pack_uim_SetPinProtection_t, 457
pack_uim_UnblockPin_t, 465
pinVal
uim_changeUIMPIN, 558
uim_verifyUIMPIN, 575
pinValue
uim_setPINProtection, 571
PkgDescLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
PkgDescription
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
PkgName
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
PkgNameLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
pkgver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 633
PktErrRate
unpack_qos_swiQosFlow_t, 814
plasmaIDString
pack_dms_SLQSSwiSetHostDevInfo_t, 338
unpack_dms_SLQSSwiGetHostDevInfo_t, 636
plmn
nas_GERANInfo, 178
nas_LTEInfoIntrafreq, 203
nas_UMTSInfo, 288
PlmnLen
nas_QmisNasSlqsNasPCICellInfo, 252
polarityIncluded
voice_lineCtrlInfo, 1015
port
loc_IPv4Info, 133
loc_IPv6Info, 134
unpack_qos_Port_t, 797
posDataNmea

pack_pds_SetEventReportCallback_t, 410
posSrc
unpack_pds_SetEventReport_Ind_t, 787
PositionDataNMEA
unpack_pds_SetEventReport_Ind_t, 787
positionSrc
pack_loc_SLQSLOCInjectPosition_t, 364
PowerOffMode
unpack_dms_SLQSDmsSwiGetPCInfo_t, 619
pprofileid3gpp
pack_wds_SLQSStartDataSession_t, 507
pprofileid3gpp2
pack_wds_SLQSStartDataSession_t, 507
prDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 969
prDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 969
prPCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 969
prPCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 969
Precedence
unpack_qos_swiQosFilter_t, 810
precedenceClass
LibPackGPRSRequestedQoS, 93
wds_GPRSQoS, 1039
prefNetwork
unpack_wds_SLQSGetCurrDataSystemStat_t, 947
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
preference
pack_pds_SetXTRANetwork_t, 413
Preferred
nas_QmiNas3GppNetworkInfo, 250
prefixLen
unpack_qos_IPv6Addr_t, 796
presentationInd
voice_ECTNum, 1011
voice_remotePartyNum, 1023
priCSCFPort
ims_PCSCFPortInfo, 77
priChA
nas_CDMAChannel, 151
priChB
nas_CDMAChannel, 151
priSize
unpack_dms_GetFirmwareRevisions_t, 598
pridns
unpack_wds_GetDefaultProfile_t, 931
pridnsv6
unpack_wds_GetDefaultProfile_t, 931
primaryDNS
pack_wds_SetDefaultProfile_t, 492
PrimaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 955
PrimaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 955
primaryHA
unpack_wds_GetMobileIPProfile_t, 935
privacyPref
pack_voice_SLQSVoiceSetPreferredPrivacy_t, 482
priver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 633
priversion_str
unpack_dms_GetFirmwareInfo_t, 596
Profile
pack_audio_SLQSGetAudioPathConfig_t, 319
pack_audio_SLQSGetAudioVolTLBConfig_t, 320
pack_audio_SLQSSetAudioPathConfig_t, 322
pack_audio_SLQSSetAudioProfile_t, 323
pack_audio_SLQSSetAudioVolTLBConfig_t, 324
pack_swiaudio_SLQSGetM2MAudioVolume_t, 433
pack_swiaudio_SLQSGetM2MAVMute_t, 434
pack_swiaudio_SLQSGetM2MSpkrGain_t, 434
pack_swiaudio_SLQSSetM2MAudioAVCFG_t, 435
pack_swiaudio_SLQSSetM2MAudioProfile_t, 436
pack_swiaudio_SLQSSetM2MAudioVolume_t, 437
pack_swiaudio_SLQSSetM2MAVMute_t, 438
pack_swiaudio_SLQSSetM2MSpkrGain_t, 438
unpack_audio_SLQSGetAudioProfile_t, 579
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 840
ProfileID
unpack_wds_SLQSGetRuntimeSettings_t, 955
ProfileId
pack_wds_SLQSGetProfileSettings_t, 500
profileId
wds_DHCPProfileIdTlv, 1034
wds_DHCPv4ProfileId, 1037
wdsDhcpv4ProfileId, 1051
ProfileId3GPP2
unpack_qos_swiQosFlow_t, 814
ProfileIdTlv
unpack_wds_DHCPv4ClientLease_ind_t, 925
ProfileIndex
PackCreateProfileOut, 511
profileIndex
pack_wds_SLQSDeleteProfile_t, 497
wds_ProfileIdentifier, 1043
profileList
unpack_wds_SLQSGet3GPPConfigItem_t, 946
ProfileName
unpack_wds_SLQSGetRuntimeSettings_t, 955
ProfileType
pack_wds_SLQSGetProfileSettings_t, 500
PackCreateProfileOut, 511
unpack_wds_SLQSGetProfileSettings_t, 952
profileType
pack_wds_SetDefaultProfile_t, 492
pack_wds_SLQSDeleteProfile_t, 497

wds_DHCPProfileIdTlv, 1034
wds_DHCPv4ProfileId, 1037
wds_ProfileIdentifier, 1044
wdsDhcpv4ProfileId, 1051
profiletype
pack_wds_GetDefaultProfile_t, 486
PromptFwDownload
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
PromptFwUpdate
pack_swiavms_SLQSAVMSSetSettings_t, 441
pack_swiavms_SLQSAVMSSetSettings_v2_t, 443
Protocol
unpack_nas_GetCDMANetworkParameters_t, 705
ProvisionStatus
voice_CLIPResp, 1002
voice_CLIRResp, 1003
voice_CNAPResp, 1004
voice_COLPResp, 1004
voice_COLRResp, 1005
psAttachState
nas_servSystem, 272
NASServingSystemInfo, 317
psBarStatus
nas_CallBarringSysInfo, 149
nas_callBarStatus, 150
psState
nas_CommInfo, 162
psc
nas_UMTSInfo, 288
nas_wcdmaCellInfo, 292
nas_WCDMASysInfo, 297
pscValid
nas_WCDMASysInfo, 297
pscsfIPv4Addr
wds_PCSCFIPv4ServerAddressList, 1043
psetting
unpack_wds_GetAutoconnect_t, 925
psid
pack_wds_SLQSStopDataSession_t, 508
unpack_wds_SLQSStartDataSession_t, 966
puk1Retries
appStats, 38
uim_appStatus, 554
puk2Retries
appStats, 38
uim_appStatus, 554
pukLen
uim_unblockUIMPIN, 574
pukVal
uim_unblockUIMPIN, 575
pukValue
pack_dms_UIMUnblockPIN_t, 344
pwrDenialTime
voice_lineCtrlInfo, 1015

QMI_LITE_WDS_CURRENT_CALL_DB_MASK
wds.h, 1463
QMI_LITE_WDS_LAST_CALL_DB_MASK
wds.h, 1463
QCI
LibPackQosClassID, 118
QFlowState
unpack_qos_QosFlowInfo_t, 799
QLIC
nas_DeviceConfigDetail, 169
QMI pack/unpack (lite-qmi), 29
qaGobiApiTableBandClasses.h, 1267
qaGobiApiTableCallControlReturnReasons.h, 1270
qaGobiApiTableCallEndReasons.h, 1271
qaGobiApiTableCarrierCodes.h, 1287
qaGobiApiTableCodingScheme.h, 1289
qaGobiApiTableGpsCapabilityCodes.h, 1292
qaGobiApiTablePowerModes.h, 1293
qaGobiApiTableRadioInterfaces.h, 1293
qaGobiApiTableRegionCodes.h, 1294
qaGobiApiTableServiceOptions.h, 1294
qaGobiApiTableSupServiceInfoClasses.h, 1297
qaGobiApiTableSwiAudio.h, 1297
qaGobiApiTableSwiOMADMSessionStatus.h, 1298
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h, 1299
qaGobiApiTableVoiceCallEndReasons.h, 1300
qm_wds_ds_profile_extended_err_codes
qmerrno.h, 1313
qmTlvResult, 517
DeviceError, 517
DeviceResult, 517
TlvPresenceMask, 517
TlvResultCode, 517
qmerrno.h
eQCWWAN_ERR_API_MUTEX_TIMEOUT, 1310
eQCWWAN_ERR_BUFFER_SZ, 1309
eQCWWAN_ERR_CANCEL_OP, 1310
eQCWWAN_ERR_DRIVER, 1310
eQCWWAN_ERR_ENUM_BEGIN, 1309
eQCWWAN_ERR_ENUM_END, 1310
eQCWWAN_ERR_FILE_COPY, 1309
eQCWWAN_ERR_FILE_OPEN, 1309
eQCWWAN_ERR_GENERAL, 1309
eQCWWAN_ERR_INTERNAL, 1309
eQCWWAN_ERR_INVALID_ARG, 1309
eQCWWAN_ERR_INVALID_DEVID, 1309
eQCWWAN_ERR_INVALID_FILE, 1309
eQCWWAN_ERR_INVALID_QMI_RSP, 1309
eQCWWAN_ERR_INVALID_XID, 1310
eQCWWAN_ERR_MALFORMED_QMI_RSP, 1309
eQCWWAN_ERR_MEMORY, 1309
eQCWWAN_ERR_MULTIPLE_DEVICES, 1310
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED, 1310
eQCWWAN_ERR_NO_CANCELABLE_OP, 1310
eQCWWAN_ERR_NO_CONNECTION, 1309
eQCWWAN_ERR_NO_DEVICE, 1309
eQCWWAN_ERR_NO_SIGNAL, 1310
eQCWWAN_ERR_NONE, 1309

eQCWWAN_ERR_NULL_TLV, 1313
eQCWWAN_ERR_OFFLINE, 1310
eQCWWAN_ERR_PDU_GENERATION, 1310
eQCWWAN_ERR_QMI_ABORTED, 1310
eQCWWAN_ERR_QMI_ACCESS_DENIED, 1312
eQCWWAN_ERR_QMI_ACK_NOT_SENT, 1312
eQCWWAN_ERR_QMI_ARG_TOO_LONG, 1310
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED, 1311
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK, 1311
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED, 1312
eQCWWAN_ERR_QMI_CALL_FAILED, 1310
eQCWWAN_ERR_QMI_CARD_BUSY_RSP, 1313
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED, 1312
eQCWWAN_ERR_QMI_CAT_END, 1313
eQCWWAN_ERR_QMI_CAT_START, 1313
eQCWWAN_ERR_QMI_CAUSE_CODE, 1311
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED, 1310
eQCWWAN_ERR_QMI_CONNECT, 1309
eQCWWAN_ERR_QMI_DEVICE_IN_USE, 1310
eQCWWAN_ERR_QMI_DEVICE_NOT_READY, 1311
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL, 1311
eQCWWAN_ERR_QMI_DISABLED, 1311
eQCWWAN_ERR_QMI_ENCODING, 1311
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE, 1313
eQCWWAN_ERR_QMI_EVENT_REG_FAILED, 1313
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL, 1312
eQCWWAN_ERR_QMI_FDN_RESTRICT, 1312
eQCWWAN_ERR_QMI_FLOW_SUSPENDED, 1311
eQCWWAN_ERR_QMI_GENERAL, 1311
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED, 1312
eQCWWAN_ERR_QMI_IFACE, 1309
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE, 1312
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER, 1311
eQCWWAN_ERR_QMI_INCORRECT_PIN, 1310
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE, 1312
eQCWWAN_ERR_QMI_INJECT_TIMEOUT, 1312
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES, 1311
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND, 1311
eQCWWAN_ERR_QMI_INTERNAL, 1310
eQCWWAN_ERR_QMI_INVALID_ARG, 1311
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID, 1310
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT, 1311
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD, 1313
eQCWWAN_ERR_QMI_INVALID_HANDLE, 1310
eQCWWAN_ERR_QMI_INVALID_ID, 1311
eQCWWAN_ERR_QMI_INVALID_INDEX, 1311
eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF, 1311
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE, 1311
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID, 1311
eQCWWAN_ERR_QMI_INVALID_OPERATION, 1312
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE, 1310
eQCWWAN_ERR_QMI_INVALID_PINID, 1310
eQCWWAN_ERR_QMI_INVALID_PROFILE, 1310
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE, 1310
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION, 1311
eQCWWAN_ERR_QMI_INVALID_QMI_CMD, 1312
eQCWWAN_ERR_QMI_INVALID_QOS_ID, 1311
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION, 1311
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE, 1310
eQCWWAN_ERR_QMI_INVALID_TECH_PREF, 1310
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP, 1313
eQCWWAN_ERR_QMI_INVALID_TRANSITION, 1311
eQCWWAN_ERR_QMI_INVALID_TX_ID, 1310
eQCWWAN_ERR_QMI_MALFORMED_MSG, 1310
eQCWWAN_ERR_QMI_MAX, 1312
eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE, 1311
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE, 1311
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE, 1311
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT, 1311
eQCWWAN_ERR_QMI_MISSING_ARG, 1310
eQCWWAN_ERR_QMI_MSG_BLOCKED, 1312
eQCWWAN_ERR_QMI_NETWORK_ABORTED, 1312
eQCWWAN_ERR_QMI_NETWORK_NOT_READY, 1311
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE, 1311

eQCWWAN_ERR_QMI_NO_EFFECT, 1310
eQCWWAN_ERR_QMI_NO_ENTRY, 1311
eQCWWAN_ERR_QMI_NO_FREE_PROFILE, 1310
eQCWWAN_ERR_QMI_NO_MEMORY, 1310
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND, 1310
eQCWWAN_ERR_QMI_NO_RADIO, 1312
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION, 1312
eQCWWAN_ERR_QMI_NO_THRESHOLDS, 1310
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE, 1311
eQCWWAN_ERR_QMI_NOT_PROVISIONED, 1310
eQCWWAN_ERR_QMI_NOT_SUPPORTED, 1312
eQCWWAN_ERR_QMI_OFFSET, 1310
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED, 1310
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED, 1310
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE, 1312
eQCWWAN_ERR_QMI_OUT_OF_CALL, 1310
eQCWWAN_ERR_QMI_PIN_BLOCKED, 1311
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED, 1311
eQCWWAN_ERR_QMI_POLICY_MISMATCH, 1312
eQCWWAN_ERR_QMI_REQ, 1309
eQCWWAN_ERR_QMI_REQ_SCH, 1309
eQCWWAN_ERR_QMI_REQ_TO, 1309
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED, 1311
eQCWWAN_ERR_QMI_RSP, 1309
eQCWWAN_ERR_QMI_RSP_TO, 1309
eQCWWAN_ERR_QMI_SEGMENT_ORDER, 1312
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG, 1312
eQCWWAN_ERR_QMI_SESSION_INACTIVE, 1311
eQCWWAN_ERR_QMI_SESSION_INVALID, 1311
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP, 1311
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND, 1312
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED, 1311
eQCWWAN_ERR_QMI_SMSC_ADDR, 1312
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE, 1312
eQCWWAN_ERR_QMI_TPDU_TYPE, 1312
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION, 1310
eQCWWAN_ERR_QMI_UNKNOWN, 1311
eQCWWAN_ERR_QMI_WIDTH, 1313
eQCWWAN_ERR_RESET, 1310
eQCWWAN_ERR_SWICM_AM_VERS_ERROR, 1312
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS, 1312
eQCWWAN_ERR_SWICM_END, 1313
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS, 1312
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID, 1312
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID, 1312
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID, 1312
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED, 1312
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED, 1312
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED, 1312
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS, 1312
eQCWWAN_ERR_SWICM_SOCKET_IN_USE, 1312
eQCWWAN_ERR_SWICM_START, 1312
eQCWWAN_ERR_SWICM_TIMEOUT, 1312
eQCWWAN_ERR_SWICM_V4DWN_V6DWN, 1312
eQCWWAN_ERR_SWICM_V4DWN_V6UP, 1312
eQCWWAN_ERR_SWICM_V4UP_V6DWN, 1312
eQCWWAN_ERR_SWICM_V4UP_V6UP, 1312
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED, 1313
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND, 1313
eQCWWAN_ERR_SWIDCS_END, 1313
eQCWWAN_ERR_SWIDCS_FILEIO_ERR, 1313
eQCWWAN_ERR_SWIDCS_IOCTL_ERR, 1313
eQCWWAN_ERR_SWIDCS_START, 1313
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE, 1313
eQCWWAN_ERR_SWIIM_END, 1313
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND, 1313
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED, 1313
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE, 1313
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE, 1313
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX, 1313
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH, 1313
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE, 1313

eQCWWAN_ERR_SWIIM_FW_SECBOOT_IMAGE_NOT_SIGNED, 1313
eQCWWAN_ERR_SWIIM_FW_SECBOOT_INVALID_CERT_CHAIN, 1313
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL, 1313
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS, 1313
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT, 1313
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE, 1313
eQCWWAN_ERR_SWIIM_INVALID_PATH, 1313
eQCWWAN_ERR_SWIIM_OPENING_DIR, 1313
eQCWWAN_ERR_SWIIM_OPENING_FILE, 1313
eQCWWAN_ERR_SWIIM_START, 1313
eQCWWAN_ERR_SWISM_END, 1313
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND, 1313
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED, 1313
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG, 1313
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED, 1313
eQCWWAN_ERR_SWISMS_START, 1313
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE, 1314
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR, 1314
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED, 1314
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES, 1314
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY, 1314
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET, 1314
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET, 1314
eWDS_ERR_PROFILE_REG_END, 1314
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID, 1314
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED, 1314
eWDS_ERR_PROFILE_REG_RESULT_FAIL, 1314
eWDS_ERR_PROFILE_REG_RESULT_LIST_END, 1314
qmerrno.h, 1307
eQCWWANError, 1309
qm_wds_ds_profile_extended_err_codes, 1313
qmiSmsMessageList, 516
messageIndex, 516
messageTag, 516
qmiWDSDataBearerTechnology, 516
currentNetwork, 517
ratMask, 517
soMask, 517
qmuint16
SwiDataTypes.h, 1375
qmulong
SwiDataTypes.h, 1375
qos.h, 1314
pack_qos_BindDataPort, 1316
pack_qos_SLQSQosGetNetworkStatus, 1316
pack_qos_SLQSQosSwiReadApnExtraParams, 1317
pack_qos_SLQSQosSwiReadDataStats, 1318
pack_qos_SLQSSetQosEventCallback, 1318
unpack_qos_BindDataPort, 1319
unpack_qos_SLQSQosGetNetworkStatus, 1319
unpack_qos_SLQSQosSwiReadApnExtraParams, 1320
unpack_qos_SLQSQosSwiReadDataStats, 1320
unpack_qos_SLQSSetQosEventCallback, 1321
unpack_qos_SLQSSetQosEventCallback_ind, 1321
unpack_qos_SLQSSetQosNWStatusCallback_ind, 1322
unpack_qos_SLQSSetQosPriEventCallback_ind, 1322
unpack_qos_SLQSSetQosStatusCallback_ind, 1323
qos_BindDataPortMuxID_t, 517
MuxID, 518
qos_BindDataPortPeripheralEndPointID_t, 518
EndPointType, 518
IfaceID, 518
qos_BindDataPortSIODataPort_t, 518
SIODataPort, 519
qosDeliveryOrder
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
qosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 803
QosFlowInfo
unpack_qos_SLQSSetQosEventCallback_ind_t, 804

RAN
unpack_nas_GetServingNetwork_t, 714
RAT
nas_QmiNas3GppNetworkRAT, 251
RATMask
currNetworkInfo, 52
wds_currNetworkInfo, 1030
RBTTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
RFBandInfoElements, 519
activeBandClass, 519
activeChannel, 519
radioInterface, 519
unpack_nas_GetRFInfo_t, 711
RFTlv
unpack_nas_SetEventReportInd_t, 719
RMAutoConnect
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
RPCause
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
RPTlv
NASQmiCbkNasSystemSelPrefInd, 314
RRTlv
unpack_nas_SetEventReportInd_t, 719
RSRPThresListLen
nas_RSRPThresh, 264
RSRQThresListLen
nas_RSRQThresh, 265
RSSIThresListLen
nas_RSSIThresh, 266
RTIDTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
RTTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
RX_EC_IO
nas_NetworkStat1x, 226
RX_PWR
nas_NetworkStat1x, 226
nas_NetworkStatEVDO, 228
RXBOTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 944
RXChan
nas_LTEInfo, 200
rXDroppedCount
unpack_wds_GetPacketStatus_t, 939
rXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 939
rXOkBytesCount
unpack_wds_GetPacketStatus_t, 939
rXPacketErrors
unpack_wds_GetPacketStatus_t, 939
rXPacketOverflows
unpack_wds_GetPacketStatus_t, 939
rXPacketSuccesses
unpack_wds_GetPacketStatus_t, 939
radio
unpack_nas_GetSignalStrengths_t, 715
radio_if
pack_nas_SLQSNasGetTxRxInfo_t, 384
radioIf
nas_ecioListElement, 171
nas_errorRateListElement, 176
nas_rsrqInformation, 264
nas_rxSignalStrengthListElement, 269
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 760
RadioIfaces
unpack_dms_GetDeviceCap_t, 589
unpack_dms_GetDeviceCapabilities_t, 591
unpack_nas_GetServingNetwork_t, 714
RadioIfacesSize
unpack_dms_GetDeviceCap_t, 589
unpack_nas_GetServingNetwork_t, 714
radioIfacesSize
unpack_dms_GetDeviceCapabilities_t, 591
radioInterface
nas_RFBandInfoElements, 256
nas_RfBandInfoExtFormatElements, 257
nas_RfBandwidthInfoElements, 259
nas_RfDedicatedBandInfoElements, 261
nas_RFInfoTlv, 262
nas_roamIndList, 263
nas_servSystem, 272
nas_SignalStrengthTlv, 273
nas_timeInfo, 285
RFBandInfoElements, 519
radioInterfaceList
NASServingSystemInfo, 317
radioInterfaceNo
NASServingSystemInfo, 317
radioInterfaceSize
nas_RFInfoTlv, 262
randomizationWindow
dms_PSMRandomizationWindowTlv, 59
range
unpack_qos_Port_t, 797
rankIndicatorTlv
unpack_nas_SLQSSwiRandIndicatorCallback_-Ind_t, 772
rat
nas_CSGID, 163
nas_MNRInfo, 222
RatDMTlv
NASQmiCbkNasSystemSelPrefInd, 314
RatDisabledMask
NASRatDisabledMaskTlv, 315
ratDisabledMask
nas_RatDisabledMaskTlv, 255
RatHandover
unpack_imsa_SLQSImsaRatStatusCallBack_ind_-t, 670
RatHandoverStatus
imsa_RatHandoverStatusInfo, 87

RatHandoverStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
ratMask
qmiWDSDataBearerTechnology, 517
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
rawHorConfidence
pack_loc_SLQSLOCInjectPosition_t, 364
rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 364
rawLen
uim_fileAttributes, 563
rawValue
uim_fileAttributes, 563
readTransparent
pack_uim_ReadTransparent_t, 456
Reason
pack_voice_SLQSVoiceGetCallForwardingStatus_t, 474
pack_voice_SLQSVoiceSetCallBarringPassword_t, 480
reason
pack_voice_SLQSVoiceGetCallBarring_t, 473
pack_voice_SLQSVoiceSetSUPSService_t, 484
unpack_qos_SLQSSetQosStatusCallback_ind_t, 807
voice_ccSUPSType, 1001
voice_redirNumInfo, 1021
receiptAction
sms_routeEntry, 533
reconfigReqd
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 959
recordCount
uim_fileAttributes, 563
recordSize
uim_fileAttributes, 563
RedirPartyNum
voice_arrRedirPartyNum, 988
refData
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 635
refID
pack_cat_CATSendTerminalResponse_t, 326
refString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 635
ReferenceID
cat_AlPhaIdentifierTlv, 46
cat_EventIDDataTlv, 48
refpn
nas_CDMAInfo, 153
refreshComplete
pack_uim_SLQSUIMRefreshComplete_t, 462
refreshEvent
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 876
RefreshMode
cat_RefreshTlv, 49
RefreshStage
cat_RefreshTlv, 49
regAction
pack_nas_SLQSInitiateNetworkRegistration_t, 377
RegForeignNID
unpack_nas_GetCDMANetworkParameters_t, 705
RegForeignSID
unpack_nas_GetCDMANetworkParameters_t, 705
RegHomeSID
unpack_nas_GetCDMANetworkParameters_t, 706
regInd
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 835
regPrd
nas_AddCDMASysInfo, 146
regRefresh
pack_uim_SLQSUIMRefreshRegister_t, 463
regRejectInfoValid
nas_GSMSysInfo, 184
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 235
nas_WCDMASysInfo, 297
regState
nas_servSystem, 272
RegStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
registerFlag
uim_registerRefresh, 569
RegistrationState
unpack_nas_GetServingNetwork_t, 714
registrationState
NASServingSystemInfo, 317
rejCause
nas_GSMSysInfo, 184
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 235
nas_WCDMASysInfo, 297
rejectCause
nas_RejectReasonTlv, 255
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t, 760
rejectSrvDomain
nas_GSMSysInfo, 185
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 235
nas_WCDMASysInfo, 297
reliabilityClass
LibPackGPRSRequestedQoS, 93
wds_GPRSQoS, 1039
remPartyNumber
voice_remotePartyNum, 1023
RemotePartyName
voice_getAllCallRmtPtyName, 1013
RemotePartyNum
voice_getAllCallRmtPtyNum, 1013
ReqMitigationLvl
unpack_tmd_SLQSTmdGetMitigationLvl_t, 865
resBerRatio
LibPackUMTSQoS, 122

wds_UMTSMinQoS, 1049
ResCode
unpack_audio_SLQSGetAudioVolTLBConfig_t, 579
unpack_audio_SLQSSetAudioVolTLBConfig_t, 580
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 635
resetInfoInd
pack_dms_SLQSDmsSwiIndicationRegister_t, 335
resultcode
unpack_swiavms_SLQSAVMSEventReportInd_t, 843
unpack_swiavms_SLQSAVMSGetSettings_t, 846
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 848
unpack_swiavms_SLQSAVMSSendSelection_t, 849
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 850
unpack_swiavms_SLQSAvmsSetEventReport_t, 850
unpack_swiavms_SLQSAVMSSetSettings_t, 850
unpack_swiavms_SLQSAVMSSetSettings_v2_t, 851
unpack_swiavms_SLQSAVMSStartSession_t, 851
unpack_swiavms_SLQSAVMSStopSession_t, 852
RetryCount
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
revPolarity
voice_lineCtrlInfo, 1015
revTunneling
unpack_wds_GetMobileIPProfile_t, 935
ReverseMac
nas_protocolSubtypeElement, 248
RfSarState
pack_sar_SLQSSetRfSarState_t, 423
rfbandInfoList
unpack_nas_SLQSNasGetRFInfo_t, 755
RingBkTmr
ims_RngBkTmrInfo, 78
RingTmr
ims_RngTmrInfo, 79
RmTrasnferStaticsReq
pack_wds_RMSetTransferStatistics_t, 489
rmTrasnferStaticsReq, 519
bResetStatistics, 520
ulMask, 520
rms.h, 1324
pack_rms_GetSMSWake, 1324
pack_rms_SetSMSWake, 1325
unpack_rms_GetSMSWake, 1325
unpack_rms_SetSMSWake, 1325
RmtPtyNum
voice_arrRemotePartyNum, 989
roamIndicator
nas_roamIndList, 263
RoamIndicatorVal
unpack_nas_SLQSGetServingSystem_t, 733
roamOrigVoiceSO
voice_prefVoiceSO, 1020
RoamPref
nas_RoamPrefTlv, 263
NASRoamPreferenceTlv, 315
roamStatus
nas_NR5GSystemInfoTlv, 235
nas_sysInfoCommon, 280
roamStatusValid
nas_NR5GSystemInfoTlv, 235
nas_sysInfoCommon, 280
roamTimerValue
voice_roamTimer, 1024
Roaming
nas_QmiNas3GppNetworkInfo, 250
unpack_nas_GetCDMANetworkParameters_t, 706
unpack_nas_GetServingNetwork_t, 714
roaming
unpack_nas_SetRoamingIndicatorCallback_ind_t, 721
RoamingIndicatorList
unpack_nas_SLQSGetServingSystem_t, 733
routeList
sms_setRoutesReq, 537
routeStorage
sms_routeEntry, 533
RptIMSCapability
pack_dms_SetIndicationRegister_t, 333
rptPosData
pack_pds_SetEventReportCallback_t, 410
rptRate
nas_LTESigRptCfg, 214
nas_LTESigRptConfig, 215
rscp
nas_rxInfo, 267
nas_UMTSInfo, 288
tdscdmaSigInfoExt, 548
rsrp
lteSSInfo, 142
nas_cellParams, 158
nas_QmisNasSlqsNasPCIInfo, 253
nas_rxInfo, 267
nas_RxSigInfo, 268
nas_umtsLTENbrCell, 290
rsrpRx0
nas_QmisNasSlqsNasPCIInfo, 253
rsrpRx1
nas_QmisNasSlqsNasPCIInfo, 253
rsrplevel
nas_lteRsrpinformation, 212
rsrq
lteSSInfo, 143
nas_cellParams, 158
nas_QmisNasSlqsNasPCIInfo, 254
nas_rsrqInformation, 264
nas_SccRxInfo, 270
nas_umtsLTENbrCell, 290

rsrqDelta
nas_SLQSSignalStrengthsIndReq, 275
rsrqInfo
nas_SLQSSignalStrengthsInformation, 276
unpack_nas_SLQSGetSignalStrength_t, 736
rsrqRx0
nas_QmisNasSlqsNasPCIInfo, 254
rsrqRx1
nas_QmisNasSlqsNasPCIInfo, 254
rssi
cdmaSSInfo, 49
hdrSSInfo, 71
lteSSInfo, 143
nas_cellParams, 158
nas_gsmCellInfo, 180
tdscdmaSigInfoExt, 548
unpack_nas_GetSignalStrengths_t, 715
rx_bytes
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
rx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
RxByteOKCnt
wds_RXBytesOKTlv, 1044
rxChainIndex
nas_RxSigInfo, 268
RxDropConutTlv
unpack_RMTransferStatistics_ind_t, 818
rxLev
nas_GERANInfo, 178
rxOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 952
RxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 818
RxOkConutTlv
unpack_RMTransferStatistics_ind_t, 819
rxPower
nas_rxInfo, 267
nas_RxSigInfo, 268
RxQFilter
unpack_qos_QosFlowInfo_t, 799
RxQFlowGranted
unpack_qos_QosFlowInfo_t, 799
rxSignalStrength
nas_rxSignalStrengthListElement, 269
rxSignalStrengthDelta
nas_SLQSSignalStrengthsIndReq, 275
rxSignalStrengthInfo
nas_SLQSSignalStrengthsInformation, 276
rxSignalStrengthList
unpack_nas_SLQSGetSignalStrength_t, 736
rxSignalStrengthListLen
unpack_nas_SLQSGetSignalStrength_t, 736

SCI
unpack_nas_GetCDMANetworkParameters_t, 706
SCM
unpack_nas_GetCDMANetworkParameters_t, 706
SCTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
SDPTlv
NASQmiCbkNasSystemSelPrefInd, 314
SDTlv
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 665
SDU_HDR_LEN
common.h, 1065
SFTlv
unpack_ims_SLQSSMSCfgCallBack_ind_t, 662
SHORT
SwiDataTypes.h, 1375
SI
voice_calledPartyInfo, 992
voice_callFWExtInfo, 997
voice_callingPartyInfo, 1001
voice_redirNumInfo, 1021
SINTlv
unpack_ims_SLQSSMSCfgCallBack_ind_t, 662
SIODataPort
qos_BindDataPortSIODataPort_t, 519
SIPLocalPort
ims_SIPPortInfo, 81
sIntraSearch
nas_LTEInfoIntrafreq, 203
SLQSSSTlv
unpack_nas_SetEventReportInd_t, 719
SMS_CONFIG_LEN
sms.h, 1331
SMS_MAX_SMS_ROUTES
sms.h, 1332
SMS_NUM_OF_SET
sms.h, 1332
SMSC_TYPE_LEN
sms.h, 1332
sMSCAddressInfo, 538
data, 538
length, 538
sMSCAddressTlv, 538
SMSCInfo, 538
TlvPresent, 539
SMSCInfo
sMSCAddressTlv, 538
SMSCTlv
unpack_sms_SetNewSMSCallback_ind_t, 823
sMSEtwsMessageInfo, 539
data, 539
length, 539
notificationType, 539
sMSEtwsMessageTlv, 539
EtwsMessageInfo, 540
TlvPresent, 540
sMSEtwsPlmnInfo, 540
mobileCountryCode, 540
mobileNetworkCode, 540
sMSMTMessageInfo, 541
messageIndex, 541
storageType, 541

sMSMessageModeInfo, 540
messageMode, 541
sMSOnIMSInfo, 541
smsOnIMS, 541
sMSOnIMSTlv, 542
IMSInfo, 542
TlvPresent, 542
SMSSupport
pack_dms_SetCustFeature_t, 331
unpack_dms_GetCustFeature_t, 587
sMSTransferRouteMTMessageInfo, 542
ackIndicator, 543
data, 543
format, 543
length, 543
transactionID, 543
SNR
nas_NetworkStatEVDO, 228
sNonIntraSearch
nas_LTEInfoIntrafreq, 203
SO
nas_NetworkStat1x, 226
SOMask
currNetworkInfo, 52
wds_currNetworkInfo, 1030
SPC_SIZE
dms.h, 1077
SPTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
sPhyCaAggPcellInfo
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 720
sPhyCaAggScellDlBw
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 720
sPhyCaAggScellIndType
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 720
sPhyCaAggScellIndex
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 720
sPhyCaAggScellInfo
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 720
SRTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
SSInfo
unpack_nas_SetServingSystemCallback_ind_t, 722
sSLQSSignalStrengthsInfo
nas_SLQSSignalStrengthsTlv, 277
SSTlv
unpack_nas_SetEventReportInd_t, 719
STTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
SUPSInformation
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 918
SUPSType
pack_voice_SLQSVoiceManageCalls_t, 478
SWI_API
SwiDataTypes.h, 1375
SWIWWANCMAPI.h, 1399
samplesPerBatch
loc_accelAcceptReady, 124
loc_accelTempAcceptReady, 125
loc_gyroAcceptReady, 131
loc_gyroTempAcceptReady, 131
sar.h, 1326
pack_sar_SLQSGetRfSarState, 1326
pack_sar_SLQSSetRfSarState, 1326
unpack_sar_SLQSGetRfSarState, 1327
unpack_sar_SLQSSetRfSarState, 1327
unpack_sar_SLQSSetRfSarState_t, 1326
sbas_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
sbas_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
sbas_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
sbas_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 793
scell_idx
nas_PhyCaAggScellIndex, 239
NASPhyCaAggScellArray, 308
NASPhyCaAggScellIndex, 309
scell_state
nas_PhyCaAggScellIndType, 239
nas_PhyCaAggScellInfo, 242
NASPhyCaAggScellArray, 308
NASPhyCaAggScellIndType, 310
NASPhyCaAggScellInfo, 311
scrAmrEnable
ims_EnabSCRAMRInfo, 75
scrAmrWBEnable
ims_EnabSCRAMRWBInfo, 75
screeningInd
voice_connectNumInfo, 1007
sduErrorRatio
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
seDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
seDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
sePCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
sePCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 969
secActivate
uim_fileAttributes, 563
secActivateMask

uim_fileAttributes, 563
secChA
nas_CDMAChannel, 151
secChB
nas_CDMAChannel, 151
secDeactivate
uim_fileAttributes, 563
secDeactivateMask
uim_fileAttributes, 563
secIncrease
uim_fileAttributes, 563
secIncreaseMask
uim_fileAttributes, 563
SecProt
nas_protocolSubtypeElement, 248
secRead
uim_fileAttributes, 563
secReadMask
uim_fileAttributes, 563
secWrite
uim_fileAttributes, 563
secWriteMask
uim_fileAttributes, 563
secdns
unpack_wds_GetDefaultProfile_t, 931
secdnsv6
unpack_wds_GetDefaultProfile_t, 931
second
nas_timeInfo, 285
nas_UniversalTime, 291
secondaryDNS
pack_wds_SetDefaultProfile_t, 492
SecondaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 955
SecondaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 955
secondaryHA
unpack_wds_GetMobileIPProfile_t, 935
SectorIDLen
nas_NetworkStatEVDO, 228
selNetwork
nas_servSystem, 272
selected
sms_BroadcastConfig, 525
sms_CDMABroadcastConfig, 526
selectedNetwork
NASServingSystemInfo, 317
selection
pack_swiavms_SLQSAVMSSendSelection_t, 439
pack_swioma_SLQSOMADMSendSelection_t, 449
pack_swioma_SLQSOMADMSendSelectionExt_t, 450
sendStatus
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
sensorData_t, 520
flags, 521
sensorDataLen, 521
timeOfFirstSample, 521
timeOffset, 521
xAxis, 521
yAxis, 521
zAxis, 521
sensorDataLen
sensorData_t, 521
ServerAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 955
serverAddrTypeMask
pack_loc_SLQSLOCGetServer_t, 359
serverStatus
unpack_loc_GetServer_Ind_t, 685
unpack_loc_SetServer_Ind_t, 701
serverType
pack_loc_SLQSLOCGetServer_t, 359
pack_loc_SLQSLOCSetServer_t, 368
unpack_loc_GetServer_Ind_t, 685
serviceCategory
sms_CDMABroadcastConfig, 526
serviceDomain
nas_RejectReasonTlv, 255
unpack_nas_SLQSNasNetworkRejectCallback_-Ind_t, 760
ServiceStatusConfig
pack_imsa_SLQSRegisterIMSAIndication_t, 354
servingCellId
nas_LTEInfoIntrafreq, 203
ServingSystem
unpack_nas_SLQSGetServingSystem_t, 733
sessExp
ims_SessDurInfo, 80
sessionControl
pack_pds_StartPDSTrackingSessionExt_t, 419
sessionEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 959
SessionId
pack_loc_Start_t, 370
pack_loc_Stop_t, 370
sessionId
unpack_loc_PositionRpt_Ind_t, 696
sessionInfo
pack_uim_ChangePin_t, 455
pack_uim_ReadTransparent_t, 456
pack_uim_SetPinProtection_t, 457
pack_uim_SLQSUIMAuthenticate_t, 458
pack_uim_SLQSUIMGetFileAttributes_t, 460
pack_uim_SLQSUIMRefreshComplete_t, 462
pack_uim_SLQSUIMRefreshGetLastEvent_t, 462
pack_uim_SLQSUIMRefreshOK_t, 463
pack_uim_SLQSUIMRefreshRegister_t, 463
pack_uim_UnblockPin_t, 465
pack_uim_VerifyPin_t, 466
SessionInfoConfig
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 858

SessionInfoFota
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 858
SessionInfoNotification
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 858
sessionOperation
pack_pds_StartPDSTrackingSessionExt_t, 419
sessionResponse
unpack_swiavms_SLQSAVMSStartSession_t, 852
sessionServerOption
pack_pds_StartPDSTrackingSessionExt_t, 419
SessionState
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
SessionStatus
unpack_pds_SetEventReport_Ind_t, 787
sessionStatus
unpack_loc_PositionRpt_Ind_t, 696
unpack_omaDmNotificationsTlv_t, 780
SessionType
pack_swioma_SLQSOMADMGetSessionInfo_t, 449
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
sessionType
pack_pds_StartPDSTrackingSessionExt_t, 419
pack_swiavms_SLQSAVMSStartSession_t, 443
pack_swiavms_SLQSAVMSStopSession_t, 444
pack_swioma_SLQSOMADMCancelSession_t, 448
pack_swioma_SLQSOMADMCancelSessionExt_t, 448
pack_swioma_SLQSOMADMStartSession_t, 452
pack_swioma_SLQSOMADMStartSessionExt_t, 453
uim_refreshevent, 568
uim_sessionInformation, 570
uim_UIMSessionInformation, 574
unpack_omaDmFotaTlv_t, 779
set_fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 447
set_fix_type
pack_swiloc_SwiLocSetAutoStart_t, 447
set_function
pack_swiloc_SwiLocSetAutoStart_t, 447
set_max_dist
pack_swiloc_SwiLocSetAutoStart_t, 447
set_max_time
pack_swiloc_SwiLocSetAutoStart_t, 447
SetupEventList
cat_EventListTlv, 48
Severity
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
severity
unpack_omaDmFotaTlv_t, 779
shortName
nas_PLMNNetworkNameData, 246
unpack_nas_SLQSGetPLMNName_t, 728
shortNameCI
unpack_nas_SLQSGetPLMNName_t, 728
shortNameEn
unpack_nas_SLQSGetPLMNName_t, 728
shortNameLen
nas_PLMNNetworkNameData, 246
unpack_nas_SLQSGetPLMNName_t, 728
shortNameSB
unpack_nas_SLQSGetPLMNName_t, 728
shortNameSpareBits
nas_PLMNNetworkNameData, 246
sid
nas_CDMAInfo, 153
nas_sidNid, 272
unpack_nas_GetHomeNetwork_t, 709
SidNid
nas_homeSIDNID, 191
SigCompEn
ims_SigCompEnInfo, 80
SigInd
LibPackUMTSReqQoSSigInd, 123
sigInfo
nas_SccRxInfo, 270
signal
voice_signalInfo, 1024
signalStrength
nas_SignalStrengthTlv, 273
signalStrengthReqMask
unpack_nas_SLQSGetSignalStrength_t, 736
signalType
voice_signalInfo, 1024
SimCapability
unpack_dms_GetDeviceCap_t, 589
simCapability
unpack_dms_GetDeviceCapabilities_t, 591
sinr
hdrSSInfo, 71
nas_SLQSSignalStrengthsInformation, 276
tdscdmaSigInfoExt, 548
unpack_nas_SLQSGetSignalStrength_t, 736
sinrDelta
nas_SLQSSignalStrengthsIndReq, 275
sinrThresholdList
nas_SLQSSignalStrengthsIndReq, 275
sinrThresholdListLen
nas_SLQSSignalStrengthsIndReq, 275
sku_str
unpack_dms_GetFirmwareInfo_t, 596
slot
pack_uim_SLQSUIMPowerDown_t, 461
pack_uim_SLQSUIMPowerUp_t, 461
slot_t, 522
bICCID, 522
bICCIDLength, 522
bLogicalSlot, 522
uPhyCardStatus, 522

uPhySlotStatus, 522
slotInf, 523
AppStatus, 524
cardState, 524
errorState, 524
numApp, 524
upinRetries, 524
upinState, 524
upukRetries, 524
SlotInfo
uim_cardStatus, 557
slots_t, 524
uimSlotStatus, 525
slotsstatusChange
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 871
SlqsProfile3GPP
unpackWdsProfileParam, 979
wds_profileInfo, 1044
SlqsProfile3GPP2
unpackWdsProfileParam, 979
wds_profileInfo, 1044
sms.h
LITEQMI_QMI_CBK_PARAM_NOCHANGE, 1332
LITEQMI_QMI_CBK_PARAM_RESET, 1332
LITEQMI_QMI_CBK_PARAM_SET, 1332
sms.h, 1328
eqmiCbkSetStatus, 1332
MAX_MSE_TWS_MSG, 1331
MAX_SMS_LIST_SIZE, 1331
pack_sms_GetSMSCAddress, 1332
pack_sms_SLQSDeleteSMS, 1334
pack_sms_SLQSGetIndicationRegister, 1334
pack_sms_SLQSGetMessageWaiting, 1334
pack_sms_SLQSGetSMS, 1335
pack_sms_SLQSGetSMSList, 1335
pack_sms_SLQSGetSmsBroadcastConfig, 1335
pack_sms_SLQSGetTransLayerInfo, 1336
pack_sms_SLQSGetTransNWRegInfo, 1336
pack_sms_SLQSModifySMSStatus, 1336
pack_sms_SLQSSendAsyncSMS, 1337
pack_sms_SLQSSetIndicationRegister, 1337
pack_sms_SLQSSetSmsBroadcastActivation, 1337
pack_sms_SLQSSetSmsBroadcastConfig, 1338
pack_sms_SLQSSetSmsStorage, 1338
pack_sms_SLQSSmsGetMaxStorageSize, 1339
pack_sms_SLQSSmsGetMessageProtocol, 1339
pack_sms_SLQSSmsSetRoutes, 1339
pack_sms_SLQSSwiGetSMSStorage, 1340
pack_sms_SaveSMS, 1332
pack_sms_SendSMS, 1333
pack_sms_SetNewSMSCallback, 1333
pack_sms_SetSMSCAddress, 1333
SMS_CONFIG_LEN, 1331
SMS_MAX_SMS_ROUTES, 1332
SMS_NUM_OF_SET, 1332
SMSC_TYPE_LEN, 1332
unpack_sms_GetSMSCAddress, 1340
unpack_sms_SLQSDeleteSMS, 1342
unpack_sms_SLQSGetIndicationRegister, 1342
unpack_sms_SLQSGetMessageWaiting, 1343
unpack_sms_SLQSGetSMS, 1343
unpack_sms_SLQSGetSMSList, 1344
unpack_sms_SLQSGetSmsBroadcastConfig, 1343
unpack_sms_SLQSGetTransLayerInfo, 1344
unpack_sms_SLQSGetTransNWRegInfo, 1345
unpack_sms_SLQSModifySMSStatus, 1345
unpack_sms_SLQSNWRegInfoCallback_ind, 1345
unpack_sms_SLQSSendAsyncSMS, 1346
unpack_sms_SLQSSetIndicationRegister, 1346
unpack_sms_SLQSSetSmsBroadcastActivation, 1346
unpack_sms_SLQSSetSmsBroadcastConfig, 1347
unpack_sms_SLQSSetSmsStorage, 1347
unpack_sms_SLQSSmsGetMaxStorageSize, 1347
unpack_sms_SLQSSmsGetMessageProtocol, 1348
unpack_sms_SLQSSmsSetRoutes, 1348
unpack_sms_SLQSSwiGetSMSStorage, 1348
unpack_sms_SLQSTransLayerInfoCallback_ind, 1349
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind, 1349
unpack_sms_SLQSWmsMemoryFullCallBack_ind, 1350
unpack_sms_SLQSWmsMessageWaitingCallBack_ind, 1350
unpack_sms_SaveSMS, 1340
unpack_sms_SendSMS, 1341
unpack_sms_SetNewSMSCallback, 1341
unpack_sms_SetNewSMSCallback_ind, 1341
unpack_sms_SetSMSCAddress, 1342
sms_BroadcastConfig, 525
fromServiceId, 525
selected, 525
toServiceId, 525
sms_CDMABroadcastConfig, 525
language, 526
selected, 526
serviceCategory, 526
sms_getIndicationReg, 526
pRegCallStatInfoEvt, 527
pRegTransLayerInfoEvt, 527
pRegTransNWRegInfoEvt, 527
sms_getMsgWaitingInfo, 527
msgWaitInfo, 527
numInstances, 527
sms_getTransLayerInfo, 527
pRegInd, 528
pTransLayerInfo, 528
sms_getTransNWRegInfo, 528
pRegStatus, 528
sms_maxStorageSizeReq, 529
pMessageMode, 529

storageType, 529
sms_maxStorageSizeResp, 529
freeSlots, 530
maxStorageSize, 530
sms_messageWaitingInfoContent, 530
activeInd, 530
msgCount, 530
msgType, 530
sms_msgProtocolResp, 531
msgProtocol, 531
sms_qaQmi3GPP2BroadcastCfgInfo, 531
activated_ind, 532
CDMABroadcastConfig, 532
num_instances, 532
sms_qaQmi3GPPBroadcastCfgInfo, 532
activated_ind, 532
broadcastConfig, 532
num_instances, 532
sms_routeEntry, 532
messageClass, 533
messageType, 533
receiptAction, 533
routeStorage, 533
sms_sendAsyncsmsParams, 533
messageFormat, 535
messageSize, 535
pFollowOnDC, 535
pForceOnDC, 535
pLinktimer, 535
pMessage, 535
pRetryMessage, 535
pRetryMessageId, 535
pServiceOption, 535
pSmsOnIms, 535
pUserData, 535
sms_setIndicationReg, 536
pRegCallStatInfoEvt, 536
pRegTransLayerInfoEvt, 536
pRegTransNWRegInfoEvt, 536
sms_setRoutesReq, 536
numOfRoutes, 537
pTransferStatusReport, 537
routeList, 537
sms_transLayerInfo, 537
TransCap, 537
TransType, 538
smsFormat
ims_SMSFmtInfo, 82
smsOnIMS
sMSOnIMSInfo, 541
SmsRat
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
SmsRatVal
imsa_SmsRatInfo, 88
SmsService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
SmsServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
SmsServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
smsStorage
pack_sms_SLQSSetSmsStorage_t, 431
SmsSvcStatus
imsa_SmsSvcStatusInfo, 88
smsoIPNW
ims_SMSoIPNwInfo, 82
snr
loc_satelliteInfo, 138
lteSSInfo, 143
nas_SccRxInfo, 270
snrlevel
nas_lteSnrinformation, 216
soMask
qmiWDSDataBearerTechnology, 517
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
Source
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
source
altSrcInfo_t, 35
unpack_dms_GetNetworkTime_t, 602
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 620
unpack_dms_SLQSDmsSwiGetResetInfo_t, 621
sourceIPMask
LibPackTFTIDParams, 120
SourceLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
SourceRAT
imsa_RatHandoverStatusInfo, 87
spc
pack_dms_ResetToFactoryDefaults_t, 328
pack_nas_SetACCOLC_t, 371
pack_wds_SetMobileIPProfile_t, 496
spn
nas_serviceProviderName, 270
unpack_nas_SLQSGetPLMNName_t, 728
spnEncoding
unpack_nas_SLQSGetPLMNName_t, 728
spnLength
nas_serviceProviderName, 270
unpack_nas_SLQSGetPLMNName_t, 729
srcPortRangeEnd
LibPackTFTIDParams, 120
srcPortRangeStart
LibPackTFTIDParams, 120
srvCapValid
nas_NR5GSystemInfoTlv, 235
srvCapability
nas_detailSvcInfo, 168
nas_sysInfoCommon, 280
srvCapabilityValid

nas_sysInfoCommon, 280
srvDomain
nas_NR5GSystemInfoTlv, 235
nas_sysInfoCommon, 280
SrvDomainPref
nas_SrvDomainPrefTlv, 277
NASServDomainPrefTlv, 315
srvDomainValid
nas_NR5GSystemInfoTlv, 235
nas_sysInfoCommon, 280
srvOption
voice_arrSvcOption, 989
srvStatus
nas_detailSvcInfo, 168
nas_GSMSrvStatusInfo, 182
nas_NR5GSerStatTlv, 231
nas_SrvStatusInfo, 278
srvcapability
nas_NR5GSystemInfoTlv, 235
srxlev
nas_cellParams, 158
nas_gsmCellInfo, 180
nas_umtsLTENbrCell, 290
nas_wcdmaCellInfo, 292
stage
uim_refreshevent, 568
State
nas_NetworkStat1x, 226
nas_NetworkStatEVDO, 228
state
pack_loc_SLQSLOCSetCradleMountConfig_t, 367
unpack_dms_GetActivationState_t, 582
unpack_dms_SLQSUIMGetState_t, 640
unpack_omaDmConfigTlv_t, 777
unpack_omaDmFotaTlv_t, 779
unpack_qos_QosFlowInfoState_t, 799
statmask
pack_wds_GetPacketStatus_t, 489
StatsMask
wds_transferStatInd, 1045
statsMask
wds_TrStatInd, 1046
StatsPeriod
wds_transferStatInd, 1045
statsPeriod
wds_TrStatInd, 1046
Status
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
status
nas_lteEARFCN, 197
nas_ltePCI, 211
nas_wcdmaUARFCN, 298
pack_sms_SetNewSMSCallback_t, 425
unpack_loc_BestAvailPos_Ind_t, 679
unpack_loc_DeleteAssistData_Ind_t, 681
unpack_loc_InjectPositionCallback_Ind_t, 686
unpack_loc_InjectUTCTimeCallback_Ind_t, 689
unpack_loc_SetExtPowerConfig_Ind_t, 698
unpack_loc_SetOperationMode_Ind_t, 699
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 805
unpack_qos_SLQSSetQosStatusCallback_ind_t, 807
storageIndex
FMSImageIdElement, 68
storageType
pack_sms_SaveSMS_t, 424
pack_sms_SLQSDeleteSMS_t, 426
pack_sms_SLQSGetSMS_t, 427
pack_sms_SLQSGetSMSList_t, 428
pack_sms_SLQSModifySMSStatus_t, 429
sms_maxStorageSizeReq, 529
sMSMTMessageInfo, 541
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 838
Streaming Download Protocol (lite-fw), 30
String
unpack_dms_GetDeviceHardwareRev_t, 591
unpack_dms_GetDeviceMfr_t, 592
unpack_dms_GetFSN_t, 599
unpack_dms_UIMGetICCID_t, 643
stringSize
unpack_dms_GetDeviceHardwareRev_t, 591
unpack_dms_GetDeviceMfr_t, 592
unpack_dms_UIMGetICCID_t, 643
subAddr
voice_calledPartySubAdd, 993
subAddrLen
voice_calledPartySubAdd, 993
subAddrType
voice_calledPartySubAdd, 993
subnetMask
unpack_qos_IPv4Addr_t, 795
SubnetMaskV4
unpack_wds_SLQSGetRuntimeSettings_t, 955
subsType
pack_voice_SLQSVoiceBindSubscription_t, 469
subscrTmr
ims_SubscrTmrInfo, 83
SupUSBComps
unpack_dms_GetUSBComp_t, 608
SuppOA
voice_CUGInfo, 1007
SuppPrefCUG
voice_CUGInfo, 1007
svInfoMask
loc_satelliteInfo, 138
svListLen
loc_satelliteInfo, 138
svStatus
loc_satelliteInfo, 138
svc
pack_qmi_t, 420
SvcClass
voice_callFWExtInfo, 997

voice_callFWInfo, 998
SvcStatus
voice_callFWExtInfo, 997
voice_callFWInfo, 998
svcType
voice_ccSUPSType, 1002
voice_SUPSInfo, 1025
sw1
uim_cardResult, 556
sw2
uim_cardResult, 556
swVerString
pack_dms_SLQSSwiSetHostDevInfo_t, 338
unpack_dms_SLQSSwiGetHostDevInfo_t, 636
swi_uint256_clear_bit
switype_256bit.h, 1398
swi_uint256_get_bit
switype_256bit.h, 1398
swi_uint256_print_mask
switype_256bit.h, 1398
swi_uint256_set_bit
switype_256bit.h, 1399
swi_uint256_t, 543
word, 543
SwiDataTypes.h, 1374
BOOL, 1375
BYTE, 1375
CHAR, 1375
FLOAT, 1375
INT32, 1375
INT8, 1375
LPCSTR, 1375
qmuint16, 1375
qmulong, 1375
SHORT, 1375
SWI_API, 1375
ULONG, 1375
ULONGLONG, 1375
UNUSEDPARAM, 1375
USHORT, 1375
WORD, 1375
swiaudio.h, 1350
pack_swiaudio_SLQSGetM2MAVMute, 1353
pack_swiaudio_SLQSGetM2MAudioProfile, 1352
pack_swiaudio_SLQSGetM2MAudioVolume, 1352
pack_swiaudio_SLQSGetM2MSpkrGain, 1353
pack_swiaudio_SLQSSetM2MAVMute, 1355
pack_swiaudio_SLQSSetM2MAudioAVCFG, 1354
pack_swiaudio_SLQSSetM2MAudioLPBK, 1354
pack_swiaudio_SLQSSetM2MAudioNVDef, 1354
pack_swiaudio_SLQSSetM2MAudioProfile, 1355
pack_swiaudio_SLQSSetM2MAudioVolume, 1355
pack_swiaudio_SLQSSetM2MSpkrGain, 1356
unpack_swiaudio_SLQSGetM2MAVMute, 1357
unpack_swiaudio_SLQSGetM2MAudioProfile, 1356
unpack_swiaudio_SLQSGetM2MAudioVolume, 1357
unpack_swiaudio_SLQSGetM2MSpkrGain, 1357
unpack_swiaudio_SLQSSetM2MAVMute, 1359
unpack_swiaudio_SLQSSetM2MAVMute_t, 1352
unpack_swiaudio_SLQSSetM2MAudioAVCFG, 1358
unpack_swiaudio_SLQSSetM2MAudioAVCFG_t, 1352
unpack_swiaudio_SLQSSetM2MAudioLPBK, 1358
unpack_swiaudio_SLQSSetM2MAudioLPBK_t, 1352
unpack_swiaudio_SLQSSetM2MAudioNVDef, 1358
unpack_swiaudio_SLQSSetM2MAudioNVDef_t, 1352
unpack_swiaudio_SLQSSetM2MAudioProfile, 1359
unpack_swiaudio_SLQSSetM2MAudioProfile_t, 1352
unpack_swiaudio_SLQSSetM2MAudioVolume, 1359
unpack_swiaudio_SLQSSetM2MAudioVolume_t, 1352
unpack_swiaudio_SLQSSetM2MSpkrGain, 1360
unpack_swiaudio_SLQSSetM2MSpkrGain_t, 1352
swiaudio_PCMparams, 543
iFaceTab, 544
iFaceTabLen, 544
swiavms.h, 1360
pack_swiavms_SLQSAVMSGetSettings, 1362
pack_swiavms_SLQSAVMSGetSettings_v2, 1363
pack_swiavms_SLQSAVMSSendSelection, 1363
pack_swiavms_SLQSAVMSSessionGetInfo, 1364
pack_swiavms_SLQSAVMSSetSettings, 1365
pack_swiavms_SLQSAVMSSetSettings_v2, 1366
pack_swiavms_SLQSAVMSSetSettingsNoAuto-RebootField, 1366
pack_swiavms_SLQSAVMSStartSession, 1367
pack_swiavms_SLQSAVMSStopSession, 1368
pack_swiavms_SLQSAvmsSetEventReport, 1365
unpack_swiavms_SLQSAVMSEventReportInd, 1368
unpack_swiavms_SLQSAVMSGetSettings, 1369
unpack_swiavms_SLQSAVMSGetSettings_v2, 1369
unpack_swiavms_SLQSAVMSSendSelection, 1370
unpack_swiavms_SLQSAVMSSessionGetInfo, 1370
unpack_swiavms_SLQSAVMSSetSettings, 1371
unpack_swiavms_SLQSAVMSSetSettings_v2, 1372
unpack_swiavms_SLQSAVMSStartSession, 1372
unpack_swiavms_SLQSAVMSStopSession, 1373
unpack_swiavms_SLQSAVMSStopSession_avc2, 1373
unpack_swiavms_SLQSAvmsSetEventReport, 1371
swidms.h, 1375

pack_swidms_SLQSSwiDmsGetMTU, 1376
pack_swidms_SLQSSwiDmsGetUsbComp, 1377
pack_swidms_SLQSSwiDmsGetUsbNetNum, 1377
pack_swidms_SLQSSwiDmsSetMTU, 1377
pack_swidms_SLQSSwiDmsSetUsbComp, 1378
pack_swidms_SLQSSwiDmsSetUsbNetNum, 1378
unpack_swidms_SLQSSwiDmsGetMTU, 1378
unpack_swidms_SLQSSwiDmsGetUsbComp, 1379
unpack_swidms_SLQSSwiDmsGetUsbNetNum, 1379
unpack_swidms_SLQSSwiDmsSetMTU, 1379
unpack_swidms_SLQSSwiDmsSetUsbComp, 1380
unpack_swidms_SLQSSwiDmsSetUsbNetNum, 1380
unpack_swidms_SLQSSwiDmsSetUsbNetNum_t, 1376
swidms_ehrpdMTUSizeTlv, 544
ehrpdMTUSize, 544
TlvPresent, 544
swidms_hrpdMTUSizeTlv, 544
hrpdMTUSize, 545
TlvPresent, 545
swidms_intfaceCfgTlv, 545
CfgValue, 546
CurrentCfgType, 546
TlvPresent, 546
swidms_mtuSize3gppTlv, 546
MTUSize3gpp, 547
TlvPresent, 547
swidms_supportedIntBitmaskTlv, 547
TlvPresent, 547
ValidBitmasks, 547
swidms_usbMTUSizeTlv, 547
TlvPresent, 548
UsbMTUSize, 548
swiloc.h, 1381
pack_swiloc_SwiLocGetAutoStart, 1381
pack_swiloc_SwiLocSetAutoStart, 1381
unpack_swiloc_SwiLocGetAutoStart, 1382
unpack_swiloc_SwiLocSetAutoStart, 1382
unpack_swiloc_SwiLocSetAutoStart_t, 1381
swioma.h, 1382
pack_swioma_SLQSOMADMAlertCallback, 1384
pack_swioma_SLQSOMADMCancelSession, 1384
pack_swioma_SLQSOMADMGetSessionInfo, 1385
pack_swioma_SLQSOMADMGetSettings, 1386
pack_swioma_SLQSOMADMSendSelection, 1386
pack_swioma_SLQSOMADMSetSettings, 1387
pack_swioma_SLQSOMADMStartSession, 1388
unpack_swioma_SLQSOMADMAlertCallback, 1388
unpack_swioma_SLQSOMADMAlertCallback_ind, 1389
unpack_swioma_SLQSOMADMAlertCallback_t, 1384
unpack_swioma_SLQSOMADMCancelSession, 1389
unpack_swioma_SLQSOMADMCancelSession_t, 1384
unpack_swioma_SLQSOMADMGetSessionInfo, 1390
unpack_swioma_SLQSOMADMGetSettings, 1390
unpack_swioma_SLQSOMADMSendSelection, 1391
unpack_swioma_SLQSOMADMSendSelection_t, 1384
unpack_swioma_SLQSOMADMSetSettings, 1391
unpack_swioma_SLQSOMADMSetSettings_t, 1384
unpack_swioma_SLQSOMADMStartSession, 1391
swiomaext.h, 1392
pack_swioma_SLQSOMADMCancelSessionExt, 1393
pack_swioma_SLQSOMADMSendSelectionExt, 1394
pack_swioma_SLQSOMADMSetSettingsExt, 1394
pack_swioma_SLQSOMADMStartSessionExt, 1395
unpack_swioma_SLQSOMADMCancelSessionExt, 1395
unpack_swioma_SLQSOMADMCancelSessionExt_t, 1393
unpack_swioma_SLQSOMADMSendSelectionExt, 1396
unpack_swioma_SLQSOMADMSendSelectionExt_t, 1393
unpack_swioma_SLQSOMADMSetSettingsExt, 1396
unpack_swioma_SLQSOMADMSetSettingsExt_t, 1393
unpack_swioma_SLQSOMADMStartSessionExt, 1397
unpack_swioma_SLQSOMADMStartSessionExt_t, 1393
switchOption
pack_voice_SLQSVoiceALSSetLineSwitching_t, 468
switype_256bit.h, 1397
swi_uint256_clear_bit, 1398
swi_uint256_get_bit, 1398
swi_uint256_print_mask, 1398
swi_uint256_set_bit, 1399
sysForbidden
nas_NR5GSystemInfoTlv, 235
sysForbiddenValid
nas_NR5GSystemInfoTlv, 235
sysInfoCDMA
nas_CDMASysInfo, 157
sysInfoGSM
nas_GSMSysInfo, 185

sysInfoHDR
nas_HDRSysInfo, 191
sysInfoLTE
nas_LTESysInfo, 219
sysInfoWCDMA
nas_WCDMASysInfo, 297
system
loc_satelliteInfo, 138
loc_SV, 140
systemDiscontinuities
pack_pds_PDSInjectTimeReference_t, 409
SystemID
unpack_nas_SLQSGetServingSystem_t, 733
systemID
nas_CDMASysInfo, 157
systemMode
nas_CommInfo, 162
systemTime
pack_pds_PDSInjectTimeReference_t, 409
szAPN
PackSwiAvmsSetSettingsAPNInfo, 513
szAlertMsg
UnpackSwiAvmsEventReportConfig, 972
szCarrier_str
_litefw_FirmwareInfo_, 33
szCarrierPriversion_str
_litefw_FirmwareInfo_, 34
szDescription
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
szFwversion_str
_litefw_FirmwareInfo_, 34
szModelid_str
_litefw_FirmwareInfo_, 34
szName
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
szPWD
PackSwiAvmsSetSettingsAPNInfo, 513
szPackageid_str
_litefw_FirmwareInfo_, 34
szPartno_str
_litefw_FirmwarePartNo_, 34
szSku_str
_litefw_FirmwareInfo_, 34
szUname
PackSwiAvmsSetSettingsAPNInfo, 513
szVersion
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971

t3396_apn
unpack_nas_SLQSNasTimerCallback_ind_t, 765
t3396_plmn_id
unpack_nas_SLQSNasTimerCallback_ind_t, 765
t3396_val
unpack_nas_SLQSNasTimerCallback_ind_t, 765
TCPDstPort
unpack_qos_swiQosFilter_t, 810
TCPSrcPort
unpack_qos_swiQosFilter_t, 810
TDSCDMAECIOThreshListLen
nas_TDSCDMAECIOThresh, 281
TDSCDMARSCPThreshListLen
nas_TDSCDMARSCPThresh, 281
TDSCDMARSSIThreshListLen
nas_TDSCDMARSSIThresh, 282
TDSCDMASINRThreshListLen
nas_TDSCDMASINRThresh, 283
TMD_MAX_DEV_LIST
tmd.h, 1400
TMTlv
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 657
TPCause
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
TRMessageTlv
unpack_sms_SetNewSMSCallback_ind_t, 823
TT1Tlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
TT2Tlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
TTfTlv
unpack_ims_SLQSSIPCfgCallBack_ind_t, 661
TX_PWR
nas_NetworkStat1x, 226
TXBOTlv
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 944
TXChan
nas_LTEInfo, 200
tXDroppedCount
unpack_wds_GetPacketStatus_t, 939
tXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 939
tXOkBytesCount
unpack_wds_GetPacketStatus_t, 939
tXPacketErrors
unpack_wds_GetPacketStatus_t, 939
tXPacketOverflows
unpack_wds_GetPacketStatus_t, 939
tXPacketSuccesses
unpack_wds_GetPacketStatus_t, 939
Tables, 31
tac
nas_LTEInfoIntrafreq, 203
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 235
tacValid
nas_LTESysInfo, 219
nas_NR5GSystemInfoTlv, 235
TargetRAT
imsa_RatHandoverStatusInfo, 87
TdsBandCapability
unpack_dms_SLQSGetBandCapability_t, 627
tdscdmaSigInfoExt, 548
ecio, 548

rscp, 548
rssi, 548
sinr, 548
techName
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 959
Technology
nas_DeviceConfigDetail, 169
unpack_wds_SLQSGetRuntimeSettings_t, 955
TechnologyPref
pack_nas_SetNetworkPreference_t, 374
tempData_t, 548
temperature, 549
temperatureDataLen, 549
timeOfFirstSample, 549
timeOffset, 549
timeSource, 549
TempStat
dms_TemperatureTlv, 60
TempTlv
unpack_dms_SwiEventReportCallBack_ind_t, 641
Temperature
dms_TemperatureTlv, 60
temperature
nas_CommInfo, 162
tempData_t, 549
temperatureDataLen
tempData_t, 549
threshGsmHigh
nas_lteGsmCellInfo, 198
threshGsmLow
nas_lteGsmCellInfo, 198
threshServingLow
nas_LTEInfoIntrafreq, 203
threshXHigh
nas_infoInterFreq, 192
threshXLow
nas_infoInterFreq, 193
threshXhigh
nas_lteWcdmaCellInfo, 220
threshXlow
nas_lteWcdmaCellInfo, 220
Time
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
time
NASTimeInfoTlv, 317
Time_uncert_ms
unpack_pds_SLQSGetGPSStateInfo_t, 793
timeBase
pack_pds_SLQSPDSInjectAbsoluteTimeReference-_t, 414
TimeLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
timeMsec
pack_loc_SLQSLOCInjectUTCTime_t, 366
pack_pds_SLQSPDSInjectAbsoluteTimeReference-_t, 414
timeOfFirstSample
sensorData_t, 521
tempData_t, 549
timeOffset
sensorData_t, 521
tempData_t, 549
timeSource
tempData_t, 549
TimeStmp_gps_week
unpack_pds_SLQSGetGPSStateInfo_t, 793
TimeStmp_tow_ms
unpack_pds_SLQSGetGPSStateInfo_t, 793
timeSyncRefCounter
unpack_loc_EventTimeSyncCallback_Ind_t, 683
timeTlv
NASQmiCbkNasSwiOTAMessageInd, 312
timeUncMsec
pack_loc_SLQSLOCInjectUTCTime_t, 366
pack_pds_SLQSPDSInjectAbsoluteTimeReference-_t, 415
timeZone
nas_timeInfo, 285
timeout
pack_pds_SetPDSDefaults_t, 411
pack_qmi_t, 420
Timers
PackSwiAvmsSetSettingsConnectionRetryTimers, 513
PackSwiAVMSSettingsConnectionRetryTimers, 515
timestamp
unpack_dms_GetNetworkTime_t, 602
timestampAge
pack_loc_SLQSLOCInjectPosition_t, 364
timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 364
timingAdvance
nas_GERANInfo, 178
TlvPresenceMask
qmTlvResult, 517
TlvPresent
cat_commonEventTlv, 46
dms_ActivationStatusTlv, 53
dms_OperatingModeTlv, 54
dms_PSMActiveTimerIndTlv, 54
dms_PSMActiveTimerTlv, 55
dms_PSMDurationDueToOOSTlv, 55
dms_PSMDurationThresholdTlv, 56
dms_PSMEarlyWakeupTimeTlv, 56
dms_PSMEnableStateIndTlv, 57
dms_PSMEnableStateTlv, 57
dms_PSMPeriodicUpdateTimerIndTlv, 58
dms_PSMPeriodicUpdateTimerTlv, 58
dms_PSMRandomizationWindowTlv, 59
dms_TemperatureTlv, 60
dms_UimAutoSwitchActSlotTlv, 60

dms_UimStatusTlv, 61
dms_VoltageTlv, 62
eTWSPLMNInfoTlv, 66
ims_AMRModeInfo, 72
ims_AMROctAlgnInfo, 72
ims_AMRWBModeInfo, 73
ims_AMRWBOctAlgnInfo, 74
ims_CSCFPortNameInfo, 74
ims_EnabAMRWBInfo, 74
ims_EnabSCRAMRInfo, 75
ims_EnabSCRAMRWBInfo, 75
ims_IMSDomainInfo, 76
ims_IMSTestModeInfo, 76
ims_MinSessExpInfo, 77
ims_PCSCFPortInfo, 77
ims_PhCtxtURIInfo, 78
ims_RngBkTmrInfo, 78
ims_RngTmrInfo, 79
ims_RTPRTCPInactTmrDurInfo, 79
ims_SessDurInfo, 80
ims_SigCompEnInfo, 80
ims_SIPPortInfo, 81
ims_SIPRegnTmrInfo, 81
ims_SMSFmtInfo, 82
ims_SMSoIPNwInfo, 82
ims_SubscrTmrInfo, 83
ims_TmrT1Info, 83
ims_TmrT2Info, 84
ims_TmrTfInfo, 84
imsa_IMSFailErrCodeTlv, 85
imsa_IMSRegStatusErrorCodeInfo, 85
imsa_IMSRegStatusInfo, 86
imsa_NewIMSRegStatusInfo, 86
imsa_RatHandoverStatusInfo, 87
imsa_SmsRatInfo, 88
imsa_SmsSvcStatusInfo, 88
imsa_UtRatInfo, 89
imsa_UtSvcStatusInfo, 89
imsa_VoipRatInfo, 90
imsa_VoipSvcStatusInfo, 90
imsa_VtRatInfo, 91
imsa_VtSvcStatusInfo, 92
loc_IPv4Info, 133
loc_IPv6Info, 134
loc_urlAddr, 141
messageModeTlv, 143
nas_AcqOrderPrefTlv, 145
nas_BandPrefInfoTlv, 147
nas_BandPrefTlv, 149
nas_CiotAcqOrderPrefTlv, 160
nas_CiotLteOpModePrefTlv, 160
nas_CsgId, 163
nas_EdrxCiotLteMode, 172
nas_EdrxCycleLength, 172
nas_EdrxEnableType, 173
nas_EdrxPagingTimeWindow, 173
nas_EdrxRatType, 174
nas_EmerModeTlv, 175
nas_ForbiddenNetworks3GPP, 177
nas_GWAcqOrderPrefTlv, 185
nas_LTEBandPrefTlv, 196
nas_LteCiotOpModeTlv, 196
nas_LteM1BandPrefTlv, 207
nas_LteNb1BandPrefTlv, 209
nas_LTEOperationMode, 209
nas_LteOpMode, 210
nas_lteOpModeTlv, 211
nas_ModePrefTlv, 222
nas_NetSelPrefTlv, 224
nas_NR5GCellStatusTlv, 230
nas_NR5GSerStatTlv, 231
nas_NR5GSystemInfoTlv, 235
nas_PhyCaAggPcellInfo, 238
nas_PhyCaAggScellDlBw, 238
nas_PhyCaAggScellIndex, 239
nas_PhyCaAggScellIndType, 239
nas_PhyCaAggScellInfo, 242
nas_PlmnID, 244
nas_PRLPrefTlv, 247
nas_RankIndicatorTlv, 254
nas_RatDisabledMaskTlv, 255
nas_RejectReasonTlv, 255
nas_RfBandInfoExtFormat, 257
nas_RfBandwidthInfo, 259
nas_RfDedicatedBandInfo, 260
nas_RFInfoTlv, 262
nas_RoamPrefTlv, 263
nas_SccRxInfo, 270
nas_SignalStrengthTlv, 273
nas_SLQSSignalStrengthsTlv, 277
nas_SrvDomainPrefTlv, 277
nas_timeInfo, 285
NASAcqOrderPrefTlv, 298
NASBandPreferenceTlv, 298
NASCiotAcqOrderPrefTlv, 299
NASCiotLteOpModePrefTlv, 299
NASEmergencyModeTlv, 300
NASGWAcqOrderPrefTlv, 301
NASLTEBandPreferenceTlv, 302
NASLteM1BandPrefTlv, 302
NASLteNasReleaseInfoTlv, 303
NASLteNB1BandPrefTlv, 303
NASModePreferenceTlv, 304
NASNetSelPreferenceTlv, 304
NASNr5gBandPrefTlv, 305
NASOTAMessageTlv, 306
NASPhyCaAggPcellInfo, 306
NASPhyCaAggScellArray, 308
NASPhyCaAggScellDlBw, 308
NASPhyCaAggScellIndex, 309
NASPhyCaAggScellIndType, 310
NASPhyCaAggScellInfo, 311
NASPRLPreferenceTlv, 311
NASRatDisabledMaskTlv, 315
NASRoamPreferenceTlv, 315
NASServDomainPrefTlv, 315

NASTimeInfoTlv, 317
newMTMessageTlv, 318
sMSCAddressTlv, 539
sMSEtwsMessageTlv, 540
sMSOnIMSTlv, 542
swidms_ehrpdMTUSizeTlv, 544
swidms_hrpdMTUSizeTlv, 545
swidms_intfaceCfgTlv, 546
swidms_mtuSize3gppTlv, 547
swidms_supportedIntBitmaskTlv, 547
swidms_usbMTUSizeTlv, 548
transferRouteMessageTlv, 551
unpack_pds_SetPdsState_Ind_t, 788
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 876
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
UnpackSwiAvmsEventReportConfig, 973
UnpackSwiAvmsEventReportConnectionRequest, 973
UnpackSwiAvmsEventReportDataSessionStatus, 974
UnpackSwiAvmsEventReportHTTPStatus, 975
UnpackSwiAvmsEventReportNotification, 977
UnpackSwiAvmsEventReportPackageID, 977
UnpackSwiAvmsEventReportRegStatus, 977
UnpackSwiAvmsEventReportSessionType, 978
UnpackSwiAvmsEventReportWAMSParam-Change, 979
wds_ChannelRateTlv, 1028
wds_ConnStatusTlv, 1029
wds_DataBearTechTlv, 1031
wds_DataULongLongTlv, 1032
wds_DataULongTlv, 1032
wds_DHCPLeaseOptTlv, 1033
wds_DHCPLeaseStateTlv, 1033
wds_DHCPProfileIdTlv, 1034
wds_DormStatTlv, 1038
wds_IPv4AdTlv, 1040
wds_LastMdmCallEndRsnTlv, 1041
wds_RXBytesOKTlv, 1044
wds_TXBytesOKTlv, 1046

tlvPresent
voice_USSDNotificationNetworkInfo, 1026

TlvResultCode
qmTlvResult, 517

Tlvresult
pack_dms_GetCustFeaturesV2_t, 327
pack_dms_SetCustFeaturesV2_t, 332
pack_dms_SetPower_t, 334
pack_dms_SetUSBComp_t, 334
pack_dms_UIMGetICCID_t, 341
pack_fms_GetImagesPreference_t, 345
pack_fms_GetStoredImages_t, 345
pack_fms_SetImagesPreference_t, 346
pack_loc_Delete_Assist_Data_t, 355
pack_loc_EventRegister_t, 357
pack_loc_SetExtPowerState_t, 357
pack_loc_SetOperationMode_t, 358
pack_loc_SLQSLOCGetBestAvailPos_t, 358
pack_loc_Start_t, 370
pack_loc_Stop_t, 370
pack_nas_SetNetworkPreference_t, 374
pack_uim_ChangePin_t, 455
pack_uim_ReadTransparent_t, 456
pack_uim_SetPinProtection_t, 457
pack_uim_UnblockPin_t, 465
pack_uim_VerifyPin_t, 466
unpack_cat_SetCATEventCallback_t, 581
unpack_dms_GetBandCapability_t, 584
unpack_dms_GetCrashAction_t, 585
unpack_dms_GetCustFeature_t, 587
unpack_dms_GetCustFeaturesV2_t, 588
unpack_dms_GetDeviceCap_t, 589
unpack_dms_GetDeviceHardwareRev_t, 591
unpack_dms_GetDeviceMfr_t, 592
unpack_dms_GetDeviceSerialNumbers_t, 594
unpack_dms_GetFirmwareInfo_t, 596
unpack_dms_GetFirmwareRevision_t, 597
unpack_dms_GetFirmwareRevisions_t, 598
unpack_dms_GetFSN_t, 599
unpack_dms_GetIMSI_t, 600
unpack_dms_GetManufacturer_t, 600
unpack_dms_GetModelID_t, 601
unpack_dms_GetNetworkTime_t, 602
unpack_dms_GetOfflineReason_t, 603
unpack_dms_GetPower_t, 604
unpack_dms_GetPRLVersion_t, 605
unpack_dms_GetUSBComp_t, 608
unpack_dms_GetVoiceNumber_t, 609
unpack_dms_PSMCfgChange_ind_t, 610
unpack_dms_ResetToFactoryDefaults_t, 611
unpack_dms_SetActivationStatusCallback_t, 611
unpack_dms_SetCustFeature_t, 612
unpack_dms_SetCustFeaturesV2_t, 613
unpack_dms_SetEventReport_ind_t, 614
unpack_dms_SetEventReport_t, 614
unpack_dms_SetFirmwarePreference_t, 615
unpack_dms_SetIndicationRegister_t, 615
unpack_dms_SetPower_t, 616
unpack_dms_SetUSBComp_t, 616
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 620
unpack_dms_SLQSDmsSwiGetResetInfo_t, 621
unpack_dms_SLQSDmsSwiIndicationRegister_t, 623
unpack_dms_SLQSGetERIFile_t, 628
unpack_dms_SLQSSetPowerSaveModeConfig_t, 630
unpack_dms_SLQSSwiClearDyingGaspStatistics-_t, 630
unpack_dms_SLQSSwiGetCrashInfo_t, 631
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 632
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 632
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 635

unpack_dms_SLQSSwiGetHostDevInfo_t, 636
unpack_dms_SLQSSwiGetOSInfo_t, 637
unpack_dms_SLQSSwiGetSerialNoExt_t, 638
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 638
unpack_dms_SLQSSwiSetHostDevInfo_t, 639
unpack_dms_SLQSSwiSetOSInfo_t, 639
unpack_dms_SLQSUIMGetState_t, 640
unpack_dms_SwiSetEventReport_t, 641
unpack_dms_SwiUimSelect_t, 642
unpack_dms_UIMGetControlKeyStatus_t, 643
unpack_dms_UIMGetICCID_t, 643
unpack_dms_UIMGetPINStatus_t, 645
unpack_dms_UIMSetControlKeyProtection_t, 646
unpack_dms_UIMSetPINProtection_t, 647
unpack_dms_UIMUnblockControlKey_t, 647
unpack_fms_GetImagesPreference_t, 648
unpack_fms_GetStoredImages_t, 649
unpack_fms_SetImagesPreference_t, 649
unpack_loc_BestAvailPos_Ind_t, 679
unpack_loc_Delete_Assist_Data_t, 680
unpack_loc_DeleteAssistData_Ind_t, 681
unpack_loc_EngineState_Ind_t, 682
unpack_loc_EventNMEA_Ind_t, 682
unpack_loc_EventRegister_t, 683
unpack_loc_GetServer_Ind_t, 685
unpack_loc_GnssSvInfo_Ind_t, 686
unpack_loc_PositionRpt_Ind_t, 696
unpack_loc_SetExtPowerConfig_Ind_t, 698
unpack_loc_SetExtPowerState_t, 698
unpack_loc_SetOperationMode_Ind_t, 699
unpack_loc_SetOperationMode_t, 700
unpack_loc_SetServer_Ind_t, 701
unpack_loc_SLQSLOCGetBestAvailPos_t, 701
unpack_loc_Start_t, 702
unpack_loc_Stop_t, 702
unpack_nas_GetNetworkPreference_t, 710
unpack_nas_SetNetworkPreference_t, 721
unpack_nas_SetServingSystemCallback_ind_t, 722
unpack_nas_SlqsGetLTECphyCAInfo_t, 724
unpack_nas_SLQSNasGetRFInfo_t, 755
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 765
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 766
unpack_result_t, 816
unpack_swidms_SLQSSwiDmsSetMTU_t, 855
unpack_swidms_SLQSSwiDmsSetUsbComp_t, 855
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 864
unpack_tmd_SLQSTmdGetMitigationDevList_t, 865
unpack_tmd_SLQSTmdGetMitigationLvl_t, 865
unpack_tmd_SLQSTmdRegNotMitigationLvl_t, 867
unpack_uim_ChangePin_t, 868
unpack_uim_GetCardStatus_t, 868
unpack_uim_ReadTransparent_t, 869
unpack_uim_SetPinProtection_t, 870
unpack_uim_UnblockPin_t, 878
unpack_uim_VerifyPin_t, 878
unpack_wds_SLQSCreateProfile_t, 942
unpack_wds_SLQSGetProfileSettings_t, 952
unpack_wds_SLQSSetIPFamilyPreference_t, 957
tmd.h, 1399
pack_tmd_SLQSTmdDeRegNotMitigationLvl, 1400
pack_tmd_SLQSTmdGetMitigationDevList, 1400
pack_tmd_SLQSTmdGetMitigationLvl, 1400
pack_tmd_SLQSTmdRegNotMitigationLvl, 1401
TMD_MAX_DEV_LIST, 1400
unpack_tmd_SLQSTmdDeRegNotMitigationLvl, 1401
unpack_tmd_SLQSTmdGetMitigationDevList, 1401
unpack_tmd_SLQSTmdGetMitigationLvl, 1402
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind, 1402
unpack_tmd_SLQSTmdRegNotMitigationLvl, 1403
tmd_mitigationDevList, 550
maxMitigationLevel, 550
mitigationDevId, 550
mitigationDevIdLen, 550
tmrSIPRegn
ims_SIPRegnTmrInfo, 81
tmrT1
ims_TmrT1Info, 83
tmrT2
ims_TmrT2Info, 84
tmrTf
ims_TmrTfInfo, 84
toServiceId
sms_BroadcastConfig, 525
toggleMode
voice_lineCtrlInfo, 1015
TokenBucket
unpack_qos_swiQosFlow_t, 814
tokenRate
unpack_qos_tokenBucket_t, 815
tosMask
LibPackTFTIDParams, 120
total_rx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 803
total_rx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 803
total_tx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 803
total_tx_bytes_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 803
total_tx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 803
total_tx_pkt_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 803
TrackAreaCode
unpack_nas_SLQSGetServingSystem_t, 733
TrackingStatus

unpack_pds_SetPdsState_Ind_t, 788
TrafficClass
unpack_qos_swiQosFlow_t, 814
trafficClass
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
trafficPriority
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
TranDstPort
unpack_qos_swiQosFilter_t, 810
TranSrcPort
unpack_qos_swiQosFilter_t, 810
TransCap
sms_transLayerInfo, 537
TransType
sms_transLayerInfo, 538
transactionID
sMSTransferRouteMTMessageInfo, 543
transferDelay
LibPackUMTSQoS, 122
wds_UMTSMinQoS, 1049
TransferRouteMTMessageInfo
transferRouteMessageTlv, 551
transferRouteMessageTlv, 550
TlvPresent, 551
TransferRouteMTMessageInfo, 551
transferStats
pack_wds_SLQSSetWdsEventCallback_t, 505
trueSrvStatus
nas_GSMSrvStatusInfo, 182
nas_NR5GSerStatTlv, 231
tx_bytes
unpack_QosFlowStat_t, 816
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
tx_bytes_drp
unpack_QosFlowStat_t, 816
tx_pkt
unpack_QosFlowStat_t, 816
tx_pkt_drp
unpack_QosFlowStat_t, 816
tx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
TxByteOKCnt
wds_TXBytesOKTlv, 1046
TxDropConutTlv
unpack_RMTransferStatistics_ind_t, 819
txOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 952
TxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 819
TxOkConutTlv
unpack_RMTransferStatistics_ind_t, 819
txPower
nas_txInfo, 286
TxQFilter
unpack_qos_QosFlowInfo_t, 799
TxQFlowGranted
unpack_qos_QosFlowInfo_t, 799
type
pack_wds_GetDefaultProfileNum_t, 486
pack_wds_SetDefaultProfileNum_t, 492
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 620
unpack_dms_SLQSDmsSwiGetResetInfo_t, 621
unpack_qmi_t, 794
typeSize
unpack_sms_GetSMSCAddress_t, 820

u16PRLVersion
unpack_dms_GetPRLVersion_t, 605
u8PRLPreference
unpack_dms_GetPRLVersion_t, 605
UDPDstPort
unpack_qos_swiQosFilter_t, 810
UDPSrcPort
unpack_qos_swiQosFilter_t, 810
ULONG
SwiDataTypes.h, 1375
ULONGLONG
SwiDataTypes.h, 1375
UMTSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 956
UMTSInstInfo
nas_UMTSInfo, 288
UMTSLTENbrCell
nas_WCDMAInfoLTENeighborCell, 293
UMTSReqQoS
LibPackUMTSReqQoSSigInd, 123
UNIQUE_ID_LEN
dms.h, 1077
UNUSEDPARAM
common.h, 1065
SwiDataTypes.h, 1375
uPhyCardStatus
slot_t, 522
uPhySlotStatus
slot_t, 522
USBComp
pack_dms_SetUSBComp_t, 334
unpack_dms_GetUSBComp_t, 608
USHORT
SwiDataTypes.h, 1375
USSDNotificationNetworkInfo
unpack_voice_USSDNotificationCallback_ind_t, 920
USSInformation
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 479
UUSData
voice_UUSInfo, 1027
UUSDatalen
voice_UUSInfo, 1027
UUSDcs
voice_UUSInfo, 1027
UUSType

voice_UUSInfo, 1027
uarfcn
nas_lteWcdmaCellInfo, 220
nas_UMTSInfo, 288
nas_wcdmaUARFCN, 298
ueInIdle
nas_LTEInfoInterfreq, 201
nas_LTEInfoIntrafreq, 203
nas_LTEInfoNeighboringGSM, 204
nas_LTEInfoNeighboringWCDMA, 205
uim.h, 1403
MAX_ICCID_LENGTH, 1406
MAX_NO_OF_SLOTS, 1406
MAX_SLOTS_STATUS, 1406
pack_uim_ChangePin, 1407
pack_uim_GetCardStatus, 1407
pack_uim_ReadTransparent, 1407
pack_uim_SLQSUIMAuthenticate, 1408
pack_uim_SLQSUIMDepersonalization, 1408
pack_uim_SLQSUIMEventRegister, 1409
pack_uim_SLQSUIMGetConfiguration, 1409
pack_uim_SLQSUIMGetFileAttributes, 1410
pack_uim_SLQSUIMGetSlotsStatus, 1410
pack_uim_SLQSUIMPowerDown, 1410
pack_uim_SLQSUIMPowerUp, 1411
pack_uim_SLQSUIMRefreshComplete, 1411
pack_uim_SLQSUIMRefreshGetLastEvent, 1411
pack_uim_SLQSUIMRefreshOK, 1412
pack_uim_SLQSUIMRefreshRegister, 1412
pack_uim_SLQSUIMReset, 1413
pack_uim_SLQSUIMSwitchSlot, 1413
pack_uim_SetPinProtection, 1408
pack_uim_UnblockPin, 1413
pack_uim_VerifyPin, 1414
unpack_uim_ChangePin, 1414
unpack_uim_GetCardStatus, 1414
unpack_uim_ReadTransparent, 1415
unpack_uim_SLQSUIMAuthenticate, 1416
unpack_uim_SLQSUIMDepersonalization, 1416
unpack_uim_SLQSUIMEventRegister, 1417
unpack_uim_SLQSUIMGetConfiguration, 1417
unpack_uim_SLQSUIMGetFileAttributes, 1417
unpack_uim_SLQSUIMGetSlotsStatus, 1418
unpack_uim_SLQSUIMPowerDown, 1418
unpack_uim_SLQSUIMPowerDown_t, 1406
unpack_uim_SLQSUIMPowerUp, 1418
unpack_uim_SLQSUIMPowerUp_t, 1406
unpack_uim_SLQSUIMRefreshCallback_Ind, 1419
unpack_uim_SLQSUIMRefreshComplete, 1419
unpack_uim_SLQSUIMRefreshComplete_t, 1406
unpack_uim_SLQSUIMRefreshGetLastEvent, 1420
unpack_uim_SLQSUIMRefreshOK, 1420
unpack_uim_SLQSUIMRefreshOK_t, 1407
unpack_uim_SLQSUIMRefreshRegister, 1420
unpack_uim_SLQSUIMRefreshRegister_t, 1407
unpack_uim_SLQSUIMReset, 1421
unpack_uim_SLQSUIMReset_t, 1407
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind, 1421
unpack_uim_SLQSUIMSwitchSlot, 1421
unpack_uim_SLQSUIMSwitchSlot_t, 1407
unpack_uim_SetPinProtection, 1415
unpack_uim_SetUimSlotStatusChangeCallback_ind, 1416
unpack_uim_UnblockPin, 1422
unpack_uim_VerifyPin, 1422
uim_UIMSessionInformation, 573
aid, 574
aidLength, 574
sessionType, 574
uim_appStatus, 551
aidLength, 553
aidVal, 553
appState, 553
appType, 553
persoFeature, 553
persoRetries, 553
persoState, 553
persoUnblockRetries, 553
pin1Retries, 553
pin1State, 553
pin2Retries, 553
pin2State, 554
puk1Retries, 554
puk2Retries, 554
univPin, 554
uim_authenticateResult, 554
content, 554
contentLen, 554
uim_authenticationData, 554
context, 555
data, 555
dataLen, 555
uim_cardResult, 555
sw1, 556
sw2, 556
uim_cardStatus, 556
index1xPri, 557
index1xSec, 557
indexGwPri, 557
indexGwSec, 557
numSlot, 557
SlotInfo, 557
uim_changeUIMPIN, 557
oldPINLen, 558
oldPINVal, 558
pinID, 558
pinLen, 558
pinVal, 558
uim_depersonalizationInformation, 558
ckLen, 559
ckVal, 559
feature, 559
operation, 559
uim_encryptedPIN1, 559

pin1Len, 560
pin1Val, 560
uim_fileAttributes, 560
fileID, 563
fileSize, 563
fileType, 563
rawLen, 563
rawValue, 563
recordCount, 563
recordSize, 563
secActivate, 563
secActivateMask, 563
secDeactivate, 563
secDeactivateMask, 563
secIncrease, 563
secIncreaseMask, 563
secRead, 563
secReadMask, 563
secWrite, 563
secWriteMask, 563
uim_fileInfo, 563
fileID, 564
path, 564
pathLen, 564
uim_hotSwapStatus, 564
hotSwap, 564
hotSwapLength, 564
uim_personalizationStatus, 564
feature, 565
numFeatures, 565
unblockLeft, 565
verifyLeft, 565
uim_readResult, 565
content, 566
contentLen, 566
uim_readTransparentInfo, 566
length, 566
offset, 566
uim_refreshevent, 566
aid, 568
aidLength, 568
arrfileInfo, 568
mode, 568
numOfFiles, 568
sessionType, 568
stage, 568
uim_registerRefresh, 568
arrfileInfo, 569
numFiles, 569
registerFlag, 569
voteForInit, 569
uim_remainingRetries, 569
unblockLeft, 569
verifyLeft, 569
uim_select
pack_dms_SwiUimSelect_t, 339
uim_sessionInformation, 570
aid, 570
aidLength, 570
sessionType, 570
uim_setPINProtection, 570
pinID, 571
pinLength, 571
pinOperation, 571
pinValue, 571
uim_slotInfo, 571
AppStatus, 573
cardState, 573
errorState, 573
numApp, 573
upinRetries, 573
upinState, 573
upukRetries, 573
uim_unblockUIMPIN, 574
newPINLen, 574
newPINVal, 574
pinID, 574
pukLen, 574
pukVal, 575
uim_verifyUIMPIN, 575
pinID, 575
pinLen, 575
pinVal, 575
uimAutoSwitchActSlot
dms_UimAutoSwitchActSlotTlv, 60
uimSelect
unpack_dms_SLQSDmsSwiGetUimSelection_t, 622
uimSlotStatus
slots_t, 525
UimStatusTlv
unpack_dms_SwiEventReportCallBack_ind_t, 641
ulData
wds_DataULongTlv, 1032
ulMask
rmTrasnferStaticsReq, 520
ulMax
PackSwiAvmsSetSettingsPeriodInfo, 514
ulMin
PackSwiAvmsSetSettingsPeriodInfo, 514
ulPhysicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 464
ulPkgDownloadComplete
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 971
ulPkgDownloadSize
UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 972
ullData
wds_DataULongLongTlv, 1032
umtsEcio
nas_UMTSinstInfo, 289
umtsInst
nas_UMTSInfo, 288
umtsLTENbrCellLen
nas_WCDMAInfoLTENeighborCell, 293

umtsPsc
nas_UMTSinstInfo, 289
umtsRscp
nas_UMTSinstInfo, 289
umtsUarfcn
nas_UMTSinstInfo, 289
UnPackGetProfileSettingOut, 969
curProfile, 970
pExtErrCode, 970
unblockLeft
uim_personalizationStatus, 565
uim_remainingRetries, 569
unblockRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 643
unpack_dms_UIMSetPINProtection_t, 647
unpack_dms_UIMUnblockControlKey_t, 647
uniqueID
image_info_t, 71
univPin
appStats, 38
uim_appStatus, 554
universalTime
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 761
unpack_QosFlowStat_t, 815
bearerId, 816
tx_bytes, 816
tx_bytes_drp, 816
tx_pkt, 816
tx_pkt_drp, 816
unpack_RMTransferStatistics_ind_t, 818
ParamPresenceMask, 818
RxDropConutTlv, 818
RxOkByteCountTlv, 818
RxOkConutTlv, 819
TxDropConutTlv, 819
TxOkByteCountTlv, 819
TxOkConutTlv, 819
unpack_audio_SLQSGetAudioPathConfig
audio.h, 1057
unpack_audio_SLQSGetAudioPathConfig_t, 575
pCodecSTGain, 577
pECMode, 577
pMICGainSelect, 577
pNSEnable, 577
pRXAGCList, 577
pRXAVCList, 577
pTXAGCList, 577
pTXAVCSwitch, 577
pTXGain, 577
ParamPresenceMask, 577
unpack_audio_SLQSGetAudioProfile
audio.h, 1057
unpack_audio_SLQSGetAudioProfile_t, 578
EarMute, 578
MicMute, 579
ParamPresenceMask, 579
Profile, 579
Volume, 579
unpack_audio_SLQSGetAudioVolTLBConfig
audio.h, 1057
unpack_audio_SLQSGetAudioVolTLBConfig_t, 579
ParamPresenceMask, 579
ResCode, 579
unpack_audio_SLQSSetAudioPathConfig
audio.h, 1058
unpack_audio_SLQSSetAudioProfile
audio.h, 1058
unpack_audio_SLQSSetAudioVolTLBConfig
audio.h, 1058
unpack_audio_SLQSSetAudioVolTLBConfig_t, 579
ParamPresenceMask, 580
ResCode, 580
unpack_cat_CATSendEnvelopeCommand
cat.h, 1061
unpack_cat_CATSendEnvelopeCommand_t
cat.h, 1060
unpack_cat_CATSendTerminalResponse
cat.h, 1061
unpack_cat_CATSendTerminalResponse_t
cat.h, 1060
unpack_cat_SetCATEventCallback
cat.h, 1062
unpack_cat_SetCATEventCallback_t, 580
errorMask, 581
ParamPresenceMask, 581
Tlvresult, 581
unpack_cat_SetCatEventCallback_ind
cat.h, 1062
unpack_cat_SetCatEventCallback_ind_t, 580
CCETlv, 580
event_Index, 580
ParamPresenceMask, 580
unpack_dms_ActivateAutomatic
dms.h, 1102
unpack_dms_GetActivationState
dms.h, 1103
unpack_dms_GetActivationState_t, 581
ParamPresenceMask, 582
state, 582
unpack_dms_GetBandCapability
dms.h, 1103
unpack_dms_GetBandCapability_t, 582
BandCapability, 584
ParamPresenceMask, 584
Tlvresult, 584
unpack_dms_GetCrashAction
dms.h, 1103
unpack_dms_GetCrashAction_t, 584
DevCrashState, 584
ParamPresenceMask, 584
Tlvresult, 585
unpack_dms_GetCustFeature
dms.h, 1104
unpack_dms_GetCustFeature_t, 585
DHCPRelayEnabled, 587

DisableIMSI, 587
GPSLPM, 587
GPSSel, 587
GpsEnable, 587
IPFamSupport, 587
IsVoiceEnabled, 587
ParamPresenceMask, 587
RMAutoConnect, 587
SMSSupport, 587
Tlvresult, 587
unpack_dms_GetCustFeaturesV2
dms.h, 1104
unpack_dms_GetCustFeaturesV2_t, 587
GetCustomFeatureV2, 588
ParamPresenceMask, 588
Tlvresult, 588
unpack_dms_GetDeviceCap
dms.h, 1104
unpack_dms_GetDeviceCap_t, 588
DataServiceCapability, 589
MaxRXChannelRate, 589
MaxTXChannelRate, 589
ParamPresenceMask, 589
RadioIfaces, 589
RadioIfacesSize, 589
SimCapability, 589
Tlvresult, 589
unpack_dms_GetDeviceCapabilities
dms.h, 1105
unpack_dms_GetDeviceCapabilities_t, 589
dataServiceCaCapability, 590
maxRxChannelRate, 591
maxTxChannelRate, 591
ParamPresenceMask, 591
RadioIfaces, 591
radioIfacesSize, 591
simCapability, 591
unpack_dms_GetDeviceHardwareRev
dms.h, 1105
unpack_dms_GetDeviceHardwareRev_t, 591
ParamPresenceMask, 591
String, 591
stringSize, 591
Tlvresult, 591
unpack_dms_GetDeviceMfr
dms.h, 1105
unpack_dms_GetDeviceMfr_t, 592
ParamPresenceMask, 592
String, 592
stringSize, 592
Tlvresult, 592
unpack_dms_GetDeviceSerialNumbers
dms.h, 1106
unpack_dms_GetDeviceSerialNumbers_t, 592
ESNString, 594
esnSize, 594
IMEIString, 594
imeiSize, 594
imeiSvnSize, 594
ImeiSvnString, 594
MEIDString, 594
meidSize, 594
ParamPresenceMask, 594
Tlvresult, 594
unpack_dms_GetFSN
dms.h, 1107
unpack_dms_GetFSN_t, 598
ParamPresenceMask, 599
String, 599
Tlvresult, 599
unpack_dms_GetFirmwareInfo
dms.h, 1106
unpack_dms_GetFirmwareInfo_t, 594
appversion_str, 595
bootversion_str, 595
carrier_str, 595
cur_carr_name, 596
cur_carr_rev, 596
modelid_str, 596
packageid_str, 596
ParamPresenceMask, 596
priversion_str, 596
sku_str, 596
Tlvresult, 596
unpack_dms_GetFirmwareRevision
dms.h, 1106
unpack_dms_GetFirmwareRevision_t, 596
AMSSString, 597
amssSize, 597
PRIString, 597
ParamPresenceMask, 597
Tlvresult, 597
unpack_dms_GetFirmwareRevisions
dms.h, 1107
unpack_dms_GetFirmwareRevisions_t, 597
AMSSString, 598
amssSize, 598
bootSize, 598
BootString, 598
PRIString, 598
ParamPresenceMask, 598
priSize, 598
Tlvresult, 598
unpack_dms_GetHardwareRevision
dms.h, 1107
unpack_dms_GetHardwareRevision_t, 599
hwVer, 599
ParamPresenceMask, 599
unpack_dms_GetIMSI
dms.h, 1108
unpack_dms_GetIMSI_t, 599
imsi, 600
ParamPresenceMask, 600
Tlvresult, 600
unpack_dms_GetManufacturer
dms.h, 1108

unpack_dms_GetManufacturer_t, 600
manufacturer, 600
ParamPresenceMask, 600
Tlvresult, 600
unpack_dms_GetModelID
dms.h, 1108
unpack_dms_GetModelID_t, 601
modelid, 601
ParamPresenceMask, 601
Tlvresult, 601
unpack_dms_GetNetworkTime
dms.h, 1109
unpack_dms_GetNetworkTime_t, 601
ParamPresenceMask, 602
source, 602
timestamp, 602
Tlvresult, 602
unpack_dms_GetOfflineReason
dms.h, 1109
unpack_dms_GetOfflineReason_t, 602
pReasonMask, 603
ParamPresenceMask, 603
pbPlatform, 603
Tlvresult, 603
unpack_dms_GetPRLVersion
dms.h, 1110
unpack_dms_GetPRLVersion_t, 604
ParamPresenceMask, 605
Tlvresult, 605
u16PRLVersion, 605
u8PRLPreference, 605
unpack_dms_GetPower
dms.h, 1109
unpack_dms_GetPower_t, 603
HardwareControlledMode, 604
OfflineReason, 604
OperationMode, 604
ParamPresenceMask, 604
Tlvresult, 604
unpack_dms_GetSerialNumbers
dms.h, 1110
unpack_dms_GetSerialNumbers_t, 605
esn, 606
imei_no, 606
imeisv_svn, 606
meid, 606
ParamPresenceMask, 606
unpack_dms_GetUSBComp
dms.h, 1110
unpack_dms_GetUSBComp_t, 606
NumSupUSBComps, 608
ParamPresenceMask, 608
SupUSBComps, 608
Tlvresult, 608
USBComp, 608
unpack_dms_GetVoiceNumber
dms.h, 1111
unpack_dms_GetVoiceNumber_t, 608
MIN, 609
minSize, 609
ParamPresenceMask, 609
Tlvresult, 609
VoiceNumber, 609
voiceNumberSize, 609
unpack_dms_PSMCfgChange_ind
dms.h, 1111
unpack_dms_PSMCfgChange_ind_t, 610
ActiveTimer, 610
EnableState, 610
ParamPresenceMask, 610
PeriodicUpdateTimer, 610
Tlvresult, 610
unpack_dms_ResetToFactoryDefaults
dms.h, 1111
unpack_dms_ResetToFactoryDefaults_t, 610
ParamPresenceMask, 611
Tlvresult, 611
unpack_dms_SLQSDmsSwiGetPCInfo
dms.h, 1115
unpack_dms_SLQSDmsSwiGetPCInfo_t, 616
has_LpmFlag, 618
has_PersistentLpm, 618
has_Wdisable, 618
LpmFlag, 619
opMode, 619
ParamPresenceMask, 619
PersistentLpm, 619
PowerOffMode, 619
Wdisable, 619
unpack_dms_SLQSDmsSwiGetResetInfo
dms.h, 1116
unpack_dms_SLQSDmsSwiGetResetInfo_Ind
dms.h, 1116
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 619
ParamPresenceMask, 620
source, 620
Tlvresult, 620
type, 620
unpack_dms_SLQSDmsSwiGetResetInfo_t, 620
ParamPresenceMask, 621
source, 621
Tlvresult, 621
type, 621
unpack_dms_SLQSDmsSwiGetUimSelection
dms.h, 1116
unpack_dms_SLQSDmsSwiGetUimSelection_t, 621
pUimAutoSwitchActSlot, 622
ParamPresenceMask, 622
uimSelect, 622
unpack_dms_SLQSDmsSwiIndicationRegister
dms.h, 1117
unpack_dms_SLQSDmsSwiIndicationRegister_t, 622
ParamPresenceMask, 623
Tlvresult, 623
unpack_dms_SLQSGetBandCapability
dms.h, 1117

unpack_dms_SLQSGetBandCapability_t, 623
bandCapability, 627
LteBandCapability, 627
ParamPresenceMask, 627
TdsBandCapability, 627
unpack_dms_SLQSGetERIFile
dms.h, 1117
unpack_dms_SLQSGetERIFile_t, 627
eriFile, 628
ParamPresenceMask, 628
Tlvresult, 628
unpack_dms_SLQSGetPowerSaveModeConfig
dms.h, 1118
unpack_dms_SLQSGetPowerSaveModeConfig_t, 628
pActiveTimer, 629
pDurationThreshold, 629
pEarlyWakeupTime, 629
pPeriodicUpdateTimer, 629
pPsmEnableState, 629
pRandomizationWindow, 629
ParamPresenceMask, 629
unpack_dms_SLQSSetPowerSaveModeConfig
dms.h, 1118
unpack_dms_SLQSSetPowerSaveModeConfig_t, 629
ParamPresenceMask, 630
Tlvresult, 630
unpack_dms_SLQSSwiClearDyingGaspStatistics
dms.h, 1119
unpack_dms_SLQSSwiClearDyingGaspStatistics_t, 630
ParamPresenceMask, 630
Tlvresult, 630
unpack_dms_SLQSSwiGetCrashInfo
dms.h, 1119
unpack_dms_SLQSSwiGetCrashInfo_t, 630
crashInfoParam, 631
ParamPresenceMask, 631
Tlvresult, 631
unpack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 1119
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 631
pGetDyingGaspCfg, 632
ParamPresenceMask, 632
Tlvresult, 632
unpack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 1120
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 632
pGetDyingGaspStatistics, 632
ParamPresenceMask, 632
Tlvresult, 632
unpack_dms_SLQSSwiGetFirmwareCurr
dms.h, 1120
unpack_dms_SLQSSwiGetFirmwareCurr_t, 632
carrier, 633
fwvers, 633
numEntries, 633
pCurrImgInfo, 633
ParamPresenceMask, 633
pkgver, 633
priver, 633
unpack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 1120
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 633
imgType, 635
logString, 635
ParamPresenceMask, 635
refData, 635
refString, 635
ResCode, 635
Tlvresult, 635
unpack_dms_SLQSSwiGetHostDevInfo
dms.h, 1121
unpack_dms_SLQSSwiGetHostDevInfo_t, 635
hostID, 636
manString, 636
modelString, 636
ParamPresenceMask, 636
plasmaIDString, 636
swVerString, 636
Tlvresult, 636
unpack_dms_SLQSSwiGetOSInfo
dms.h, 1121
unpack_dms_SLQSSwiGetOSInfo_t, 636
nameString, 637
ParamPresenceMask, 637
Tlvresult, 637
versionString, 637
unpack_dms_SLQSSwiGetSerialNoExt
dms.h, 1121
unpack_dms_SLQSSwiGetSerialNoExt_t, 637
meidString, 638
ParamPresenceMask, 638
Tlvresult, 638
unpack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 1122
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 638
ParamPresenceMask, 638
Tlvresult, 638
unpack_dms_SLQSSwiSetHostDevInfo
dms.h, 1122
unpack_dms_SLQSSwiSetHostDevInfo_t, 638
ParamPresenceMask, 639
Tlvresult, 639
unpack_dms_SLQSSwiSetOSInfo
dms.h, 1122
unpack_dms_SLQSSwiSetOSInfo_t, 639
ParamPresenceMask, 639
Tlvresult, 639
unpack_dms_SLQSUIMGetState
dms.h, 1123
unpack_dms_SLQSUIMGetState_t, 639
ParamPresenceMask, 640
state, 640
Tlvresult, 640
unpack_dms_SetActivationStatusCallback
dms.h, 1112

unpack_dms_SetActivationStatusCallback_t, 611
ParamPresenceMask, 611
Tlvresult, 611
unpack_dms_SetCrashAction
dms.h, 1112
unpack_dms_SetCrashAction_t, 611
notused, 612
ParamPresenceMask, 612
unpack_dms_SetCustFeature
dms.h, 1112
unpack_dms_SetCustFeature_t, 612
ParamPresenceMask, 612
Tlvresult, 612
unpack_dms_SetCustFeaturesV2
dms.h, 1113
unpack_dms_SetCustFeaturesV2_t, 612
ParamPresenceMask, 613
Tlvresult, 613
unpack_dms_SetEventReport
dms.h, 1113
unpack_dms_SetEventReport_ind
dms.h, 1113
unpack_dms_SetEventReport_ind_t, 613
ActivationStatusTlv, 613
OperatingModeTlv, 613
ParamPresenceMask, 614
Tlvresult, 614
unpack_dms_SetEventReport_t, 614
ParamPresenceMask, 614
Tlvresult, 614
unpack_dms_SetFirmwarePreference
dms.h, 1114
unpack_dms_SetFirmwarePreference_t, 614
ParamPresenceMask, 615
Tlvresult, 615
unpack_dms_SetIndicationRegister
dms.h, 1114
unpack_dms_SetIndicationRegister_t, 615
ParamPresenceMask, 615
Tlvresult, 615
unpack_dms_SetPower
dms.h, 1115
unpack_dms_SetPower_t, 615
ParamPresenceMask, 616
Tlvresult, 616
unpack_dms_SetUSBComp
dms.h, 1115
unpack_dms_SetUSBComp_t, 616
ParamPresenceMask, 616
Tlvresult, 616
unpack_dms_SwiEventReportCallBack_ind
dms.h, 1123
unpack_dms_SwiEventReportCallBack_ind_t, 640
ParamPresenceMask, 641
TempTlv, 641
UimStatusTlv, 641
VoltTlv, 641
unpack_dms_SwiSetEventReport
dms.h, 1123
unpack_dms_SwiSetEventReport_t, 641
ParamPresenceMask, 641
Tlvresult, 641
unpack_dms_SwiUimSelect
dms.h, 1124
unpack_dms_SwiUimSelect_t, 641
ParamPresenceMask, 642
Tlvresult, 642
unpack_dms_UIMChangePIN
dms.h, 1124
unpack_dms_UIMGetControlKeyStatus
dms.h, 1124
unpack_dms_UIMGetControlKeyStatus_t, 642
facilityState, 643
ParamPresenceMask, 643
Tlvresult, 643
unblockRetriesLeft, 643
verifyRetriesLeft, 643
unpack_dms_UIMGetICCID
dms.h, 1125
unpack_dms_UIMGetICCID_t, 643
ParamPresenceMask, 643
String, 643
stringSize, 643
Tlvresult, 643
unpack_dms_UIMGetPINStatus
dms.h, 1125
unpack_dms_UIMGetPINStatus_t, 644
p1Status, 645
p1UnblockRetriesLeft, 645
p1VerifyRetriesLeft, 645
p2Status, 645
p2UnblockRetriesLeft, 645
p2VerifyRetriesLeft, 645
ParamPresenceMask, 645
Tlvresult, 645
unpack_dms_UIMSetControlKeyProtection
dms.h, 1125
unpack_dms_UIMSetControlKeyProtection_t, 645
ParamPresenceMask, 646
Tlvresult, 646
verifyRetriesLeft, 646
unpack_dms_UIMSetPINProtection
dms.h, 1126
unpack_dms_UIMSetPINProtection_t, 646
ParamPresenceMask, 647
Tlvresult, 647
unblockRetriesLeft, 647
verifyRetriesLeft, 647
unpack_dms_UIMUnblockControlKey
dms.h, 1126
unpack_dms_UIMUnblockControlKey_t, 647
ParamPresenceMask, 647
Tlvresult, 647
unblockRetriesLeft, 647
unpack_dms_UIMUnblockPIN
dms.h, 1126

unpack_dms_UIMVerifyPIN
dms.h, 1127
unpack_dms_ValidateSPC
dms.h, 1127
unpack_fms_GetImagesPreference
fms.h, 1130
unpack_fms_GetImagesPreference_t, 647
ImageListSize, 648
pImageList, 648
ParamPresenceMask, 648
Tlvresult, 648
unpack_fms_GetStoredImages
fms.h, 1130
unpack_fms_GetStoredImages_t, 648
imageList, 649
imagelistSize, 649
ParamPresenceMask, 649
Tlvresult, 649
unpack_fms_SetImagesPreference
fms.h, 1131
unpack_fms_SetImagesPreference_t, 649
ImageTypes, 649
ImageTypesSize, 649
ParamPresenceMask, 649
Tlvresult, 649
unpack_ims_SLQSGetIMSSMSConfig
ims.h, 1138
unpack_ims_SLQSGetIMSSMSConfig_t, 649
pSettingResp, 650
ParamPresenceMask, 650
unpack_ims_SLQSGetIMSUserConfig
ims.h, 1138
unpack_ims_SLQSGetIMSUserConfig_t, 650
pIMSDomain, 651
pSettingResp, 651
ParamPresenceMask, 651
unpack_ims_SLQSGetIMSVoIPConfig
ims.h, 1139
unpack_ims_SLQSGetIMSVoIPConfig_t, 651
pAmrMode, 653
pRingingTimer, 654
pSettingResp, 654
ParamPresenceMask, 654
unpack_ims_SLQSGetRegMgrConfig
ims.h, 1139
unpack_ims_SLQSGetRegMgrConfig_t, 654
pIMSTestMode, 655
pPCSCFPort, 655
pSettingResp, 655
ParamPresenceMask, 655
unpack_ims_SLQSGetSIPConfig
ims.h, 1139
unpack_ims_SLQSGetSIPConfig_t, 655
pSIPLocalPort, 656
pSettingResp, 656
pSigCompEnabled, 656
pSubscribeTimer, 656
pTimerSIPReg, 656
pTimerT1, 656
pTimerT2, 656
pTimerTf, 656
ParamPresenceMask, 656
unpack_ims_SLQSImsConfigIndicationRegister
ims.h, 1140
unpack_ims_SLQSImsConfigIndicationRegister_t
ims.h, 1133
unpack_ims_SLQSRegMgrCfgCallBack_ind
ims.h, 1140
unpack_ims_SLQSRegMgrCfgCallBack_ind_t, 656
PCTlv, 657
PNTlv, 657
ParamPresenceMask, 657
TMTlv, 657
unpack_ims_SLQSSIPCfgCallBack_ind
ims.h, 1142
unpack_ims_SLQSSIPCfgCallBack_ind_t, 660
ParamPresenceMask, 661
SCTlv, 661
SPTlv, 661
SRTlv, 661
STTlv, 661
TT1Tlv, 661
TT2Tlv, 661
TTfTlv, 661
unpack_ims_SLQSSMSCfgCallBack_ind
ims.h, 1142
unpack_ims_SLQSSMSCfgCallBack_ind_t, 661
ParamPresenceMask, 662
SFTlv, 662
unpack_ims_SLQSSetIMSSMSConfig
ims.h, 1140
unpack_ims_SLQSSetIMSSMSConfig_t, 657
pSettingResp, 658
ParamPresenceMask, 658
unpack_ims_SLQSSetIMSUserConfig
ims.h, 1141
unpack_ims_SLQSSetIMSUserConfig_t, 658
pSettingResp, 658
ParamPresenceMask, 658
unpack_ims_SLQSSetIMSVoIPConfig
ims.h, 1141
unpack_ims_SLQSSetIMSVoIPConfig_t, 658
pSettingResp, 659
ParamPresenceMask, 659
unpack_ims_SLQSSetRegMgrConfig
ims.h, 1141
unpack_ims_SLQSSetRegMgrConfig_t, 659
pSettingResp, 659
ParamPresenceMask, 659
unpack_ims_SLQSSetSIPConfig
ims.h, 1142
unpack_ims_SLQSSetSIPConfig_t, 659
pSettingResp, 660
ParamPresenceMask, 660
unpack_ims_SLQSUserCfgCallBack_ind
ims.h, 1143

unpack_ims_SLQSUserCfgCallBack_ind_t, 662
IDTlv, 663
ParamPresenceMask, 663
unpack_ims_SLQSVoIPCfgCallBack_ind
ims.h, 1143
unpack_ims_SLQSVoIPCfgCallBack_ind_t, 663
AMTlv, 664
ParamPresenceMask, 665
RTTlv, 665
SDTlv, 665
unpack_imsa_SLQSGetIMSARegStatus
imsa.h, 1146
unpack_imsa_SLQSGetIMSARegStatus_t, 665
ImsRegErrCode, 666
ImsRegStatus, 666
NewImsRegStatus, 666
ParamPresenceMask, 666
unpack_imsa_SLQSGetIMSAServiceStatus
imsa.h, 1147
unpack_imsa_SLQSGetIMSAServiceStatus_t, 666
ParamPresenceMask, 668
SmsServiceRat, 668
SmsServiceStatus, 668
UtServiceRat, 668
UtServiceStatus, 668
VoipServiceRat, 668
VoipServiceStatus, 668
VsServiceRat, 668
VsServiceStatus, 668
VtServiceRat, 668
VtServiceStatus, 668
unpack_imsa_SLQSImsaPdpStatusCallBack_ind
imsa.h, 1147
unpack_imsa_SLQSImsaPdpStatusCallBack_ind_t, 668
FailErrCode, 669
ParamPresenceMask, 669
PdpConnState, 669
unpack_imsa_SLQSImsaRatStatusCallBack_ind
imsa.h, 1147
unpack_imsa_SLQSImsaRatStatusCallBack_ind_t, 669
ParamPresenceMask, 670
RatHandover, 670
unpack_imsa_SLQSImsaRegStatusCallBack_ind
imsa.h, 1148
unpack_imsa_SLQSImsaRegStatusCallBack_ind_t, 670
IMSRegistration, 670
ParamPresenceMask, 670
unpack_imsa_SLQSImsaSvcStatusCallBack_ind
imsa.h, 1148
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 671
ParamPresenceMask, 672
SmsRat, 672
SmsService, 672
UtRat, 672
UtService, 672
VoipRat, 672
VoipService, 672
VtRat, 672
VtService, 672
unpack_imsa_SLQSRegisterIMSAIndication
imsa.h, 1149
unpack_imsa_SLQSRegisterIMSAIndication_t
imsa.h, 1145
unpack_loc_BestAvailPos_Ind
loc.h, 1171
unpack_loc_BestAvailPos_Ind_t, 672
pAltitudeWrtEllipsoid, 678
pAltitudeWrtMeanSeaLevel, 678
pGpsTime, 678
pHeading, 678
pHeadingUnc, 678
pHorCirConf, 678
pHorEllpConf, 678
pHorReliability, 678
pHorUncCircular, 678
pHorUncEllipseOrientAzimuth, 678
pHorUncEllipseSemiMajor, 678
pHorUncEllipseSemiMinor, 678
pLatitude, 678
pLongitude, 678
pMagneticDeviation, 678
pPrecisionDilution, 678
pSensorDataUsage, 678
pSpeedHorizontal, 678
pSpeedUnc, 678
pSpeedVertical, 678
pSpeedVerticalUnc, 678
pSvUsedforFix, 678
pTechnologyMask, 678
pTimeSrc, 678
pTimeUnc, 678
pTimestampUtc, 678
pVertConfidence, 678
pVertReliability, 679
pVertUnc, 679
pXid, 679
ParamPresenceMask, 678
status, 679
Tlvresult, 679
unpack_loc_CradleMountCallback_Ind
loc.h, 1172
unpack_loc_CradleMountCallback_Ind_t, 679
cradleMountConfigStatus, 679
ParamPresenceMask, 679
unpack_loc_Delete_Assist_Data_t, 679
ParamPresenceMask, 680
Tlvresult, 680
unpack_loc_DeleteAssistData
loc.h, 1172
unpack_loc_DeleteAssistData_Ind
loc.h, 1173
unpack_loc_DeleteAssistData_Ind_t, 680
ParamPresenceMask, 681
status, 681
Tlvresult, 681

unpack_loc_DeleteAssistData_t
loc.h, 1166
unpack_loc_EngineState_Ind
loc.h, 1173
unpack_loc_EngineState_Ind_t, 681
engineState, 682
ParamPresenceMask, 682
Tlvresult, 682
unpack_loc_EventNMEA_Ind
loc.h, 1173
unpack_loc_EventNMEA_Ind_t, 682
NMEAData, 682
ParamPresenceMask, 682
Tlvresult, 682
unpack_loc_EventRegister
loc.h, 1174
unpack_loc_EventRegister_t, 683
ParamPresenceMask, 683
Tlvresult, 683
unpack_loc_EventTimeSyncCallback_Ind
loc.h, 1174
unpack_loc_EventTimeSyncCallback_Ind_t, 683
ParamPresenceMask, 683
timeSyncRefCounter, 683
unpack_loc_GetServer_Ind
loc.h, 1174
unpack_loc_GetServer_Ind_t, 683
pIPv4AddrInfo, 685
pIPv6AddrInfo, 685
pURL, 685
ParamPresenceMask, 685
serverStatus, 685
serverType, 685
Tlvresult, 685
unpack_loc_GnssSvInfo_Ind
loc.h, 1175
unpack_loc_GnssSvInfo_Ind_t, 685
altitudeAssumed, 686
pSatelliteInfo, 686
ParamPresenceMask, 686
Tlvresult, 686
unpack_loc_InjectPositionCallback_Ind
loc.h, 1175
unpack_loc_InjectPositionCallback_Ind_t, 686
ParamPresenceMask, 686
status, 686
unpack_loc_InjectSensorDataCallback_Ind
loc.h, 1175
unpack_loc_InjectSensorDataCallback_Ind_t, 686
injectSensorDataStatus, 688
pAccelSamplesAccepted, 688
pAccelTempSamplesAccepted, 688
pGyroSamplesAccepted, 688
pGyroTempSamplesAccepted, 688
pOpaqueIdentifier, 688
ParamPresenceMask, 688
unpack_loc_InjectTimeSyncDataCallback_Ind
loc.h, 1176
unpack_loc_InjectTimeSyncDataCallback_Ind_t, 688
injectTimeSyncStatus, 689
ParamPresenceMask, 689
unpack_loc_InjectUTCTimeCallback_Ind
loc.h, 1176
unpack_loc_InjectUTCTimeCallback_Ind_t, 689
ParamPresenceMask, 689
status, 689
unpack_loc_PositionRpt_Ind
loc.h, 1176
unpack_loc_PositionRpt_Ind_t, 689
pAltitudeAssumed, 695
pAltitudeWrtEllipsoid, 695
pAltitudeWrtMeanSeaLevel, 695
pFixId, 695
pGpsTime, 695
pHeading, 695
pHeadingUnc, 695
pHorConfidence, 695
pHorReliability, 695
pHorUncCircular, 695
pHorUncEllipseOrientAzimuth, 695
pHorUncEllipseSemiMajor, 695
pHorUncEllipseSemiMinor, 695
pLatitude, 695
pLeapSeconds, 695
pLongitude, 695
pMagneticDeviation, 695
pPrecisionDilution, 696
pSensorDataUsage, 696
pSpeedHorizontal, 696
pSpeedUnc, 696
pSpeedVertical, 696
pSvUsedforFix, 696
pTechnologyMask, 696
pTimeSrc, 696
pTimeUnc, 696
pTimestampUtc, 696
pVertConfidence, 696
pVertReliability, 696
pVertUnc, 696
ParamPresenceMask, 695
sessionId, 696
sessionStatus, 696
Tlvresult, 696
unpack_loc_SLQSLOCGetBestAvailPos
loc.h, 1179
unpack_loc_SLQSLOCGetBestAvailPos_t, 701
ParamPresenceMask, 701
Tlvresult, 701
unpack_loc_SLQSLOCGetServer
loc.h, 1179
unpack_loc_SLQSLOCGetServer_t
loc.h, 1166
unpack_loc_SLQSLOCInjectPosition
loc.h, 1179
unpack_loc_SLQSLOCInjectPosition_t
loc.h, 1166

unpack_loc_SLQSLOCInjectSensorData
loc.h, 1180
unpack_loc_SLQSLOCInjectSensorData_t
loc.h, 1166
unpack_loc_SLQSLOCInjectUTCTime
loc.h, 1180
unpack_loc_SLQSLOCInjectUTCTime_t
loc.h, 1166
unpack_loc_SLQSLOCSetCradleMountConfig
loc.h, 1180
unpack_loc_SLQSLOCSetCradleMountConfig_t
loc.h, 1166
unpack_loc_SLQSLOCSetServer
loc.h, 1181
unpack_loc_SensorStreamingCallback_Ind
loc.h, 1177
unpack_loc_SensorStreamingCallback_Ind_t, 696
pAccelAcceptReady, 697
pAccelTempAcceptReady, 697
pGyroAcceptReady, 697
pGyroTempAcceptReady, 697
ParamPresenceMask, 697
unpack_loc_SetExtPowerConfig_Ind
loc.h, 1177
unpack_loc_SetExtPowerConfig_Ind_t, 697
ParamPresenceMask, 698
status, 698
Tlvresult, 698
unpack_loc_SetExtPowerState
loc.h, 1177
unpack_loc_SetExtPowerState_t, 698
ParamPresenceMask, 698
Tlvresult, 698
unpack_loc_SetOperationMode
loc.h, 1178
unpack_loc_SetOperationMode_Ind
loc.h, 1178
unpack_loc_SetOperationMode_Ind_t, 699
ParamPresenceMask, 699
status, 699
Tlvresult, 699
unpack_loc_SetOperationMode_t, 699
ParamPresenceMask, 700
Tlvresult, 700
unpack_loc_SetServer_Ind
loc.h, 1178
unpack_loc_SetServer_Ind_t, 700
ParamPresenceMask, 701
serverStatus, 701
Tlvresult, 701
unpack_loc_Start
loc.h, 1181
unpack_loc_Start_t, 701
ParamPresenceMask, 701
Tlvresult, 702
unpack_loc_Stop
loc.h, 1181
unpack_loc_Stop_t, 702
ParamPresenceMask, 702
Tlvresult, 702
unpack_nas_GetACCOLC
nas.h, 1217
unpack_nas_GetACCOLC_t, 702
pAccolc, 702
ParamPresenceMask, 703
unpack_nas_GetANAAAAuthenticationStatus
nas.h, 1217
unpack_nas_GetANAAAAuthenticationStatus_t, 703
pAuthStatus, 703
ParamPresenceMask, 703
unpack_nas_GetCDMANetworkParameters
nas.h, 1218
unpack_nas_GetCDMANetworkParameters_t, 703
Application, 705
Broadcast, 705
CustomSCP, 705
ForceRev0, 705
ParamPresenceMask, 705
Protocol, 705
RegForeignNID, 705
RegForeignSID, 705
RegHomeSID, 706
Roaming, 706
SCI, 706
SCM, 706
unpack_nas_GetHomeNetwork
nas.h, 1218
unpack_nas_GetHomeNetwork3GPP2
nas.h, 1219
unpack_nas_GetHomeNetwork3GPP2_t, 706
nameSize, 708
pMCC, 708
pMNC, 708
pNID, 708
pName, 708
pNw2DescDisp, 708
pNw2DescEnc, 708
pNw2DescLen, 708
pNw2MCC, 708
pNw2MNC, 708
pNw2Name, 708
pSID, 708
ParamPresenceMask, 708
unpack_nas_GetHomeNetwork_t, 708
mcc, 709
mnc, 709
name, 709
nid, 709
ParamPresenceMask, 709
sid, 709
unpack_nas_GetNetworkPreference
nas.h, 1219
unpack_nas_GetNetworkPreference_t, 709
ActiveTechPref, 710
Duration, 710
ParamPresenceMask, 710

PersistentTechPref, 710
TlvresultI, 710
unpack_nas_GetRFInfo
nas.h, 1219
unpack_nas_GetRFInfo_t, 710
instancesSize, 711
ParamPresenceMask, 711
RFBandInfoElements, 711
unpack_nas_GetServingNetwork
nas.h, 1220
unpack_nas_GetServingNetwork_t, 711
CSDomain, 713
DataCaps, 713
DataCapsLen, 713
MCC, 713
MNC, 713
Name, 713
nameSize, 714
PSDomain, 714
ParamPresenceMask, 714
RAN, 714
RadioIfaces, 714
RadioIfacesSize, 714
RegistrationState, 714
Roaming, 714
unpack_nas_GetServingNetworkCapabilities
nas.h, 1220
unpack_nas_GetServingNetworkCapabilities_t, 714
DataCaps, 715
DataCapsLen, 715
ParamPresenceMask, 715
unpack_nas_GetSignalStrengths
nas.h, 1220
unpack_nas_GetSignalStrengths_t, 715
len, 715
ParamPresenceMask, 715
radio, 715
rssi, 715
unpack_nas_InitiateDomainAttach
nas.h, 1221
unpack_nas_InitiateDomainAttach_t
nas.h, 1194
unpack_nas_PerformNetworkScan
nas.h, 1221
unpack_nas_PerformNetworkScan_t, 716
p3GppNetworkInfoInstances, 717
p3GppNetworkInstanceSize, 717
pLteOpModeTlv, 717
pPCIInfo, 717
pPCSInstance, 717
pPCSInstanceSize, 717
pRATINstance, 717
pRATInstanceSize, 717
pScanResult, 717
ParamPresenceMask, 717
unpack_nas_SLQSConfigSigInfo
nas.h, 1225
unpack_nas_SLQSConfigSigInfo_t
nas.h, 1195
unpack_nas_SLQSGetErrorRate
nas.h, 1225
unpack_nas_SLQSGetErrorRate_t, 722
pCDMAFrameErrRate, 723
pGSMBER, 723
pHDRPackErrRate, 723
pWCDMABER, 723
ParamPresenceMask, 723
unpack_nas_SLQSGetNetworkTime
nas.h, 1226
unpack_nas_SLQSGetNetworkTime_t, 724
p3GPP2TimeInfo, 724
p3GPPTimeInfo, 724
ParamPresenceMask, 724
unpack_nas_SLQSGetOperatorNameData
nas.h, 1226
unpack_nas_SLQSGetOperatorNameData_t, 724
pNITZInformation, 725
pOperatorNameString, 725
pOperatorPLMNList, 725
pPLMNNetworkName, 725
pSrvcProviderName, 725
ParamPresenceMask, 725
unpack_nas_SLQSGetPLMNName
nas.h, 1227
unpack_nas_SLQSGetPLMNName_t, 726
longName, 728
longNameCI, 728
longNameEn, 728
longNameLen, 728
longNameSB, 728
ParamPresenceMask, 728
shortName, 728
shortNameCI, 728
shortNameEn, 728
shortNameLen, 728
shortNameSB, 728
spn, 728
spnEncoding, 728
spnLength, 729
unpack_nas_SLQSGetServingSystem
nas.h, 1227
unpack_nas_SLQSGetServingSystem_t, 729
BasestationID, 732
BasestationLatitude, 732
BasestationLongitude, 732
CDMASystemInfoExt, 732
CallBarStatus, 732
CellID, 732
ConcSvcInfo, 733
CurrentPLMN, 733
DTMInd, 733
DataSrvCapabilities, 733
DefaultRoamInd, 733
DetailedSvcInfo, 733
Gpp2TimeZone, 733
GppNetworkDSTAdjustment, 733

GppTimeZone, 733
HdrPersonality, 733
Lac, 733
NetworkID, 733
PRLInd, 733
ParamPresenceMask, 733
RoamIndicatorVal, 733
RoamingIndicatorList, 733
ServingSystem, 733
SystemID, 733
TrackAreaCode, 733
unpack_nas_SLQSGetSignalStrength
nas.h, 1227
unpack_nas_SLQSGetSignalStrength_t, 733
ecioList, 735
ecioListLen, 735
errorRateList, 735
errorRateListLen, 735
Io, 735
ltersrp, 736
ltesnr, 736
ParamPresenceMask, 736
rsrqInfo, 736
rxSignalStrengthList, 736
rxSignalStrengthListLen, 736
signalStrengthReqMask, 736
sinr, 736
unpack_nas_SLQSGetSysInfo
nas.h, 1228
unpack_nas_SLQSGetSysInfo_t, 736
pAddCDMASysInfo, 739
pAddGSMSysInfo, 739
pAddHDRSysInfo, 739
pAddLTESysInfo, 739
pCDMASrvStatusInfo, 739
pCDMASysInfo, 739
pCampedCiotLteOpMode, 739
pGSMCallBarringSysInfo, 739
pGSMCipherDomainSysInfo, 739
pGSMSrvStatusInfo, 739
pGSMSysInfo, 739
pHDRSrvStatusInfo, 739
pHDRSysInfo, 739
pLTESrvStatusInfo, 739
pLTESysInfo, 739
pLTEVoiceSupportSysInfo, 739
pWCDMASysInfo, 740
ParamPresenceMask, 739
unpack_nas_SLQSGetSysSelectionPref
nas.h, 1228
unpack_nas_SLQSGetSysSelectionPref_t, 740
pBandPref, 744
pEmerMode, 744
pGWAcqOrderPref, 744
pLTEBandPref, 744
pModePref, 744
pNetSelPref, 744
pPRLPref, 744
pRoamPref, 744
pSrvDomainPref, 744
ParamPresenceMask, 744
unpack_nas_SLQSGetSysSelectionPrefExt
nas.h, 1229
unpack_nas_SLQSGetSysSelectionPrefExt_t, 744
pAcqOrderPref, 746
pBandPref, 746
pCiotAcqOrderPref, 746
pCiotLteOpModePref, 746
pEmerMode, 747
pGWAcqOrderPref, 747
pLTEBandPref, 747
pLteM1BandPref, 747
pLteNb1BandPref, 747
pModePref, 747
pNetSelPref, 747
pNr5gBandPref, 747
pPRLPref, 747
pRatDisabledMask, 747
pRoamPref, 747
pSrvDomainPref, 747
unpack_nas_SLQSInitiateNetworkRegistration
nas.h, 1229
unpack_nas_SLQSInitiateNetworkRegistration_t
nas.h, 1195
unpack_nas_SLQSNASGetForbiddenNetworks
nas.h, 1232
unpack_nas_SLQSNASGetForbiddenNetworks_t, 753
ParamPresenceMask, 754
unpack_nas_SLQSNASGeteDRXParams
nas.h, 1231
unpack_nas_SLQSNASGeteDRXParams_t, 751
pCycleLen, 752
pEdrxEnable, 752
ParamPresenceMask, 752
unpack_nas_SLQSNASGeteDRXParamsExt
nas.h, 1231
unpack_nas_SLQSNASGeteDRXParamsExt_t, 752
unpack_nas_SLQSNASSeteDRXParams
nas.h, 1235
unpack_nas_SLQSNASSeteDRXParams_t
nas.h, 1195
unpack_nas_SLQSNASSwiGetChannelLock
nas.h, 1235
unpack_nas_SLQSNASSwiGetChannelLock_t, 763
ParamPresenceMask, 763
unpack_nas_SLQSNASSwiSetChannelLock
nas.h, 1237
unpack_nas_SLQSNASSwiSetChannelLock_t
nas.h, 1195
unpack_nas_SLQSNasConfigSigInfo2
nas.h, 1229
unpack_nas_SLQSNasConfigSigInfo2_t
nas.h, 1195
unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind
nas.h, 1230

unpack_nas_SLQSNasEdrxChangeInfoCallBack_Ind_t, 747
unpack_nas_SLQSNasGet3GPP2Subscription
nas.h, 1230
unpack_nas_SLQSNasGet3GPP2Subscription_t, 748
pDirNum, 749
ParamPresenceMask, 749
unpack_nas_SLQSNasGetCellLocationInfo
nas.h, 1230
unpack_nas_SLQSNasGetCellLocationInfo_t, 749
pCDMAInfo, 751
pLTEInfoInterfreq, 751
pLTEInfoIntrafreq, 751
pUMTSInfo, 751
ParamPresenceMask, 750
unpack_nas_SLQSNasGetHDRColorCode
nas.h, 1232
unpack_nas_SLQSNasGetHDRColorCode_t, 754
pColorCode, 754
ParamPresenceMask, 754
unpack_nas_SLQSNasGetRFInfo
nas.h, 1232
unpack_nas_SLQSNasGetRFInfo_t, 754
pLTEOperationMode, 755
pRfBandInfoExtFormat, 755
pRfBandwidthInfo, 755
pRfDedicatedBandInfo, 755
rfbandInfoList, 755
Tlvresult, 755
unpack_nas_SLQSNasGetSigInfo
nas.h, 1233
unpack_nas_SLQSNasGetSigInfo_t, 755
CDMASSInfo, 756
GSMSSInfo, 756
HDRSSInfo, 756
LTESSInfo, 756
ParamPresenceMask, 756
unpack_nas_SLQSNasGetTxRxInfo
nas.h, 1233
unpack_nas_SLQSNasGetTxRxInfo_t, 756
pRXChain0Info, 757
pRXChain1Info, 757
pTXInfo, 757
ParamPresenceMask, 757
unpack_nas_SLQSNasIndicationRegisterExt
nas.h, 1233
unpack_nas_SLQSNasIndicationRegisterExt_t
nas.h, 1195
unpack_nas_SLQSNasNetworkRejectCallback_Ind
nas.h, 1234
unpack_nas_SLQSNasNetworkRejectCallback_Ind_t, 757
pCsgId, 760
pLteOpMode, 760
pPlmnId, 760
ParamPresenceMask, 760
radioIf, 760
rejectCause, 760
serviceDomain, 760
unpack_nas_SLQSNasNetworkTimeCallBack_ind
nas.h, 1234
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 760
pDayltSavAdj, 761
pRadioInterface, 761
pTimeZone, 761
ParamPresenceMask, 761
universalTime, 761
unpack_nas_SLQSNasSigInfoCallback_ind
nas.h, 1235
unpack_nas_SLQSNasSigInfoCallback_ind_t, 761
pRscp, 762
ParamPresenceMask, 762
unpack_nas_SLQSNasSwiIndicationRegister
nas.h, 1236
unpack_nas_SLQSNasSwiIndicationRegister_t
nas.h, 1195
unpack_nas_SLQSNasSwiModemStatus
nas.h, 1236
unpack_nas_SLQSNasSwiModemStatus_t, 763
commonInfo, 764
pLTEInfo, 764
ParamPresenceMask, 764
unpack_nas_SLQSNasSwiOTAMessageCallback_ind
nas.h, 1236
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 764
Info, 765
Tlvresult, 765
unpack_nas_SLQSNasSysInfoCallback_ind
nas.h, 1237
unpack_nas_SLQSNasTimerCallback_ind
nas.h, 1237
unpack_nas_SLQSNasTimerCallback_ind_t, 765
ParamPresenceMask, 765
t3396_apn, 765
t3396_plmn_id, 765
t3396_val, 765
unpack_nas_SLQSSetBandPreference
nas.h, 1238
unpack_nas_SLQSSetBandPreference_t
nas.h, 1195
unpack_nas_SLQSSetSignalStrengthsCallback
nas.h, 1238
unpack_nas_SLQSSetSignalStrengthsCallback_t
nas.h, 1195
unpack_nas_SLQSSetSysSelectionPref
nas.h, 1238
unpack_nas_SLQSSetSysSelectionPref_t
nas.h, 1195
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind
nas.h, 1239
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 765
Info, 766
Tlvresult, 766

unpack_nas_SLQSSetSysSelectionPrefExt
nas.h, 1239
unpack_nas_SLQSSetSysSelectionPrefExt_t
nas.h, 1195
unpack_nas_SLQSSwiGetHDRPersonality
nas.h, 1239
unpack_nas_SLQSSwiGetHDRPersonality_t, 766
pCurrentPersonality, 767
pPersonalityListLength, 767
pProtocolSubtypeElement, 767
ParamPresenceMask, 767
unpack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 1240
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 767
pAppSubType, 768
pCurrentPrsnlty, 768
pPersonalityListLength, 768
ParamPresenceMask, 768
unpack_nas_SLQSSwiGetHRPDStats
nas.h, 1240
unpack_nas_SLQSSwiGetHRPDStats_t, 768
pPilotSetData, 769
pUATI, 769
ParamPresenceMask, 769
unpack_nas_SLQSSwiGetLteCQI
nas.h, 1240
unpack_nas_SLQSSwiGetLteCQI_t, 769
ParamPresenceMask, 770
ValidityCW0, 770
ValidityCW1, 770
unpack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 1241
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 770
pSccRxInfo, 770
ParamPresenceMask, 770
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind
nas.h, 1241
unpack_nas_SLQSSwiHDRPersonalityCallback_Ind_t, 770
unpack_nas_SLQSSwiNetworkDebug
nas.h, 1241
unpack_nas_SLQSSwiNetworkDebug_t, 771
pDataStatusDetail, 772
pDeviceConfigDetail, 772
pNetworkStat1x, 772
pNetworkStatEVDO, 772
pObjectVer, 772
ParamPresenceMask, 772
unpack_nas_SLQSSwiPSDetach
nas.h, 1242
unpack_nas_SLQSSwiPSDetach_t
nas.h, 1195
unpack_nas_SLQSSwiRandIndicatorCallback_Ind
nas.h, 1242
unpack_nas_SLQSSwiRandIndicatorCallback_Ind_t, 772
ParamPresenceMask, 772
rankIndicatorTlv, 772
unpack_nas_SLQSSysInfoCallback_ind_t, 772
pGSMSysInfo, 776
pHDRSysInfo, 776
pLTESysInfo, 776
pLteCiotOpModeTlv, 776
pNR5GCellStatus, 776
pSysInfoNoChange, 776
ParamPresenceMask, 776
unpack_nas_SetACCOLC
nas.h, 1221
unpack_nas_SetACCOLC_t
nas.h, 1195
unpack_nas_SetCDMANetworkParameters
nas.h, 1222
unpack_nas_SetCDMANetworkParameters_t
nas.h, 1195
unpack_nas_SetDataCapabilitiesCallback_ind
nas.h, 1222
unpack_nas_SetDataCapabilitiesCallback_ind_t, 717
dataCaps, 718
dataCapsSize, 718
ParamPresenceMask, 718
unpack_nas_SetEventReportInd
nas.h, 1222
unpack_nas_SetEventReportInd_t, 718
ParamPresenceMask, 719
RFTlv, 719
RRTlv, 719
SLQSSSTlv, 719
SSTlv, 719
unpack_nas_SetLURejectCallback
nas.h, 1223
unpack_nas_SetLURejectCallback_t
nas.h, 1195
unpack_nas_SetNasLTECphyCaIndCallback_ind
nas.h, 1223
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 719
ParamPresenceMask, 720
unpack_nas_SetNetworkPreference
nas.h, 1224
unpack_nas_SetNetworkPreference_t, 720
ParamPresenceMask, 720
Tlvresult, 721
unpack_nas_SetRFInfoCallback
nas.h, 1224
unpack_nas_SetRFInfoCallback_t
nas.h, 1195
unpack_nas_SetRoamingIndicatorCallback_ind
nas.h, 1224
unpack_nas_SetRoamingIndicatorCallback_ind_t, 721
ParamPresenceMask, 721
roaming, 721
unpack_nas_SetServingSystemCallback_ind
nas.h, 1225
unpack_nas_SetServingSystemCallback_ind_t, 721
ParamPresenceMask, 722
SSInfo, 722
Tlvresult, 722

unpack_nas_SlqsGetLTECphyCAInfo
nas.h, 1226
unpack_nas_SlqsGetLTECphyCAInfo_t, 723
LTECphyCaInfo, 724
ParamPresenceMask, 724
Tlvresult, 724
unpack_omaDmConfigTlv_t, 776
alertmsg, 777
alertmsglength, 777
state, 777
userInputReq, 777
userInputTimeout, 777
unpack_omaDmFotaTlv_t, 777
description, 779
descriptionlength, 779
fwdloadsize, 779
fwloadComplete, 779
namelength, 779
package_name, 779
sessionType, 779
severity, 779
state, 779
updateCompleteStatus, 779
userInputReq, 779
userInputTimeout, 779
version, 779
versionlength, 779
unpack_omaDmNotificationsTlv_t, 779
notification, 780
sessionStatus, 780
unpack_pds_ForceXTRADownload
pds.h, 1257
unpack_pds_ForceXTRADownload_t
pds.h, 1247
unpack_pds_GetPDSDefaults
pds.h, 1257
unpack_pds_GetPDSDefaults_t, 780
pAccuracy, 781
pInterval, 781
pOperation, 781
pTimeout, 781
ParamPresenceMask, 781
unpack_pds_GetPDSState
pds.h, 1258
unpack_pds_GetPDSState_t, 781
pEnabledStatus, 782
pTrackingStatus, 782
ParamPresenceMask, 782
unpack_pds_GetPortAutomaticTracking
pds.h, 1258
unpack_pds_GetPortAutomaticTracking_t, 782
ParamPresenceMask, 782
pbAuto, 783
unpack_pds_GetServiceAutomaticTracking
pds.h, 1258
unpack_pds_GetServiceAutomaticTracking_t, 783
ParamPresenceMask, 783
pbAuto, 783
unpack_pds_GetXTRAAutomaticDownload
pds.h, 1259
unpack_pds_GetXTRAAutomaticDownload_t, 783
pInterval, 784
ParamPresenceMask, 784
pbEnabled, 784
unpack_pds_GetXTRANetwork
pds.h, 1259
unpack_pds_GetXTRANetwork_t, 784
pPreference, 784
ParamPresenceMask, 784
unpack_pds_GetXTRAValidity
pds.h, 1260
unpack_pds_GetXTRAValidity_t, 784
pDuration, 785
pGPSWeek, 785
pGPSWeekOffset, 785
ParamPresenceMask, 785
unpack_pds_PDSInjectTimeReference
pds.h, 1260
unpack_pds_PDSInjectTimeReference_t
pds.h, 1247
unpack_pds_ResetPDSData
pds.h, 1260
unpack_pds_ResetPDSData_t
pds.h, 1247
unpack_pds_SLQSGetAGPSConfig
pds.h, 1264
unpack_pds_SLQSGetAGPSConfig_t, 788
pServerAddress, 789
pServerPort, 789
pServerURL, 789
ParamPresenceMask, 789
unpack_pds_SLQSGetGPSStateInfo
pds.h, 1264
unpack_pds_SLQSGetGPSStateInfo_t, 789
Altitude, 793
EngineState, 793
HorizontalUncertainty, 793
Iono_valid, 793
Latitude, 793
Longitude, 793
ParamPresenceMask, 793
Time_uncert_ms, 793
ValidMask, 793
VerticalUncertainty, 793
unpack_pds_SLQSPDSDeterminePosition
pds.h, 1264
unpack_pds_SLQSPDSDeterminePosition_t
pds.h, 1248
unpack_pds_SLQSPDSInjectAbsoluteTimeReference
pds.h, 1265
unpack_pds_SLQSPDSInjectAbsoluteTimeReference_t
pds.h, 1248
unpack_pds_SLQSPDSInjectPositionData
pds.h, 1265
unpack_pds_SLQSPDSInjectPositionData_t
pds.h, 1248

unpack_pds_SLQSSetAGPSConfig
pds.h, 1265
unpack_pds_SLQSSetAGPSConfig_t
pds.h, 1248
unpack_pds_SLQSSetPositionMethodState
pds.h, 1266
unpack_pds_SLQSSetPositionMethodState_t
pds.h, 1248
unpack_pds_SetEventReport_Ind
pds.h, 1261
unpack_pds_SetEventReport_Ind_t, 785
dLatitude, 787
dLongitude, 787
has_PositionDataNMEA, 787
has_SessionStatus, 787
has_dLatitude, 787
has_dLongitude, 787
has_posSrc, 787
ParamPresenceMask, 787
posSrc, 787
PositionDataNMEA, 787
SessionStatus, 787
unpack_pds_SetEventReportCallback
pds.h, 1261
unpack_pds_SetEventReportCallback_t
pds.h, 1247
unpack_pds_SetPDSDefaults
pds.h, 1261
unpack_pds_SetPDSDefaults_t
pds.h, 1247
unpack_pds_SetPDSState
pds.h, 1262
unpack_pds_SetPDSState_t
pds.h, 1247
unpack_pds_SetPdsState_Ind
pds.h, 1262
unpack_pds_SetPdsState_Ind_t, 787
EnabledStatus, 788
ParamPresenceMask, 788
TlvPresent, 788
TrackingStatus, 788
unpack_pds_SetPortAutomaticTracking
pds.h, 1262
unpack_pds_SetPortAutomaticTracking_t
pds.h, 1247
unpack_pds_SetServiceAutomaticTracking
pds.h, 1263
unpack_pds_SetServiceAutomaticTracking_t
pds.h, 1248
unpack_pds_SetXTRAAutomaticDownload
pds.h, 1263
unpack_pds_SetXTRAAutomaticDownload_t
pds.h, 1248
unpack_pds_SetXTRANetwork
pds.h, 1263
unpack_pds_SetXTRANetwork_t
pds.h, 1248
unpack_pds_StartPDSTrackingSessionExt
pds.h, 1266
unpack_pds_StartPDSTrackingSessionExt_t
pds.h, 1248
unpack_pds_StopPDSTrackingSession
pds.h, 1266
unpack_pds_StopPDSTrackingSession_t
pds.h, 1248
unpack_qmi_t, 794
msgid, 794
type, 794
xid, 794
unpack_qos_BindDataPort
qos.h, 1319
unpack_qos_BindDataPort_t, 794
ParamPresenceMask, 794
unpack_qos_IPv4Addr_t, 795
addr, 795
subnetMask, 795
unpack_qos_IPv6Addr_t, 795
addr, 796
prefixLen, 796
unpack_qos_IPv6TrafCls_t, 796
mask, 796
val, 796
unpack_qos_Port_t, 797
port, 797
range, 797
unpack_qos_QosFlowInfo_t, 797
BearerID, 798
is_RxQFlowGranted_Available, 798
is_TxQFlowGranted_Available, 798
NumRxFilters, 798
NumTxFilters, 799
QFlowState, 799
RxQFilter, 799
RxQFlowGranted, 799
TxQFilter, 799
TxQFlowGranted, 799
unpack_qos_QosFlowInfoState_t, 799
id, 799
isNewFlow, 799
state, 799
unpack_qos_SLQSQosGetNetworkStatus
qos.h, 1319
unpack_qos_SLQSQosGetNetworkStatus_t, 799
NWQoSStatus, 800
ParamPresenceMask, 800
unpack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 1320
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 800
ambr_dl, 801
ambr_dl_ext, 801
ambr_dl_ext2, 801
ambr_ul, 801
ambr_ul_ext, 801
ambr_ul_ext2, 801
apnId, 801

ParamPresenceMask, 801
unpack_qos_SLQSQosSwiReadDataStats
qos.h, 1320
unpack_qos_SLQSQosSwiReadDataStats_t, 802
apnId, 803
numQosFlow, 803
ParamPresenceMask, 803
qosFlow, 803
total_rx_bytes, 803
total_rx_pkt, 803
total_tx_bytes, 803
total_tx_pkt, 803
unpack_qos_SLQSSetQosEventCallback
qos.h, 1321
unpack_qos_SLQSSetQosEventCallback_ind
qos.h, 1321
unpack_qos_SLQSSetQosEventCallback_ind_t, 803
NumFlows, 804
ParamPresenceMask, 804
QosFlowInfo, 804
unpack_qos_SLQSSetQosEventCallback_t, 804
ParamPresenceMask, 804
unpack_qos_SLQSSetQosNWStatusCallback_ind
qos.h, 1322
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 804
status, 805
unpack_qos_SLQSSetQosPriEventCallback_ind
qos.h, 1322
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 805
event, 805
ParamPresenceMask, 805
unpack_qos_SLQSSetQosStatusCallback_ind
qos.h, 1323
unpack_qos_SLQSSetQosStatusCallback_ind_t, 805
event, 807
id, 807
ParamPresenceMask, 807
reason, 807
status, 807
unpack_qos_Tos_t, 815
mask, 815
val, 815
unpack_qos_dataRate_t, 795
dataRateMax, 795
guaranteedRate, 795
unpack_qos_pktErrRate_t, 796
exponent, 797
multiplier, 797
unpack_qos_swiQosFilter_t, 807
EspSpi, 809
IPv4DstAddr, 809
IPv4SrcAddr, 809
IPv4Tos, 809
IPv6DstAddr, 809
IPv6Label, 809
IPv6SrcAddr, 809
IPv6TrafCls, 809
Id, 809
index, 809
is_EspSpi_Available, 809
is_IPv4DstAddr_Available, 809
is_IPv4SrcAddr_Available, 809
is_IPv4Tos_Available, 809
is_IPv6DstAddr_Available, 809
is_IPv6Label_Available, 809
is_IPv6SrcAddr_Available, 809
is_IPv6TrafCls_Available, 809
is_Id_Available, 809
is_NxtHdrProto_Available, 810
is_Precedence_Available, 810
is_TCPDstPort_Available, 810
is_TCPSrcPort_Available, 810
is_TranDstPort_Available, 810
is_TranSrcPort_Available, 810
is_UDPDstPort_Available, 810
is_UDPSrcPort_Available, 810
NxtHdrProto, 810
Precedence, 810
TCPDstPort, 810
TCPSrcPort, 810
TranDstPort, 810
TranSrcPort, 810
UDPDstPort, 810
UDPSrcPort, 810
version, 810
unpack_qos_swiQosFlow_t, 810
DataRate, 813
index, 813
is_DataRate_Available, 813
is_Jitter_Available, 813
is_Latency_Available, 813
is_LteQci_Available, 813
is_MaxAllowedPktSz_Available, 813
is_MinPolicedPktSz_Available, 813
is_PktErrRate_Available, 813
is_ProfileId3GPP2_Available, 813
is_TokenBucket_Available, 813
is_TrafficClass_Available, 813
is_val_3GPP2Pri_Available, 813
is_val_3GPPImCn_Available, 813
is_val_3GPPSigInd_Available, 814
Jitter, 814
Latency, 814
LteQci, 814
MaxAllowedPktSz, 814
MinPolicedPktSz, 814
PktErrRate, 814
ProfileId3GPP2, 814
TokenBucket, 814
TrafficClass, 814
val_3GPP2Pri, 814
val_3GPPImCn, 814
val_3GPPResResidualBER, 814
val_3GPPSigInd, 814
val_3GPPTraHdlPri, 814

unpack_qos_tokenBucket_t, 814
bucketSz, 815
peakRate, 815
tokenRate, 815
unpack_result_code_only
common.h, 1068
unpack_result_t, 816
ParamPresenceMask, 816
Tlvresult, 816
unpack_rms_GetSMSWake
rms.h, 1325
unpack_rms_GetSMSWake_t, 817
enabled, 817
ParamPresenceMask, 817
wake_mask, 817
unpack_rms_SetSMSWake
rms.h, 1325
unpack_rms_SetSMSWake_t, 817
ParamPresenceMask, 818
unpack_sar_SLQSGetRfSarState
sar.h, 1327
unpack_sar_SLQSGetRfSarState_t, 819
pSarRFState, 819
ParamPresenceMask, 819
unpack_sar_SLQSSetRfSarState
sar.h, 1327
unpack_sar_SLQSSetRfSarState_t
sar.h, 1326
unpack_sms_GetSMSCAddress
sms.h, 1340
unpack_sms_GetSMSCAddress_t, 819
addressSize, 820
pSMSCAddress, 820
pSMSCType, 820
ParamPresenceMask, 820
typeSize, 820
unpack_sms_SLQSDeleteSMS
sms.h, 1342
unpack_sms_SLQSDeleteSMS_t, 824
ParamPresenceMask, 824
unpack_sms_SLQSGetIndicationRegister
sms.h, 1342
unpack_sms_SLQSGetIndicationRegister_t, 824
pGetIndicationRegInfo, 825
ParamPresenceMask, 825
unpack_sms_SLQSGetMessageWaiting
sms.h, 1343
unpack_sms_SLQSGetMessageWaiting_t, 825
pGetMsgWaitingInfoResp, 825
ParamPresenceMask, 825
unpack_sms_SLQSGetSMS
sms.h, 1343
unpack_sms_SLQSGetSMS_t, 825
message, 826
messageFormat, 826
messageSize, 826
messageTag, 826
ParamPresenceMask, 826
unpack_sms_SLQSGetSMSList
sms.h, 1344
unpack_sms_SLQSGetSMSList_t, 827
messageList, 828
messageListSize, 828
ParamPresenceMask, 828
unpack_sms_SLQSGetSmsBroadcastConfig
sms.h, 1343
unpack_sms_SLQSGetSmsBroadcastConfig_t, 826
pBroadcastConfig, 827
pCDMABroadcastConfig, 827
ParamPresenceMask, 827
unpack_sms_SLQSGetTransLayerInfo
sms.h, 1344
unpack_sms_SLQSGetTransLayerInfo_t, 828
pGetTransLayerInfo, 828
ParamPresenceMask, 828
unpack_sms_SLQSGetTransNWRegInfo
sms.h, 1345
unpack_sms_SLQSGetTransNWRegInfo_t, 828
ParamPresenceMask, 829
unpack_sms_SLQSModifySMSStatus
sms.h, 1345
unpack_sms_SLQSModifySMSStatus_t, 829
ParamPresenceMask, 829
unpack_sms_SLQSNWRegInfoCallback_ind
sms.h, 1345
unpack_sms_SLQSNWRegInfoCallback_ind_t, 829
NWRegStat, 830
ParamPresenceMask, 830
unpack_sms_SLQSSendAsyncSMS
sms.h, 1346
unpack_sms_SLQSSendAsyncSMS_t, 830
ParamPresenceMask, 830
unpack_sms_SLQSSetIndicationRegister
sms.h, 1346
unpack_sms_SLQSSetIndicationRegister_t, 830
ParamPresenceMask, 831
unpack_sms_SLQSSetSmsBroadcastActivation
sms.h, 1346
unpack_sms_SLQSSetSmsBroadcastActivation_t, 831
ParamPresenceMask, 831
unpack_sms_SLQSSetSmsBroadcastConfig
sms.h, 1347
unpack_sms_SLQSSetSmsBroadcastConfig_t, 831
ParamPresenceMask, 832
unpack_sms_SLQSSetSmsStorage
sms.h, 1347
unpack_sms_SLQSSetSmsStorage_t, 832
ParamPresenceMask, 832
unpack_sms_SLQSSmsGetMaxStorageSize
sms.h, 1347
unpack_sms_SLQSSmsGetMaxStorageSize_t, 832
pMaxStorageSizeResp, 833
ParamPresenceMask, 832
unpack_sms_SLQSSmsGetMessageProtocol
sms.h, 1348
unpack_sms_SLQSSmsGetMessageProtocol_t, 833

pMessageProtocol, 833
ParamPresenceMask, 833
unpack_sms_SLQSSmsSetRoutes
sms.h, 1348
unpack_sms_SLQSSmsSetRoutes_t, 833
ParamPresenceMask, 833
unpack_sms_SLQSSwiGetSMSStorage
sms.h, 1348
unpack_sms_SLQSSwiGetSMSStorage_t, 834
pSmsStorage, 834
ParamPresenceMask, 834
unpack_sms_SLQSTransLayerInfoCallback_ind
sms.h, 1349
unpack_sms_SLQSTransLayerInfoCallback_ind_t, 834
pTransLayerInfo, 835
ParamPresenceMask, 835
regInd, 835
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind
sms.h, 1349
unpack_sms_SLQSWmsAsyncRawSendCallBack_ind_t, 835
causeCode, 837
errorClass, 837
messageID, 837
RPCause, 837
sendStatus, 837
TPCause, 837
userData, 837
unpack_sms_SLQSWmsMemoryFullCallBack_ind
sms.h, 1350
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 837
messageMode, 838
ParamPresenceMask, 838
storageType, 838
unpack_sms_SLQSWmsMessageWaitingCallBack_ind
sms.h, 1350
unpack_sms_SLQSWmsMessageWaitingCallBack_ind_t, 838
msgWaitInfo, 838
numInstances, 838
ParamPresenceMask, 838
unpack_sms_SaveSMS
sms.h, 1340
unpack_sms_SaveSMS_t, 820
pMessageIndex, 821
ParamPresenceMask, 821
unpack_sms_SendSMS
sms.h, 1341
unpack_sms_SendSMS_t, 821
messageFailureCode, 821
messageID, 821
ParamPresenceMask, 821
unpack_sms_SetNewSMSCallback
sms.h, 1341
unpack_sms_SetNewSMSCallback_ind
sms.h, 1341
unpack_sms_SetNewSMSCallback_ind_t, 822
ETWSTlv, 823
IMSTlv, 823
MMTlv, 823
NewMMTlv, 823
ParamPresenceMask, 823
SMSCTlv, 823
TRMessageTlv, 823
unpack_sms_SetNewSMSCallback_t, 823
ParamPresenceMask, 823
unpack_sms_SetSMSCAddress
sms.h, 1342
unpack_sms_SetSMSCAddress_t, 823
ParamPresenceMask, 824
unpack_swiaudio_SLQSGetM2MAVMute
swiaudio.h, 1357
unpack_swiaudio_SLQSGetM2MAVMute_t, 840
CwtMute, 841
EarMute, 841
MicMute, 841
ParamPresenceMask, 841
unpack_swiaudio_SLQSGetM2MAudioProfile
swiaudio.h, 1356
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 838
CwtMute, 839
EarMute, 839
Generator, 839
MicMute, 839
ParamPresenceMask, 840
Profile, 840
Volume, 840
unpack_swiaudio_SLQSGetM2MAudioVolume
swiaudio.h, 1357
unpack_swiaudio_SLQSGetM2MAudioVolume_t, 840
Level, 840
ParamPresenceMask, 840
unpack_swiaudio_SLQSGetM2MSpkrGain
swiaudio.h, 1357
unpack_swiaudio_SLQSGetM2MSpkrGain_t, 841
ParamPresenceMask, 842
Value, 842
unpack_swiaudio_SLQSSetM2MAVMute
swiaudio.h, 1359
unpack_swiaudio_SLQSSetM2MAVMute_t
swiaudio.h, 1352
unpack_swiaudio_SLQSSetM2MAudioAVCFG
swiaudio.h, 1358
unpack_swiaudio_SLQSSetM2MAudioAVCFG_t
swiaudio.h, 1352
unpack_swiaudio_SLQSSetM2MAudioLPBK
swiaudio.h, 1358
unpack_swiaudio_SLQSSetM2MAudioLPBK_t
swiaudio.h, 1352
unpack_swiaudio_SLQSSetM2MAudioNVDef
swiaudio.h, 1358
unpack_swiaudio_SLQSSetM2MAudioNVDef_t
swiaudio.h, 1352
unpack_swiaudio_SLQSSetM2MAudioProfile
swiaudio.h, 1359

unpack_swiaudio_SLQSSetM2MAudioProfile_t
swiaudio.h, 1352
unpack_swiaudio_SLQSSetM2MAudioVolume
swiaudio.h, 1359
unpack_swiaudio_SLQSSetM2MAudioVolume_t
swiaudio.h, 1352
unpack_swiaudio_SLQSSetM2MSpkrGain
swiaudio.h, 1360
unpack_swiaudio_SLQSSetM2MSpkrGain_t
swiaudio.h, 1352
unpack_swiavms_SLQSAVMSEventReportInd
swiavms.h, 1368
unpack_swiavms_SLQSAVMSEventReportInd_t, 842
pConfig, 843
pConnectionRequest, 843
pDataSessionStatus, 843
pNotification, 843
pPackageID, 843
pRegStatus, 843
pSessionType, 843
ParamPresenceMask, 843
resultcode, 843
unpack_swiavms_SLQSAVMSGetSettings
swiavms.h, 1369
unpack_swiavms_SLQSAVMSGetSettings_t, 843
AutoConnect, 845
AutoReboot, 845
FwAutoSDM, 845
FwPromptUpdate, 846
FwPromptdownload, 845
pAPNInfo, 846
pConnectionRetryTimers, 846
pNotificationStore, 846
pPeroidsInfo, 846
pPollingTimer, 846
ParamPresenceMask, 846
resultcode, 846
unpack_swiavms_SLQSAVMSGetSettings_v2
swiavms.h, 1369
unpack_swiavms_SLQSAVMSGetSettings_v2_t, 846
AutoConnect, 848
FwPromptUpdate, 848
FwPromptdownload, 848
pAutoReboot, 848
pNotificationStore, 848
pPeroidsInfo, 848
pPollingTimer, 848
ParamPresenceMask, 848
resultcode, 848
unpack_swiavms_SLQSAVMSSendSelection
swiavms.h, 1370
unpack_swiavms_SLQSAVMSSendSelection_t, 848
ParamPresenceMask, 849
resultcode, 849
unpack_swiavms_SLQSAVMSSessionGetInfo
swiavms.h, 1370
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 849
pConfig, 849
pNotification, 850
pPackageID, 850
ParamPresenceMask, 849
resultcode, 850
unpack_swiavms_SLQSAVMSSetSettings
swiavms.h, 1371
unpack_swiavms_SLQSAVMSSetSettings_t, 850
ParamPresenceMask, 850
resultcode, 850
unpack_swiavms_SLQSAVMSSetSettings_v2
swiavms.h, 1372
unpack_swiavms_SLQSAVMSSetSettings_v2_t, 851
ParamPresenceMask, 851
resultcode, 851
unpack_swiavms_SLQSAVMSStartSession
swiavms.h, 1372
unpack_swiavms_SLQSAVMSStartSession_t, 851
ParamPresenceMask, 851
resultcode, 851
sessionResponse, 852
unpack_swiavms_SLQSAVMSStopSession
swiavms.h, 1373
unpack_swiavms_SLQSAVMSStopSession_avc2
swiavms.h, 1373
unpack_swiavms_SLQSAVMSStopSession_t, 852
ParamPresenceMask, 852
resultcode, 852
unpack_swiavms_SLQSAvmsSetEventReport
swiavms.h, 1371
unpack_swiavms_SLQSAvmsSetEventReport_t, 850
ParamPresenceMask, 850
resultcode, 850
unpack_swidms_SLQSSwiDmsGetMTU
swidms.h, 1378
unpack_swidms_SLQSSwiDmsGetMTU_t, 852
pMTUSize3gpp, 853
ParamPresenceMask, 853
unpack_swidms_SLQSSwiDmsGetUsbComp
swidms.h, 1379
unpack_swidms_SLQSSwiDmsGetUsbComp_t, 853
pInterfaceCfg, 854
pSupportedBitmasks, 854
ParamPresenceMask, 853
unpack_swidms_SLQSSwiDmsGetUsbNetNum
swidms.h, 1379
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t, 854
ParamPresenceMask, 854
usbNetNum, 854
unpack_swidms_SLQSSwiDmsSetMTU
swidms.h, 1379
unpack_swidms_SLQSSwiDmsSetMTU_t, 854
ParamPresenceMask, 855
Tlvresult, 855
unpack_swidms_SLQSSwiDmsSetUsbComp
swidms.h, 1380
unpack_swidms_SLQSSwiDmsSetUsbComp_t, 855
ParamPresenceMask, 855
Tlvresult, 855

unpack_swidms_SLQSSwiDmsSetUsbNetNum
swidms.h, 1380
unpack_swidms_SLQSSwiDmsSetUsbNetNum_t
swidms.h, 1376
unpack_swiloc_SwiLocGetAutoStart
swiloc.h, 1382
unpack_swiloc_SwiLocGetAutoStart_t, 855
fix_rate, 857
fix_rate_reported, 857
fix_type, 857
fix_type_reported, 857
function, 857
function_reported, 857
max_dist, 857
max_dist_reported, 857
max_time, 857
max_time_reported, 857
ParamPresenceMask, 857
unpack_swiloc_SwiLocSetAutoStart
swiloc.h, 1382
unpack_swiloc_SwiLocSetAutoStart_t
swiloc.h, 1381
unpack_swioma_SLQSOMADMAlertCallback
swioma.h, 1388
unpack_swioma_SLQSOMADMAlertCallback_ind
swioma.h, 1389
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 857
eventType, 858
unpack_swioma_SLQSOMADMAlertCallback_t
swioma.h, 1384
unpack_swioma_SLQSOMADMCancelSession
swioma.h, 1389
unpack_swioma_SLQSOMADMCancelSession_t
swioma.h, 1384
unpack_swioma_SLQSOMADMCancelSessionExt
swiomaext.h, 1395
unpack_swioma_SLQSOMADMCancelSessionExt_t
swiomaext.h, 1393
unpack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 1390
unpack_swioma_SLQSOMADMGetSessionInfo_t, 858
Date, 860
DateLength, 860
ParamPresenceMask, 861
PkgDescLength, 861
PkgDescription, 861
PkgName, 861
PkgNameLength, 861
RetryCount, 861
SessionState, 861
SessionType, 861
Severity, 861
Source, 861
SourceLength, 861
Status, 861
Time, 861
TimeLength, 861
UpdateCompleteStatus, 861
unpack_swioma_SLQSOMADMGetSettings
swioma.h, 1390
unpack_swioma_SLQSOMADMGetSettings_t, 861
Autosdm, 862
FOTAdownload, 862
FwAutoCheck, 862
ParamPresenceMask, 863
unpack_swioma_SLQSOMADMSendSelection
swioma.h, 1391
unpack_swioma_SLQSOMADMSendSelection_t
swioma.h, 1384
unpack_swioma_SLQSOMADMSendSelectionExt
swiomaext.h, 1396
unpack_swioma_SLQSOMADMSendSelectionExt_t
swiomaext.h, 1393
unpack_swioma_SLQSOMADMSetSettings
swioma.h, 1391
unpack_swioma_SLQSOMADMSetSettings_t
swioma.h, 1384
unpack_swioma_SLQSOMADMSetSettingsExt
swiomaext.h, 1396
unpack_swioma_SLQSOMADMSetSettingsExt_t
swiomaext.h, 1393
unpack_swioma_SLQSOMADMStartSession
swioma.h, 1391
unpack_swioma_SLQSOMADMStartSession_t, 863
FwAvailability, 863
ParamPresenceMask, 863
unpack_swioma_SLQSOMADMStartSessionExt
swiomaext.h, 1397
unpack_swioma_SLQSOMADMStartSessionExt_t
swiomaext.h, 1393
unpack_tmd_SLQSTmdDeRegNotMitigationLvl
tmd.h, 1401
unpack_tmd_SLQSTmdDeRegNotMitigationLvl_t, 863
ParamPresenceMask, 864
Tlvresult, 864
unpack_tmd_SLQSTmdGetMitigationDevList
tmd.h, 1401
unpack_tmd_SLQSTmdGetMitigationDevList_t, 864
MitigationDevList, 864
MitigationDevListLen, 864
ParamPresenceMask, 865
Tlvresult, 865
unpack_tmd_SLQSTmdGetMitigationLvl
tmd.h, 1402
unpack_tmd_SLQSTmdGetMitigationLvl_t, 865
CurrentmitigationLvl, 865
ParamPresenceMask, 865
ReqMitigationLvl, 865
Tlvresult, 865
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind
tmd.h, 1402
unpack_tmd_SLQSTmdMitigationLvlRptCallback_ind_t, 865
deviceID, 866
deviceIdLen, 866

lvl, 866
ParamPresenceMask, 866
unpack_tmd_SLQSTmdRegNotMitigationLvl
tmd.h, 1403
unpack_tmd_SLQSTmdRegNotMitigationLvl_t, 866
ParamPresenceMask, 867
Tlvresult, 867
unpack_uim_ChangePin
uim.h, 1414
unpack_uim_ChangePin_t, 867
pEncryptedPIN1, 867
pIndicationToken, 867
pRemainingRetries, 867
ParamPresenceMask, 867
Tlvresult, 868
unpack_uim_GetCardStatus
uim.h, 1414
unpack_uim_GetCardStatus_t, 868
pCardStatus, 868
pHotSwapStatus, 868
ParamPresenceMask, 868
Tlvresult, 868
unpack_uim_ReadTransparent
uim.h, 1415
unpack_uim_ReadTransparent_t, 868
pCardResult, 869
pEncryptedData, 869
pIndicationToken, 869
pReadResult, 869
ParamPresenceMask, 869
Tlvresult, 869
unpack_uim_SLQSUIMAuthenticate
uim.h, 1416
unpack_uim_SLQSUIMAuthenticate_t, 871
pAuthenticateResult, 871
pCardResult, 872
pIndicationToken, 872
ParamPresenceMask, 871
unpack_uim_SLQSUIMDepersonalization
uim.h, 1416
unpack_uim_SLQSUIMDepersonalization_t, 872
pRemainingRetries, 872
ParamPresenceMask, 872
unpack_uim_SLQSUIMEventRegister
uim.h, 1417
unpack_uim_SLQSUIMEventRegister_t, 872
eventMask, 873
ParamPresenceMask, 873
unpack_uim_SLQSUIMGetConfiguration
uim.h, 1417
unpack_uim_SLQSUIMGetConfiguration_t, 873
pAutoSelection, 874
pHaltSubscription, 874
pPersonalizationStatus, 874
ParamPresenceMask, 873
unpack_uim_SLQSUIMGetFileAttributes
uim.h, 1417
unpack_uim_SLQSUIMGetFileAttributes_t, 874
pCardResult, 874
pFileAttributes, 874
pIndicationToken, 874
ParamPresenceMask, 874
unpack_uim_SLQSUIMGetSlotsStatus
uim.h, 1418
unpack_uim_SLQSUIMGetSlotsStatus_t, 874
pNumberOfPhySlot, 875
pUimSlotsStatus, 875
ParamPresenceMask, 875
unpack_uim_SLQSUIMPowerDown
uim.h, 1418
unpack_uim_SLQSUIMPowerDown_t
uim.h, 1406
unpack_uim_SLQSUIMPowerUp
uim.h, 1418
unpack_uim_SLQSUIMPowerUp_t
uim.h, 1406
unpack_uim_SLQSUIMRefreshCallback_Ind
uim.h, 1419
unpack_uim_SLQSUIMRefreshCallback_Ind_t, 875
ParamPresenceMask, 876
refreshEvent, 876
TlvPresent, 876
unpack_uim_SLQSUIMRefreshComplete
uim.h, 1419
unpack_uim_SLQSUIMRefreshComplete_t
uim.h, 1406
unpack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 1420
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 876
pRefreshEvent, 876
ParamPresenceMask, 876
unpack_uim_SLQSUIMRefreshOK
uim.h, 1420
unpack_uim_SLQSUIMRefreshOK_t
uim.h, 1407
unpack_uim_SLQSUIMRefreshRegister
uim.h, 1420
unpack_uim_SLQSUIMRefreshRegister_t
uim.h, 1407
unpack_uim_SLQSUIMReset
uim.h, 1421
unpack_uim_SLQSUIMReset_t
uim.h, 1407
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind
uim.h, 1421
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind-
_t, 876
unpack_uim_SLQSUIMSwitchSlot
uim.h, 1421
unpack_uim_SLQSUIMSwitchSlot_t
uim.h, 1407
unpack_uim_SetPinProtection
uim.h, 1415
unpack_uim_SetPinProtection_t, 869
pEncryptedPIN1, 870
pIndicationToken, 870

pRemainingRetries, 870
ParamPresenceMask, 870
Tlvresult, 870
unpack_uim_SetUimSlotStatusChangeCallback_ind
uim.h, 1416
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 870
bNumberOfPhySlots, 871
ParamPresenceMask, 871
slotsstatusChange, 871
unpack_uim_UnblockPin
uim.h, 1422
unpack_uim_UnblockPin_t, 877
pEncryptedPIN1, 877
pIndicationToken, 878
pRemainingRetries, 878
ParamPresenceMask, 877
Tlvresult, 878
unpack_uim_VerifyPin
uim.h, 1422
unpack_uim_VerifyPin_t, 878
pEncryptedPIN1, 878
pIndicationToken, 878
pRemainingRetries, 878
ParamPresenceMask, 878
Tlvresult, 878
unpack_valid_nas_GetCDMANetworkParameters
nas.h, 1242
unpack_valid_nas_SLQSGetServingSystem
nas.h, 1243
unpack_valid_nas_SLQSGetSignalStrength
nas.h, 1244
unpack_valid_nas_SLQSNasGetSigInfo
nas.h, 1244
unpack_voice_AnswerUSSD
voice.h, 1441
unpack_voice_AnswerUSSD_t
voice.h, 1428
unpack_voice_CancelUSSD
voice.h, 1441
unpack_voice_CancelUSSD_t
voice.h, 1428
unpack_voice_DTMFEventCallback_ind
voice.h, 1441
unpack_voice_DTMFEventCallback_ind_t, 881
DTMFInformation, 881
pOffLength, 881
pOnLength, 881
ParamPresenceMask, 881
unpack_voice_OTASPStatusCallback_ind
voice.h, 1442
unpack_voice_OTASPStatusCallback_ind_t, 882
callID, 883
OTASPStatus, 883
ParamPresenceMask, 883
unpack_voice_OriginateUSSD
voice.h, 1442
unpack_voice_OriginateUSSD_t
voice.h, 1428
unpack_voice_SLQSOriginateUSSD
voice.h, 1442
unpack_voice_SLQSOriginateUSSD_t, 883
pAlphaIDInfo, 885
pCCSuppsType, 885
pCallId, 885
pCcResultType, 885
pUSSDInfo, 885
ParamPresenceMask, 885
pfailureCause, 885
unpack_voice_SLQSVoiceALSSelectLine
voice.h, 1443
unpack_voice_SLQSVoiceALSSelectLine_t
voice.h, 1428
unpack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 1443
unpack_voice_SLQSVoiceALSSetLineSwitching_t
voice.h, 1428
unpack_voice_SLQSVoiceAnswerCall
voice.h, 1444
unpack_voice_SLQSVoiceAnswerCall_t, 885
pCallId, 886
ParamPresenceMask, 886
unpack_voice_SLQSVoiceBindSubscription
voice.h, 1444
unpack_voice_SLQSVoiceBindSubscription_t
voice.h, 1428
unpack_voice_SLQSVoiceBurstDTMF
voice.h, 1444
unpack_voice_SLQSVoiceBurstDTMF_t, 886
pCallID, 886
ParamPresenceMask, 886
unpack_voice_SLQSVoiceDialCall
voice.h, 1445
unpack_voice_SLQSVoiceDialCall_t, 886
pAlphaIDInfo, 887
pCCResultType, 887
pCCSUPSType, 887
pCallID, 887
ParamPresenceMask, 887
unpack_voice_SLQSVoiceEndCall
voice.h, 1445
unpack_voice_SLQSVoiceEndCall_t, 887
pCallId, 888
ParamPresenceMask, 888
unpack_voice_SLQSVoiceGetAllCallInfo
voice.h, 1445
unpack_voice_SLQSVoiceGetAllCallInfo_t, 888
pArrAlertingPattern, 891
pArrAlertingType, 891
pArrAlphaID, 891
pArrCallEndReason, 891
pArrCallInfo, 891
pArrCalledPartyNum, 891
pArrConnectPartyNum, 891
pArrDiagInfo, 891
pArrRedirPartyNum, 891

pArrRemotePartyName, 891
pArrRemotePartyNum, 891
pArrSvcOption, 891
pArrUUSInfo, 891
pVoicePrivacy, 891
ParamPresenceMask, 891
unpack_voice_SLQSVoiceGetCLIP
voice.h, 1447
unpack_voice_SLQSVoiceGetCLIP_t, 899
pAlphaIDInfo, 900
pCCResType, 900
pCallID, 900
pFailCause, 901
ParamPresenceMask, 900
unpack_voice_SLQSVoiceGetCLIR
voice.h, 1447
unpack_voice_SLQSVoiceGetCLIR_t, 901
pAlphaIDInfo, 902
pCCResType, 902
pCallID, 902
pFailCause, 902
ParamPresenceMask, 902
unpack_voice_SLQSVoiceGetCNAP
voice.h, 1448
unpack_voice_SLQSVoiceGetCNAP_t, 902
pAlphaIDInfo, 903
pCCResType, 903
pCallID, 903
pFailCause, 903
ParamPresenceMask, 903
unpack_voice_SLQSVoiceGetCOLP
voice.h, 1448
unpack_voice_SLQSVoiceGetCOLP_t, 904
pAlphaIDInfo, 905
pCCResType, 905
pCallID, 905
pFailCause, 905
ParamPresenceMask, 905
unpack_voice_SLQSVoiceGetCOLR
voice.h, 1448
unpack_voice_SLQSVoiceGetCOLR_t, 905
pAlphaIDInfo, 906
pCCResType, 906
pCallID, 906
pFailCause, 906
ParamPresenceMask, 906
unpack_voice_SLQSVoiceGetCallBarring
voice.h, 1446
unpack_voice_SLQSVoiceGetCallBarring_t, 892
pAlphaIDInfo, 893
pCCResType, 893
pCallID, 893
pFailCause, 893
pSvcClass, 893
ParamPresenceMask, 893
unpack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 1446
unpack_voice_SLQSVoiceGetCallForwardingStatus_t, 893
pAlphaIDInfo, 894
pCCResType, 894
pCallID, 894
pFailCause, 894
ParamPresenceMask, 894
unpack_voice_SLQSVoiceGetCallInfo
voice.h, 1446
unpack_voice_SLQSVoiceGetCallInfo_t, 895
pAlertType, 897
pAlertingPattern, 897
pAlphaIDInfo, 897
pCallInfo, 897
pConnectNumInfo, 897
pDiagInfo, 897
pOTASPStatus, 897
pRemotePartyName, 897
pRemotePartyNum, 897
pSrvOpt, 897
pUUSInfo, 898
pVoicePrivacy, 898
ParamPresenceMask, 897
unpack_voice_SLQSVoiceGetCallWaiting
voice.h, 1447
unpack_voice_SLQSVoiceGetCallWaiting_t, 898
pAlphaIDInfo, 899
pCCResType, 899
pCallID, 899
pFailCause, 899
pSvcClass, 899
ParamPresenceMask, 899
unpack_voice_SLQSVoiceGetConfig
voice.h, 1449
unpack_voice_SLQSVoiceGetConfig_t, 906
pAirTimerCnt, 908
pAutoAnswerStat, 908
pCurAMRConfig, 908
pCurPrefVoiceSO, 908
pCurVoiceDomainPref, 908
pCurVoicePrivacyPref, 908
pCurrTTYMode, 908
pRoamTimerCnt, 908
ParamPresenceMask, 908
unpack_voice_SLQSVoiceIndicationRegister
voice.h, 1449
unpack_voice_SLQSVoiceIndicationRegister_t
voice.h, 1428
unpack_voice_SLQSVoiceManageCalls
voice.h, 1449
unpack_voice_SLQSVoiceManageCalls_t, 908
pFailCause, 909
ParamPresenceMask, 909
unpack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 1450
unpack_voice_SLQSVoiceOrigUSSDNoWait_t
voice.h, 1428
unpack_voice_SLQSVoiceSUPSCallback_ind

voice.h, 1452
unpack_voice_SLQSVoiceSUPSCallback_ind_t, 915
pDataSrc, 918
pFailCause, 918
pReason, 918
pSvcClass, 918
ParamPresenceMask, 917
unpack_voice_SLQSVoiceSendFlash
voice.h, 1450
unpack_voice_SLQSVoiceSendFlash_t, 909
pCallID, 909
ParamPresenceMask, 909
unpack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 1450
unpack_voice_SLQSVoiceSetCallBarringPassword_t, 910
pAlphaIDInfo, 911
pCCResType, 911
pCallID, 911
pFailCause, 911
ParamPresenceMask, 911
unpack_voice_SLQSVoiceSetConfig
voice.h, 1451
unpack_voice_SLQSVoiceSetConfig_t, 911
pAirTimerStatus, 912
pAutoAnsStatus, 912
pPrefVoiceSOStatus, 912
pRoamTimerStatus, 912
pTTYConfigStatus, 912
pVoiceDomainPrefStatus, 912
ParamPresenceMask, 912
unpack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 1451
unpack_voice_SLQSVoiceSetPreferredPrivacy_t
voice.h, 1428
unpack_voice_SLQSVoiceSetSUPSService
voice.h, 1451
unpack_voice_SLQSVoiceSetSUPSService_t, 913
pCallID, 914
pFailCause, 914
ParamPresenceMask, 914
unpack_voice_SLQSVoiceStartContDTMF
voice.h, 1452
unpack_voice_SLQSVoiceStartContDTMF_t, 914
pCallID, 914
ParamPresenceMask, 914
unpack_voice_SLQSVoiceStopContDTMF
voice.h, 1452
unpack_voice_SLQSVoiceStopContDTMF_t, 914
callID, 915
ParamPresenceMask, 915
unpack_voice_SUPSNotificationCallback_ind
voice.h, 1453
unpack_voice_SUPSNotificationCallback_ind_t, 918
callID, 920
notifType, 920
pCUGIndex, 920
pECTNum, 920
ParamPresenceMask, 920
unpack_voice_USSDNotificationCallback_ind
voice.h, 1453
unpack_voice_USSDNotificationCallback_ind_t, 920
notification_Type, 920
ParamPresenceMask, 920
unpack_voice_VoiceInfoRecCallback_ind
voice.h, 1453
unpack_voice_VoiceInfoRecCallback_ind_t, 920
callID, 923
pCLIRCause, 923
pCallWaitInd, 923
pCalledPartyInfo, 923
pCallerIDInfo, 923
pCallerNameInfo, 923
pCallingPartyInfo, 923
pConnectNumInfo, 923
pDispInfo, 923
pExtDispInfo, 923
pExtDispRecInfo, 923
pLineCtrlInfo, 923
pNSSAudioCtrl, 923
pNSSRelease, 923
pRedirNumInfo, 923
pSignalInfo, 923
ParamPresenceMask, 923
unpack_voice_allCallStatusCallback_ind
voice.h, 1440
unpack_voice_allCallStatusCallback_ind_t, 879
arrCallInfomation, 880
pArrAlertingPattern, 880
pArrAlertingType, 880
pArrAlphaID, 880
pArrCallEndReason, 880
pArrCalledPartyNum, 880
pArrConnectPartyNum, 880
pArrDiagInfo, 880
pArrRedirPartyNum, 880
pArrRemotePartyName, 880
pArrRemotePartyNum, 880
pArrSvcOption, 880
ParamPresenceMask, 880
unpack_voice_voicePrivacyChangeCallback_ind
voice.h, 1454
unpack_voice_voicePrivacyChangeCallback_ind_t, 923
callID, 924
ParamPresenceMask, 924
voicePrivacy, 924
unpack_wds_DHCPv4ClientLease_ind
wds.h, 1483
unpack_wds_DHCPv4ClientLease_ind_t, 924
DHCPv4LeaseOptTlv, 925
DHCPv4LeaseStateTlv, 925
IPv4AddrTlv, 925
ParamPresenceMask, 925
ProfileIdTlv, 925
unpack_wds_DHCPv4ClientLeaseChange
wds.h, 1483

unpack_wds_DHCPv4ClientLeaseChange_t
wds.h, 1462
unpack_wds_GetAutoconnect
wds.h, 1484
unpack_wds_GetAutoconnect_t, 925
ParamPresenceMask, 925
psetting, 925
unpack_wds_GetByteTotals
wds.h, 1484
unpack_wds_GetByteTotals_t, 925
pRXTotalBytes, 926
pTXTotalBytes, 926
ParamPresenceMask, 926
unpack_wds_GetConnectionRate
wds.h, 1484
unpack_wds_GetConnectionRate_t, 926
currentChannelRXRate, 927
currentChannelTXRate, 927
maxChannelRXRate, 927
maxChannelTXRate, 927
ParamPresenceMask, 927
unpack_wds_GetDataBearerTechnology
wds.h, 1485
unpack_wds_GetDataBearerTechnology_t, 927
pDataBearer, 928
ParamPresenceMask, 928
unpack_wds_GetDefaultProfile
wds.h, 1485
unpack_wds_GetDefaultProfile_t, 928
apnname, 930
apnsize, 930
auth, 930
ipaddr, 930
ipaddrv6, 931
name, 931
namesize, 931
ParamPresenceMask, 931
pdptype, 931
pridns, 931
pridnsv6, 931
secdns, 931
secdnsv6, 931
username, 931
usersize, 931
unpack_wds_GetDefaultProfileNum
wds.h, 1485
unpack_wds_GetDefaultProfileNum_t, 931
index, 931
ParamPresenceMask, 931
unpack_wds_GetDormancyState
wds.h, 1486
unpack_wds_GetDormancyState_t, 931
dormancyState, 932
ParamPresenceMask, 932
unpack_wds_GetLastMobileIPError
wds.h, 1486
unpack_wds_GetLastMobileIPError_t, 932
error, 932
ParamPresenceMask, 933
unpack_wds_GetMobileIP
wds.h, 1487
unpack_wds_GetMobileIP_t, 933
mipMode, 933
ParamPresenceMask, 933
unpack_wds_GetMobileIPProfile
wds.h, 1487
unpack_wds_GetMobileIPProfile_t, 933
AAASPI, 935
AAAState, 935
address, 935
enabled, 935
HASPI, 935
HAState, 935
NAI, 935
naiSize, 935
ParamPresenceMask, 935
primaryHA, 935
revTunneling, 935
secondaryHA, 935
unpack_wds_GetPacketStatistics
wds.h, 1487
unpack_wds_GetPacketStatistics_t, 935
pRXDroppedCount, 937
pRXOKBytesLastCall, 937
pRXOkBytesCount, 937
pRXPacketErrors, 937
pRXPacketOverflows, 937
pRXPacketSuccesses, 937
pTXDroppedCount, 937
pTXOKBytesLastCall, 937
pTXOkBytesCount, 937
pTXPacketErrors, 937
pTXPacketOverflows, 937
pTXPacketSuccesses, 937
ParamPresenceMask, 937
unpack_wds_GetPacketStatus
wds.h, 1488
unpack_wds_GetPacketStatus_t, 937
ParamPresenceMask, 939
rXDroppedCount, 939
rXOKBytesLastCall, 939
rXOkBytesCount, 939
rXPacketErrors, 939
rXPacketOverflows, 939
rXPacketSuccesses, 939
tXDroppedCount, 939
tXOKBytesLastCall, 939
tXOkBytesCount, 939
tXPacketErrors, 939
tXPacketOverflows, 939
tXPacketSuccesses, 939
unpack_wds_GetSessionDuration
wds.h, 1488
unpack_wds_GetSessionDuration_t, 939
callDuration, 940
ParamPresenceMask, 940

unpack_wds_GetSessionState
wds.h, 1488
unpack_wds_GetSessionState_t, 940
connectionStatus, 940
ParamPresenceMask, 940
unpack_wds_RMSetTransferStatistics
wds.h, 1489
unpack_wds_RMSetTransferStatistics_t, 940
ParamPresenceMask, 941
unpack_wds_RMTransferStatistics_ind
wds.h, 1489
unpack_wds_SLQSCreateProfile
wds.h, 1492
unpack_wds_SLQSCreateProfile_t, 941
pCreateProfileOut, 942
pProfileID, 942
ParamPresenceMask, 942
Tlvresult, 942
unpack_wds_SLQSDUNCallInfoCallBack_ind
wds.h, 1492
unpack_wds_SLQSDUNCallInfoCallBack_ind_t, 943
unpack_wds_SLQSDeleteProfile
wds.h, 1492
unpack_wds_SLQSDeleteProfile_t, 942
extendedErrorCode, 942
ParamPresenceMask, 942
unpack_wds_SLQSGet3GPPConfigItem
wds.h, 1493
unpack_wds_SLQSGet3GPPConfigItem_t, 944
_3gppRelease, 945
defaultPDNEnabled, 945
LTEAttachProfile, 945
ParamPresenceMask, 946
profileList, 946
unpack_wds_SLQSGetCurrDataSystemStat
wds.h, 1493
unpack_wds_SLQSGetCurrDataSystemStat_t, 946
currNetworkInfo, 946
networkInfoLen, 946
ParamPresenceMask, 946
prefNetwork, 947
unpack_wds_SLQSGetCurrentChannelRate
wds.h, 1493
unpack_wds_SLQSGetCurrentChannelRate_t, 947
max_channel_rx_rate, 948
max_channel_tx_rate, 948
ParamPresenceMask, 948
unpack_wds_SLQSGetDUNCallInfo
wds.h, 1494
unpack_wds_SLQSGetDUNCallInfo_t, 949
callEndReason, 951
channelRate, 951
connectionStatus, 951
dataBearerTech, 951
dormancyStatus, 951
lastCallDataBearerTech, 951
mdmCallDurationActive, 952
ParamPresenceMask, 952
rxOKBytesCount, 952
txOKBytesCount, 952
unpack_wds_SLQSGetDataBearerTechnology
wds.h, 1494
unpack_wds_SLQSGetDataBearerTechnology_t, 948
curDataBearerTechnology, 948
dataBearerMask, 948
lastCallDataBearerTechnology, 949
ParamPresenceMask, 949
unpack_wds_SLQSGetProfileSettings
wds.h, 1494
unpack_wds_SLQSGetProfileSettings_t, 952
pProfileSettings, 952
ParamPresenceMask, 952
ProfileType, 952
Tlvresult, 952
unpack_wds_SLQSGetRuntimeSettings
wds.h, 1495
unpack_wds_SLQSGetRuntimeSettings_t, 952
APNName, 955
Authentication, 955
DomainList, 955
GPRSGrantedQoS, 955
GWAddressV4, 955
IMCNflag, 955
IPFamilyPreference, 955
IPV6AddrInfo, 955
IPV6GWAddrInfo, 955
IPv4, 955
Mtu, 955
PDPType, 955
ParamPresenceMask, 955
PrimaryDNSV4, 955
PrimaryDNSV6, 955
ProfileID, 955
ProfileName, 955
SecondaryDNSV4, 955
SecondaryDNSV6, 955
ServerAddrList, 955
SubnetMaskV4, 955
Technology, 955
UMTSGrantedQoS, 956
Username, 956
unpack_wds_SLQSModifyProfile
wds.h, 1495
unpack_wds_SLQSModifyProfile_t, 956
pExtErrorCode, 956
ParamPresenceMask, 956
unpack_wds_SLQSResetPacketStatics
wds.h, 1495
unpack_wds_SLQSResetPacketStatics_t
wds.h, 1463
unpack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 1497
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 964
pHwConfig, 965
ParamPresenceMask, 965
unpack_wds_SLQSSGetLoopback

wds.h, 1498
unpack_wds_SLQSSGetLoopback_t, 965
ByteLoopbackMode, 966
ByteLoopbackMultiplier, 966
ParamPresenceMask, 966
unpack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 1498
unpack_wds_SLQSSSetDHCPv4ClientConfig_t
wds.h, 1463
unpack_wds_SLQSSSetLoopback
wds.h, 1498
unpack_wds_SLQSSSetLoopback_t
wds.h, 1463
unpack_wds_SLQSSet3GPPConfigItem
wds.h, 1496
unpack_wds_SLQSSet3GPPConfigItem_t
wds.h, 1463
unpack_wds_SLQSSetIPFamilyPreference
wds.h, 1496
unpack_wds_SLQSSetIPFamilyPreference_t, 956
ParamPresenceMask, 957
Tlvresult, 957
unpack_wds_SLQSSetPacketSrvStatusCallback
wds.h, 1496
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 957
bearerID, 958
conn_status, 958
ipFamily, 959
ParamPresenceMask, 959
reconfigReqd, 959
sessionEndReason, 959
techName, 959
verboseSessnEndReason, 959
verboseSessnEndReasonType, 959
unpack_wds_SLQSSetWdsEventCallback
wds.h, 1497
unpack_wds_SLQSSetWdsEventCallback_ind
wds.h, 1497
unpack_wds_SLQSSetWdsEventCallback_ind_t, 959
currDBTechAvail, 963
currNWInfo, 963
dBTechAvail, 963
dBTechnology, 964
dBtechExtAvail, 963
dBtechnologyExt, 964
dataSysStatAvail, 963
dormancyStatAvail, 964
dormancyStatus, 964
mipStatus, 964
mipstatAvail, 964
netInfoLen, 964
ParamPresenceMask, 964
prefNetwork, 964
ratMask, 964
rx_bytes, 964
rx_pkts, 964
soMask, 964
tx_bytes, 964
tx_pkts, 964
xferStatAvail, 964
unpack_wds_SLQSSetWdsEventCallback_t
wds.h, 1463
unpack_wds_SLQSStartDataSession
wds.h, 1499
unpack_wds_SLQSStartDataSession_t, 966
pFailureReason, 966
pVerboseFailReasonType, 967
pVerboseFailureReason, 967
ParamPresenceMask, 966
psid, 966
unpack_wds_SLQSStopDataSession
wds.h, 1499
unpack_wds_SLQSStopDataSession_t
wds.h, 1463
unpack_wds_SLQSWdsGoActive
wds.h, 1499
unpack_wds_SLQSWdsGoActive_t
wds.h, 1463
unpack_wds_SLQSWdsGoDormant
wds.h, 1500
unpack_wds_SLQSWdsGoDormant_t
wds.h, 1463
unpack_wds_SLQSWdsSetEventReport
wds.h, 1500
unpack_wds_SLQSWdsSetEventReport_t
wds.h, 1463
unpack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 1500
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 967
apnName, 969
bearerId, 969
contextId, 969
ipv4Address, 969
ipv4GWAddress, 969
ipv6Address, 969
ipv6GWAddress, 969
ParamPresenceMask, 969
unpack_wds_SetAutoconnect
wds.h, 1489
unpack_wds_SetAutoconnect_t
wds.h, 1462
unpack_wds_SetDefaultProfile
wds.h, 1490
unpack_wds_SetDefaultProfile_t
wds.h, 1462
unpack_wds_SetDefaultProfileNum
wds.h, 1490
unpack_wds_SetDefaultProfileNum_t
wds.h, 1462
unpack_wds_SetMobileIP
wds.h, 1490
unpack_wds_SetMobileIP_t
wds.h, 1462
unpack_wds_SetMobileIPParameters
wds.h, 1491

unpack_wds_SetMobileIPParameters_t
wds.h, 1462
unpack_wds_SetMobileIPProfile
wds.h, 1491
unpack_wds_SetMobileIPProfile_t, 941
ParamPresenceMask, 941
unpack_wds_SetMuxID
wds.h, 1491
unpack_wds_SetMuxID_t
wds.h, 1462
UnpackQmiProfileInfo
wds.h, 1463
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 970
bBinaryType, 971
bSerity, 971
bState, 971
bUserInputRequest, 971
szDescription, 971
szName, 971
szVersion, 971
TlvPresent, 971
ulPkgDownloadComplete, 971
ulPkgDownloadSize, 972
wDescriptionLength, 972
wNameLength, 972
wUpdateCompeteStatus, 972
wUserInputTimeout, 972
wVersionLength, 972
UnpackSwiAvmsEventReportConfig, 972
bState, 972
bUserInputRequest, 972
szAlertMsg, 972
TlvPresent, 973
wAlertMsgLength, 973
wUserInputTimeout, 973
UnpackSwiAvmsEventReportConnectionRequest, 973
bUserInputRequest, 973
TlvPresent, 973
wUserInputTimeout, 973
UnpackSwiAvmsEventReportDataSessionStatus, 973
bType, 974
TlvPresent, 974
wErrorCode, 974
UnpackSwiAvmsEventReportHTTPStatus, 974
TlvPresent, 975
wHTTPStatus, 975
UnpackSwiAvmsEventReportNotification, 976
bNotification, 976
TlvPresent, 977
wSessionStatus, 977
UnpackSwiAvmsEventReportPackageID, 977
bPackageID, 977
TlvPresent, 977
UnpackSwiAvmsEventReportRegStatus, 977
bRegStatus, 977
TlvPresent, 977
UnpackSwiAvmsEventReportSessionType, 978
bType, 978
TlvPresent, 978
UnpackSwiAvmsEventReportWAMSParamChange, 978
TlvPresent, 979
wWamsChangeMask, 979
unpackWdsProfileParam, 979
SlqsProfile3GPP, 979
SlqsProfile3GPP2, 979
upLink
voice_NSSAudioCtrl, 1016
UpdateCompleteStatus
unpack_swioma_SLQSOMADMGetSessionInfo_t, 861
updateCompleteStatus
unpack_omaDmFotaTlv_t, 779
upinRetries
slotInf, 524
uim_slotInfo, 573
upinState
slotInf, 524
uim_slotInfo, 573
upukRetries
slotInf, 524
uim_slotInfo, 573
urlAddr
loc_URLAddrInfo, 142
usageMask
loc_sensorDataUsage, 139
UsbMTUSize
swidms_usbMTUSizeTlv, 548
usbNetNum
unpack_swidms_SLQSSwiDmsGetUsbNetNum_t, 854
userData
unpack_sms_SLQSWmsAsyncRawSendCallBack-_ind_t, 837
userInputReq
unpack_omaDmConfigTlv_t, 777
unpack_omaDmFotaTlv_t, 779
userInputTimeout
unpack_omaDmConfigTlv_t, 777
unpack_omaDmFotaTlv_t, 779
Username
unpack_wds_SLQSGetRuntimeSettings_t, 956
username
unpack_wds_GetDefaultProfile_t, 931
usersize
unpack_wds_GetDefaultProfile_t, 931
ussDCS
pack_voice_SLQSOriginateUSSD_t, 468
voice_USSInfo, 1026
ussData
pack_voice_SLQSOriginateUSSD_t, 468
voice_USSInfo, 1026
ussLen
pack_voice_SLQSOriginateUSSD_t, 468
voice_USSInfo, 1026
UtRat

unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
UtRatVal
imsa_UtRatInfo, 89
UtService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
UtServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
UtServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
UtSvcStatus
imsa_UtSvcStatusInfo, 89
uusInfo
voice_allCallsUUSInfo, 981

VDOP
loc_precisionDilution, 136
val
unpack_qos_IPv6TrafCls_t, 796
unpack_qos_Tos_t, 815
val_3GPP2Pri
unpack_qos_swiQosFlow_t, 814
val_3GPPImCn
unpack_qos_swiQosFlow_t, 814
val_3GPPResResidualBER
unpack_qos_swiQosFlow_t, 814
val_3GPPSigInd
unpack_qos_swiQosFlow_t, 814
val_3GPPTraHdlPri
unpack_qos_swiQosFlow_t, 814
ValidBitmasks
swidms_supportedIntBitmaskTlv, 547
ValidMask
unpack_pds_SLQSGetGPSStateInfo_t, 793
validMask
loc_satelliteInfo, 138
ValidityCW0
unpack_nas_SLQSSwiGetLteCQI_t, 770
ValidityCW1
unpack_nas_SLQSSwiGetLteCQI_t, 770
Value
pack_swiaudio_SLQSSetM2MSpkrGain_t, 438
unpack_swiaudio_SLQSGetM2MSpkrGain_t, 842
value
pack_dms_UIMSetPINProtection_t, 343
pack_dms_UIMVerifyPIN_t, 344
value_length
DMScustSettingInfo, 63
pack_dms_SetCustFeaturesV2_t, 332
verboseSessnEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 959
verboseSessnEndReasonType
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 959
verifyLeft
uim_personalizationStatus, 565
uim_remainingRetries, 569
verifyPIN
pack_uim_VerifyPin_t, 466
verifyRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 643
unpack_dms_UIMSetControlKeyProtection_t, 646
unpack_dms_UIMSetPINProtection_t, 647
version
unpack_omaDmFotaTlv_t, 779
unpack_qos_swiQosFilter_t, 810
versionString
pack_dms_SLQSSwiSetOSInfo_t, 338
unpack_dms_SLQSSwiGetOSInfo_t, 637
versionlength
unpack_omaDmFotaTlv_t, 779
vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 364
vertReliability
pack_loc_SLQSLOCInjectPosition_t, 364
vertUnc
pack_loc_SLQSLOCInjectPosition_t, 364
VerticalUncertainty
unpack_pds_SLQSGetGPSStateInfo_t, 793
VirtStream
nas_protocolSubtypeElement, 248
voice.h
LITE_VOICE_SUPS_SRV_CLASS_DATA, 1429
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC, 1429
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC, 1429
LITE_VOICE_SUPS_SRV_CLASS_FAX, 1429
LITE_VOICE_SUPS_SRV_CLASS_NONE, 1429
LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS, 1429
LITE_VOICE_SUPS_SRV_CLASS_PADACCESS, 1429
LITE_VOICE_SUPS_SRV_CLASS_SMS, 1429
LITE_VOICE_SUPS_SRV_CLASS_VOICE, 1429
voice.h, 1422
liteServiceClassInformation, 1428
MAXVOICEUSSDLENGTH, 1428
pack_voice_AnswerUSSD, 1429
pack_voice_CancelUSSD, 1429
pack_voice_OriginateUSSD, 1430
pack_voice_SLQSOriginateUSSD, 1430
pack_voice_SLQSVoiceALSSelectLine, 1430
pack_voice_SLQSVoiceALSSetLineSwitching, 1431
pack_voice_SLQSVoiceAnswerCall, 1431
pack_voice_SLQSVoiceBindSubscription, 1431
pack_voice_SLQSVoiceBurstDTMF, 1432
pack_voice_SLQSVoiceDialCall, 1432
pack_voice_SLQSVoiceEndCall, 1432
pack_voice_SLQSVoiceGetAllCallInfo, 1433
pack_voice_SLQSVoiceGetCLIP, 1435
pack_voice_SLQSVoiceGetCLIR, 1435
pack_voice_SLQSVoiceGetCNAP, 1435
pack_voice_SLQSVoiceGetCOLP, 1436

pack_voice_SLQSVoiceGetCOLR, 1436
pack_voice_SLQSVoiceGetCallBarring, 1433
pack_voice_SLQSVoiceGetCallForwardingStatus, 1434
pack_voice_SLQSVoiceGetCallInfo, 1434
pack_voice_SLQSVoiceGetCallWaiting, 1434
pack_voice_SLQSVoiceGetConfig, 1436
pack_voice_SLQSVoiceIndicationRegister, 1437
pack_voice_SLQSVoiceManageCalls, 1437
pack_voice_SLQSVoiceOrigUSSDNoWait, 1437
pack_voice_SLQSVoiceSendFlash, 1438
pack_voice_SLQSVoiceSetCallBarringPassword, 1438
pack_voice_SLQSVoiceSetConfig, 1438
pack_voice_SLQSVoiceSetPreferredPrivacy, 1439
pack_voice_SLQSVoiceSetSUPSService, 1439
pack_voice_SLQSVoiceStartContDTMF, 1440
pack_voice_SLQSVoiceStopContDTMF, 1440
unpack_voice_AnswerUSSD, 1441
unpack_voice_AnswerUSSD_t, 1428
unpack_voice_CancelUSSD, 1441
unpack_voice_CancelUSSD_t, 1428
unpack_voice_DTMFEventCallback_ind, 1441
unpack_voice_OTASPStatusCallback_ind, 1442
unpack_voice_OriginateUSSD, 1442
unpack_voice_OriginateUSSD_t, 1428
unpack_voice_SLQSOriginateUSSD, 1442
unpack_voice_SLQSVoiceALSSelectLine, 1443
unpack_voice_SLQSVoiceALSSelectLine_t, 1428
unpack_voice_SLQSVoiceALSSetLineSwitching, 1443
unpack_voice_SLQSVoiceALSSetLineSwitching_t, 1428
unpack_voice_SLQSVoiceAnswerCall, 1444
unpack_voice_SLQSVoiceBindSubscription, 1444
unpack_voice_SLQSVoiceBindSubscription_t, 1428
unpack_voice_SLQSVoiceBurstDTMF, 1444
unpack_voice_SLQSVoiceDialCall, 1445
unpack_voice_SLQSVoiceEndCall, 1445
unpack_voice_SLQSVoiceGetAllCallInfo, 1445
unpack_voice_SLQSVoiceGetCLIP, 1447
unpack_voice_SLQSVoiceGetCLIR, 1447
unpack_voice_SLQSVoiceGetCNAP, 1448
unpack_voice_SLQSVoiceGetCOLP, 1448
unpack_voice_SLQSVoiceGetCOLR, 1448
unpack_voice_SLQSVoiceGetCallBarring, 1446
unpack_voice_SLQSVoiceGetCallForwarding-Status, 1446
unpack_voice_SLQSVoiceGetCallInfo, 1446
unpack_voice_SLQSVoiceGetCallWaiting, 1447
unpack_voice_SLQSVoiceGetConfig, 1449
unpack_voice_SLQSVoiceIndicationRegister, 1449
unpack_voice_SLQSVoiceIndicationRegister_t, 1428
unpack_voice_SLQSVoiceManageCalls, 1449
unpack_voice_SLQSVoiceOrigUSSDNoWait, 1450
unpack_voice_SLQSVoiceOrigUSSDNoWait_t, 1428
unpack_voice_SLQSVoiceSUPSCallback_ind, 1452
unpack_voice_SLQSVoiceSendFlash, 1450
unpack_voice_SLQSVoiceSetCallBarringPassword, 1450
unpack_voice_SLQSVoiceSetConfig, 1451
unpack_voice_SLQSVoiceSetPreferredPrivacy, 1451
unpack_voice_SLQSVoiceSetPreferredPrivacy_t, 1428
unpack_voice_SLQSVoiceSetSUPSService, 1451
unpack_voice_SLQSVoiceStartContDTMF, 1452
unpack_voice_SLQSVoiceStopContDTMF, 1452
unpack_voice_SUPSNotificationCallback_ind, 1453
unpack_voice_USSDNotificationCallback_ind, 1453
unpack_voice_VoiceInfoRecCallback_ind, 1453
unpack_voice_allCallStatusCallback_ind, 1440
unpack_voice_voicePrivacyChangeCallback_ind, 1454
voice_CLIPResp, 1002
ActiveStatus, 1002
ProvisionStatus, 1002
voice_CLIRResp, 1002
ActiveStatus, 1003
ProvisionStatus, 1003
voice_CNAPResp, 1003
ActiveStatus, 1004
ProvisionStatus, 1004
voice_COLPResp, 1004
ActiveStatus, 1004
ProvisionStatus, 1004
voice_COLRResp, 1004
ActiveStatus, 1005
ProvisionStatus, 1005
voice_CUGInfo, 1007
CUGIndex, 1007
SuppOA, 1007
SuppPrefCUG, 1007
voice_DTMFInfo, 1009
callID, 1009
DTMFEvent, 1009
digitBuff, 1009
digitCnt, 1009
voice_DTMFLengths, 1009
DTMFInterdigitInterval, 1010
DTMFPulseWidth, 1010
voice_ECTNum, 1010
ECTCallState, 1011
number, 1011
presentationInd, 1011
voice_NSSAudioCtrl, 1016
downLink, 1016
upLink, 1016
voice_SUPSInfo, 1025

isModByCC, 1025
svcType, 1025
voice_USSDNotificationNetworkInfo, 1025
networkInfo, 1026
tlvPresent, 1026
voice_USSInfo, 1026
ussDCS, 1026
ussData, 1026
ussLen, 1026
voice_UUSInfo, 1026
UUSData, 1027
UUSDatalen, 1027
UUSDcs, 1027
UUSType, 1027
voice_airTimer, 979
airTimerValue, 980
namID, 980
voice_allCallsAlphaIDInfo, 980
AlphaIDInfo, 980
callID, 980
voice_allCallsDiagInfo, 980
callID, 981
DiagInfo, 981
voice_allCallsUUSInfo, 981
callID, 981
uusInfo, 981
voice_alphaIDInfo, 981
alphaDcs, 982
alphaLen, 982
alphaText, 982
voice_arrAlertingPattern, 982
alertingPattern, 983
callID, 983
numInstances, 983
voice_arrAlertingType, 983
AlertingType, 984
callID, 984
numInstances, 984
voice_arrAlphaID, 984
allCallsAlphaIDInfoArr, 984
numInstances, 984
voice_arrCallEndReason, 985
callEndReason, 985
callID, 985
numInstances, 986
voice_arrCallInfo, 986
getAllCallInfo, 986
numInstances, 986
voice_arrCalledPartyNum, 984
CalledPartyNum, 985
numInstances, 985
voice_arrConnectPartyNum, 986
ConnectedPartyNum, 987
numInstances, 987
voice_arrDiagInfo, 987
DiagInfo, 987
numInstances, 987
voice_arrRedirPartyNum, 987
numInstances, 988
RedirPartyNum, 988
voice_arrRemotePartyName, 988
GetAllCallRmtPtyName, 988
numInstances, 988
voice_arrRemotePartyNum, 988
numInstances, 989
RmtPtyNum, 989
voice_arrSvcOption, 989
callID, 989
numInstances, 989
srvOption, 989
voice_arrUUSInfo, 989
AllCallsUUSInfo, 990
numInstances, 990
voice_burstDTMFInfo, 990
digitCnt, 991
pCallID, 991
pDigitBuff, 991
voice_callFWExtInfo, 995
noReplyTimer, 996
numLen, 996
numPlan, 997
numType, 997
number, 996
PI, 997
SI, 997
SvcClass, 997
SvcStatus, 997
voice_callFWInfo, 997
noReplyTimer, 997
numLen, 998
number, 997
SvcClass, 998
SvcStatus, 998
voice_callFwdTypeAndPlan, 994
numberPlan, 995
numberType, 995
voice_callInfo, 998
callID, 999
callState, 999
callType, 999
direction, 999
mode, 999
voice_calledPartyInfo, 991
numLen, 992
numPlan, 992
numType, 992
number, 992
PI, 992
SI, 992
voice_calledPartySubAdd, 992
extBit, 993
oddEvenInd, 993
subAddr, 993
subAddrLen, 993
subAddrType, 993
voice_callerIDInfo, 993

callerID, 994
callerIDLen, 994
PI, 994
voice_callingPartyInfo, 999
numLen, 1001
numPlan, 1001
numType, 1001
number, 1001
PI, 1001
SI, 1001
voice_ccSUPSType, 1001
reason, 1001
svcType, 1002
voice_connectNumInfo, 1005
callerID, 1006
callerIDLen, 1006
numPlan, 1006
numPresInd, 1006
numType, 1007
screeningInd, 1007
voice_curAMRConfig, 1007
gsmAmrStat, 1008
wcdmaAmrStat, 1008
voice_diagInfo, 1008
diagInfoLen, 1008
diagnosticInfo, 1008
voice_extDispRecInfo, 1011
dispType, 1011
extDispInfo, 1011
extDispInfoLen, 1011
voice_getAllCallInformation, 1012
ALS, 1012
CallInfo, 1012
isEmpty, 1012
voice_getAllCallRmtPtyName, 1012
callID, 1013
RemotePartyName, 1013
voice_getAllCallRmtPtyNum, 1013
callID, 1013
RemotePartyNum, 1013
voice_getCallFWExtInfo, 1013
CallFWExtInfo, 1014
numInstances, 1014
voice_getCallFWInfo, 1014
CallFWInfo, 1014
numInstances, 1014
voice_lineCtrlInfo, 1014
polarityIncluded, 1015
pwrDenialTime, 1015
revPolarity, 1015
toggleMode, 1015
voice_newPwdData, 1015
newPwd, 1016
newPwdAgain, 1016
voice_peerNumberInfo, 1016
callID, 1017
numLen, 1018
numPI, 1018
numPlan, 1018
numSI, 1018
numType, 1018
number, 1018
voice_prefVoiceSO, 1018
evrcCapability, 1020
homeOrigVoiceSO, 1020
homePageVoiceSO, 1020
namID, 1020
roamOrigVoiceSO, 1020
voice_redirNumInfo, 1020
numLen, 1021
numPlan, 1021
numType, 1021
number, 1021
PI, 1021
reason, 1021
SI, 1021
voice_remotePartyName, 1022
callerName, 1022
codingScheme, 1022
nameLen, 1022
namePI, 1022
voice_remotePartyNum, 1022
numLen, 1023
presentationInd, 1023
remPartyNumber, 1023
voice_roamTimer, 1023
namID, 1024
roamTimerValue, 1024
voice_signalInfo, 1024
alertPitch, 1024
signal, 1024
signalType, 1024
VoiceNumber
unpack_dms_GetVoiceNumber_t, 609
voiceNumberSize
unpack_dms_GetVoiceNumber_t, 609
voicePrivacy
unpack_voice_voicePrivacyChangeCallback_ind_t, 924
voiceSvc
pack_voice_SLQSVoiceSetSUPSService_t, 484
VoipRat
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
VoipRatVal
imsa_VoipRatInfo, 90
VoipService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
VoipServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
VoipServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
VoipSvcStatus
imsa_VoipSvcStatusInfo, 90
VolValue

pack_audio_SLQSSetAudioVolTLBConfig_t, 324
VoltStat
dms_VoltageTlv, 62
VoltTlv
unpack_dms_SwiEventReportCallBack_ind_t, 641
Voltage
dms_VoltageTlv, 62
Volume
pack_audio_SLQSGetAudioVolTLBConfig_t, 320
pack_audio_SLQSSetAudioProfile_t, 323
pack_audio_SLQSSetAudioVolTLBConfig_t, 324
unpack_audio_SLQSGetAudioProfile_t, 579
unpack_swiaudio_SLQSGetM2MAudioProfile_t, 840
voteForInit
uim_registerRefresh, 569
VsServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
VsServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
VtRat
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
VtRatVal
imsa_VtRatInfo, 91
VtService
unpack_imsa_SLQSImsaSvcStatusCallBack_ind_t, 672
VtServiceRat
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
VtServiceStatus
unpack_imsa_SLQSGetIMSAServiceStatus_t, 668
VtSvcStatus
imsa_VtSvcStatusInfo, 92

wAlertMsgLength
UnpackSwiAvmsEventReportConfig, 973
WCDMACellInfo
nas_lteWcdmaCellInfo, 220
WCDMAECIOThreshListLen
nas_WCDMAECIOThresh, 292
WCDMARSSIThreshListLen
nas_WCDMARSSIThresh, 294
WCDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 756
WDS_PROFILE_3GPP
wds.h, 1462
WDS_PROFILE_3GPP2
wds.h, 1462
wDescriptionLength
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 972
wErrorCode
UnpackSwiAvmsEventReportDataSessionStatus, 974
wHTTPStatus
UnpackSwiAvmsEventReportHTTPStatus, 975
wNameLength
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 972
WORD
SwiDataTypes.h, 1375
wSessionStatus
UnpackSwiAvmsEventReportNotification, 977
wUpdateCompeteStatus
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 972
wUserInputTimeout
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 972
UnpackSwiAvmsEventReportConfig, 973
UnpackSwiAvmsEventReportConnectionRequest, 973
wVersionLength
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 972
wWamsChangeMask
UnpackSwiAvmsEventReportWAMSParamChange, 979
wake_mask
pack_rms_SetSMSWake_t, 422
unpack_rms_GetSMSWake_t, 817
wcdmaAmrStat
voice_curAMRConfig, 1008
wcdmaRRCState
nas_WCDMAInfoLTENeighborCell, 293
Wdisable
unpack_dms_SLQSDmsSwiGetPCInfo_t, 619
wds.h
QMI_LITE_WDS_CURRENT_CALL_DB_MASK, 1463
QMI_LITE_WDS_LAST_CALL_DB_MASK, 1463
wds.h, 1454
BYT_STAT_STAT_MASK, 1462
liteQmiDataBearerMasks, 1463
PACK_WDS_IPV4, 1462
PACK_WDS_IPV6, 1462
pack_wds_DHCPv4ClientLeaseChange, 1463
pack_wds_GetAutoconnect, 1464
pack_wds_GetByteTotals, 1464
pack_wds_GetConnectionRate, 1465
pack_wds_GetDataBearerTechnology, 1465
pack_wds_GetDefaultProfile, 1465
pack_wds_GetDefaultProfileNum, 1466
pack_wds_GetDormancyState, 1466
pack_wds_GetLastMobileIPError, 1467
pack_wds_GetMobileIP, 1467
pack_wds_GetMobileIPProfile, 1468
pack_wds_GetPacketStatistics, 1468
pack_wds_GetPacketStatus, 1468
pack_wds_GetSessionDuration, 1469
pack_wds_GetSessionState, 1469
pack_wds_RMSetTransferStatistics, 1470
pack_wds_SLQSCreateProfile, 1473
pack_wds_SLQSDeleteProfile, 1473
pack_wds_SLQSGet3GPPConfigItem, 1474

pack_wds_SLQSGetCurrDataSystemStat, 1474
pack_wds_SLQSGetCurrentChannelRate, 1475
pack_wds_SLQSGetDUNCallInfo, 1476
pack_wds_SLQSGetDataBearerTechnology, 1475
pack_wds_SLQSGetProfileSettings, 1476
pack_wds_SLQSGetRuntimeSettings, 1476
pack_wds_SLQSModifyProfile, 1477
pack_wds_SLQSResetPacketStatics, 1477
pack_wds_SLQSSGetDHCPv4ClientConfig, 1479
pack_wds_SLQSSGetLoopback, 1479
pack_wds_SLQSSSetDHCPv4ClientConfig, 1480
pack_wds_SLQSSSetLoopback, 1480
pack_wds_SLQSSet3GPPConfigItem, 1478
pack_wds_SLQSSetIPFamilyPreference, 1478
pack_wds_SLQSSetWdsEventCallback, 1478
pack_wds_SLQSStartDataSession, 1480
pack_wds_SLQSStopDataSession, 1481
pack_wds_SLQSWdsGoActive, 1481
pack_wds_SLQSWdsGoDormant, 1482
pack_wds_SLQSWdsSetEventReport, 1482
pack_wds_SLQSWdsSwiPDPRuntimeSettings, 1483
pack_wds_SetAutoconnect, 1470
pack_wds_SetDefaultProfile, 1470
pack_wds_SetDefaultProfileNum, 1471
pack_wds_SetMobileIP, 1471
pack_wds_SetMobileIPParameters, 1472
pack_wds_SetMobileIPProfile, 1472
pack_wds_SetMuxID, 1473
unpack_wds_DHCPv4ClientLease_ind, 1483
unpack_wds_DHCPv4ClientLeaseChange, 1483
unpack_wds_DHCPv4ClientLeaseChange_t, 1462
unpack_wds_GetAutoconnect, 1484
unpack_wds_GetByteTotals, 1484
unpack_wds_GetConnectionRate, 1484
unpack_wds_GetDataBearerTechnology, 1485
unpack_wds_GetDefaultProfile, 1485
unpack_wds_GetDefaultProfileNum, 1485
unpack_wds_GetDormancyState, 1486
unpack_wds_GetLastMobileIPError, 1486
unpack_wds_GetMobileIP, 1487
unpack_wds_GetMobileIPProfile, 1487
unpack_wds_GetPacketStatistics, 1487
unpack_wds_GetPacketStatus, 1488
unpack_wds_GetSessionDuration, 1488
unpack_wds_GetSessionState, 1488
unpack_wds_RMSetTransferStatistics, 1489
unpack_wds_RMTransferStatistics_ind, 1489
unpack_wds_SLQSCreateProfile, 1492
unpack_wds_SLQSDUNCallInfoCallBack_ind, 1492
unpack_wds_SLQSDeleteProfile, 1492
unpack_wds_SLQSGet3GPPConfigItem, 1493
unpack_wds_SLQSGetCurrDataSystemStat, 1493
unpack_wds_SLQSGetCurrentChannelRate, 1493
unpack_wds_SLQSGetDUNCallInfo, 1494
unpack_wds_SLQSGetDataBearerTechnology, 1494
unpack_wds_SLQSGetProfileSettings, 1494
unpack_wds_SLQSGetRuntimeSettings, 1495
unpack_wds_SLQSModifyProfile, 1495
unpack_wds_SLQSResetPacketStatics, 1495
unpack_wds_SLQSResetPacketStatics_t, 1463
unpack_wds_SLQSSGetDHCPv4ClientConfig, 1497
unpack_wds_SLQSSGetLoopback, 1498
unpack_wds_SLQSSSetDHCPv4ClientConfig, 1498
unpack_wds_SLQSSSetDHCPv4ClientConfig_t, 1463
unpack_wds_SLQSSSetLoopback, 1498
unpack_wds_SLQSSSetLoopback_t, 1463
unpack_wds_SLQSSet3GPPConfigItem, 1496
unpack_wds_SLQSSet3GPPConfigItem_t, 1463
unpack_wds_SLQSSetIPFamilyPreference, 1496
unpack_wds_SLQSSetPacketSrvStatusCallback, 1496
unpack_wds_SLQSSetWdsEventCallback, 1497
unpack_wds_SLQSSetWdsEventCallback_ind, 1497
unpack_wds_SLQSSetWdsEventCallback_t, 1463
unpack_wds_SLQSStartDataSession, 1499
unpack_wds_SLQSStopDataSession, 1499
unpack_wds_SLQSStopDataSession_t, 1463
unpack_wds_SLQSWdsGoActive, 1499
unpack_wds_SLQSWdsGoActive_t, 1463
unpack_wds_SLQSWdsGoDormant, 1500
unpack_wds_SLQSWdsGoDormant_t, 1463
unpack_wds_SLQSWdsSetEventReport, 1500
unpack_wds_SLQSWdsSetEventReport_t, 1463
unpack_wds_SLQSWdsSwiPDPRuntimeSettings, 1500
unpack_wds_SetAutoconnect, 1489
unpack_wds_SetAutoconnect_t, 1462
unpack_wds_SetDefaultProfile, 1490
unpack_wds_SetDefaultProfile_t, 1462
unpack_wds_SetDefaultProfileNum, 1490
unpack_wds_SetDefaultProfileNum_t, 1462
unpack_wds_SetMobileIP, 1490
unpack_wds_SetMobileIP_t, 1462
unpack_wds_SetMobileIPParameters, 1491
unpack_wds_SetMobileIPParameters_t, 1462
unpack_wds_SetMobileIPProfile, 1491
unpack_wds_SetMuxID, 1491
unpack_wds_SetMuxID_t, 1462
UnpackQmiProfileInfo, 1463
WDS_PROFILE_3GPP, 1462
WDS_PROFILE_3GPP2, 1462
wds_ChannelRateTlv, 1028
ChannelRate, 1028
TlvPresent, 1028
wds_ConnStatusTlv, 1029
MDMConnStatus, 1029
TlvPresent, 1029
wds_DHCPLeaseOptTlv, 1032
numOpt, 1032

optList, 1032
optListData, 1033
TlvPresent, 1033
wds_DHCPLeaseStateTlv, 1033
leaseState, 1033
TlvPresent, 1033
wds_DHCPOpt, 1033
optCode, 1034
optValLen, 1034
pOptVal, 1034
wds_DHCPProfileIdTlv, 1034
profileId, 1034
profileType, 1034
TlvPresent, 1034
wds_DHCPv4HWConfig, 1034
chaddr, 1035
chaddrLen, 1035
hwType, 1035
wds_DHCPv4Option, 1035
optCode, 1036
optVal, 1036
optValLen, 1036
wds_DHCPv4OptionList, 1036
numOpt, 1036
pOptList, 1036
wds_DHCPv4ProfileId, 1036
profileId, 1037
profileType, 1037
wds_DataBearTechTlv, 1030
DataBearerTech, 1031
TlvPresent, 1031
wds_DataULongLongTlv, 1031
TlvPresent, 1032
ullData, 1032
wds_DataULongTlv, 1032
TlvPresent, 1032
ulData, 1032
wds_Domain, 1037
domainLen, 1037
domainName, 1037
wds_DomainNameList, 1037
domain, 1038
numInstances, 1038
wds_DormStatTlv, 1038
DormancyStat, 1038
TlvPresent, 1038
wds_GPRSQoS, 1038
delayClass, 1039
meanThroughputClass, 1039
peakThroughputClass, 1039
precedenceClass, 1039
reliabilityClass, 1039
wds_IPV6AddressInfo, 1040
IPAddressV6, 1040
IPV6PrefixLen, 1040
wds_IPV6GWAddressInfo, 1040
gwAddressV6, 1041
gwV6PrefixLen, 1041
wds_IPv4AdTlv, 1039
IPv4Addr, 1040
TlvPresent, 1040
wds_LastMdmCallEndRsnTlv, 1041
CallEndReason, 1041
TlvPresent, 1041
wds_PCSCFFQDNAddress, 1041
fqdnAddr, 1042
fqdnLen, 1042
wds_PCSCFFQDNAddressList, 1042
numInstances, 1042
pcsfFQDNAddress, 1042
wds_PCSCFIPv4ServerAddressList, 1042
numInstances, 1043
pscsfIPv4Addr, 1043
wds_ProfileIdentifier, 1043
profileIndex, 1043
profileType, 1044
wds_RXBytesOKTlv, 1044
RxByteOKCnt, 1044
TlvPresent, 1044
wds_TXBytesOKTlv, 1046
TlvPresent, 1046
TxByteOKCnt, 1046
wds_TrStatInd, 1045
statsMask, 1046
statsPeriod, 1046
wds_UMTSMinQoS, 1047
deliveryErrSDU, 1048
grntDownlinkBitrate, 1048
grntUplinkBitrate, 1049
maxDownlinkBitrate, 1049
maxSDUSize, 1049
maxUplinkBitrate, 1049
qosDeliveryOrder, 1049
resBerRatio, 1049
sduErrorRatio, 1049
trafficClass, 1049
trafficPriority, 1049
transferDelay, 1049
wds_channelRate, 1028
CurrChanRxRate, 1028
CurrChanTxRate, 1028
wds_currNetworkInfo, 1029
NetworkType, 1030
RATMask, 1030
SOMask, 1030
wds_profileInfo, 1044
SlqsProfile3GPP, 1044
SlqsProfile3GPP2, 1044
wds_transferStatInd, 1045
StatsMask, 1045
StatsPeriod, 1045
wdsDhcpv4HwConfig, 1049
chaddr, 1049
chaddrLen, 1049
hwType, 1049
wdsDhcpv4Option, 1050

optCode, 1050
optVal, 1050
optValLen, 1050
wdsDhcpv4OptionList, 1050
numOpt, 1050
pOptList, 1051
wdsDhcpv4ProfileId, 1051
profileId, 1051
profileType, 1051
word
swi_uint256_t, 543

xAxis
sensorData_t, 521
xferStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 964
xid
pack_loc_SLQSLOCGetBestAvailPos_t, 358
pack_qmi_t, 420
unpack_qmi_t, 794
xtra_start_gps_minutes
unpack_pds_SLQSGetGPSStateInfo_t, 793
xtra_start_gps_week
unpack_pds_SLQSGetGPSStateInfo_t, 794
xtra_valid_duration_hours
unpack_pds_SLQSGetGPSStateInfo_t, 794

yAxis
sensorData_t, 521
year
nas_timeInfo, 285
nas_UniversalTime, 291

zAxis
sensorData_t, 521