



CDMA 1xEV-DO CnS Reference

Proprietary and Confidential

Products supported:

| | |
|---------------------|-------------|
| MiniCard | MC5728V |
| ExpressCard/PC Card | AirCard 402 |
| USB modem | USB 598 |



SIERRA
WIRELESS

2130754
Rev 2

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.

Safety and hazards

Do not operate the Sierra Wireless modem in areas where blasting is in progress, where explosive atmospheres may be present, near medical equipment, near life support equipment, or any equipment which may be susceptible to any form of radio interference. In such areas, the Sierra Wireless modem **MUST BE POWERED OFF**. The Sierra Wireless modem can transmit signals that could interfere with this equipment.

Do not operate the Sierra Wireless modem in any aircraft, whether the aircraft is on the ground or in flight. In aircraft, the Sierra Wireless modem **MUST BE POWERED OFF**. When operating, the Sierra Wireless modem can transmit signals that could interfere with various onboard systems.

Note: Some airlines may permit the use of cellular phones while the aircraft is on the ground and the door is open. Sierra Wireless modems may be used at this time.

The driver or operator of any vehicle should not operate the Sierra Wireless modem while in control of a vehicle. Doing so will detract from the driver or operator's control and operation of that vehicle. In some states and provinces, operating such communications devices while in control of a vehicle is an offence.

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.

Patents

This product may contain technology developed by or for Sierra Wireless Inc. This product includes technology licensed from QUALCOMM®. This product is manufactured or sold by Sierra Wireless Inc. or its affiliates under one or more patents licensed from InterDigital Group.

Copyright

©2012 Sierra Wireless. All rights reserved.

Trademarks

AirCard® is a registered trademark of Sierra Wireless. Sierra Wireless™, AirPrime™, AirLink™, AirVantage™, Watcher™ and the Sierra Wireless logo are trademarks of Sierra Wireless.

Windows® and Windows Vista® are registered trademarks of Microsoft Corporation.

Macintosh and Mac OS are registered trademarks of Apple Inc., registered in the U.S. and other countries.

QUALCOMM® is a registered trademark of QUALCOMM Incorporated. Used under license.

Other trademarks are the property of the respective owners.

Contact Information

| | | |
|--------------------|---------|--|
| Sales Desk: | Phone: | 1-604-232-1488 |
| | Hours: | 8:00 AM to 5:00 PM Pacific Time |
| | E-mail: | sales@sierrawireless.com |

| | |
|---------------------------|--|
| Technical Support: | Included with the purchase of the AirCard402/ MC5728V/ USB 598 Development Kit you receive five hours of tier 3 engineering integration support. You will have received instructions by e-mail on how to access the OEM Customer Support web site. For more details, please contact your account manager, or the Sierra Wireless sales desk (see above). |
| RMA Support: | repairs@sierrawireless.com |
| Post: | Sierra Wireless 13811 Wireless Way Richmond, BC Canada V6V 3A4 |
| Fax: | 1-604-231-1109 |
| Web: | www.sierrawireless.com |

For up-to-date product descriptions, documentation, application notes, firmware upgrades, troubleshooting tips, and press releases, consult our website:

www.sierrawireless.com

Revision history

| Version | Summary of changes | Internal document |
|-----------------------|--|-------------------|
| 1.0 Apr 07 | Initial release | |
| 1.1 Apr 08 | <p>Added AirCard 597E ExpressCard, Compass 597 USB modem, MC5727 Mini Card.</p> <p>Fixed the values for query of the PRI SKU revision field (Get/set NV Item, starting on page 133).</p> <p>HDR Miscellaneous Notification Status (page 154), HDR Tech Notification Status (page 172), and Call Notification Status (page 196): removed parameter 07 (Notify).</p> <p>HDR Miscellaneous Notification Status (page 154) and HDR Tech Notification Status (page 172): fixed Len (correct value is 2, instead of 4).</p> | |
| 1.2 Aug 08 | <p>New sections:</p> <ul style="list-style-type: none"> Device Management (page 109); Appendix B: LBS Message Flow (page 313); Appendix C: PRL Message Flow (page 315) [Bugzilla item 10706] <p>New objects:</p> <ul style="list-style-type: none"> DM objects (see "Device Management" starting on page 109); Audio Profile (page 78); Device Time (page 81); PRI Information (page 82); Flash Image Information (page 83); Removable Media Configuration (page 87); | |

| Version | Summary of changes | Internal document |
|---------|--|--|
| | <p>Remote Diagnostics Agent (page 88); Satellite Information (page 286); Clear GPS Assistance Data (page 286); System Selection (page 159) [Bugzilla item 10789].</p> <p>New figure: “SMS retrieve and delete: CnS message flow” (page 302)</p> <p>Sierra Wireless Modem Type (page 54): added 0x24, 0x25, and 0x3A; added new fields.</p> <p>Get/set NV Item (page 133): added GSP info (0x0190; 0x0EAC; 0x1004; 0x608D).</p> <p>Supported Object Releases (page 58): added 0x06.</p> <p>Current Radio Band (page 67): added 0xFF.</p> <p>Supported Object Releases (page 58) and Current Radio Band (page 67): added maximum list length.</p> <p>Get Activation Date (page 137): changed Get parameter.</p> <p>HDR Single-User Forward Traffic Channel DRC packet termination slot count (page 180): adjusted the description of TC DRC8.</p> <p>LBS status (page 262): now supports notifications.</p> <p>Position Fix Session End (page 272) : added 0x0400 and 0x0401 to Table 32 (page 272).</p> <p>Added the MC5727V MiniCard.</p> <p>Added a reference to the Linux CnS Reference, and to document 2130907.</p> | |
| 1.3 | <p>Added AirCard 402, USB 598, MC5728V.</p> <p>Removed references to NAM 1. Only NAM 0 is supported.</p> <p>Removed the list of QUALCOMM patents from the preface.</p> <p>Removed “Appendix F: Source Documents”. Added “Internal document” column to this table.</p> <p>Changes to Audio Profile (page 78) and Audio Mode (page 107): added support for HAC (Hearing Aid Compatibility) profile.</p> | |
| | <p>New objects: DM DL Progress (page 120); DM Download Alert (page 121); DM Download Description (page 123); DM Download — Installation Status (page 124).</p> <p>Changes to: DM Configuration (page 110) – HFA attempted flag is deprecated; Start DM Session (page 112) – DM-TLV Account node name is currently not used; Network-Initiated Alert (page 118) - Special UI mode is deprecated.</p> <p>Renamed “Supported Object Versions” to “Supported Object Releases”, and “object set version” to “object set Release”.</p> | 2111259_CWE_DM_Control_and_Status_Objects_R2_V1.5_CLEAN.2009.01.23.doc |

| Version | Summary of changes | Internal document |
|------------------------------|--|---|
| | <p>Added 3 right-most columns to Table 2 on page 36.</p> <p>New object: SWOC Configuration (page 89).</p> <p>Changes to: Sierra Wireless Modem Type (page 54) – Added values for Modem type.</p> | <p>Protocol Interface Specification for CWE Control and Status Objects [Object Revision 13]_1.39_CLEAN.doc</p> |
| | <p>Changes to: Position Fix Data (page 267)</p> | <p>2111248_CWE_LBS_Control_and_Status_Objects_R4_v2.3_CLEAN.doc</p> |
| <p>1.4 Jun 10</p> | <p>Fixed inaccuracy: changed the number of text messages that can be buffered from 19 to 98 (section “Incoming messages” on page 293).</p> <p>Fixed inaccuracy for Radio Temperature (page 52): changed “Type” for current temperature from UINT16 to INT16.</p> <p>SMS Write (page 236): Corrected info for “03 (Set) 1st”.</p> <p>Changes to Audio Profile (page 78) and Audio Mode (page 107) (MC5728V [firmware revision 1.04 and higher])</p> | |
| | <p>Added Notify support to Current CDMA System Time (page 153).</p> <p>Added Get support to HDR DDTM Preference (page 169).</p> <p>Removed value of 0x0002 (“No change in preference”).</p> | <p>Interface Specification for CDMA Control and Status Generic Objects [Object Revision 4].doc, rev 1.4 (Filehold date 10aug09)</p> |
| | <p>New objects:</p> <ul style="list-style-type: none"> • SMS Receive Status (r2) and Storage Capacity (page 241) • SMS Read (r2) (page 242) • SMS Delete (r2) (page 245) • SMS Full (page 241) <p>Renamed:</p> <ul style="list-style-type: none"> • SMS Receive Status to SMS Receive Status (r1) • SMS Read to SMS Read (r1) • SMS Delete to SMS Delete (r1) <p>Edits to Appendix A: SMS Overview (page 293) (related to support of r2 of SMS objects). Added Sample SMS read sequence (for r2 objects) (page 307) and Sample SMS delete sequence (for r2 objects) (page 310); figure: “SMS delete: CnS message flow (for r2 objects)” (page 303).</p> | <p>“Interface Specification for CDMA Control and Status SMS Objects [Object Revision 2]” (Filehold date 16mar10)</p> |
| | <p>Supported Object Releases (page 58): added 0x07 through 0x0A; adjusted the related objects (entry in their “Version information” table).</p> | <p>Interface Specification for CWE Control and Status Objects [Object Revision 14]</p> |

| Version | Summary of changes | Internal document |
|---------|--|-------------------|
| | Applied new look-and-feel (document formatting). Removed reference to document 2131024 (FKA Linux CnS Reference) | |
| 2 | Fixed inaccuracy for: <ul style="list-style-type: none">• Audio Profile (page 78) and Audio Mode (page 107): removed external PCM settings.• Clear GPS Assistance Data (page 286): changed the acknowledgement (Ack). Removed products that have reached end-of-life (Compass 597, AirCard 595, 595U, 597E, MC 5725, 5725V, 5727, 5727V). Changes to the trademarks section. | |

Table of Contents

| | |
|---|-----------|
| 1: About This Guide | 18 |
| Introduction | 18 |
| Document structure | 18 |
| Currency | 20 |
| Upgrading..... | 20 |
| References | 20 |
| Terminology and acronyms..... | 21 |
| Conventions used in this document | 21 |
| Reference tables | 21 |
| 2: Overview | 23 |
| Core Wireless Engine (CWE) | 23 |
| Modem features..... | 23 |
| Communication methodology | 23 |
| HIP and CnS | 23 |
| 3: HIP Basics | 24 |
| HIP overview..... | 24 |
| HIP format..... | 24 |
| Packet framing | 25 |
| Escaping | 25 |
| HIP interface rules | 26 |
| Message types..... | 26 |
| Requests | 26 |
| Responses | 26 |
| Notifications..... | 27 |
| HIP message support..... | 27 |
| HIP message reference | 27 |
| Modem CnS notification | 27 |
| Host CnS notification..... | 28 |
| 4: CnS Basics | 29 |
| CnS overview..... | 29 |
| Features | 29 |
| Conversations..... | 29 |
| Request and response | 29 |
| Notification | 30 |
| Sequences | 30 |

| | |
|---|-----------|
| Request and Response..... | 31 |
| Request and Error..... | 31 |
| Notifications..... | 31 |
| CnS format..... | 32 |
| HIP encapsulation..... | 33 |
| Reference conventions..... | 34 |
| Reference tables..... | 34 |
| 5: Numeric Reference | 36 |
| Introduction..... | 36 |
| 6: Hardware Characteristics..... | 45 |
| Introduction..... | 45 |
| Hardware summary..... | 45 |
| Hardware reference..... | 47 |
| Heartbeat..... | 47 |
| Firmware Version..... | 48 |
| Firmware Date..... | 49 |
| Boot Version..... | 49 |
| Radio Power..... | 50 |
| Radio Temperature..... | 52 |
| Radio Supply Voltage..... | 53 |
| Sierra Wireless Modem Type..... | 54 |
| NVRAM Backup/Restore..... | 56 |
| Supported Object Releases..... | 58 |
| Boot Build Date..... | 62 |
| Headset Icon..... | 63 |
| Reset Device..... | 63 |
| Interface Unlock..... | 64 |
| NVRAM Update from File..... | 65 |
| Interface Set Lock..... | 66 |
| Current Radio Band..... | 67 |
| Reset Device Data Statistics..... | 68 |
| Fetch Device Data Statistics..... | 69 |
| Audio Profile..... | 78 |
| Device Time..... | 81 |
| PRI Information..... | 82 |
| Flash Image Information..... | 83 |
| Removable Media Configuration..... | 87 |
| Remote Diagnostics Agent..... | 88 |
| SWOC Configuration..... | 89 |
| ESN..... | 91 |
| PRL Version..... | 92 |

| | |
|---|------------|
| Reset Modem | 92 |
| Cumulative Call Duration | 93 |
| Power Down Modem..... | 93 |
| ERI File Name..... | 94 |
| ERI Write | 95 |
| Get/Set PRL Size | 96 |
| PRL Write..... | 97 |
| Hardware Temperature | 98 |
| Temperature Threshold..... | 99 |
| Modem Too Hot Notification | 99 |
| Wakeup Enable..... | 100 |
| Wakeup Reason..... | 101 |
| 7: Configuration | 103 |
| Introduction | 103 |
| Configuration summary | 103 |
| Configuration reference | 104 |
| Get/Set Mic Codec Gain (Mic Level)..... | 104 |
| Get/Set Speaker Codec Gain (Speaker Level)..... | 105 |
| Get/Set Firmware Autoconnect..... | 106 |
| IPR2 | 106 |
| Audio Mode..... | 107 |
| Get/Set IOTA Feature | 108 |
| 8: Device Management | 109 |
| Introduction | 109 |
| Device Management summary..... | 109 |
| Device Management reference | 110 |
| DM Configuration | 110 |
| Start DM Session | 112 |
| Cancel DM Session..... | 113 |
| DM Session State | 114 |
| Network-Initiated Alert..... | 118 |
| DM DL Progress..... | 120 |
| DM Download Alert | 121 |
| DM Download Description..... | 123 |
| DM Download — Installation Status | 124 |
| 9: Account Management..... | 127 |
| Introduction | 127 |
| Account summary | 127 |
| Account reference..... | 128 |
| Phone Number | 128 |

| | |
|--|------------|
| Activation Status | 128 |
| Set MSL/OTSL Activation Validation Challenge: Step 1 of 2..... | 129 |
| Set MSL/OTSL Activation Validation Challenge: Step 2 of 2..... | 130 |
| Terminate Activation | 131 |
| Change Active NAM..... | 132 |
| Get Active NAM..... | 132 |
| Get/set NV Item..... | 133 |
| Get Activation Date | 137 |
| Get/Set Mobile Directory Number | 138 |
| Get/Set Mobile MIN..... | 139 |
| 10:Network Information | 140 |
| Introduction | 140 |
| Radio summary..... | 140 |
| Network summary..... | 141 |
| Radio reference | 142 |
| RSSI..... | 142 |
| Channel Number | 143 |
| Channel State | 144 |
| Ec/Io Values | 144 |
| RSSI and Ec/Io for CDMA and HDR..... | 145 |
| HDR RSSI | 146 |
| Frame Error Rate | 146 |
| Current Band Class..... | 147 |
| HDR Sector | 148 |
| HDR Channel | 148 |
| HDR RF Statistics | 149 |
| Network reference | 150 |
| Service Indication..... | 150 |
| Roaming Status..... | 151 |
| Protocol Revision Number | 152 |
| Current CDMA System Time..... | 153 |
| HDR Miscellaneous Notification Status | 154 |
| HDR Roam Status..... | 155 |
| HDR Service State | 156 |
| HDR Hybrid Preference | 157 |
| HDR Idle Digital Mode..... | 158 |
| System Selection | 159 |
| SID Value | 160 |
| NID Value..... | 161 |
| Base Station Information..... | 161 |
| HDR Protocol Revision Number | 162 |
| 11:Data Services | 163 |

| | |
|--|------------|
| Introduction | 163 |
| Data summary | 163 |
| Data reference | 166 |
| Get/Set Answer Mode | 166 |
| Mobile IP Error Code..... | 167 |
| HDR DDTM Preference | 169 |
| Close HDR Session | 170 |
| Bytes Sent and Received..... | 170 |
| Dormant Packet Call | 171 |
| KB Sent and Received..... | 172 |
| HDR Tech Notification Status | 172 |
| HDR State | 174 |
| HDR Session Info..... | 175 |
| HDR Authentication Status | 176 |
| HDR DRC Value | 176 |
| HDR DRC Cover | 177 |
| HDR RRI | 178 |
| HDR Single-user Forward statistics | 179 |
| HDR Single-User Forward Traffic Channel DRC packet termination slot count..... | 180 |
| HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count..... | 182 |
| HDR Multi-user Forward statistics | 183 |
| Current personality data..... | 186 |
| Negotiated protocols and application subtypes | 187 |
| Set Dormant..... | 189 |
| Determining which service indicator to display..... | 190 |
| 12:Voice Call Management..... | 192 |
| Introduction | 192 |
| Call management summary..... | 192 |
| Headset controls..... | 193 |
| Dialing (DTMF) controls..... | 194 |
| Voice management reference | 195 |
| Get/Set E911 Status | 195 |
| Call Notification Status | 196 |
| Call Duration | 197 |
| Incoming Call | 197 |
| Answer Incoming Call | 199 |
| Caller ID Information Received | 199 |
| Originate Call | 200 |
| Extended Caller ID Information..... | 200 |
| Call Connected..... | 201 |
| Connection State..... | 203 |

| | |
|--|------------|
| Call Disconnected | 204 |
| Disconnect All Calls | 206 |
| Connection Error | 207 |
| Send Flash | 209 |
| Call Connecting | 210 |
| Call Privacy | 211 |
| OTASP State | 212 |
| Headset control reference | 213 |
| Get/Set Speaker Volume | 213 |
| Mute/Unmute Speaker | 214 |
| Mute/Unmute Mic | 214 |
| Get/Set Tone Level | 215 |
| Mute/Unmute DTMF Tones | 216 |
| Get/Set Echo Cancellation Level | 216 |
| Enable/Disable TTY Option | 217 |
| Sidetone Gain Level | 218 |
| Dial control reference | 219 |
| Play Specified DTMF Tone | 219 |
| Get/Set Default DTMF Tone Duration | 220 |
| Send Out Pressed Key | 221 |
| Set All Keys Up | 222 |
| Generate DTMF Burst | 223 |
| DTMF Playback Mode | 224 |
| 13:SMS Messages | 225 |
| Introduction | 225 |
| Terminology | 225 |
| Summary | 226 |
| Data structures | 226 |
| SMS packet header | 226 |
| SMS header | 228 |
| Reference | 232 |
| SMS Receive Status (r1) | 232 |
| SMS Read (r1) | 233 |
| SMS Delete (r1) | 235 |
| SMS Write | 236 |
| SMS Send | 237 |
| SMS Send Status | 238 |
| SMS Voice Message Delete | 240 |
| SMS Full | 241 |
| SMS Receive Status (r2) and Storage Capacity | 241 |
| SMS Read (r2) | 242 |
| SMS Delete (r2) | 245 |

| | |
|---|------------|
| 14:IOTA Messages | 247 |
| Introduction | 247 |
| IOTA messages summary | 247 |
| IOTA reference | 247 |
| Start IOTA Session | 247 |
| Stop IOTA Session..... | 248 |
| IOTA Status..... | 249 |
| IOTA WAP Push | 251 |
| 15:Location Based Services | 252 |
| Introduction | 252 |
| Types of LBS clients | 252 |
| LBS messages summary..... | 252 |
| LBS reference..... | 254 |
| Location notification status..... | 254 |
| Download notification status | 255 |
| Parameter notification status | 256 |
| PD download..... | 257 |
| PD get position..... | 260 |
| LBS status | 262 |
| PD track | 263 |
| PD end session | 265 |
| PA parameter settings..... | 266 |
| Position Fix Data | 267 |
| Position Fix Session Done | 271 |
| Position Fix Session End | 272 |
| Position Fix Session Begin | 274 |
| PD Update Failed..... | 274 |
| Download Data Session Begin | 275 |
| Download Data..... | 275 |
| Download Data Session Done | 277 |
| Download Data Session End | 277 |
| TCP/IP Address | 278 |
| GPS Lock | 279 |
| Transport Mechanism | 280 |
| Port ID | 281 |
| Privacy Level..... | 282 |
| Base Station information | 283 |
| Network Access Permissions..... | 284 |
| Satellite Information | 285 |
| Clear GPS Assistance Data | 286 |
| 16:Additional Features | 288 |

| | |
|--|------------|
| Introduction | 288 |
| Feature summary..... | 288 |
| Reference | 289 |
| Modem Lock Status | 289 |
| Lock/Unlock Modem..... | 290 |
| Set Locking PIN | 290 |
| Enable/Disable Power-Up Auto-Lock..... | 291 |
| Scratchpad | 292 |
| 17:Appendix A: SMS Overview | 293 |
| SMS overview..... | 293 |
| Incoming messages | 293 |
| Voice mail notification | 294 |
| Outgoing messages | 295 |
| Segmentation..... | 296 |
| Terminology | 296 |
| SMS packet types | 297 |
| Managing segmentation..... | 297 |
| Read segment acknowledgement..... | 297 |
| CnS SMS packet structure..... | 298 |
| SMS packet header..... | 299 |
| SMS packet payload | 299 |
| Receiving SMS messages | 300 |
| Sending SMS messages..... | 311 |
| 18:Appendix B: LBS Message Flow | 313 |
| Mobile Originated Single Position Fix Session..... | 313 |
| Mobile Originated Position Tracking Session..... | 314 |
| 19:Appendix C: PRL Message Flow | 315 |
| Writing a PRL..... | 315 |
| 20:Appendix D: Objects That Have Been Replaced, and May Be Removed From Future Releases | 316 |
| 21:Appendix E: ASCII Table | 317 |
| 22:Appendix F: Glossary | 318 |
| 23:Index | 326 |
| General index | 326 |
| By object name..... | 342 |

Tables

| | |
|---|-----|
| Table 1: Sample object description | 35 |
| Table 2: Numeric object listing | 36 |
| Table 3: Modem device characteristics | 45 |
| Table 4: Error codes for Set ERI File Name | 94 |
| Table 5: Error codes for PRL Write | 97 |
| Table 6: Configuration messages..... | 103 |
| Table 7: Device Management messages | 109 |
| Table 8: Result codes for the last DM session..... | 116 |
| Table 9: Download status codes | 124 |
| Table 10: Account management messages | 127 |
| Table 11: Radio signal information..... | 140 |
| Table 12: Network information | 141 |
| Table 13: Signal strength interpretations..... | 143 |
| Table 14: Data call management | 163 |
| Table 15: MIP error codes..... | 167 |
| Table 16: Determining which service indicator to display | 190 |
| Table 17: Voice call management..... | 192 |
| Table 18: Headset controls | 193 |
| Table 19: DTMF controls..... | 194 |
| Table 20: Call types | 198 |
| Table 21: Call State | 202 |
| Table 22: End Reason | 204 |
| Table 23: OTASP State..... | 212 |
| Table 24: SMS message handling objects | 226 |
| Table 25: SMS header structure | 228 |
| Table 26: SMS send status Cause Codes..... | 239 |
| Table 27: IOTA messages | 247 |
| Table 28: LBS messages | 252 |
| Table 29: PD Error Type values | 258 |
| Table 30: Location fix methods | 260 |
| Table 31: Position Uncertainty | 269 |
| Table 32: End Status values | 272 |
| Table 33: PA Error Type values | 278 |
| Table 34: Additional feature messages | 288 |
| Table 35: Objects that have been replaced, and may be removed from future releases..... | 316 |
| Table 36: Acronyms..... | 318 |

Figures

| | |
|---|-----|
| Figure 1: SMS retrieve and delete flowchart..... | 301 |
| Figure 2: SMS retrieve and delete: CnS message flow (for r1 objects)..... | 302 |
| Figure 3: SMS retrieve: CnS message flow (for r2 objects) | 303 |
| Figure 4: SMS delete: CnS message flow (for r2 objects) | 304 |
| Figure 5: LBS single fix session | 313 |
| Figure 6: LBS tracking session..... | 314 |
| Figure 7: Writing a PRL..... | 315 |

1: About This Guide

Introduction

This guide provides a description and reference for using the Host Interface Protocol (HIP) and Control and Status language (CnS) on the [CDMA 1X/1xEV-DO AirCard 402 ExpressCard/PC Card](#), USB 598 modems, and MC5728V MiniCard modem. These are proprietary [protocols](#) for managing the control and status of the modem. Unless specified otherwise, the word “modem” applies to all of these Sierra Wireless products.

HIP is an encapsulating [protocol](#) intended to carry a variety of other protocol [packets](#) across a single link layer. **CnS** is one of the types of packets that can be carried inside HIP.

This document provides information on how the modem transmits and receives control and status messages. Also provided is some explanation of how the host computer communicates with the modem, the types of transfers, and different transaction sequences. There is also a detailed description of each of the supported CnS messages.

Document structure

This reference covers more than 150 CnS message objects. There are two methods to locate a message of interest:

- By desired task—to find a message that performs a desired task, such as generating a DTMF tone.
- By object number—when a specific message is being sent or received and you need to find out what the message means.

Most readers need to learn about the messages required to perform a desired task. Therefore this reference divides the messages up into chapters related to classes of activity:

- [Hardware characteristics](#) (page 45)
- [Configuration of the hardware](#) (page 103)
- [Account activation and management](#) (page 127)
- [Radio and network characteristics](#) (page 140)
- [Data and fax operations](#) (page 162)
- [Voice call, headset, and dial \(DTMF\) management](#) (page 192)
- [SMS messaging](#) (page 225)
- [IOTA](#) (page 247)
- [Location Based Services](#) (page 252)
- [Miscellaneous special features](#) (PIN security and the scratchpad) (page 288)

To accommodate the second method—looking up a message by its object number—[Table 2](#) on page 36 lists all objects by number, with PDF links and page number references.

Before getting into the details of the messages, this reference provides chapters on the general structure and rules for the required communication [protocols](#).

This document is divided into these chapters:

- [About This Guide](#) (this chapter) —Provides introductory information on the subject, including the [firmware](#) it is current with and typographic conventions used in the document.
- [Overview](#)—Introduces the CDMA [1X/1xEV-DO](#) products: details on how the modems transmit and receive data wirelessly over the [CDMA](#) network, how the host computer communicates to the modems, the types of transfers using the modems, and different transaction sequences (page 23).
- [HIP Basics](#)—Describes the Host Interface Protocol. This is the protocol for encapsulating other protocols to and from the modem. This chapter also includes a reference of the message types you may use in this protocol (page 24).
- [CnS Basics](#)—Provides the general format and process of communicating with Wireless Interface Control Language (page 29).
- [Numeric Reference](#)—Provides a summary of the [CnS](#) messages in object number order. This allows you to locate the description of a message given only an object ID (page 36).
- [Hardware Characteristics](#)—Describes the control and status messages used to determine elements of the hardware characteristics and control fundamental hardware elements (such as resets) (page 45).
- [Configuration](#)—Details the CnS messages related to configuring the modem for particular integrations and for conducting tests. These are not used in end-user applications (page 103).
- [Device Management](#)—CnS messages related to the device management (DM) protocol specified by the Open Mobile Alliance (OMA) (page 109).
- [Account Management](#)—CnS messages related to managing CDMA accounts and activations (page 127).
- [Network Information](#)—CnS messages related to information about the network currently providing service to the modem (such as signal strength, [roaming](#), base station location, and type of service available) (page 140).
- [Data Services](#)—CnS messages related to managing data and fax connections (page 163).
- [Voice Call Management](#)—CnS messages related to managing voice calls are described in this chapter (page 192).
- [SMS Messages](#)—CnS messages related to managing Short Message Service traffic (page 225).
- [IOTA Messages](#)—CnS messages related to Internet Over The Air—an automated feature to perform account setup (page 247).
- [Location Based Services](#)—CnS messages related to Location-based services (page 252).

- [Additional Features](#)—Details the CnS messages that apply to miscellaneous modem features, such as PIN security and the scratchpad (page 288).
- [Appendix A: SMS Overview](#) (page 293)
- [Appendix B: LBS Message Flow](#) (page 313).
- [Appendix C: PRL Message Flow](#) (page 315)
- [Appendix D: Objects That Have Been Replaced, and May Be Removed From Future Releases](#) (page 316)
- [Appendix E: ASCII Table](#) (page 317)
- [Appendix F: Glossary](#) (page 318)
- [Index](#) (page 326)

Currency

This document may be revised from time to time as new [firmware](#) features become available. For revisions, please check our web site.

This document is current with the modem [firmware](#) revisions shown in the following table:

| Product | Firmware version |
|------------------------------------|---------------------------|
| AirCard 402 ExpressCard/PC Card | SWI6085_FP_01.00 or newer |
| USB 598 modem | SWI6085_FP_01.00 or newer |
| MC5728V Mini Card | SWI6085_FP_01.00 or newer |

The revision can be determined using the AT command: **AT+GMR**. The modem responds with version information for software, [firmware](#), and hardware.

Upgrading

If your modem [firmware](#) is an earlier version, you can acquire updated firmware by contacting your account manager.

References

For background information on [CDMA](#) technology, visit the CDMA Development Group at: www.cdg.org.

[1] “*Software Integration Guide*” (document 2130759, from Sierra Wireless)

[2] “*Position Determination Service Standard for Dual Mode Spread Spectrum Systems*”, TIA/EIA/IS-801

[3] “*OMA-TS-DM_FUMO-V1_0-20070209-A.doc*” (Firmware Update Management Object), Open Mobile Alliance

Terminology and acronyms

This document makes wide use of acronyms that are in common use in data communications and cellular technology.

Many of the terms and acronyms used in this document are listed in [Appendix F: Glossary](#) (page 318).

Our web site also provides a Glossary (document #2110032) that may be helpful in understanding some acronyms and terminology used in this guide.

Conventions used in this document

Numeric values are decimal unless prefixed as noted below.

Hexadecimal values are shown with a prefix of 0x, that is, in the form 0x3D.

Binary values are shown with a prefix of 0b, that is, in the form 0b00111101.

Color changes are used to aid in identifying components of a string or structure. The colors are not related to specific field types; they change in a simple cycle.

Citations of command names, object values, or particular samples are shown in a bold font: **Heartbeat**.

Parameter and data element samples are generally noted in a sans serif font: Rx-y-z. In this sample, the use of italics and lowercase indicates the item(s) can vary in specifics and should not be taken literally.

Character codes or keystrokes that are described with words or standard abbreviations are shown within angle brackets using a different font: such as <CR> for Carriage Return and <space> for a blank space character. This font is also used to highlight options, technical references such as web sites or e-mail addresses.

Note: You can view this guide online or print it to keep on hand. If you're viewing it online, simply click a topic in the [Table of Contents](#), or any page reference, table reference, section reference, or object name. (Most text that is blue is a clickable link.) The PDF automatically displays the appropriate page.

Reference tables

The conventions used to summarize objects and provide detailed object descriptions are defined in “[Reference conventions](#)” on page 34.

Shaded entries in summary tables are objects that you, as the [OEM](#), would use in configuring the unit at your factory. Your end-users should not have access to these operations, so they should not be used in any user applications.



Individual objects present the syntax in a table showing the object value, the supported message types and the parameters for each type. If the object is for factory configuration (**shaded** in the summary tables) it displays the icon shown at

left.

2: Overview

This edition of the [CnS](#) reference applies only to the Sierra Wireless AirCard402/ MC5728V/ USB 598 CDMA [1X/1xEV-DO](#) modems.

Core Wireless Engine (CWE)

Sierra Wireless has developed the Core Wireless Engine™ (CWE) design to create a family of products that share many characteristics. To this end, there is a core set of CnS messages supported by all these products. However, differences in specific network infrastructures, and specific product features, require that the core set of messages be extended to manage these individual characteristics.

Modem features

The modem supports these primary features:

- [1X](#) and [1xEV-DO](#) packet switched data
- IS-95 circuit switched data
- [Voice](#) (supported only by the MC5728V product)
- [Short Message Service](#) (SMS) (both transmitting and receiving)
- Fax
- [Location Based Services](#)

There are other features related to modem power control, reset, and PIN security that are also covered by CnS.

All wireless products require some form of user activation to allow them to operate with a wireless service provider or carrier. CnS can be used to manage this activation process as well.

Communication methodology

For details on hardware and [port](#) configuration, see the *Software Integration Guide* (document 2130759).

HIP and CnS

To permit easier portability of software and [firmware](#), Sierra Wireless has created the proprietary Host Interface Protocol (HIP). It serves as a standard method to communicate over whatever physical link layer is used by the product. This allows the same applications to access devices using different connection mechanisms.

3: HIP Basics

HIP overview

The Host Interface Protocol (HIP) is a multiplexing layer designed to allow several types of message streams to share the same physical link layer. CnS is only one of the supported streams.

HIP is used to carry control and status data between the modem and the host to provide management of the modem. This management consists of:

- Non-volatile configuration of the modem
- Run-time configuration of the modem
- Status reporting and monitoring of the modem

This protocol does not include checksums. It relies on a high-reliability physical interface between the host and the modem to reduce the processing burden on the microprocessor.

Although only two message types are supported by the modem, it is useful to have a broader view of the protocol to permit you to design your system to be easily ported to other Sierra Wireless modems.

The host must be able to encapsulate CnS messages within HIP and extract the CnS payload from HIP packets received from the modem.

HIP format

A HIP packet has the following structure:

| Byte offset | Content |
|-------------|----------------------------|
| 0 | Framing character (0x7E) |
| 1–2 | Length of optional payload |
| 3 | Message ID |
| 4 | Parameter |
| [5–n] | Optional payload |
| n+1 | Framing character (0x7E) |

The minimum length of a HIP packet is six bytes, for a packet without the optional payload. Note that such a packet would have the first field set to 0x0000—no payload.

Framing All HIP packets open and close with a framing character. This is the HDLC frame character (0x7E). Framing is discussed in more detail in “[Packet framing](#)” on page 25.

Length The maximum length of the optional payload field is 2000 bytes, making the largest HIP packet 2004 bytes total.

Message ID The message ID field is actually composed of a 1-bit flag and 7-bit identifier. The flag bit, if set, indicates that the optional parameter field begins with a pad byte of random data (possibly added to make the data word-aligned). The pad byte is included in the length given in the first field.

The 7-bit message ID itself provides information about the type of HIP packet. Different IDs are used for messages going to, and coming from, the modem. This prevents echoing across the interface.

Parameter The parameter is rarely used, but some messages do carry an informative flag in this field.

Payload The payload varies by message type and purpose.

Packet framing

All HIP packets are variable length. The link layer over the COM port requires marking the HIP packets with start and end flags, and escaping the data within.

The HDLC protocol mechanism is used by the modem; the value of the flag is 0b01111110 (0x7E). Only one flag byte is required for back-to-back packets. Two consecutive flags constitute an empty frame, which is silently discarded.

Escaping

As the framing character (0x7E) can occur within the frame itself, you need to use escaping/un-escaping to distinguish the data value from the frame character.

The escape character is 0x7D. Since this character is also treated as part of the protocol process, it too must be escaped if it appears in the data stream.

If either of the values 0x7D or 0x7E occur in the frame, they must be replaced by two (2) bytes: 0x7D (the escape character) and then the data character exclusive-ORed with 0x20.

Therefore a binary data stream of:

0x30 0x37 0x7E 0x65 0x7D 0x66

would be sent as:

0x30 0x37 0x7D 0x5E 0x65 0x7D 0x5D 0x66

In the above example, the offending frame character 0x7E is replaced by the escape character 0x7D followed by the exclusive-or result of 0x7E with 0x20 (0x5E). Since 0x7D is the escape flag and also occurs in the sample above, it too must be replaced using the same method.

HIP interface rules

If either side of the interface receives a message containing an undefined message ID, the message is quietly discarded.

If the maximum length of a message (2004 bytes) is ever exceeded, the receiver discards the entire packet and ignores the message.

All multi-byte fields used in this protocol are in network byte order (Big-Endian). That is, the most significant byte is sent first, followed by less significant bytes. The least significant byte of a multi-byte field is sent last. Note that this differs from the way Intel processors normally store multi-byte fields; special methods must be used to write multi-byte fields.

The host never sends a message to the modem with the pad byte flag set (in the message ID). The modem ignores the pad byte flag.

Message types

There are three fundamental types of HIP messages that flow between the host and the modem:

- Requests
- Responses
- Notifications

Requests

Note: *The modem does not currently support the HIP request message types.*

A request message provides a means for the host (master) to ask the modem (slave) to either perform an operation or return a value. The slave always responds to a request message with a response message. With this master/slave relationship, the modem is not allowed to make requests of the host. One, and only one, response is sent for every request.

Note that, to provide flow control to the modem, a response to a specific request must be received before a second request of the same type can be issued. This effectively provides stop-and-wait flow control on a per request basis. If the host never receives a response to a request (timeout), this is a failure case, and thus the host can assume that the modem has failed and should be recovered from a restart state.

Request message IDs are in the range of 00–31 (0x00–0x1F).

Responses

Note: *The modem does not currently issue any HIP response message types.*

A response message is always sent from the modem, in reply to a request from the host. The format and type of data in the response varies according to the type of response. Responses can never be sent unsolicited (that is, every response is paired with a specific request).

Response message IDs are in the range of 64–95 (0x40–0x5F).

Notifications

Notification messages provide a means to send unsolicited data from either side of the interface.

Notifications are used when no reply or return data is required from the receiver (or conversations do not require stop-and-wait flow control).

No acknowledgement or reply is ever sent by the receiver of a notification. The encapsulated payload [protocol](#) may require acknowledgements or replies (as many [CnS](#) messages do), but this is outside of the HIP specification.

Notification message IDs sent from the host are in the range 32–63 (0x20–0x3F). Notifications from the modem have a message ID in the range of 96–127 (0x60–0x7F).

More information on notifications is provided on page [30](#).

HIP message support

The only type of HIP messages supported by the modem is in the notification category. All CnS messages are treated as asynchronous events and can originate from either side (host or modem).

Although CnS may also enforce a request and reply transaction, the HIP encapsulation of CnS is independent, and has no knowledge of the payload content. Therefore, from the point of view of HIP, the CnS conversation is handled by asynchronous notification messages.

HIP message reference

There are only two HIP message IDs supported by the modem: the CnS notifications to and from the modem. The CnS message payload of these [packets](#) is the subject of the following chapters.

Modem CnS notification

The modem [CnS](#) notification is an unsolicited message sent by the host to the modem for transport of CnS messages.

| | |
|----------------|-------------|
| ID | 0x2B (43) |
| Parm | None |
| Payload | CnS message |

Host CnS notification

The host CnS notification is an unsolicited message sent by the modem to the host for transport of CnS messages.

| | |
|----------------|-------------|
| ID | 0x6B (107) |
| Parm | None |
| Payload | CnS message |

4: CnS Basics

CnS overview

Control and Status (CnS) language is a defined “dictionary” of objects and parameters combined with a “grammar” of message structure and conversation. It allows an application to:

- Query the modem for status
- Set parameters and configuration of the modem
- Control the traffic of event notifications from the modem
- Receive event notifications

Features

- Supports binary data in variable length messages
- Supports request and reply
- Supports asynchronous notifications

Conversations

CnS conversations fall into one of two categories:

- Request-and-response, in which the host issues a command or request (poll) and the modem replies (response)
- Notification, a monologue in which the modem forwards an information report without a specific request

Request and response

This allows an application to poll the modem and issue commands to control the modem. A request is sent to the modem and a response is returned by the modem to acknowledge the request. Depending on the context of the message, the response may contain data, such as status information, or it may simply be an acknowledgement that the requested command was received.

Some requests may yield an error response. If the response has set bit 7 of the Operation Type (described later), the operation failed. The parameter may include an explanation for the failure.

Acknowledgements to Set commands generally return the value of the setting. They are issued after execution. Any exceptions are indicated in the notes within each command’s description.

Notification

Notifications consist of unsolicited, asynchronous, status updates that are forwarded to the host from the modem based on events. Upon power-up, notifications are always disabled. Forwarding notifications must be explicitly enabled using the request and response mechanism. Different types of information can be automatically forwarded to the host. The criterion for an unsolicited update depends on the specific object being forwarded.

CnS notifications are classified as high priority or low priority. **High priority notifications** are those that, once enabled, are always forwarded to the host regardless if it is asleep or not. These notifications do wake up the host if it is asleep. **Low priority notifications**, once enabled, are suppressed if host is asleep, but forwarded as soon as host wakes up, containing current data. CnS objects capable of high priority notifications are noted in the sections that follow.

In addition to notification classification, the modem supports three host power-save configurations:

- Host is always awake; thus, all enabled notifications are always forwarded
- Host is in power-save mode, but wants to receive all notifications; modem reduces the frequency of notifications but does not suppress any
- Host is fully asleep; thus, only high priority notifications are forwarded to the host, and low priority events are suppressed as described above

This configuration is controlled via **PRI** and is permanently stored in the modem's **NV** memory.

Unless otherwise stated in the individual command references, the modem reviews the status of most objects at two-second intervals. If the value of an object has changed, and notification is enabled for that object, then a notification message is sent according to the above stated power-save rules regarding host and notification priority. If an object value is unchanged, no notification is generated.

Notifications stop if the host sends the Forward Stop Request for the particular object.

Note: A modem reset or power cycle disables notifications for all object types. Desired notifications have to be re-enabled.

Sequences

Conversation sequences include:

- Request and response
- Request (or notification enable) and error
- Notification enable, notification **packets**, notification disable, and disable acknowledge.

Request and Response

| Step | Host to Modem | Modem to Host |
|------|--------------------------------|----------------------------------|
| 1 | Get (request) or Set (command) | - |
| 2 | - | Reply (Get) or Acknowledge (Set) |

Request and Error

| Step | Host to Modem | Modem to Host |
|------|---|----------------|
| 1 | Get (request) or Set (command) or Notification Enable or Notification Disable | - |
| 2 | - | Error Response |

Notifications

| Step | Host to Modem | Modem to Host |
|------|----------------------|---------------------|
| 1 | Notification Enable | - |
| 2 | - | Enable Response |
| 3 | - | Notification |
| 4 | - | Notification |
| ... | - | ... |
| n | - | Notification |
| n+1 | Notification Disable | - |
| n+2 | - | Disable Acknowledge |

CnS format

All **CnS** sentences are composed of the following elements:

| Byte offset | Content |
|-------------|--------------------------------------|
| 0-1 | Object ID |
| 2 | Operation Type |
| 3 | Reserved |
| 4-7 | Application ID |
| 8-9 | Length of payload (0–246) |
| [10-255] | Parameter (if needed for the object) |

The maximum length of a CnS message is 255 bytes. The minimum length is 10.

Object ID This is the object (noun) that the message relates to. Object IDs are expressed as hex numbers.

Operation Type This is the verb of the message. It indicates whether the message is a Get (poll/request), Set (command), reply (to a Get), acknowledge (to a Set), error, or notification. The supported types are shown in the following table.

| Code | Operation type | Direction |
|-------|--------------------------------|--------------|
| 0x01 | Get request (poll) | Host ⇨ Modem |
| 0x02 | Reply to Get (with data) | Host ⇐ Modem |
| 0x03 | Set (command) | Host ⇨ Modem |
| 0x04 | Acknowledge to Set | Host ⇐ Modem |
| 0x05 | Notification Enable | Host ⇨ Modem |
| 0x06 | Enable Response | Host ⇐ Modem |
| 0x07 | Notification | Host ⇐ Modem |
| 0x08 | Notification Disable | Host ⇨ Modem |
| 0x09 | Acknowledge to Forward Disable | Host ⇐ Modem |
| 0x80* | Error response | Host ⇐ Modem |

*Errors are noted by ORing the original Operation Type with 0x80 (setting bit 7). This preserves the original operation type in the error response.

Application ID This is intended to allow the application to assign a number (perhaps indicating the process or thread that originated a request). Whatever value is used in this field will appear in the subsequent response from the modem. If a notification is not in response to a specific poll, the field contains zeroes (the modem does not retain the original enabler's ID).

Length Only the length of an object parameter is needed. This field is zero if there is no parameter. Poll responses and notifications use this to identify the length of the data being returned.

Parameter Any data associated with the object is carried here. This is used in Set commands (and some acknowledgements), some Get requests and replies to Get requests, and notifications. Error responses may use this to convey the reason for a failure.

HIP encapsulation

All CnS messages are encapsulated in a HIP [packet](#) to be sent over the physical link. Thus the complete packet sent to the modem, including the HIP [header](#), is in the following form:

| Byte offset | Content |
|-------------|---|
| 0 | Frame character (0x7E) |
| 1–2 | Length of CnS format packet including parameter length (10–255) |
| 3 | Message ID: 0x2B—for host to modem 0x6B—for modem to host |
| 4 | Parameter 0x00—no HIP parameter for CnS messages |
| 5-6 | CnS Object ID |
| 7 | Operation Type |
| 8 | Reserved (0x00) |
| 9–12 | Application ID |
| 13 | Reserved (0x00) |
| 14 | Length of CnS parameter (0–245) |
| [15–259] | CnS Parameter (if needed for the object) |
| | Frame character (0x7E) |

This reference covers only the material from offset 4 onward—the actual CnS message content. The HIP [header](#) is always constructed from the length of the

CnS message, the ID type (either host to modem or modem to host), and a zero for the HIP parameter.

Reference conventions

Reference tables

Each reference chapter begins with a summary table, identifying the objects in the chapter and which message types are supported by it: Set (S), Get (G), and Notification (N).

| Object | ID | Description | S | G | N |
|------------|--------|---|---|---|---|
| Scratchpad | 0x5000 | Read or write to the scratchpad | ✓ | ✓ | ✗ |
| IPR2 | 0x5001 | Set or read the data rate on the secondary port | ✓ | ✓ | ✗ |

Shaded entries are objects that you, as the **OEM**, would use in configuring the unit at your factory. Your end-users should not have access to these operations, so they should not be used in any user applications.

Individual objects present the syntax in a table showing the object value, the supported operation types and the parameters.

The “Operations” entry in the table shows, with checkmarks (✓) and crosses (✗), the supported Operation Types for the object. Where Get or Set operations are supported, the corresponding replies and acknowledgements are also supported. Where notifications are supported, all five types (enable, response to enable, notification, disable, and response to disable) are supported.

The parameter section provides columns for Length of the parameter block, then the offset in the block for each field of the parameter, the data type of the field, and the description of each field.

Table 1: Sample object description

| | | | | |
|-----------------------------------|---------------|--------------------|-----------------|----------------------|
| Object ID | 0x2000 | SMS Receive Status | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 6 | 0 | UINT16 | Urgent message count |
| | | 2 | UINT16 | Normal message count |
| | | 4 | UINT16 | Voice message count |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

In the sample above:

- A host request (**0x01** message type) does not have parameters, so the CnS parameter length is 0.
- When the modem replies to the Get request, it provides a parameter block with three fields. The length of the parameter block is 6 bytes. Each field is listed in order, with the offset within the parameter block.
- The notification parameters appear in both message type **0x06** (Notification Enable Acknowledge) and **0x07** (Notification).

Unless noted otherwise, the parameters for the notification types **0x06** and **0x07** are the same as those for the reply (**0x02**) to a Get request. There are no parameters associated with the other notification types: **0x05** (enable), **0x08** (disable), and **0x09** (acknowledge).

Object version (in the “Version information” table): Each CnS object has a version number, which gets incremented in a firmware release if the structure for that particular object has changed.

Object set Release: All of the CnS objects of a certain type in a firmware release comprise an Object Set (in the sample table above – the CDMA SMS object set), which has an associated version number. For example, the CDMA SMS object set Release number is incremented in a firmware release, if a new object is added to the CDMA SMS object set.

For more information on object sets, see [Supported Object Releases](#) (page 58).

5: Numeric Reference

Introduction

This chapter provides a cross-reference listing of all the **CnS** messages supported by the modem.

Message objects are listed in numeric order by object ID.

Table 2: Numeric object listing

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|--|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x0000 | Heartbeat | 47 | N/a | CWE 1 | CWE 1 |
| 0x0001 | Firmware Version | 48 | N/a | CWE 1 | CWE 1 |
| 0x0002 | Firmware Date | 49 | N/a | CWE 1 | CWE 1 |
| 0x0004 | Boot Version | 49 | N/a | CWE 1 | CWE 1 |
| 0x0005 | See "Appendix D: Objects That Have Been Replaced, and May Be Removed From Future Releases" | 316 | N/a | | |
| 0x0006 | See "Appendix D: Objects That Have Been Replaced, and May Be Removed From Future Releases" | 316 | N/a | | |
| 0x0007 | Radio Power | 50 | 2 | CWE 3 | CWE 3 |
| 0x0008 | Radio Temperature | 52 | 1 | CWE 3 | CWE 3 |
| 0x0009 | Radio Supply Voltage | 53 | 1 | CWE 3 | CWE 3 |
| 0x000A | Sierra Wireless Modem Type | 54 | 1 | CWE 3 | CWE 3 |
| 0x000B | NVRAM Backup/Restore | 56 | 1 | CWE 3 | CWE 3 |
| 0x000C | Supported Object Releases | 58 | 3 | CWE 14 | CWE 3 |
| 0x000D | Boot Build Date | 62 | 1 | CWE 3 | CWE 3 |
| 0x000F | Headset Icon | 63 | 1 | CWE 3 | CWE 3 |
| 0x0010 | Reset Device | 63 | 1 | CWE 3 | CWE 3 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|-----------------------------------|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x0011 | NVRAM Update from File | 65 | 1 | CWE 3 | CWE 3 |
| 0x0012 | Interface Unlock | 64 | 1 | CWE 3 | CWE 3 |
| 0x0015 | Interface Set Lock | 66 | 1 | CWE 3 | CWE 3 |
| 0x0016 | Current Radio Band | 67 | 1 | CWE 3 | CWE 3 |
| 0x0018 | Reset Device Data Statistics | 68 | 1 | CWE 4 | CWE 4 |
| 0x0019 | Fetch Device Data Statistics | 69 | 1 | CWE 4 | CWE 4 |
| 0x001A | Audio Profile | 78 | 2 | CWE 10 | CWE 6 |
| 0x001B | Device Time | 81 | 1 | CWE 10 | CWE 10 |
| 0x0023 | PRI Information | 82 | 1 | CWE 10 | CWE 10 |
| 0x0024 | Flash Image Information | 83 | 1 | CWE 10 | CWE 10 |
| 0x0025 | Removable Media Configuration | 87 | 1 | CWE 10 | CWE 10 |
| 0x0026 | Remote Diagnostics Agent | 88 | 1 | CWE 10 | CWE 10 |
| 0x0027 | SWOC Configuration | 89 | 1 | CWE 12 | CWE 12 |
| 0x0E00 | DM Configuration | 110 | 1 | DM 1 | DM 1 |
| 0x0E01 | Start DM Session | 112 | 1 | DM 1 | DM 1 |
| 0x0E02 | Cancel DM Session | 113 | 1 | DM 1 | DM 1 |
| 0x0E03 | DM Session State | 114 | 1 | DM 1 | DM 1 |
| 0x0E04 | Network-Initiated Alert | 118 | 1 | DM 1 | DM 1 |
| 0x0E10 | DM DL Progress | 120 | 1 | DM 2 | DM 2 |
| 0x0E11 | DM Download Alert | 121 | 1 | DM 2 | DM 2 |
| 0x0E12 | DM Download Description | 123 | 1 | DM 2 | DM 2 |
| 0x0E13 | DM Download — Installation Status | 124 | 1 | DM 2 | DM 2 |
| 0x0F01 | PD download | 257 | 1 | LBS 1 | LBS 1 |
| 0x0F02 | PD get position | 260 | 1 | LBS 1 | LBS 1 |
| 0x0F03 | LBS status | 262 | 1 | LBS 1 | LBS 1 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|-------------------------------|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x0F04 | PD track | 263 | 1 | LBS 1 | LBS 1 |
| 0x0F05 | PD end session | 265 | 1 | LBS 1 | LBS 1 |
| 0x0F06 | PA parameter settings | 266 | 1 | LBS 1 | LBS 1 |
| 0x0F07 | Location notification status | 254 | 1 | LBS 1 | LBS 1 |
| 0x0F08 | Download notification status | 255 | 1 | LBS 1 | LBS 1 |
| 0x0F09 | Parameter notification status | 256 | 1 | LBS 1 | LBS 1 |
| 0x0F0A | Position Fix Data | 267 | 2 | LBS 2 | LBS 1 |
| 0x0F0B | Position Fix Session Done | 271 | 1 | LBS 1 | LBS 1 |
| 0x0F0C | Position Fix Session End | 272 | 1 | LBS 1 | LBS 1 |
| 0x0F0D | Position Fix Session Begin | 274 | 1 | LBS 1 | LBS 1 |
| 0x0F0E | PD Update Failed | 274 | 1 | LBS 1 | LBS 1 |
| 0x0F0F | Download Data Session Begin | 275 | 1 | LBS 1 | LBS 1 |
| 0x0F10 | Download Data | 275 | 1 | LBS 1 | LBS 1 |
| 0x0F11 | Download Data Session Done | 277 | 1 | LBS 1 | LBS 1 |
| 0x0F12 | Download Data Session End | 277 | 1 | LBS 1 | LBS 1 |
| 0x0F13 | TCP/IP Address | 278 | 1 | LBS 1 | LBS 1 |
| 0x0F14 | GPS Lock | 279 | 1 | LBS 1 | LBS 1 |
| 0x0F15 | Transport Mechanism | 280 | 1 | LBS 1 | LBS 1 |
| 0x0F16 | Port ID | 281 | 1 | LBS 1 | LBS 1 |
| 0x0F17 | Privacy Level | 282 | 1 | LBS 1 | LBS 1 |
| 0x0F18 | Network Access Permissions | 284 | 1 | LBS 1 | LBS 1 |
| 0x0F19 | Base Station information | 283 | 1 | LBS 1 | LBS 1 |
| 0x0F1F | Satellite Information | 286 | 1 | LBS 1 | LBS 1 |
| 0x0F20 | Clear GPS Assistance Data | 286 | 1 | LBS 1 | LBS 1 |
| 0x1000 | ESN | 91 | 1 | CDMA 1 | CDMA 1 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|---|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x1001 | RSSI | 142 | 1 | CDMA 1 | CDMA 1 |
| 0x1002 | Phone Number | 128 | 1 | CDMA 1 | CDMA 1 |
| 0x1004 | Channel Number | 143 | 1 | CDMA 1 | CDMA 1 |
| 0x1005 | Channel State | 144 | 1 | CDMA 1 | CDMA 1 |
| 0x1006 | Service Indication | 150 | 1 | CDMA 1 | CDMA 1 |
| 0x1007 | Roaming Status | 151 | 1 | CDMA 1 | CDMA 1 |
| 0x1008 | PRL Version | 92 | 1 | CDMA 1 | CDMA 1 |
| 0x1009 | Activation Status | 128 | 1 | CDMA 1 | CDMA 1 |
| 0x100A | Ec/Io Values | 144 | 1 | CDMA 1 | CDMA 1 |
| 0x100F | Reset Modem | 92 | 1 | CDMA 1 | CDMA 1 |
| 0x1014 | Get/Set Answer Mode | 166 | 1 | CDMA 1 | CDMA 1 |
| 0x1015 | Set MSL/OTSL Activation Validation Challenge: Step 1 of 2 | 129 | 1 | CDMA 1 | CDMA 1 |
| 0x1016 | Set MSL/OTSL Activation Validation Challenge: Step 2 of 2 | 130 | 1 | CDMA 1 | CDMA 1 |
| 0x1017 | Terminate Activation | 131 | 1 | CDMA 1 | CDMA 1 |
| 0x1018 | Modem Lock Status | 289 | 1 | CDMA 1 | CDMA 1 |
| 0x1019 | Lock/Unlock Modem | 290 | 1 | CDMA 1 | CDMA 1 |
| 0x101A | Set Locking PIN | 290 | 1 | CDMA 1 | CDMA 1 |
| 0x101B | Cumulative Call Duration | 93 | 1 | CDMA 1 | CDMA 1 |
| 0x101C | Protocol Revision Number | 152 | 1 | CDMA 1 | CDMA 1 |
| 0x101E | Change Active NAM | 132 | 1 | CDMA 1 | CDMA 1 |
| 0x101F | Get Active NAM | 132 | 1 | CDMA 1 | CDMA 1 |
| 0x1022 | Power Down Modem | 93 | 1 | CDMA 1 | CDMA 1 |
| 0x1026 | Get/Set Speaker Volume | 213 | 1 | CDMA 1 | CDMA 1 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|--|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x1027 | Mute/Unmute Speaker | 214 | 1 | CDMA 1 | CDMA 1 |
| 0x1028 | Get/Set Mic Codec Gain (Mic Level) | 104 | 1 | CDMA 1 | CDMA 1 |
| 0x1029 | Mute/Unmute Mic | 214 | 1 | CDMA 1 | CDMA 1 |
| 0x102A | Get/Set Tone Level | 215 | 1 | CDMA 1 | CDMA 1 |
| 0x102D | Mute/Unmute DTMF Tones | 216 | 1 | CDMA 1 | CDMA 1 |
| 0x102E | Play Specified DTMF Tone | 219 | 1 | CDMA 1 | CDMA 1 |
| 0x102F | Get/Set Default DTMF Tone Duration | 220 | 1 | CDMA 1 | CDMA 1 |
| 0x1030 | Send Out Pressed Key | 221 | 1 | CDMA 1 | CDMA 1 |
| 0x1031 | Set All Keys Up | 222 | 1 | CDMA 1 | CDMA 1 |
| 0x1032 | Current CDMA System Time | 153 | 1 | CDMA 1 | CDMA 1 |
| 0x1033 | Get/set NV Item | 133 | 1 | CDMA 1 | CDMA 1 |
| 0x1034 | Generate DTMF Burst | 223 | 1 | CDMA 1 | CDMA 1 |
| 0x1035 | Get/Set Echo Cancellation Level | 216 | 1 | CDMA 1 | CDMA 1 |
| 0x1036 | Get/Set Speaker Codec Gain (Speaker Level) | 105 | 1 | CDMA 1 | CDMA 1 |
| 0x1037 | Get Activation Date | 137 | 1 | CDMA 1 | CDMA 1 |
| 0x1038 | Get/Set Mobile Directory Number | 138 | 1 | CDMA 1 | CDMA 1 |
| 0x1039 | Get/Set Mobile MIN | 139 | 1 | CDMA 1 | CDMA 1 |
| 0x1042 | DTMF Playback Mode | 224 | 1 | CDMA 1 | CDMA 1 |
| 0x1043 | Enable/Disable Power-Up Auto-Lock | 291 | 1 | CDMA 1 | CDMA 1 |
| 0x1045 | Get/Set E911 Status | 195 | 1 | CDMA 1 | CDMA 1 |
| 0x1046 | Enable/Disable TTY Option | 217 | 1 | CDMA 1 | CDMA 1 |
| 0x104A | ERI File Name | 94 | 1 | CDMA 1 | CDMA 1 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|--|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x104C | ERI Write | 95 | 1 | CDMA 1 | CDMA 1 |
| 0x1052 | Get/Set PRL Size | 96 | 1 | CDMA 1 | CDMA 1 |
| 0x1054 | PRL Write | 97 | 1 | CDMA 1 | CDMA 1 |
| 0x1056 | Mobile IP Error Code | 167 | 1 | CDMA 1 | CDMA 1 |
| 0x1064 | Get/Set Firmware Autoconnect | 106 | 1 | CDMA 1 | CDMA 1 |
| 0x1065 | RSSI and Ec/Io for CDMA and HDR | 145 | 1 | CDMA 1 | CDMA 1 |
| 0x1066 | HDR Miscellaneous Notification Status | 154 | 1 | CDMA 1 | CDMA 1 |
| 0x1067 | HDR RSSI | 146 | 1 | CDMA 1 | CDMA 1 |
| 0x1068 | HDR Roam Status | 155 | 1 | CDMA 1 | CDMA 1 |
| 0x1069 | HDR Service State | 156 | 1 | CDMA 1 | CDMA 1 |
| 0x106A | HDR Hybrid Preference | 157 | 1 | CDMA 1 | CDMA 1 |
| 0x106B | HDR Idle Digital Mode | 158 | 1 | CDMA 1 | CDMA 1 |
| 0x106C | HDR DDTM Preference | 169 | 1 | CDMA 1 | CDMA 1 |
| 0x106D | Close HDR Session | 170 | 1 | CDMA 1 | CDMA 1 |
| 0x1070 | System Selection | 159 | 1 | CDMA 1 | CDMA 1 |
| 0x2000 | SMS Receive Status (r1) | 232 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2001 | SMS Read (r1) | 233 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2002 | SMS Delete (r1) | 235 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2003 | SMS Write | 236 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2004 | SMS Send | 237 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2005 | SMS Send Status | 238 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2006 | SMS Voice Message Delete | 240 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2007 | SMS Full | 241 | 1 | CDMA SMS 1 | CDMA SMS 1 |
| 0x2008 | SMS Receive Status (r2) and Storage Capacity | 241 | 1 | CDMA SMS 2 | CDMA SMS 2 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|--|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x2009 | SMS Read (r2) | 242 | 1 | CDMA SMS 2 | CDMA SMS 2 |
| 0x200A | SMS Delete (r2) | 245 | 1 | CDMA SMS 2 | CDMA SMS 2 |
| 0x2100 | Start IOTA Session | 247 | 1 | CDMA 1 | CDMA 1 |
| 0x2101 | Stop IOTA Session | 248 | 1 | CDMA 1 | CDMA 1 |
| 0x2102 | IOTA Status | 249 | 1 | CDMA 1 | CDMA 1 |
| 0x2103 | IOTA WAP Push | 251 | 1 | CDMA 1 | CDMA 1 |
| 0x2104 | Get/Set IOTA Feature | 108 | 1 | CDMA 1 | CDMA 1 |
| 0x3000 | Call Notification Status | 196 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3001 | Bytes Sent and Received (in the current data call) | 170 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3003 | Call Duration | 197 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3004 | Incoming Call | 197 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3005 | Answer Incoming Call | 199 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3006 | Caller ID Information Received | 199 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3007 | Originate Call | 200 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3009 | Extended Caller ID Information | 200 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x300A | Call Connected | 201 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x300B | Connection State | 203 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x300C | Call Disconnected | 204 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x300D | Disconnect All Calls | 206 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x300E | Connection Error | 207 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x300F | Send Flash | 209 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3011 | Call Connecting | 210 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3012 | Dormant Packet Call | 171 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3013 | Call Privacy | 211 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x3014 | OTASP State | 212 | 1 | CDMA CALL 1 | CDMA CALL 1 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|---|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x3015 | KB Sent and Received (cumulative byte counter) | 172 | 1 | CDMA CALL 1 | CDMA CALL 1 |
| 0x4001 | SID Value | 160 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4002 | NID Value | 161 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4003 | Base Station Information | 161 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4004 | Hardware Temperature | 98 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4005 | Frame Error Rate | 146 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4008 | Current Band Class | 147 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x401C | HDR Tech Notification Status | 172 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x401D | HDR Protocol Revision Number | 162 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x401E | HDR Sector | 148 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x401F | HDR State | 174 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4020 | HDR Channel | 148 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4021 | HDR Session Info | 175 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4022 | HDR Authentication Status | 176 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4023 | HDR DRC Value | 176 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4024 | HDR DRC Cover | 177 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4025 | HDR RRI | 178 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4026 | HDR RF Statistics | 149 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4027 | HDR Single-user Forward statistics | 179 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4028 | HDR Single-User Forward Traffic Channel DRC packet termination slot count | 180 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4029 | HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count | 182 | 1 | CDMA Tech 1 | CDMA Tech 1 |

| Object | Message | Page | Most current object: | | Object first appeared in object set Release: |
|--------|--|------|----------------------|--------------------------------|--|
| | | | Object version: | Updated in object set Release: | |
| 0x402A | HDR Multi-user Forward statistics | 183 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x402B | Current personality data | 186 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x402C | Negotiated protocols and application subtypes | 187 | 1 | CDMA Tech 1 | CDMA Tech 1 |
| 0x4500 | Modem Too Hot Notification | 99 | 1 | CDMA Temperature Protection 1 | CDMA Temperature Protection 1 |
| 0x4501 | Temperature Threshold | 99 | 1 | CDMA Temperature Protection 1 | CDMA Temperature Protection 1 |
| 0x5000 | Scratchpad | 292 | 1 | CDMA 1 | CDMA 1 |
| 0x5001 | IPR2 | 106 | 1 | CDMA 1 | CDMA 1 |
| 0x5006 | Audio Mode | 107 | N/a | CDMA 1 | CDMA 1 |
| 0x5007 | Wakeup Enable | 100 | 1 | CDMA 1 | CDMA 1 |
| 0x5008 | Wakeup Reason | 101 | 1 | CDMA 1 | CDMA 1 |
| 0x5009 | Sidetone Gain Level | 218 | 1 | CDMA 1 | CDMA 1 |
| 0x500B | Set Dormant | 189 | 1 | CDMA 1 | CDMA 1 |
| 0x7000 | Replaced by object 0x2100 "Start IOTA Session" | 247 | N/a | | |
| 0x7001 | Replaced by object 0x2101 "Stop IOTA Session" | 248 | N/a | | |
| 0x7002 | Replaced by object 0x2102 "IOTA Status" | 249 | N/a | | |
| 0x7003 | Replaced by object 0x2103 "IOTA WAP Push" | 251 | N/a | | |
| 0x7004 | Replaced by object 0x2104 "Get/Set IOTA Feature" | 108 | N/a | | |

6: Hardware Characteristics

Introduction

This chapter provides reference for [CnS](#) messages that deal almost entirely with modem characteristics. Only a few objects relate to operational activity.

Hardware summary

Table 3: Modem device characteristics

| Object | ID | Description | S | G | N |
|--|--------|--|---|---|---|
| Heartbeat | 0x0000 | Modem presence | X | X | ✓ |
| Firmware Version | 0x0001 | Version of modem firmware | X | ✓ | X |
| Firmware Date | 0x0002 | Date of the modem firmware | X | ✓ | X |
| Boot Version | 0x0004 | Version of the modem loader firmware | X | ✓ | X |
| Radio Power | 0x0007 | Modem power | ✓ | ✓ | ✓ |
| Radio Temperature | 0x0008 | Current temperature state of the modem | X | ✓ | ✓ |
| Radio Supply Voltage | 0x0009 | Current supply voltage of the modem | X | ✓ | ✓ |
| Sierra Wireless Modem Type | 0x000A | Product type and the type of the wireless technology | X | ✓ | X |
| NVRAM Backup/Restore | 0x000B | Backup or restore the NV data on modem | ✓ | X | ✓ |
| Supported Object Releases | 0x000C | Oldest and newest versions of CnS object sets that the modem firmware supports | X | ✓ | X |
| Boot Build Date | 0x000D | Build date of the boot loader | X | ✓ | X |
| Headset Icon | 0x000F | Headset information | X | ✓ | ✓ |
| Reset Device | 0x0010 | Resets the modem | ✓ | X | X |

| Object | ID | Description | S | G | N |
|-------------------------------|--------|--|---|---|---|
| NVRAM Update from File | 0x0011 | Write the contents of a file to NV | ✓ | ✗ | ✓ |
| Interface Unlock | 0x0012 | Unlock access to password-protected device command interfaces | ✓ | ✗ | ✗ |
| Interface Set Lock | 0x0015 | Sets the command interface unlock key values used by the Interface Unlock object (0x0012) | ✓ | ✗ | ✗ |
| Current Radio Band | 0x0016 | List of bands the modem is currently using, and which radio each band is associated with | ✗ | ✓ | ✓ |
| Reset Device Data Statistics | 0x0018 | Clear the modem's internal data statistic counters | ✓ | ✗ | ✗ |
| Fetch Device Data Statistics | 0x0019 | Returns the modem's internal data statistic counters | ✓ | ✗ | ✗ |
| Audio Profile | 0x001A | Mode of the audio (voice) circuitry | ✓ | ✓ | ✓ |
| Device Time | 0x001B | System time of the modem | ✗ | ✓ | ✓ |
| PRI Information | 0x0023 | PRI version information and the SKU ID or part number | ✗ | ✓ | ✗ |
| Flash Image Information | 0x0024 | Information about an image in the flash memory of the modem | ✓ | ✗ | ✗ |
| Removable Media Configuration | 0x0025 | Enable SD card support | ✓ | ✓ | ✗ |
| Remote Diagnostics Agent | 0x0026 | Enable/disable any remote diagnostics agent on the modem | ✓ | ✓ | ✗ |
| SWOC Configuration | 0x0027 | Configure SWOC features | ✓ | ✓ | ✗ |
| ESN | 0x1000 | Read the Electronic Serial Number | ✗ | ✓ | ✗ |
| PRL Version | 0x1008 | Preferred Roaming List version | ✗ | ✓ | ✗ |
| Reset Modem | 0x100F | Force the modem to reset | ✓ | ✗ | ✗ |
| Cumulative Call Duration | 0x101B | Total time of all calls since modem activation | ✗ | ✓ | ✗ |
| Power Down Modem | 0x1022 | Power down the modem | ✓ | ✗ | ✗ |
| ERI File Name | 0x104A | ERI file name and size | ✓ | ✓ | ✗ |

| Object | ID | Description | S | G | N |
|----------------------------|--------|---|---|---|---|
| ERI Write | 0x104C | Write ERI file | ✓ | ✗ | ✗ |
| Get/Set PRL Size | 0x1052 | Preferred Roaming List size | ✓ | ✓ | ✗ |
| PRL Write | 0x1054 | Write Preferred Roaming List | ✓ | ✗ | ✗ |
| Hardware Temperature | 0x4004 | Temperature of the radio | ✗ | ✓ | ✗ |
| Modem Too Hot Notification | 0x4500 | Notify of a radio overheat condition | ✗ | ✗ | ✓ |
| Temperature Threshold | 0x4501 | Threshold at which the modem's temperature will be considered "too hot" | ✓ | ✓ | ✗ |
| Wakeup Enable | 0x5007 | Event mask to use for modem wakeup | ✓ | ✓ | ✗ |
| Wakeup Reason | 0x5008 | Reason for the last modem wakeup | ✗ | ✓ | ✗ |

Hardware reference

Heartbeat

Provides a "heartbeat" notification indicating normal operation of the modem.

| | | | | |
|--------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x0000 | Heartbeat | | |
| Operations | ✗ Set | ✗ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 0 | | | None |

Version information:

| | |
|--|-----|
| Object version | N/a |
| Added in CWE CnS object set Release | 1 |

The heartbeat is a mechanism to maintain the application's confidence in the connection to the modem. An application can enable the heartbeat notification from the modem. The modem then issues the heartbeat notification at intervals of seven (7) seconds during normal operations.

If the application fails to receive a heartbeat after a reasonable period, it can assume the modem has stopped normal operation.

Failure of the heartbeat can indicate:

- A modem reset
- Port disconnection
- Modem shutdown or deep sleep operation
- A fatal modem error

Firmware Version

Reports the version of the modem application [firmware](#).

| | | | | |
|-------------------|---------------|------------------|-----------------|---|
| Object ID | 0x0001 | Firmware Version | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 0–8 | 0 | ASCII | Version string as ASCII (null-terminated, up to 8 characters) |

Version information:

| | |
|---|-----|
| Object version | N/a |
| Added in CWE CnS object set Release | 1 |

The modem firmware version string includes the revision number in the form Pxyz, where x, y, and z are ASCII numerals, either single or double digit. There may be additional information in the string.

See also "[Boot Version](#)" on page 49.

Firmware Date

Reports the date the modem's application [firmware](#) was built.

| | | | | |
|-------------------|---------------|---------------|-----------------|---|
| Object ID | 0x0002 | Firmware Date | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 8 | 0 | ASCII | Date string as 8 ASCII characters (not null-terminated) |

Version information:

| | |
|--|-----|
| Object version | N/a |
| Added in CWE CnS object set Release | 1 |

Boot Version

Reports the version of the modem bootstrap/loader [firmware](#).

| | | | | |
|-------------------|---------------|---------------|-----------------|--|
| Object ID | 0x0004 | Boot Version | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 0-84 | 0 | ASCII | Version string as ASCII (null-terminated, up to 84 characters) |

Version information:

| | |
|--|-----|
| Object version | N/a |
| Added in CWE CnS object set Release | 1 |

The modem firmware includes both a bootstrap/firmware loader and an application. This version string reports the bootstrap/loader version number. The

revision is in the form Rx-y-z, where x, y, and z are ASCII numerals, either single or double digit. There may be additional information in the string.

See also “[Firmware Version](#)” on page 48.

Radio Power

Monitor the modem power, initiate a modem power change, and be notified of the modem’s request to power down the modem.

To initiate a modem power change from low power mode to online or vice versa, use a Set request.

A notification is sent whenever the power mode of the modem changes. The modem also uses this notification to complete a shutdown, by requesting that the host cut power to the modem.

This object is useful only for MiniCards.

| Object ID | 0x0007 | Radio Power | | |
|------------|--------|-------------|----------|---|
| Operations | ✓ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Current power mode: <ul style="list-style-type: none"> 0x00: Low power mode 0x01: Online mode |
| | | 4 | UINT16 | Low power mode reason (bitmask): <ul style="list-style-type: none"> 0x0000: W_DISABLE is asserted 0x0002: User request 0x0004: Out of operational temperature range 0x0008: Out of operational supply voltage range |
| | | 6 | UINT8 | Reserved |
| 03 (Set) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Requested power state: <ul style="list-style-type: none"> 0x00: Low power mode 0x01: Online mode |

| | | | | |
|--------------------|----|---|--------|---|
| | | 4 | UINT8 | Reserved |
| 04 (Ack) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Current power mode: <ul style="list-style-type: none"> • 0x00: Low power mode • 0x01: Online mode |
| | | 4 | UINT16 | Low power mode reason: <ul style="list-style-type: none"> • 0x0000: W_DISABLE is asserted • 0x0002: User request • 0x0003: Out of operational temperature range • 0x0004: Out of operational supply voltage range |
| | | 6 | UINT8 | Reserved |
| 07 (Notify) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Current power mode: <ul style="list-style-type: none"> • 0x00: Low power mode • 0x01: Online mode |
| | | 4 | UINT16 | Low power mode reason (bitmask): <ul style="list-style-type: none"> • 0x0000: W_DISABLE is asserted • 0x0002: User request • 0x0004: Out of operational temperature range • 0x0008: Out of operational supply voltage range |
| | | 6 | UINT16 | Power-me-down request (added in Object version 2) <ul style="list-style-type: none"> • 0x0000: No request being made • 0x0001: Modem requests that the host power it down |
| | | 8 | UINT16 | Reason for power-down request (added in Object version 2) <ul style="list-style-type: none"> • 0x0000: No reason given (for example, the previous field is 0) • 0x0001: Temperature exceeds limits • 0x0002: Voltage exceeds limits • 0x0003: Network-related reason • 0x0004: All other reasons |




| | | | | |
|--|--|----|-------|----------|
| | | 10 | UINT8 | Reserved |
|--|--|----|-------|----------|

Version information:

| | |
|--|--|
| Object version | 2 (added fields "Power-me-down request" and "Reason for power-down request") |
| Added in CWE CnS object set Release | 3 |

Radio Temperature

Request the current temperature state of the modem.

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x0008 | Radio Temperature | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Temperature state: <ul style="list-style-type: none"> • 0x0000: Normal state- • 0x0001: High temperature warning state • 0x0002: High temperature critical state • 0x0003: Low temperature critical state |
| | | 4 | INT16 | Current temperature (°C) |
| | | 6 | UINT8 | Reserved |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

Radio Supply Voltage

Request the current supply voltage of the modem.

| | | | | |
|-----------------------------------|---------------|----------------------|-----------------|--|
| Object ID | 0x0009 | Radio Supply Voltage | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Supply voltage state: <ul style="list-style-type: none"> • 0x0000: Normal state • 0x0001: Low battery warning state • 0x0002: Low battery critical state • 0x0003: High battery critical state |
| | | 4 | UINT16 | Current supply voltage (mV) |
| | | 6 | UINT8 | Reserved |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

Sierra Wireless Modem Type

Returns the product type and the type of the wireless technology that the modem supports (CDMA, or GSM/UMTS).

Based on this object, the host device can determine how to interact with the modem. (Several Sierra Wireless products share a common range of object IDs [0x1000 to 0x1FFF] that can have, depending on the product, completely different functions).

| | | | | |
|-------------------|---------------|----------------------------|-----------------|--|
| Object ID | 0x000A | Sierra Wireless Modem Type | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 40 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Modem type: <ul style="list-style-type: none"> • 0x00: AirCard 710 • 0x01: AirCard 720 • 0x02: AirCard 750 • 0x03: SB750 • 0x04: SB750A • 0x05: AirCard 755 • 0x06: AirCard 775 • 0x07: AirCard 77 • 0x08: PC7000 • 0x09: PC7200 • 0x0A: AirCard 850 • 0x0B: AirCard 860 • 0x0C: MC8755 • 0x0D: MC8765 • 0x0E: MC5720 • 0x0F: EM5625 • 0x10: AirCard 595 • 0x11: MC5725/MC5725V • 0x12: AirCard 597E • 0x13: AirCard 875 • 0x14: MC8775 • 0x15: MC8775V • 0x16: EM5725 • 0x17: AirCard 597 • 0x18: AirCard 880 |

| | | |
|--|--|--|
| | | <ul style="list-style-type: none"> • 0x19: AirCard 881 • 0x1A: MC8780 • 0x1B: MC8781 • 0x1C: MC8780V • 0x1D: MC8781V • 0x1E: EC880 • 0x1F: EC881 • 0x20: EM8780 • 0x21: EM8781 • 0x22: M81A • 0x23: M81B • 0x24: AirCard 595U • 0x25: MC5727 • 0x26: Compass 597 • 0x27: MC8785 • 0x28: MC8785V • 0x29: AirCard 885E • 0x2A: Compass 885 • 0x2B: MC8790 • 0x2C: MC8790V • 0x2D: AirCard 501 • 0x2E: Compass 888 • 0x2F: Compass 889 • 0x30: K888 • 0x31: K889 • 0x32: MC8791V • 0x33: MC8792V • 0x34: C22 • 0x35: C23 • 0x36: USB 598 • 0x37: T11 • 0x38: unassigned • 0x39: AirCard 402 • 0x3A: MC5727V • 0x3B: MC5728 • 0x3C: MC5728V • 0x3D: K22 • 0x3E: K23 • 0x3F: USB 305 • 0x40: C25 • 0x41: USB 303 • 0x42: USB 304 • 0x43: USB 306 • 0x44: MC8777V • 0x45: MC8700 |
|--|--|--|

| | | |
|----|-----------|--|
| | | <ul style="list-style-type: none"> • 0x46: AirCard 502 • 0x47: USB 301 • 0x48: USB 307 • 0x49: USB 308 • 0x4A: USB 309 • 0x4B: USB 311 • 0x4C: USB 312 |
| 4 | UINT32 | Technology supported: <ul style="list-style-type: none"> • 0x00000001: CDMA • 0x00000002: GSM/UMTS |
| 8 | UINT8[16] | Unused bytes (contains zeros) |
| 24 | UINT8 | Length of the Product string (see the next field); 0—15; does not include the null character. |
| 25 | UINT8[15] | Product string. ASCII string, null padded. |

Version information:

| | |
|-------------------------------------|---|
| Object version | 2 |
| Added in CWE CnS object set Release | 3 |

NVRAM Backup/Restore

Backup or restore the [NV](#) data on modem.

The Set response indicates that the modem is processing the request.

The host should not send another Set request until the notification has been received (the notification indicates the completion and success of the Set request).

The host should not make any CnS requests to the modem while an “NV restore” operation is under way.

The User settings are backed up and restored every time firmware is downloaded to the modem.

A backup or restore operation can take as long as 3 minutes.

| | | | | |
|--------------------|--|----------------------|-------------|---|
| Object ID | 0x000B | NVRAM Backup/Restore | | |
| Operations | <input checked="" type="checkbox"/> Set <input checked="" type="checkbox"/> Get <input checked="" type="checkbox"/> Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | NV operation: <ul style="list-style-type: none"> • 0x02: Back up the NV • 0x03: Restore the NV |
| | | 3 | UINT8 | NV category: <ul style="list-style-type: none"> • 0x02: User settings |
| | | 4 | UINT8 | Reserved |
| 04 (Ack) | 2 | 0 | UINT16 | Object version The 04 (Ack) response indicates that the modem is processing the request. |
| 07 (Notify) | 40 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | NV operation status: <ul style="list-style-type: none"> • 0x00: Successful • 0x01: Unsuccessful |
| | | 3 | UINT8 | NV operation: <ul style="list-style-type: none"> • 0x02: Backup • 0x03: Restore |
| | | 4 | UINT8 | NV category: <ul style="list-style-type: none"> • 0x02: User settings |
| | | 5 | UINT32 | Item count: <ul style="list-style-type: none"> • If NV operation = Backup: Number of items stored • If NV operation = Restore: Number of items restored |
| | | 9 | UINT32 | Items skipped. (If NV operation = Restore: 0) |
| | | 13 | UINT32 | Items with 0 length (If NV operation = Restore: 0) |

| | | | | |
|--|--|----|-------|----------|
| | | 17 | UINT8 | Reserved |
|--|--|----|-------|----------|

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

Supported Object Releases

Reports the oldest and newest Releases of CnS object sets that the modern [firmware](#) supports.

| Object ID | 0x000C | Supported Object Releases | | |
|------------|--|---|---|--|
| Operations | ✗ Set | <input checked="" type="checkbox"/> Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 245 | 0 | UINT16 | Object version. Version number of this object |
| | | 2 | UINT8 | Number of entries (from 1 to 11) in Supported Object Sets list, described below. Maximum 35. |
| | | 3 | Variable length list | Supported Object Sets list (array of version information entries). ListLength = 10 bytes * Number_of_Entries |
| | | | | Each entry in Supported Object Sets has the following fields. |

| | | |
|----------------|------------|--|
| | UINT8 | <p>Object Set identifier</p> <ul style="list-style-type: none"> • 0x00: CWE (Object IDs 0x0000-0x0DFF) • 0x01: LBS (Object IDs 0x0F00-0x0FFF) • 0x03: CDMA CnS (Object IDs 0x1000-0x5FFF) • 0x06: CWE DM (Object IDs 0x0E00-0x0EFF) • 0x07: CDMA SMS (Object IDs 0x2000-0x2FFF) • 0x08: CDMA CALL (Object IDs 0x3000-0x3FFF) • 0x09: CDMA TECH (Object IDs 0x4000-0x44FF) • 0x0A: CDMA Temperature Protection (Object IDs 0x4500-0x4FFF) <p>For more information on the Object Sets, see the text after this table.</p> |
| | UINT16 | <p>The oldest Release of the Object Set that the firmware supports.</p> <p>For example, if the current field is "2", and "Object Set identifier" is 0, then the oldest Release of CWE objects that the firmware supports is version 2.</p> |
| | UINT16 | <p>Latest Release of this Object Set that the firmware supports.</p> |
| | UINT8[5] | <p>Reserved</p> |
| 3 + ListLength | UINT8[...] | <p>Remaining unused space (complete record length = 245 bytes)</p> |

Object version: Each CnS object has a version number, which gets incremented in a firmware release if the structure for that particular object has changed.

Each CnS object supported by CDMA modems is one of the following types:

- **CWE**—this type of object (Core Wireless Engine object) can also be used in GSM modems
- **LBS**—this type of object is used for location-based services
- **CDMA**— this type of object is specific to CDMA modems.
- **DM**—this type of object (Device Management object) can also be used in GSM modems
- **CDMA SMS**— this type of object is used for Short Messaging Service (SMS), and is specific to CDMA modems
- **CDMA CALL**— this type of object is used for data or voice calls, and is specific to CDMA modems
- **CDMA Tech**— this type of object is used for technical/statistical information, and is specific to CDMA modems
- **CDMA Temperature Protection**— this type of object is used for detection of modem overheating, and is specific to CDMA modems

object set Release: All of the CnS objects of a certain type (for example, all of the LBS objects) in a firmware release comprise an Object Set (in our example – the LBS CnS object set), which has an associated Release (version) number. In our example, the LBS CnS object set Release number would get incremented in a firmware release, if a new object is added to the LBS CnS object set.

After the description of the structure of each object, the “Version information” table shows the object version (for that object), and in which Release of the relevant Object set (CWE, LBS, CDMA, or DM) this object was added. For obsolete objects or objects that may become obsolete, “[Appendix D: Objects That Have Been Replaced, and May Be Removed From Future Releases](#)” (page 316) indicates, in which version of the Object set this object was removed.

As of this document’s publication, the following are valid version numbers:

| Object Set | Releases available | Releases supported by the modem |
|---------------------|--------------------|---|
| CWE | 1 through 14 | 14 |
| LBS | 1 through 4 | 4 |
| CDMA | 1 | 1 |
| DM | 1 through 2 | 2 |
| CDMA SMS | 1 through 2 | 1 and 2. Note: Release 2 includes, for some objects, a release 1 (“r1”) and a release 2 (“r2”) version: <ul style="list-style-type: none"> • SMS Read (r1) and SMS Read (r2) • SMS Receive Status (r1) and |

| Object Set | Releases available | Releases supported by the modem |
|-----------------------------|--------------------|--|
| | | <p>SMS Receive Status (r2) and Storage Capacity</p> <ul style="list-style-type: none"> • SMS Delete (r1) and SMS Delete (r2) <p>If the modem supports release 2 of SMS objects, your application can use, from the above list, either release 1 versions, or release 2 versions, but not a combination of both. For example, your application can use SMS Read (r1) and SMS Delete (r1), but not SMS Read (r1) and SMS Delete (r2).</p> |
| CDMA CALL | 1 | 1 |
| CDMA Tech | 1 | 1 |
| CDMA Temperature Protection | 1 | 1 |

At startup, your application should interrogate this **Supported Object Releases** object. If, for any of the Object Sets, the modem reports a lower Release than the one your application supports, AND your application is using that Object Set, then you must upgrade the firmware on the modem to bring it to a level equal to, or above, the level supported by your application.

Version information for **Supported Object Releases** object:

| | |
|---|----|
| Object version | 3 |
| Added in CWE CnS object set Release | 14 |

Boot Build Date

The build date of the boot loader.

| | | | | |
|-------------------|---------------|-----------------|-----------------|---|
| Object ID | 0x000D | Boot Build Date | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 20 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Date string as 8 ASCII characters (not null-terminated) |
| | | 3 | UINT8 | Reserved |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

Headset Icon

Returns the headset information (MC5728V modem only).

| | | | | |
|-----------------------------------|---------------|----------------------------------|-----------------|---|
| Object ID | 0x000F | USB Descriptor Build Information | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Headset state: <ul style="list-style-type: none"> • 0x00: Headset not plugged in • 0x01: Headset plugged in • 0x02: Headset information not available |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

Reset Device

Resets the modem.

This object replaces object 0x100F [Reset Modem](#) (page 92).

| | | | | |
|------------------------------|---------------|---------------|-----------------|--------------------------------|
| Object ID | 0x0010 | Reset Device | | |
| Operations | ✓ Set | X Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) 04 (Ack) | 2 | 0 | UINT16 | Object version |

The modem issues the Set acknowledgment, performs a graceful disconnection from the network, and then resets.

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 3 |



Interface Unlock

Unlocks access to various password-protected device command interfaces. These commands can be used to configure carrier-specific parameters and to enable limited diagnostic capabilities. These commands are used by OEMs and device provisioning applications.

If the unlock operation is successful, the host has access to the password-protected device commands until the device is reset. After a device reset, to access the password-protected device commands this object must be used again.

| | | | | |
|-------------------|--------------------------------|---------------|-------------|---|
| Object ID | 0x0012 | | | |
| Operations | ✓ Set ✗ Get ✗ Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 14 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Unlock level: <ul style="list-style-type: none"> • 0x00: OEM unlock • Other values: Reserved for future use |
| | | 3 | UINT8 | Length of password string |
| | | 4 | UINT8[10] | Password string as ASCII characters (not null-terminated) |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success; interface is unlocked • 0x01: Invalid password |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

NVRAM Update from File

Write the contents of a file to [NV](#).

Note: This should be done only during firmware upgrade.

An NV update operation can take as long as 3 minutes.

| | | | | |
|--------------------|---------------|------------------------|-------------|--|
| Object ID | 0x0011 | NVRAM Update from File | | |
| Operations | ✓ Set | ✗ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 10 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Name of file—8 ASCII characters (not null-terminated) |
| 04 (Ack) | 2 | 0 | UINT16 | Object version The 04 (Ack) response indicates that the modem is processing the request. |
| 07 (Notify) | 11 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result of NV update <ul style="list-style-type: none"> • 0x00: Successful • 0x01: File not found • 0x02: File size error • 0x03: File header error • 0x04: File revision error • 0x05: File close error • 0x06: File read error • 0x07: NV write error • 0x08: Unknown error |
| | | 3 | UINT32 | Number of items updated |
| | | 7 | UINT32 | Number of items skipped |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |



Interface Set Lock

Sets the command interface unlock key values used by the [Interface Unlock](#) object (page 64). Before you use the **Interface Set Lock** object, the modem must be unlocked to at least the level of the key being updated.

If Key length = 0, the unlock key is disabled.




| | | | | |
|-------------------|---------------|--------------------|-------------|--|
| Object ID | 0x0015 | Interface Set Lock | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 14 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Unlock level: <ul style="list-style-type: none"> • 0x00: OEM unlock • Other values: Reserved for future use |
| | | 3 | UINT8 | Key length |
| | | 4 | UINT8[10] | Key string as ASCII characters (not null-terminated) |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Invalid password • 0x02: Unlock first before setting new key • 0x03: Key commit failed |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

Current Radio Band

Returns a list of bands the modem is currently using, and which radio each band is associated with.

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x0016 | Current Radio Band | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | Variable (minimum 3) | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Number of bands in the list. Maximum 25. |
| | | 3 | Variable length list | Band info list Each entry in the list has the following fields. |
| | | | UINT8 | Radio type <ul style="list-style-type: none"> • 0x00: 1X • 0x01: 1XEV-DO |
| | | | UINT8 | Current band: <ul style="list-style-type: none"> • 0x00: Band Class 0 • 0x01: Band Class 1 • 0x02: Band Class 2 • 0x03: Band Class 3 • 0x04: Band Class 4 • 0x05: Band Class 5 • 0x06: Band Class 6 • 0x07: Band Class 7 • 0x08: Band Class 8 • 0x09: Band Class 9 |

| | | | |
|--|--|--|---|
| | | | <ul style="list-style-type: none"> • 0x0A: Band Class 10 • 0x0B: Band Class 11 • 0x0C: Band Class 12 • 0xFF: Invalid Band |
|--|--|--|---|

Version information:

| | |
|-------------------------------------|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 3 |

Reset Device Data Statistics

Clear the modem's internal data statistic counters. (To retrieve the statistics, use [Fetch Device Data Statistics](#) [page 69]).

| | | | | |
|-------------------|---------------|------------------------------|-------------|---|
| Object ID | 0x0018 | Reset Device Data Statistics | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Type of protocol/link statistics to reset: <ul style="list-style-type: none"> • 0x00: PPP over Um link • 0x01: PPP over Rm link • 0x02: IPv4 over Um link • 0x03: ICMPv4 over Um link • 0x04: TCP over Um link • 0x05: UDP over Um link |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Invalid type of statistics • 0x02: Statistics reset failed |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 4 |

Fetch Device Data Statistics

Returns the modem's internal data statistic counters.

To reset these counters, use [Reset Device Data Statistics](#) (page 68).

Note: These data statistics are not persistent, and are only for sessions since the last modem reset.

Not all products support all the statistics reported by this object. If a modem does not support a particular statistic, the corresponding field contains the maximum value supported by that field. This indicates to the host that the modem does not support this particular statistic.

| | | | | |
|-------------------|---------------|------------------------------|-------------|--|
| Object ID | 0x0019 | Fetch Device Data Statistics | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Type of protocol/link statistics to report: <ul style="list-style-type: none"> • 0x00: PPP over Um link • 0x01: PPP over Rm link • 0x02: IPv4 over Um link • 0x03: ICMPv4 over Um link • 0x04: TCP over Um link • 0x05: UDP over Um link |
| | | 3 | UINT8 | PPP statistics type (if PPP statistics are requested): <ul style="list-style-type: none"> • 0x00: General PPP • 0x01: Van Jacobson compression • 0x02: PPP authentication • 0x03: LCP related • 0x04: IPCP related |

| | | | | |
|-----------------|--------|---|--------|--|
| 04 (Ack) | 21–168 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Fetch status <ul style="list-style-type: none"> • 0x00: Valid data statistics present • 0x01: Invalid statistics link • 0x02: Invalid statistics type • 0x03: Statistics unavailable |
| | | 3 | UINT8 | Type of protocol/link statistics reported: <ul style="list-style-type: none"> • 0x00: PPP over Um link • 0x01: PPP over Rm link • 0x02: IPv4 over Um link • 0x03: ICMPv4 over Um link • 0x04: TCP over Um link • 0x05: UDP over Um link |
| | | 4 | UINT8 | Union of protocol data statistics: { PPP info (same structure regardless of link type) { PPP statistics type: <ul style="list-style-type: none"> • 0x00: General PPP • 0x01: Van Jacobson compression • 0x02: PPP authentication • 0x03: LCP • 0x04: IPCP PPP union { General PPP info (page 71) Van Jacobson Compression info (page 72) PPP authentication info (page 72) LCP-related info (page 72) IPCP-related info (page 73) } } |

| | | | |
|--|--|--|---|
| | | | IPv4 info (page 74) ICMPv4 info (page 75) TCP info (page 76) UDP info (page 78) } |
|--|--|--|---|

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CWE CnS object set Release | 4 |

The following table describes the definitions of the members of the [protocol statistics union](#) used in the Set response.

| Union member | Length (bytes) | Type | Description. Number of: |
|------------------|-----------------------|--------|-----------------------------------|
| General PPP info | 80 | UINT32 | Packets with checksum error |
| | | UINT32 | Packets with wrong protocol field |
| | | UINT32 | Packets with unsupported protocol |
| | | UINT32 | Packets with disabled protocol |
| | | UINT32 | Packets with missing address byte |
| | | UINT32 | Packets with missing control byte |
| | | UINT32 | Incoming packets dropped |
| | | UINT32 | Incoming packets received |
| | | UINT64 | Incoming framed bytes |
| | | UINT64 | Incoming unframed bytes |
| | | UINT32 | Outgoing packets dropped |
| | | UINT32 | Outgoing packets transmitted |
| | | UINT64 | Outgoing framed bytes |
| | | UINT64 | Outgoing unframed bytes |
| | | UINT32 | Incoming IPv4 packets |
| UINT32 | Outgoing IPv4 packets | | |

| Union member | Length (bytes) | Type | Description. Number of: |
|-------------------------------|----------------|--------|---|
| Van Jacobson Compression info | 16 | UINT32 | Van Jacomson compressed packets received |
| | | UINT32 | Van Jacomson compressed packets transmitted |
| | | UINT32 | Van Jacomson un-compressed packets received |
| | | UINT32 | Van Jacomson un-compressed packets received |
| PPP authentication info | 24 | UINT32 | PAP packets received |
| | | UINT32 | PAP packets transmitted |
| | | UINT32 | PAP authentications failed |
| | | UINT32 | CHAP packets received |
| | | UINT32 | CHAP packets transmitted |
| | | UINT32 | CHAP authentication failures |
| PPP LCP-related info | 104 | UINT32 | LCP packets received |
| | | UINT32 | LCP packets transmitted |
| | | UINT32 | LCP configuration packets received |
| | | UINT32 | LCP configuration acknowledgements received |
| | | UINT32 | LCP negative acknowledgement packets received |
| | | UINT32 | LCP configuration reject packets received |
| | | UINT32 | LCP termination packets received |
| | | UINT32 | Termination acknowledgement packets received |
| | | UINT32 | LCP code reject packets received |
| | | UINT32 | LCP protocol reject packets received |
| | | UINT32 | LCP echo request packets received |
| | | UINT32 | Echo reply packets received |
| | | UINT32 | LCP discard request packets received |

| Union member | Length (bytes) | Type | Description. Number of: |
|-----------------------|----------------|--------|--|
| | | UINT32 | LCP configuration request packets transmitted |
| | | UINT32 | LCP configuration acknowledgement packets transmitted |
| | | UINT32 | LCP configuration negative acknowledgement packets transmitted |
| | | UINT32 | LCP configuration reject packets transmitted |
| | | UINT32 | LCP termination request packets transmitted |
| | | UINT32 | LCP termination acknowledgement packets transmitted |
| | | UINT32 | LCP code reject packets transmitted |
| | | UINT32 | LCP protocol reject packets transmitted |
| | | UINT32 | LCP echo request packets transmitted |
| | | UINT32 | LCP echo reply packets transmitted |
| | | UINT32 | LCP discard request packets transmitted |
| | | UINT32 | LCP identification packets received |
| | | UINT32 | LCP time remaining packets received |
| PPP IPCP-related info | 64 | UINT32 | IPCP packets received |
| | | UINT32 | IPCP packets transmitted |
| | | UINT32 | IPCP configuration request packets received |
| | | UINT32 | IPCP configuration acknowledgement packets received |
| | | UINT32 | IPCP configuration negative acknowledgement packets received |
| | | UINT32 | IPCP configuration reject packets received |
| | | UINT32 | IPCP termination request packets received |
| | | UINT32 | IPCP termination acknowledgement packets received |
| | | UINT32 | IPCP code reject packets received |

| Union member | Length (bytes) | Type | Description. Number of: |
|--------------|----------------------------|--------|---|
| | | UINT32 | IPCP configuration request packets transmitted |
| | | UINT32 | IPCP configuration acknowledgement packets transmitted |
| | | UINT32 | IPCP configuration negative acknowledgement packets transmitted |
| | | UINT32 | IPCP configuration reject packets transmitted |
| | | UINT32 | IPCP termination request packets transmitted |
| | | UINT32 | IPCP termination acknowledgement packets transmitted |
| | | UINT32 | IPCP code reject request packets transmitted |
| IPv4 info | 88 | UINT32 | Packets with invalid header length |
| | | UINT32 | Packets with bad payload lengths |
| | | UINT32 | Packets with errors in option |
| | | UINT32 | Packets with bad version in IP header |
| | | UINT32 | Packets with bad header checksum |
| | | UINT32 | IP packets that are shorter than the size of a valid IP header |
| | | UINT32 | Packets not transmitted because of absence of route |
| | | UINT32 | Incoming packets with unknown protocol |
| | | UINT32 | Incoming packets |
| | | UINT32 | Incoming packets dropped at the IP layer |
| | | UINT32 | Outgoing packets dropped at the IP layer |
| | | UINT32 | Forwarded IP packets |
| | | UINT32 | Outgoing packets |
| | | UINT32 | Incoming multicast packets |
| UINT32 | Outgoing multicast packets | | |

| Union member | Length (bytes) | Type | Description. Number of: |
|--------------|----------------------------|--------|---|
| | | UINT32 | IP packets that have been successfully fragmented |
| | | UINT32 | IP packets that failed fragmentation |
| | | UINT32 | IP fragments created |
| | | UINT32 | Incoming IP fragments that needed to be reassembled |
| | | UINT32 | Incoming IP packets successfully reassembled |
| | | UINT32 | Incoming IP packets that failed reassembly |
| | | UINT32 | Incoming packets that failed reassembly due to timeout |
| ICMPv4 info | 120 | UINT32 | Incoming ICMP messages |
| | | UINT32 | Incoming ICMP messages dropped |
| | | UINT32 | Incoming address mask replies |
| | | UINT32 | Incoming address masks |
| | | UINT32 | Incoming destination unreachable |
| | | UINT32 | Incoming echo messages |
| | | UINT32 | Incoming echo replies |
| | | UINT32 | Incoming ICMP messages with bad checksum |
| | | UINT32 | Incoming ICMP messages with unsupported type |
| | | UINT32 | Incoming messages that have problems in the ICMP parameters |
| | | UINT32 | Incoming messages with bad code |
| | | UINT32 | Incoming redirects |
| | | UINT32 | Incoming source quench messages |
| | | UINT32 | Incoming time exceeded messages |
| UINT32 | Incoming timestamp replies | | |

| Union member | Length (bytes) | Type | Description. Number of: |
|--------------|----------------|--------|---|
| | | UINT32 | Incoming timestamps |
| | | UINT32 | Outgoing ICMP messages |
| | | UINT32 | Outgoing addr mask replies |
| | | UINT32 | Outgoing address masks |
| | | UINT32 | Outgoing destination unreachable |
| | | UINT32 | Outgoing echo messages |
| | | UINT32 | Outgoing echo replies |
| | | UINT32 | Outgoing messages that have problems in the ICMP parameters |
| | | UINT32 | Outgoing redirects |
| | | UINT32 | Outgoing source quench messages |
| | | UINT32 | Outgoing time exceeded messages |
| | | UINT32 | Outgoing timestamp replies |
| | | UINT32 | Outgoing timestamps |
| | | UINT32 | Unsent messages due to rate limit |
| | | UINT32 | Outgoing messages dropped |
| TCP info | 168 | UINT32 | Current open connections |
| | | UINT32 | Incoming segments with bad header length |
| | | UINT32 | Incoming segments with bad checksum |
| | | UINT32 | short (invalid) packets |
| | | UINT32 | Total number of dropped connections |
| | | UINT32 | Connections dropped due to timeouts |
| | | UINT32 | Connection dropped in keep alive |
| | | UINT32 | Resets sent |
| | | UINT32 | Resets received |
| | | UINT32 | Incoming segments dropped due to absence of a connection |

| Union member | Length (bytes) | Type | Description. Number of: |
|--------------|----------------|--------|--|
| | | UINT32 | Incoming segments dropped |
| | | UINT64 | Bytes sent |
| | | UINT64 | Bytes received |
| | | UINT32 | Segments sent |
| | | UINT32 | Segments received |
| | | UINT64 | Bytes retransmitted |
| | | UINT32 | Segments retransmitted |
| | | UINT32 | Duplicate bytes receive |
| | | UINT32 | Duplicate segments received |
| | | UINT32 | Window probes transmitted |
| | | UINT32 | Out of order segments received |
| | | UINT64 | Out of order bytes received |
| | | UINT32 | Retransmit timeouts |
| | | UINT32 | Piggybacked acks transmitted |
| | | UINT32 | Datagrams with bad payload lengths |
| | | UINT32 | Packets with bad header checksum |
| | | UINT32 | Incoming datagrams |
| | | UINT32 | Outgoing datagrams |
| | | UINT32 | Bytes received |
| | | UINT64 | Bytes sent |
| | | UINT64 | Incoming datagrams for an unreachable port |
| | | UINT32 | Incoming datagrams dropped |
| | | UINT32 | Outgoing datagrams dropped |
| | | UINT32 | Piggybacked acks received |
| | | UINT32 | Duplicate acks received |
| | | UINT32 | Duplicate acks transmitted |

| Union member | Length (bytes) | Type | Description. Number of: |
|--------------|----------------|--------|--|
| UDP info | 44 | UINT32 | Datagrams with bad payload lengths |
| | | UINT32 | Packets with bad header checksum |
| | | UINT32 | Incoming datagrams |
| | | UINT32 | Outgoing datagrams |
| | | UINT32 | Bytes received |
| | | UINT64 | Bytes sent |
| | | UINT64 | Incoming datagrams for an unreachable port |
| | | UINT32 | Incoming datagrams dropped |
| | | UINT32 | Outgoing datagrams dropped |

Audio Profile

Sets or queries the mode of the audio (voice) circuitry. The modem supports either headset level, or line level for connection to additional audio circuits. For additional information, see the MC5728V documentation.

Note: Object structure is subject to change. Supported only by the MC5728V modem.

The audio profile setting is not persistent (that is, after a modem reset, the default settings are restored).

| Object ID | 0x001A | Audio Profile | | |
|---------------------------|--------|---------------|----------|---|
| Operations | ✓ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 7 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Current audio profile setting (MC5728V [firmware revision 1.04 and higher]): <ul style="list-style-type: none"> • 0x0000: Handset • 0x0001: Headset • 0x0002: Speaker phone • 0x0003: Car kit |

| | | | | |
|-----------------|---|-------|--|--------------------------------|
| | | | <ul style="list-style-type: none"> • 0x0004: TTY—full audio • 0x0005: HAC (Hearing Aid Compatibility) <p>Current audio profile setting (MC5728V [firmware revision 1.03 and earlier]):</p> <ul style="list-style-type: none"> • 0x0000: Handset • 0x0001: Headset • 0x0002: Speaker phone • 0x0003: Car kit • 0x0004: TTY—full audio • 0x0005: TTY—talk audio • 0x0006: TTY—hear audio • 0x0007: HAC (Hearing Aid Compatibility) | |
| | 4 | UINT8 | <p>Ear piece mute setting:</p> <ul style="list-style-type: none"> • 0x00: Unmuted • 0x01: Muted | |
| | 5 | UINT8 | <p>Microphone mute setting:</p> <ul style="list-style-type: none"> • 0x00: Unmuted • 0x01: Muted | |
| | 6 | UINT8 | <p>Audio volume level:</p> <ul style="list-style-type: none"> • 0x00: level 0 • 0x01: level 1 • 0x02: level 2 • 0x03: level 3 • 0x04: level 4 • 0x05: level 5 • 0x06: level 6 • 0x07: level 7 | |
| 03 (Set) | 8 | 0 | UINT16 | Object version |

| | | | | |
|-----------------|---|---|--------|--|
| | | 2 | UINT16 | <p>Audio profile setting (MC5728V [firmware revision 1.04 and higher]):</p> <ul style="list-style-type: none"> • 0x0000: Handset • 0x0001: Headset • 0x0002: Speaker phone • 0x0003: Car kit • 0x0004: TTY—full audio • 0x0005: HAC (Hearing Aid Compatibility) <p>Audio profile setting (MC5728V [firmware revision 1.03 and earlier]):</p> <ul style="list-style-type: none"> • 0x0000: Handset • 0x0001: Headset • 0x0002: Speaker phone • 0x0003: Car kit • 0x0004: TTY—full audio • 0x0005: TTY—talk audio • 0x0006: TTY—hear audio • 0x0007: HAC (Hearing Aid Compatibility) |
| | | 4 | UINT8 | <p>Ear piece mute setting:</p> <ul style="list-style-type: none"> • 0x00: Unmuted • 0x01: Muted |
| | | 5 | UINT8 | <p>Microphone mute setting:</p> <ul style="list-style-type: none"> • 0x00: Unmuted • 0x01: Muted |
| | | 6 | UINT8 | <p>Audio generator:</p> <ul style="list-style-type: none"> • 0x00: Voice • 0x01: Key beep |
| | | 7 | UINT8 | <p>Audio volume level:</p> <ul style="list-style-type: none"> • 0x00: level 0 • 0x01: level 1 • 0x02: level 2 • 0x03: level 3 • 0x04: level 4 • 0x05: level 5 • 0x06: level 6 • 0x07: level 7 |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |

| | | | | |
|--|--|---|-------|---|
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Fail (invalid audio profile setting) |
|--|--|---|-------|---|

Version information:




| | |
|---|--|
| Object version | <ul style="list-style-type: none"> • 2 (for MC5728V [firmware revision 1.03 and earlier]); • 3 (for MC5728V [firmware revision 1.04 and higher]) |
| Added in CWE CnS object set Release | <ul style="list-style-type: none"> • 10 (for object version 2) • 13 (for object version 3) |

Device Time

Request the system time of the modem.

The host can use this time to set its clock.

A notification is sent whenever service is acquired or reacquired, and also when the time zone information changes.

| | | | | |
|-----------------------------------|--|--|---|---|
| Object ID | 0x001B | Device Time | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 19 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Year |
| | | 4 | UINT16 | Month (1 to 12) |
| | | 6 | UINT16 | Day (1 to 31) |
| | | 8 | UINT16 | Day of the week <ul style="list-style-type: none"> • 0x0000: Monday • 0x0001: Tuesday • 0x0002: Wednesday • 0x0003: Thursday • 0x0004: Friday • 0x0005: Saturday • 0x0006: Sunday |

| | | |
|----|--------|--|
| 10 | UINT16 | Hour (0 to 23), in UTC |
| 12 | UINT16 | Minute (0 to 59), in UTC |
| 14 | UINT16 | Second (0 to 59), in UTC |
| 16 | INT16 | Time zone (–48 to 48, where 1 equals 15 minutes) <ul style="list-style-type: none"> • -4: UTC minus 60 minutes • -3: UTC minus 45 minutes • -2: UTC minus 30 minutes • -1: UTC minus 15 minutes • 0: UTC • 1: UTC plus 15 minutes • 2: UTC plus 30 minutes |
| 18 | UINT8 | Daylight Savings <ul style="list-style-type: none"> • 0x00: No adjustment • 0x01: Plus one hour • 0x02: Plus two hours |

Version information:

| | |
|-------------------------------------|----|
| Object version | 1 |
| Added in CWE CnS object set Release | 10 |

PRI Information

Returns [PRI](#) version information and the SKU ID or part number. of the modem.

Note: Some fields may not be supported, depending on your modem model.

| | | | | |
|-------------------|---------------|-----------------|-----------------|--------------------------------|
| Object ID | 0x0023 | PRI Information | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 14 | 0 | UINT16 | Object version |

| | | | |
|--|----|--------|---|
| | 2 | UINT16 | Major version (binary encoded number) |
| | 4 | UINT16 | Minor version (binary encoded number) |
| | 6 | UINT16 | Sub-minor version (binary encoded number). 0xFFFF if unused. |
| | 8 | UINT16 | Sub-OEM version (binary encoded number). 0xFFFF if unused. |
| | 10 | UINT32 | SKU ID or Part Number (binary encoded number) |

Version information:

| | |
|---|----|
| Object version | 1 |
| Added in CDMA CnS object set Release | 10 |

Flash Image Information

Read information about an image in the flash memory of the modem.

| | | | | |
|-------------------|---|---|--|--------------------------------|
| Object ID | 0x0024 | Flash Image Information | | |
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 6 | 0 | UINT16 | Object version |

| | | | | |
|-----------------|----------|---|---------------------------------|---|
| | | 2 | CHAR[4] | Image type (ASCII uppercase characters; not NULL terminated): <ul style="list-style-type: none"> • QCOM: Boot images, including: NAND image partition table, PBL (Primary Boot Loader), SBL (Secondary Boot Loader), OEMSBL (OEM Secondary Boot Loader), Configuration data used by PBL • BOOT - OEMSBL (SWI Boot Loader) • APPL - application code • USBD - USB descriptor table • SWOC - Software on Card image |
| 04 (Ack) | Variable | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Invalid Image type (see above) • 0x02: Image not found |
| | | 3 | CHAR[4] | Image type (see above). |
| | | 7 | UINT8 | Number of image information objects (described below) |
| | | 8 | UINT8 UINT8 UINT8 | Image information object { Object type (See the table that follows.) Length (of the Data field of this specific Image information object [TLV instance]). See the table that follows. Data (See the table that follows.) } |

The following table describes the fields of the [Image information object](#).

| Object type | Used by these Image types : | Length | Data, and its format |
|-------------|--|--------|--|
| 0x00 | QCOM BOOT APPL | 84 | Version of the image. <product>_<build type> [_<pid>].<major>.<minor>[.<point>] <product> — Model of modem, or product type. For example, SWI6800 or SWI6800V2. |

| Object type | Used by these Image types: | Length | Data, and its format |
|-------------|----------------------------|--------|---|
| | | | <p><build type>: FP: Normal release PP: Point release.</p> <p><pid> — USB product ID (for example, MC5728V). Does not appear in builds that are used for multiple products.</p> <ul style="list-style-type: none"> <major>, <minor>, and <point> — zero-padded two-digit decimal version numbers. <p>ASCII string.</p> <p>If the modem returns less than 84 characters, then the string is padded with NULL characters, to a total length of 84.</p> <p>If the modem returns 84 characters, then it is not NULL-terminated; your application must append NULL to it.</p> <p>Examples: SWI6800_FP.00.61 SWI6800V2_PP.00.29.01</p> |
| 0x01 | USBD | 84 | <p>Version of the USB descriptor table.</p> <p><product>_<customer>.<major>.<minor></p> <p><product> — Model of modem, or product type. For example, SWI6800 or SWI6800V2. <customer>: For example, GENERIC, or AC402. <major>, and <minor> — zero-padded two-digit decimal version numbers.</p> <p>ASCII string.</p> <p>If the modem returns less than 84 characters, then the string is padded with NULL characters, to a total length of 84.</p> <p>If the modem returns 84 characters, then it is not NULL-terminated; your application must append NULL to it.</p> <p>Examples: SWI6800V2_GENERIC.00.00 SWI6800V2_HP.00.00</p> |
| 0x04 | SWOC | 84 | <p>Product code of the SWOC (Software on Card image) image.</p> <p>CDPC_<ProductCode>_<major>.<minor>.<point release></p> <p><ProductCode> — zero-padded 5-digit decimal CDPC (CD Product Code). <major>, <minor>, and <point release> —</p> |

| Object type | Used by these Image types: | Length | Data, and its format |
|-------------|---|--------|---|
| | | | <p>zero-padded two-digit decimal numbers for CD version.</p> <p>ASCII string.</p> <p>If the modem returns less than 84 characters, then the string is padded with NULL characters, to a total length of 84.</p> <p>If the modem returns 84 characters, then it is not NULL-terminated; your application must append NULL to it.</p> |
| 0x05 | <p>QCOM</p> <p>BOOT</p> <p>APPL</p> <p>USBD</p> <p>SWOC</p> | 8 | <p>Release date (MM/DD/YY).</p> <p>ASCII string, not null-terminated.</p> |
| 0x06 | <p>QCOM</p> <p>BOOT</p> <p>APPL</p> <p>USBD</p> | 4 | <p>Modem type that the image supports.</p> <p>ASCII string, not null-terminated.</p> |
| 0x07 | <p>QCOM</p> <p>BOOT</p> <p>APPL</p> <p>USBD</p> <p>SWOC</p> | 4 | <p>Size, in bytes, of the image data. Does not include the size of CWE header.</p> <p>Format: binary.</p> |

Version information:

| | |
|-------------------------------------|----|
| Object version | 1 |
| Added in CWE CnS object set Release | 10 |

Removable Media Configuration

Enable SD card support.

| | | | | |
|-------------------|---------------|-------------------------------|-------------|--|
| Object ID | 0x0025 | Removable Media Configuration | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 6 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Type of removable media supported: <ul style="list-style-type: none"> • 0x0000: miniSD and microSD are not supported • 0x0001: miniSD is supported • 0x0002: microSD is supported • 0x0003: miniSD and microSD are supported |
| | | 4 | UINT16 | Types of removable media that are enabled: <ul style="list-style-type: none"> • 0x0000: miniSD and microSD are not enabled • 0x0001: miniSD is enabled • 0x0002: microSD is enabled • 0x0003: miniSD and microSD are enabled |
| 03 (Set) | 5 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Type of removable media to enable/disable: <ul style="list-style-type: none"> • 0x0001: miniSD • 0x0002: microSD |
| | | 4 | UINT8 | Action to perform: <ul style="list-style-type: none"> • 0x00: Disable the removable media • 0x01: Enable the removable media |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |




| | | | | |
|--|--|---|-------|--|
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Fail (media not supported) • 0x02: Fail (security access denied) • 0x03: Fail (invalid parameter) |
|--|--|---|-------|--|

Version information:

| | |
|-------------------------------------|----|
| Object version | 1 |
| Added in CWE CnS object set Release | 10 |

Remote Diagnostics Agent

Reports the status, and enables/disables any remote diagnostics agent on the modem.

| | | | | |
|-------------------|---|--------------------------|-------------|---|
| Object ID | 0x0026 | Remote Diagnostics Agent | | |
| Operations |  Set  Get  Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 69 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Status of the Remote Diagnostics Agent (RDA): <ul style="list-style-type: none"> • 0x00: Not supported • 0x01: Disabled (not running) • 0x02: Enabled |
| | | 3 | UINT8 | Length of the Vendor name (see the next field) |
| | | 4 | UINT8[32] | Vendor name (ASCII characters) |
| | | 36 | UINT8 | Length of the Agent version (see the next field) |
| | | 37 | UINT8[32] | Agent version (ASCII characters) |
| 03 (Set) | 3 | 0 | UINT16 | Object version |

| | | | | |
|-----------------|---|---|--------|--|
| | | 2 | UINT8 | Action to perform: <ul style="list-style-type: none"> • 0x00: Disable the RDA • 0x01: Enable the RDA |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Fail (reason not specified) • 0x02: Fail (security access denied) |

Version information:

| | |
|---|----|
| Object version | 1 |
| Added in CWE CnS object set Release | 10 |

SWOC Configuration

Configure [SWOC](#) features.

| | | | | |
|-------------------|---------------|--------------------|-------------|--|
| Object ID | 0x0027 | SWOC Configuration | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 10 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Bit map for supported features (based on the modem's PRI and/or SKU): <ul style="list-style-type: none"> • Bit 0: General SWOC operation. • Bit 1: Allow SWOC access in modem mode. When enabled (see Bit 1 of the next row), this feature allows the user to access and run software on the SWOC media even after the modem has switched from SWOC to modem mode. For each bit: 0 = Feature is not supported; 1 = Feature is supported. |

| | | | | |
|-----------------|---|---|--------|--|
| | | 4 | UINT16 | <p>Bit map for feature state:</p> <ul style="list-style-type: none"> • Bit 0: General SWOC operation • Bit 1: Allow SWOC access in modem mode. <p>For each bit: 0 = Feature is disabled; 1 = Feature is enabled.</p> |
| | | 6 | UINT16 | <p>Bit map for host ability to enable/disable a SWOC feature:</p> <ul style="list-style-type: none"> • Bit 0: General SWOC operation • Bit 1: Allow SWOC access in modem mode. <p>For each bit: 0 = Host cannot enable/disable this SWOC feature; 1 = Host can enable/disable this SWOC feature.</p> |
| | | 8 | UINT16 | <p>Bit map for: OEM Unlock required before the host enables/disables a SWOC feature:</p> <ul style="list-style-type: none"> • Bit 0: General SWOC operation • Bit 1: Allow SWOC access in modem mode <p>Depending on the modem model, an OEM Unlock (via Interface Unlock; page 64) might be required before a Set operation can be performed successfully.</p> <p>For each bit: 0 = OEM Unlock is not required; 1 = OEM Unlock is required.</p> |
| 03 (Set) | 5 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | <p>Feature to enable/disable:</p> <ul style="list-style-type: none"> • 0x0001: General SWOC operation • 0x0002: Allow SWOC access in modem mode. |
| | | 4 | UINT8 | <p>Action to perform:</p> <ul style="list-style-type: none"> • 0x00: Disable • 0x01: Enable |

| | | | | |
|-----------------|---|---|--------|--|
| 04 (Ack) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Fail (invalid parameter) • 0x02: Fail (feature not supported) • 0x03: Fail (enable/disable not allowed) • 0x04: Fail (“General SWOC operation” in “Feature support” is not enabled) • 0x05: Fail (security access denied) |

Version information:

| | |
|---|----|
| Object version | 1 |
| Added in CWE CnS object set Release | 12 |

ESN

Returns the Electronic Serial Number ([ESN](#)) of the modem.

| | | | | |
|-------------------|---|---|--|---|
| Object ID | 0x1000 | ESN | | |
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT32 | ESN as unsigned 32-bit integer (0xNNNNNNNN) |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

PRL Version

Returns the [PRL](#) version of the modem.

| | | | | |
|-------------------|---------------|---------------|-----------------|--------------------|
| Object ID | 0x1008 | PRL Version | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | PRL Version number |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Reset Modem

Resets the modem.

This object is replaced by object 0x0010 [Reset Device](#) (page 63). (Object **Reset Modem** (0x100F) is still supported; however, it may be removed, in a future release).

| | | | | |
|-------------------|---------------|---------------|-----------------|--------------------|
| Object ID | 0x100F | Reset Modem | | |
| Operations | ✓ Set | X Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

The modem issues the Set acknowledgment, performs a graceful disconnection from the network, and then resets.

Version information:

| | |
|---|---|
| Object version | 1 |
| Removed in CDMA CnS object set Release | 1 |

Cumulative Call Duration

Returns the total duration of all calls made using the modem, in minutes.

| | | | | |
|-------------------|---------------|---------------------|-----------------|--|
| Object ID | 0x101B | Total Call Duration | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT32 | Total cumulative call duration in minutes. |

Call duration is measured from the time the [traffic channel](#) is assigned until the channel is released. This includes data as well as voice traffic.

See also "[Call Duration](#)" on page 197.

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Power Down Modem

Powers down the modem.

| | | | | |
|------------------------------|---------------|------------------|-----------------|--------------------------------|
| Object ID | 0x1022 | Power Down Modem | | |
| Operations | ✓ Set | X Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) 04 (Ack) | 2 | 0 | UINT16 | Object version |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

ERI File Name

Queries or creates current [ERI](#) file with given name and size. This is the first step in writing new ERI file. Writing new ERI file requires provisioning via the Challenge and Rebuttal items. See [Set MSL/OTSL Activation Validation Challenge: Step 1 of 2](#) on page 129.

| Object ID | 0x104A | ERI File Name | | |
|------------|--------|---------------|----------|--|
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 8 | 0 | STRING | File Name |
| 02 (Reply) | 4 | 0 | UINT16 | Result (1: success; 0: failure) |
| | | 2 | UINT32 | File size in bytes |
| | | 6 | UINT16 | Number of blocks |
| 03 (Set) | 12 | 0 | STRING | File Name |
| | | 8 | UINT32 | File size in bytes |
| 04 (Ack) | 4 | 0 | UINT16 | Result (1: success; 0: failure) |
| | | 2 | UINT16 | Error code (see Table 4 below) |

Table 4: Error codes for Set ERI File Name

| Value | Meaning |
|-------|---------------------------------|
| 0 | Success |
| 1 | Failure—Not enough buffer space |
| 2 | Failure—Sequence number error |
| 3 | Failure—File doesn't exist |
| 4 | Failure—General error |
| 5 | Failure—Write size error |
| 6 | Failure—Operation error |

| Value | Meaning |
|-------|-----------------------|
| 7 | Failure—Pending error |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

ERI Write

Writes current [ERI](#) file. Set request must be preceded by [ERI File Name](#) (page 94). Only after the last segment is successfully written, ERI provisioning is complete and modem assembles all the pieces. Modem must be reset before it can use the new file.

| Object ID | 0x104C | ERI Write | | |
|-----------------|--------|-----------|----------|---|
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 16-114 | 0 | STRING | File Name |
| | | 8 | UINT16 | Block sequence |
| | | 10 | UINT16 | Total number of blocks |
| | | 12 | UINT32 | Data block size |
| | | 16 | UINT8 | Block data up to 114 bytes |
| 04 (Ack) | 4 | 0 | UINT16 | Write result (1: success; 0: failure) |
| | | 2 | UINT16 | Error code (see Table 4 on page 94) |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Get/Set PRL Size

Returns the size of current [PRL](#), or sets the size of the PRL to be written.

Setting the PRL size requires provisioning via the Challenge and Rebuttal items. (See [Set MSL/OTSL Activation Validation Challenge: Step 1 of 2](#) on page 129.) This is the first step in writing PRL. See [“Writing a PRL”](#) (page 315).

| | | | | |
|--------------------------------|---------------|---------------|-------------|------------------------------|
| Object ID | 0x1052 | PRL Size | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 2 | 0 | UINT16 | NAM index: 0 |
| 03 (Set) | 6 | 0 | UINT16 | NAM index: 0 |
| | | 2 | UINT32 | PRL size |
| 02 (Reply) 04 (Ack) | 6 | 0 | UINT16 | Number of blocks |
| | | 2 | UINT32 | PRL size |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

PRL Write

Writes **PRL** in segments. Set request must be preceded by **Get/Set PRL Size** (page 96). Only after the last segment successful write, PRL is committed to **NV-RAM** and used by modem.

See “**Writing a PRL**” (page 315).

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x1054 | PRL Write | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 6-126 | 0 | UINT16 | NAM index: 0 |
| | | 2 | UINT16 | Block number |
| | | 4 | UINT16 | Data block size. Number of bytes in this block (maximum size is 120 bytes). |
| | | 6 | UINT8 | Block data. Size specified by Data Block Size (above). |
| 04 (Ack) | 2 | 0 | UINT16 | Write result (see Table 5 below.) |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Table 5: Error codes for PRL Write

| Value | Meaning |
|-------|----------------------------------|
| 0 | Success |
| 1 | Failure—SPC is locked |
| 2 | Failure— PRL size not set |
| 3 | Failure—Incorrect NAM set |
| 4 | Failure—Incorrect block sequence |
| 5 | Failure—Incorrect block size |

| Value | Meaning |
|-------|-----------------------------------|
| 6 | Failure—Invalid PRL (CRC failure) |
| 7 | Failure—Invalid write |

Hardware Temperature

Returns the current hardware temperature of the modem.

| | | | | |
|-------------------|---------------|----------------------|-----------------|--|
| Object ID | 0x4004 | Hardware Temperature | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Temperature of the radio in degrees Celsius. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

The modem enforces a disconnection and shutdown if the internal temperature reaches 85°C. This is required to protect the modem and prevent drift from specifications.

Prior to the shutdown, the modem reduces the transmission duty cycle by asserting flow control. This reduction is intended to maintain the connection while allowing the transmitter to cool.

Temperature-imposed flow control uses a 300 ms duty cycle. At each cycle the temperature is checked. If the temperature exceeds 70°C, the off-time (flow control asserted) of the cycle is increased by 3% (9 ms). (The flow is not reduced below 4%.) When the temperature falls to 70°C or below, flow is restored in 3% steps until 100% on-time is restored.

This temperature controlled throttling is intended to maintain as much transmission time as possible without having the connection cut off.

Temperature Threshold

Specifies the temperature at which the modem is determined to be “too hot”.

| | | | | |
|-------------------|---------------|-----------------------|-------------|-----------------------------------|
| Object ID | 0x4501 | Temperature Threshold | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Temperature threshold in Celsius. |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA Temperature Protection object set Release | 1 |

Modem Too Hot Notification

Returns which action the modem takes when it gets too hot.

| | | | | |
|--------------------|---------------|-----------------------|-------------|---|
| Object ID | 0x4500 | Mdm TooHot Indication | | |
| Operations | ✗ Set | ✗ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 2 | 0 | UINT16 | Action Code: <ul style="list-style-type: none"> • 0x0011: Disconnect (End call) • 0x0012: Shut down the modem |

This advises the application of action the modem is taking due to excessive heat in the radio.

- **0x0011**—The modem is going to end the call. The termination process begins after the notification is sent.
- **0x0012**—The modem is initiating its shutdown procedure.

Both actions are triggered when the hardware reaches the temperature threshold. (The threshold can be accessed using [Temperature Threshold-0x4501](#); page 99.) If the modem is in a call, then action 0x0011 is taken; otherwise action 0x0012 is taken.

The temperature can be read using [Hardware Temperature](#) (0x4004) on page 98.

Prior to disconnecting a call, the modem invokes flow control to try to sustain the connection without overheating. See "[Hardware Temperature](#)" on page 98.

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA Temperature Protection object set Release | 1 |

Wakeup Enable

Set or read the event mask to use for the modem to wakeup the host.

| | | | | |
|-------------------|---------------|---------------|-------------|-----------------------------|
| Object ID | 0x5007 | Wakeup Enable | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Bit-mask setting (see text) |
| 03 (Set) | 2 | 0 | UINT16 | |
| 04 (Ack) | 0 | | | None |

The modem can be configured to wake a suspended host by toggling the [RI](#) signal on the primary [port](#). The modem issues the wakeup signal, based on detection of an enabled event.

Events are enabled by setting the appropriate bit in the event mask. If any enabled event is detected, the modem triggers the **RI** signal to wake the host.

| Bit | Meaning if Set |
|-----|---------------------------------------|
| 0 | Wake on ring |
| 1 | Wake on radio (coverage restored) |
| 2 | Wake on MT SMS (SMS message received) |
| 3–F | Reserved |

Following a wakeup signal, the host can use **Wakeup Reason** (page 101) to determine which event triggered the wakeup.

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Wakeup Reason

Read the reason for the last modem wakeup.

| | | | | |
|-------------------|---------------|---------------|-----------------|-----------------------------|
| Object ID | 0x5008 | Wakeup Reason | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Bit-mask setting (see text) |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Following a wakeup signal (**RI** on the primary **port**) from the modem, the host can use this message to determine which event triggered the wakeup.

The reason is given in a bit mask that matches that of the **Wakeup Enable** setting (page 100).

| Bit | Meaning if Set |
|-----|-------------------------------|
| 0 | Ring detected (incoming call) |
| 1 | Radio (coverage restored) |
| 2 | MT SMS (SMS message received) |
| 3–F | Reserved |

A response of zero (0) indicates that the modem did not trigger the wakeup.

The value of this object is cleared (restored to 0 pending another event) when the modem is reset or by setting the enable mask **Wakeup Enable** (0x5007).

7: Configuration

Introduction

This chapter provides reference for [CnS](#) messages used to test and configure the hardware for a particular integration. These commands are **not** for use in end-user applications.

Configuration summary

Table 6: Configuration messages

| Object | ID | Description | S | G | N |
|--|--------|---|---|---|---|
| Get/Set Mic Codec Gain (Mic Level) | 0x1028 | Set the microphone calibration | ✓ | ✓ | ✗ |
| Get/Set Speaker Codec Gain (Speaker Level) | 0x1036 | Calibrate the speaker (headset) output level | ✓ | ✓ | ✗ |
| Get/Set Firmware Autoconnect | 0x1064 | Enable/disable firmware autoconnect | ✓ | ✓ | ✗ |
| IPR2 | 0x5001 | Set or read the data rate on the secondary port | ✓ | ✓ | ✗ |
| Audio Mode | 0x5006 | Set or read the mode of the audio circuit | ✓ | ✓ | ✗ |
| Get/Set IOTA Feature | 0x2104 | Enable/disable IOTA functionality | ✓ | ✓ | ✗ |

Configuration reference



Get/Set Mic Codec Gain (Mic Level)

Gets or sets the current level of the microphone input codec, either quieter than, louder than, or at the factory default level. The current level is stored in non-volatile memory.

| | | | | |
|-------------------|---------------|---------------|-------------|--|
| Object ID | 0x1028 | Mic Level | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Level setting: <ul style="list-style-type: none"> • 0x0001 to 0x0003: Low to high gain below factory default mic codec level • 0x0000: Factory default mic codec level • 0x0004 to 0x0006: Low to high gain above factory default mic codec level In other words, 0x0001 is the lowest mic gain level (three steps below factory default) and 0x0006 is the highest (three steps above default). |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |



Get/Set Speaker Codec Gain (Speaker Level)

Gets or sets the current level of the speaker output codec, either quieter than, louder than, or at the factory default level. The current level is stored in non-volatile memory.

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x1036 | Speaker Level | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 6 | 0 | UINT16 | Level setting: <ul style="list-style-type: none"> • 0x0001 to 0x0003: Low to high gain below factory default speaker output codec level • 0x0000: Factory default speaker level • 0x0004 to 0x0006: Low to high gain above factory default speaker level In other words, 0x0001 is the lowest speaker gain (three steps below factory default) and 0x0006 is the highest (three steps above default). |
| 03 (Set) | 6 | | | Same parameters as 02 (Reply) |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |



Get/Set Firmware Autoconnect

Indicates whether the modem autoconnects to the network when the [NDIS](#) driver is detected.

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x1064 | Autoconnect | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Autoconnect |
| 03 (Set) | 2 | 0 | UINT16 | <ul style="list-style-type: none"> • 0x0000: Disabled • 0x0001: Enabled |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |



IPR2

Set or read the data rate on the secondary [port](#).

| | | | | |
|-------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x5001 | Baud Rate | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT32 | bps rate |
| 03 (Set) | 4 | 0 | UINT32 | |
| 04 (Ack) | 0 | | | None |

Valid baud rates are 2400, 4800, 9600, 19200, 38400, 57600, and 115200. The factory default from Sierra Wireless is 115200.

When the Set Request (0x03) is used, the modem makes the change in the **port** setting **before** responding. The Set Response (0x04) will be issued at the **new** setting, as will all subsequent notifications. A modem reset is not required.

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |



Audio Mode

Set or read the mode of the audio circuit.

| Object ID | 0x5006 | Audio Mode | | |
|------------|--------|------------|----------|---|
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | <p>Mode setting (MC5728V [firmware revision 1.04 and higher]):</p> <ul style="list-style-type: none"> 0x0000: Handset 0x0001: Headset 0x0002: Speaker phone 0x0003: Car kit 0x0004: TTY – full audio 0x0005: HAC (Hearing Aid Compatibility) <p>Mode setting (MC5728V [firmware revision 1.03 and earlier]):</p> <ul style="list-style-type: none"> 0x0000: Handset 0x0001: Headset (default) 0x0002: Speaker phone 0x0003: Car kit 0x0004: TTY – full audio 0x0005: TTY – talk audio 0x0006: TTY – hear audio |

| | | | | |
|-----------------|---|---|--------|--|
| | | | | <ul style="list-style-type: none"> • 0x0007: HAC (Hearing Aid Compatibility) |
| 03 (Set) | 2 | 0 | UINT16 | Same parameters as 02 (Reply) |
| 04 (Ack) | 0 | | | None |

The MC5728V voice circuitry can be operated at either the mic/speaker level for direct connection to a headset, or at line level for connection to other audio circuitry. This message allows you to set and query the mode.

The mode change is immediate; it does not require a modem reset.

Version information:

| | |
|--------------------------------------|-----|
| Object version | N/a |
| Added in CDMA CnS object set Release | 1 |

Get/Set IOTA Feature

Indicates whether the [IOTA](#) feature is enabled or disabled.

This object replaces object ID 0x7004 (the latter is still supported; however, it may be removed, in a future release).

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x2104 | IOTA Feature | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 1 | 0 | UINT8 | Disable Flag: <ul style="list-style-type: none"> • 0x0000: IOTA enabled • 0x0001: IOTA disabled |
| 03 (Set) | 1 | 0 | UINT8 | |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

8: Device Management

OMA Device Management is a device management (DM) protocol specified by the Open Mobile Alliance (OMA) Device Management Working Group and the Data Synchronization (DS) Working Group.

Introduction

This chapter provides reference for [CnS](#) messages used for Device Management.

Device Management summary

Table 7: Device Management messages

| Object | ID | Description | S | G | N |
|---|--------|--|---|---|---|
| DM Configuration | 0x0E00 | Current OMA-DM configuration of the modem | ✓ | ✓ | ✓ |
| Start DM Session | 0x0E01 | Start a User-Client Initiated (U-CI) OMA-DM session | ✓ | ✗ | ✗ |
| Cancel DM Session | 0x0E02 | Cancel a DM session | ✓ | ✗ | ✗ |
| DM Session State | 0x0E03 | State of the DM session | ✗ | ✓ | ✓ |
| Network-Initiated Alert | 0x0E04 | Deny/allow a Network-Initiated Alert queued in the modem. | ✓ | ✓ | ✓ |
| DM DL Progress | 0x0E10 | Status of the DM Download | ✗ | ✓ | ✓ |
| DM Download Alert | 0x0E11 | User response (download and/or install the package) during a FOTA/FUMO operation | ✓ | ✗ | ✓ |
| DM Download Description | 0x0E12 | Description of a FOTA/FUMO download package | ✗ | ✓ | ✗ |
| DM Download — Installation Status | 0x0E13 | Status of the most recent installation attempt of a DM download | ✗ | ✓ | ✓ |

Device Management reference

DM Configuration

Get/set the current [OMA-DM](#) configuration of the modem.

Depending on the modem model, the Set operation may return the error “**Fail: Security access denied**” if the modem is locked; in this case, use [Lock/Unlock Modem](#) (page 290) to unlock the modem, and then perform the Set operation.

| Object ID | 0x0E00 | DM Configuration | | |
|---------------------------|--------|------------------|----------|--|
| Operations | ✓ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 15 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | DM support by the modem: <ul style="list-style-type: none"> • 0x00: The modem does not support DM, or DM support has been disabled via the PRI. • 0x01: Enabled; using the DM library from Sierra Wireless • 0x02: Enabled; using the DM library from Red Bend |
| | | 3 | UINT8 | Maximum number of DM accounts supported (1 to 8) Note: Currently only one DM account is supported. |
| | | 4 | UINT8 | DM account in use (1). |
| | | 5 | BOOL | DM account change flag (valid for Notification only). TRUE if one or more DM accounts were added, replaced, modified, or deleted. |

| | | | | |
|-----------------|----|----|--------|---|
| | | 6 | UINT32 | <p>Bit Map for supported Session Types:</p> <ul style="list-style-type: none"> • Bit 0: CI Device Configuration (CIDC) • Bit 1: NI Device Configuration (NIDC) • Bit 2: CI PRL Update (CIPRL) • Bit 3: NI PRL Update (NIPRL) • Bit 4: CI FUMO (CIFUMO) • Bit 5: NI FUMO (NIFUMO) • Bit 6: Hands Free Activation (HFA) <p>For each bit: 0 = Session Type is not supported; 1 = Session Type is supported.</p> |
| | | 10 | UINT32 | <p>Bit Map for enabled Session Types:</p> <ul style="list-style-type: none"> • Bit 0: CI Device Configuration (CIDC) • Bit 1: NI Device Configuration (NIDC) • Bit 2: CI PRL Update (CIPRL) • Bit 3: NI PRL Update (NIPRL) • Bit 4: CI FUMO (CIFUMO) • Bit 5: NI FUMO (NIFUMO) • Bit 6: Hands Free Activation (HFA) <p>For each bit: 0 = Session Type is not enabled; 1 = Session Type is enabled.</p> |
| | | 14 | BOOL | HFA Attempted flag (deprecated) |
| 03 (Set) | 11 | 0 | UINT16 | Object version |
| | | 2 | UINT32 | <p>Session types.</p> <p>For the Bit Map values, see Bit Map for supported Session Types above.</p> |
| | | 6 | UINT32 | <p>Enabled session types.</p> <p>For the Bit Map values, see Bit Map for enabled Session Types above.</p> |
| | | 10 | BOOL | Clear the HFA Attempted flag (deprecated) |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |

| | | | | |
|--|--|---|-------|---|
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Fail: Reason not specified • 0x02: Fail: DM is disabled or not supported • 0x03: Fail: Invalid parameter • 0x04: Fail: Option not supported • 0x05: Fail: Security access denied |
|--|--|---|-------|---|

Version information:

| | |
|-------------------------------------|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 1 |

Start DM Session

Start an **OMA-DM** session. (This is known as a User-Client Initiated (U-CI) OMA-DM session.)

| | | | | |
|-------------------|------------------|------------------|-------------|--|
| Object ID | 0x0E01 | Start DM Session | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | Variable (≤ 245) | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Reserved. Set this field to 0. |
| | | 4 | UINT8 | User-Client Initiated (U-CI) session type: <ul style="list-style-type: none"> • 0x01: Device Configuration • 0x02: PRL Update • 0x03: FUMO |
| | | 5 | | DM-TLV Account node name. Note: Currently not used. |
| 04 (Ack) | 5 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Reserved. |

| | | | | |
|--|--|---|-------|---|
| | | 4 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Fail: Reason not specified • 0x02: Fail: DM is disabled or not supported • 0x03: Fail: Invalid parameter • 0x04: Fail: Option not supported • 0x05: Fail: Security access denied • 0x06: Fail: Invalid/busy device state • 0x07: Fail: Invalid/busy DM session state |
|--|--|---|-------|---|

Version information:

| | |
|-------------------------------------|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 1 |

Cancel DM Session

Cancel any active or retry-pending (for example, an [HFA](#) retry pending) [OMA-DM](#) session.

If the session is Network Initiated (NI), then the modem may or may not queue a session reattempt; this is determined by the SKU configuration of the modem, and reflected in the Queued for reattempt field. If the [NI](#) session is not queued for a reattempt, then any [NIA](#) associated with this NI DM session is deleted from the modem; to reattempt the same NI session, the network must resend the [NIA](#).

The modem never queues a cancelled Client Initiated (CI) session for a session reattempt; to reattempt the CI session, use [Start DM Session](#) (page 112).

| Object ID | 0x0E02 | Cancel DM Session | | |
|-----------------|---------------|-------------------|----------|-------------|
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

| | | | | |
|-----------------|---|---|--------|--|
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Success • 0x01: No active or retry-pending DM session |
| | | 3 | BOOL | Queued for reattempt |




Version information:

| | |
|-------------------------------------|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 1 |

DM Session State

State of the DM session.

A notification is sent whenever there's a change in the DM session state.

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x0E03 | DM Session State | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 11 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Session state: <ul style="list-style-type: none"> • 0x00: No session active • 0x01: Session active • 0x02: HFA retry-pending; no session active |
| | | 3 | UINT16 | Time to retry HFA — approximate time (in seconds) until a pending HFA is reattempted. Valid only when Session state (the field above) = 0x02. |

| | | |
|----|--------|---|
| 5 | UINT8 | <p>Last session type</p> <ul style="list-style-type: none"> • 0x00: Unspecified • 0x01: Device Configuration • 0x02: PRL Update • 0x03: FUMO • 0x04: Hands Free Activation (HFA) • 0x05: FUMO DD Download • 0x06: FUMO package download • 0x07: Status report |
| 6 | UINT8 | <p>Initiator of the last session</p> <ul style="list-style-type: none"> • 0x00: Unspecified • 0x01: User-Client Initiated (U-CI) • 0x02: Device-Client Initiated (D-CI) • 0x03: Network Initiated (NI) |
| 7 | UINT8 | <p>UI mode of the last session</p> <ul style="list-style-type: none"> • 0x00: Unspecified • 0x01: Hidden session • 0x02: Unhidden session |
| 8 | UINT16 | <p>Result of the last session</p> <ul style="list-style-type: none"> • 0x0000: Success • 0x0001: Unspecified error • 0x0002–0xFFFF: See Table 8 on page 116. These error codes have been defined by Sierra Wireless and do not follow a standardized set of error codes. |
| 10 | UINT8 | <p>Result for the last session. The result is session-specific.</p> <p>CIDC/NIDC/HFA:</p> <ul style="list-style-type: none"> • 0x00: Device Configuration not updated • 0x01: Device Configuration updated <p>CIPRL/NIPRL:</p> <ul style="list-style-type: none"> • 0x00: PRL not updated • 0x01: PRL updated <p>CIFUMO/NIFUMO:</p> <ul style="list-style-type: none"> • 0x00: Firmware package not downloaded • 0x01: Firmware package downloaded |

Version information:

| | |
|-------------------------------------|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 1 |

Table 8: Result codes for the last DM session

| Code | Meaning |
|-----------------------------|--|
| 0x0000 | Success |
| 0x0001 | Unspecified error |
| 0x0002 | Session was established but no commands/operations were received |
| Session abort errors | |
| 0x0100 | Client (user) aborted |
| 0x0101 | Client (device) aborted |
| 0x0102 | Server aborted |
| Memory/buffer errors | |
| 0x0200 | Memory error |
| 0x0201 | Heap buffer too small |
| 0x0202 | Buffer supplied to routine too small |
| Internal errors | |
| 0x0300 | Routine called when not allowed |
| 0x0301 | Routine called when suspended |
| 0x0302 | Badly formatted input parameter |

| Code | Meaning |
|-------------------------------|--|
| SyncML Protocol errors | |
| 0x0400 | Invalid protocol or version |
| 0x0401 | Authentication failure |
| 0x0402 | Missing start message command |
| 0x0403 | Missing end message command |
| 0x0404 | Optional feature not implemented |
| 0x0405 | User interaction alert parsing error |
| 0x0406 | Data too big to pass back as large object |
| Communication errors | |
| 0x0500 | Unsupported protocol |
| 0x0501 | MIME type mismatch |
| 0x0502 | Network (transport) failed to open (except SSL open failure) |
| 0x0503 | SSL failed to open |
| 0x0504 | Unrecoverable network error |
| Management tree errors | |
| 0x1000 | Tree node already exists |
| 0x1001 | Tree node is missing |
| 0x1002 | Error in leaf node operations |
| 0x1003 | Invalid node type |
| 0x1004 | Unknown property |

| Code | Meaning |
|-------------------|---|
| 0x1005 | Permanent node cannot be deleted |
| 0x1006 | Not allowed by access type |
| 0x1007 | Bad program name passed to tree access API |
| 0x1008 | Partial write of external data not allowed |
| 0x1009 | Write of external data not allowed at this time |
| 0x100A | Value not writable |
| 0x100B | Value not readable |
| 0x100C | Cannot execute node |
| 0x100D | Tree open error |
| 0x100E | Tree commit error |
| 0x2000– 0x2FFF | Bootstrap errors |
| 0x3000– 0x3FFF | FUMO errors |

Network-Initiated Alert

Deny/allow a Network-Initiated Alert queued in the modem.

This notification is sent when a new Network Initiated Alert (NIA) is received, or an event (in the modem) triggered the modem to report a previously received NIA that was queued in the modem.

A queued NIA session is processed according to the SKU configuration of the modem. Examples of such a state: during an active data session; the NIA may be queued until the modem enters a dormant data session state and/or the data session is terminated.

For a non-interactive **NI** alert (Background or Informative), a user response is not required, since the modem auto-launches the session being requested in the NIA.

For a standard or special interactive NIA, you can specify, via the Set operation, whether the modem should launch the **NI** session associated with the NIA.

If you reject the NIA (by setting [Interactive response](#) to 0x00), the NIA may or may not be permanently deleted from the NIA queue (for a later reattempt) in the modem, depending on the SKU configuration of the modem.

| Object ID | 0x0E04 | Network-Initiated Alert | | |
|---------------------------|--------|-------------------------|----------|--|
| Operations | ✓ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 6 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | NIA pending: <ul style="list-style-type: none"> • 0x00: No NIA pending • 0x01: Pending but blocked (i.e. device busy) • 0x02: Pending (ready to process) |
| | | 3 | UINT8 | Standard UI mode <ul style="list-style-type: none"> • 0x00: Unspecified • 0x01: Background • 0x02: Informative • 0x03: Interactive |
| | | 4 | UINT8 | Special UI mode (deprecated). <ul style="list-style-type: none"> • 0x00: Unspecified |
| | | 5 | UINT8 | NIA session type <ul style="list-style-type: none"> • 0x00: Unspecified • 0x01: Device Configuration • 0x02: PRL Update • 0x03: FUMO |
| 03 (Set) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Interactive response <ul style="list-style-type: none"> • 0x00: Deny • 0x01: Allow (launch the NI session associated with the NIA) |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |

| | | | |
|--|---|-------|---|
| | 2 | UINT8 | Result: <ul style="list-style-type: none"> • 0x00: Interactive response accepted • 0x01: Fail: Interactive response not expected • 0x02: Fail: Invalid response |
| | 3 | BOOL | Queued for reattempt (valid if Interactive response = 0x00) |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 1 |




DM DL Progress

Get the status of the [DM](#) Download.

During a download session, a notification is sent out multiple times to report the progress of the operation. The interval at which this notification is sent depends on the modem model.

Once a download is complete, the Get response indicates “No download in progress”, at which point the other fields are not valid.

The end of a DL session, due to either failure or completion of the download, is reported by the [DM Session State](#) notification (page 114).

| Object ID | 0x0E10 | DM Download Progress | | |
|-----------------------------------|---|---|--|--|
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 11 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Download type: <ul style="list-style-type: none"> • 0x00: No download in progress • 0x01: FOTA package |

| | | | | |
|--|--|---|--------|---|
| | | 3 | UINT32 | Bytes downloaded in the current download session. Valid only if Download type (above) is "FOTA package". |
| | | 7 | UINT32 | Size of the package (in bytes). Valid only if Download type (above) is "FOTA package". |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 2 |

DM Download Alert

User response (whether or not to download and/or install the package) during a [FOTA/FUMO](#) operation.

The information in the user prompt is package-specific and is specified in the notification payload. To acquire the string description of the package (if available), use [DM Download Description](#) (page 123).

| | | | | |
|-------------------|--|-------------------|-------------|--|
| Object ID | 0x0E11 | DM Download Alert | | |
| Operations | <input checked="" type="checkbox"/> Set <input checked="" type="checkbox"/> Get <input checked="" type="checkbox"/> Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | User response: <ul style="list-style-type: none"> • 0x00: Reject • 0x01: Confirm • 0x02: Delay |
| 04 (Ack) | 3 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Status: <ul style="list-style-type: none"> • 0x00: Success • 0x01: Fail: Reason not specified • 0x02: Fail: Invalid response |

| | | | | |
|--------------------|----------|----|---------|--|
| 07 (Notify) | Variable | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Reserved |
| | | 4 | UINT8 | Available responses (bitmask). Indicates which User response values are valid as a response from host (in the Set request). <ul style="list-style-type: none"> • 0x00: Reject • 0x02: Confirm • 0x04: Delay |
| | | 5 | UINT8 | DL alert type: <ul style="list-style-type: none"> • 0x00: Alert prompt to download and install. • 0x01: Alert prompt to install only (download already completed) |
| | | 6 | UINT8 | Package description availability. Indicates whether the package description is available. If it is, you can use DM Download Description (page 123) to query the description. <ul style="list-style-type: none"> • 0x00: Package description is not available. • 0x01: Package description is available |
| | | 7 | UINT16 | Estimated download time (in seconds). 0 indicates that the parameter was not specified. |
| | | 9 | UINT16 | Estimated installation time (in seconds). 0 indicates that the parameter was not specified. |
| | | 11 | UINT16 | Timeout (in seconds). Indicates when the modem will timeout the response. 0 indicates there is no timeout. |
| | | 13 | UINT16 | Length of the package name (octets); 0–230. |
| | | 15 | UTF8Str | Package name; not null-terminated. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 2 |

DM Download Description

Get a description of a [FOTA/FUMO](#) download package.

| Object ID | 0x0E12 | DM Download Description | | |
|------------|---|---|--|--|
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | Variable | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Reserved |
| | | 4 | UINT16 | Length of the description string (octets); 0–239 0 = The package does not have a description. |
| | | 6 | UTF8Str | Description string; not null-terminated. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 2 |

DM Download — Installation Status

Status of the most recent installation attempt of a [DM](#) download.

| | | | | |
|-----------------------------------|---------------|-----------------------------------|-----------------|--|
| Object ID | 0x0E13 | DM Download — Installation Status | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 7 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Estimated installation time (in seconds). |
| | | 4 | UINT8 | Installation state: <ul style="list-style-type: none"> • 0x00: No FUMO installation attempt has been made. • 0x01: The device is beginning offline installation • 0x02: Offline installation was successful • 0xFF: Error – See Download status code (below) for more information. |
| | | 5 | UINT16 | Download status code. See Table 9 on page 124 (from “OMA-TS-DM_FUMO-V1_0-20070209-A.doc” from Open Mobile Alliance). |

Table 9: Download status codes

| Value | Meaning | Usage |
|---------|-------------------------------|--|
| 200 | Successful | 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 |

| Value | Meaning | Usage |
|---------|---|--|
| 402 | Corrupted Firmware Update Package | Corrupted firmware update package, did not store correctly. Detected, for example, by mismatched CRCs between actual and expected. |
| 403 | Firmware Update Package – 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 was Required but Authentication Failure was encountered when downloading Firmware Update Package |
| 407 | Alternate Download Request Time-Out | Client has encountered a time-out 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 |
| 450–499 | Client Error – Vendor Specified | Client Error encountered for Operation with Vendor Specified ResultCode |
| 500 | Alternate Download Server Error | Alternate Download Server Error Encountered |
| 501 | Download fails due to device is out of memory | The download fails due 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 |

| Value | Meaning | Usage |
|---------|--|--|
| 550–599 | Alternate Download Server Error – Vendor Specified | Alternate Download Server Error encountered for Operation with Vendor Specified ResultCode |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA DM object set Release | 2 |

9: Account Management

Introduction

This chapter provides reference for [CnS](#) messages that relate to activation and account management.

Account summary

Table 10: Account management messages

| Object | ID | Description | S | G | N |
|---|--------|--|---|---|---|
| Phone Number | 0x1002 | Phone number used on the network | X | ✓ | X |
| Activation Status | 0x1009 | Activation status of the modem | X | ✓ | X |
| Set MSL/OTSL Activation Validation Challenge: Step 1 of 2 | 0x1015 | Step 1 to gain access to activating the modem | ✓ | X | X |
| Set MSL/OTSL Activation Validation Challenge: Step 2 of 2 | 0x1016 | Step 2 to gain access to activating the modem | ✓ | X | X |
| Terminate Activation | 0x1017 | Close (commit) the activation process | ✓ | X | X |
| Change Active NAM | 0x101E | Set the NAM account index to use for connections | ✓ | X | X |
| Get Active NAM | 0x101F | Report the active NAM index | X | ✓ | ✓ |
| Get/set NV Item | 0x1033 | Set activation elements | ✓ | X | X |
| Get Activation Date | 0x1037 | Date the current account was activated | X | ✓ | X |
| Get/Set Mobile Directory Number | 0x1038 | Set the MDN | ✓ | ✓ | X |
| Get/Set Mobile MIN | 0x1039 | Set the MIN | ✓ | ✓ | X |
| OTASP State | 0x3014 | Report OTASP progress | X | X | ✓ |

Account reference

Phone Number

Returns the phone number to be used by the modem on the carrier's [CDMA](#) network.

| | | | | |
|-------------------|---------------|---------------|-----------------|-------------------------------------|
| Object ID | 0x1002 | Phone Number | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 10–15 | 0 | CHAR | ASCII string of 10 to 15 characters |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Setting the phone number is handled in the activation process using [Get/Set Mobile Directory Number](#) (0x1038; page 138).

Activation Status

Returns the activation status of the modem.

| | | | | |
|-------------------|---------------|-------------------|-----------------|--|
| Object ID | 0x1009 | Activation Status | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Unsigned 16-bit integer in one of two states: <ul style="list-style-type: none"> 0x0000: Modem is not activated 0x0001: Modem is activated |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Set MSL/OTSL Activation Validation Challenge: Step 1 of 2

This is the first of two steps to open (unlock) an activation session for the modem. Must be followed by [Set MSL/OTSL Activation Validation Challenge: Step 2 of 2](#) (page 130).

| | | | | |
|-------------------|---------------|------------------|-------------|---|
| Object ID | 0x1015 | Secure Challenge | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 8 | | CHAR | Secure Challenge Digits (SCD) string of 8 decimal numerals in ASCII. $SCD = T_0 * K_0$ <ul style="list-style-type: none"> $K_0 = \text{MSL or OTSL}$ (6 digits, or 7 digits for code that is "000000") $T_0 = 2\text{-digit random number (1-99)}$ |
| 04 (Ack) | 10 | | CHAR | Secure Challenge Response (SCR) string of 10 decimal digits. $SCR = (T_1+1)*(SCD)$ <ul style="list-style-type: none"> $T_1 = 2\text{-digit random number (1-99)}$ |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

If the [MSL/OTSL](#) is "000000", you need to add a 1 as the 7th most significant digit and continue the calculation by setting K_0 as "100000".

Set MSL/OTSL Activation Validation Challenge: Step 2 of 2

This is the second of two steps to open (unlock) an activation session for the modem. The Secure Challenge (0x1015; [Set MSL/OTSL Activation Validation Challenge: Step 1 of 2](#)) must precede it.

| | | | | |
|-------------------|---------------|-----------------|-------------|--|
| Object ID | 0x1016 | Secure Rebuttal | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 4 | | CHAR | Secure Rebuttal Digits (SRD) string of 4 decimal numerals in ASCII. $T_1' = SCR / (T_0 * K_0) - 1$ $SRD = T_1' * T_0$ <ul style="list-style-type: none"> • T_1' = Host-calculated T_1 value |
| 04 (Ack) | 2 | | UINT16 | Secure Rebuttal Response (SRR); one of three values: <ul style="list-style-type: none"> • 0x0000: Activation code does not match the OTSL or the MSL, or the matching OTSL has been used. • 0x0001: Activation code matches the OTSL within the modem (and marks it used). • 0x0002: Activation code matches the MSL within the modem. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The values of the [OTSL](#) and [MSL](#) are set at manufacturing time. Sierra Wireless supplies them to the service provider (along with the related [ESN](#)). The service provider in turn provides them to the subscriber during activation.

Terminate Activation

This closes the activation session for the modem and locks it until the challenge and rebuttal is repeated. The Challenge and Rebuttal steps must precede it.

| | | | | |
|-------------------|---------------|----------------------|-------------|---|
| Object ID | 0x1017 | Secure Commit Update | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 4 | 0 | UINT16 | Commit Type is one of two states: <ul style="list-style-type: none"> • 0x0000: Non-provisioning secure commit • 0x0001: Provisioning secure commit |
| | | 2 | UINT16 | NAM Index : 0 |
| 04 (Ack) | 4 | 0 | UINT16 | Secure Commit Result is one of two states: <ul style="list-style-type: none"> • 0x0000: Fail • 0x0001: Succeed (success for all non-provisioning commit) |
| | | 2 | UINT16 | Secure Commit Result Mask Bit describing an error made in provisioning: <ul style="list-style-type: none"> • 0x0000: No error • 0x0001: Mobile Directory Number not provisioned • 0x0002: IMSI not provisioned • 0x0010: NAM profile is not provisioned in the right order |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Change Active NAM

Sets the [NAM](#) index to use for connections.

Note: Dual NAM is currently not supported.

| | | | | |
|-------------------|---------------|---------------|-------------|----------------------------|
| Object ID | 0x101E | Select NAM | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 2 | 0 | UINT16 | Index of the active NAM: 0 |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

This command takes time to execute (beyond the issuing of the acknowledge).
To verify the completion of the command, use [Get Active NAM](#) (described next).

Get Active NAM

Returns the index of the current [NAM](#) profile.

| | | | | |
|-----------------------------------|---------------|---------------|-------------|----------------------------|
| Object ID | 0x101F | Active NAM | | |
| Operations | ✗ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 2 | | | None |
| 02 (Reply) 07 (Notify) | 0 | 0 | UINT16 | Index of the active NAM: 0 |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The notification triggers on any change (or Set command completion) even if set to the same value (no change as a result of Set request). That is, setting NAM 0 while already active on NAM 0 will still trigger the notification.




Get/set NV Item


This object has several variations, based on the parameters used. It can be used to:

- Get information on the PRI revision and HDR SCP configuration
- Set the index of the [NAM](#) profile for making activation settings
- Set the table of [SID](#) and [NID](#) values
- Set the LBS settings.

| Object ID | 0x1033 | NV Items | | |
|------------|--------|----------|----------|---|
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 4 | 0 | UINT16 | 0x0028: Set NAM Index |
| | | 2 | UINT16 | NAM Index for activation settings: 0 |
| 04 (Ack) | 0 | | | None |
| 03 (Set) | 84 | 0 | UINT16 | 0x0103: Set SID and NID table |
| | | 2 | UINT16 | NAM Index |
| | | 4 | UINT16 | SID[0] System ID 1–32767, or 0 for any (wildcard) System ID |
| | | 6 | UINT16 | NID[0] Network ID 0–65534, or 65535 for any (wildcard) NID |
| | | 8 | UINT16 | SID[1] |

| | | | | |
|-------------------|----|-----|--------|---|
| | | 10 | UINT16 | NID[1] |
| | | ... | ... | ... |
| | | 82 | UINT16 | SID[19] |
| | | 84 | UINT16 | NID[19] |
| 04 (Ack) | 0 | | | None |
| 01 (Get) | 2 | 0 | UINT16 | 0x0D82 : HDR SCP configuration |
| 02 (Reply) | 16 | 0 | UINT16 | 0x0D82 : HDR SCP configuration |
| | | 2 | UINT16 | Flag indicating whether SCP custom configuration is active <ul style="list-style-type: none"> • 0: SCP custom configuration is not active • 1: SCP custom configuration is active |
| | | 4 | UINT32 | HDR Protocol Subtypes Enable Bitmask |
| | | 8 | UINT32 | HDR Broadcast Subtypes Enable Bitmask |
| | | 12 | UINT32 | Stream Applications Subtypes Enable Bitmask |
| 01 (Get) | 2 | 0 | UINT16 | 0x6024 : PRI revision You must first use these objects: <ul style="list-style-type: none"> • Set MSL/OTSL Activation Validation Challenge: Step 1 of 2 on page 129, and Set MSL/OTSL Activation Validation Challenge: Step 2 of 2 (page 130), or • Interface Unlock (page 64) |
| 02 (Reply) | 4 | 0 | UINT16 | 0x6024: PRI revision |
| | | 2 | UINT16 | Value of PRI revision |

| | | | | | |
|---|-------------------|---|---|--------|---|
| | 01 (Get) | 2 | 0 | UINT16 | 0x603E: PRI SKU revision You must first use these objects: <ul style="list-style-type: none"> • Set MSL/OTSL Activation Validation Challenge: Step 1 of 2 on page 129, and Set MSL/OTSL Activation Validation Challenge: Step 2 of 2 (page 130), or • Interface Unlock (page 64) |
| | 02 (Reply) | 6 | 0 | UINT16 | 0x603E: PRI SKU revision |
| | | | 2 | UINT32 | Value of PRI SKU revision |
| | 04 (Ack) | 0 | | | None |
|  | 03 (Set) | 3 | 0 | UINT16 | 0x0190: GPS Coarse Location Message (CLM) |
| | | | 2 | UINT8 | <ul style="list-style-type: none"> • 0x0000: Disable CLM • 0x0001: Enable CLM |
| | 04 (Ack) | 0 | | | None |
|  | 03 (Set) | 3 | 0 | UINT16 | 0x0EAC: QUALCOMM GPS Smart Mode (enable/disable bits) |
| | | | 2 | UINT8 | Bit description <ul style="list-style-type: none"> • 0x01: MS-Based Smart Mode • 0x02: Standalone Smart Mode |
| | 04 (Ack) | 0 | | | None |
|  | 03 (Set) | 3 | 0 | UINT16 | 0x1004: GPS NMEA output message type (enable/disable bits) |
| | | | 2 | UINT8 | Message type <ul style="list-style-type: none"> • 0x01: \$GPGGA • 0x02: \$GPGSA • 0x04: \$GPGSV • 0x08: \$GPGVTG • 0x10: \$GPRMC |
| | 04 (Ack) | 0 | | | None |

| | | | | | |
|---|-----------------|---|---|--------|---|
|  | 03 (Set) | 8 | 0 | UINT16 | 0x608D : LBS Fix type |
| | | | 2 | UINT8 | Standalone Mode: <ul style="list-style-type: none"> • 0x00: Disable • 0x01: Enable |
| | | | 3 | UINT8 | MS-Based Mode: <ul style="list-style-type: none"> • 0x00: Disable • 0x01: Enable |
| | | | 4 | UINT8 | MS-Assisted Mode: <ul style="list-style-type: none"> • 0x00: Disable • 0x01: Enable |
| | | | 5 | UINT8 | Optimal Accuracy Mode: <ul style="list-style-type: none"> • 0x00: Disable • 0x01: Enable |
| | | | 6 | UINT8 | Optimal Speed Mode: <ul style="list-style-type: none"> • 0x00: Disable • 0x01: Enable |
| | | | 7 | UINT8 | Optimal Data Mode: <ul style="list-style-type: none"> • 0x00: Disable • 0x01: Enable |
| 04 (Ack) | 0 | | | None | |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Get Activation Date

Returns the activation date. Setting the date is done automatically by the modem on first acquisition of the CDMA network's system time.

| | | | | |
|-------------------|---------------|-------------------|-----------------|--|
| Object ID | 0x1037 | Provisioning Date | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 2 | 0 | UINT16 | NAM index (0) |
| 02 (Reply) | 10 | 0 | UINT16 | NAM Index: 0 |
| | | 2 | CHAR | Provisioning Date: A string of 8 characters in the format YYYYMMDD. All zeros, if the specified NAM has never been provisioned. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Get/Set Mobile Directory Number

Returns or sets the mobile directory number (MDN). Setting the MDN requires provisioning via the Challenge and Rebuttal items. (See [“Set MSL/OTSL Activation Validation Challenge: Step 1 of 2”](#) on page 129.)

For the changes to take effect, the modem must be reset.

| | | | | |
|-------------------|---------------|----------------------|-------------|---|
| Object ID | 0x1038 | Mobile Directory Num | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 2 | 0 | UINT16 | NAM Index: 0 |
| 02 (Reply) | 5–19 | 0 | UINT16 | NAM Index: 0 |
| | | 2 | UINT16 | MDN Len: Length of the mobile directory number |
| | | 4 | CHAR | MDN: A string of at least 1 and no more than 15 digits |
| 03 (Set) | 5–19 | | | Same parameters as 02 (Reply) |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Get/Set Mobile MIN

Returns or sets the **MIN** (also known as the MSIN). Only the operating MIN is returned using a Get request. Setting the **MDN** requires provisioning via the Challenge and Rebuttal items. (See “[Set MSL/OTSL Activation Validation Challenge: Step 1 of 2](#)” on page 129.) For the changes to take effect, the modem must be reset.

| | | | | |
|-------------------|------------|---------------|-------------|--|
| Object ID | 0x1039 | IMSI | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 2 | 0 | UINT16 | NAM Index: 0 |
| 02 (Reply) | 5–14 | 0 | UINT16 | NAM Index: 0 |
| | | 2 | UINT16 | MIN Len: Length of the MIN |
| | | 4 | CHAR | MIN: A string of at least 1 and no more than 10 digits. The NMSI digits of a CDMA IMSI . (Refer to IS-95B Standard). |
| 03 (Set) | 5–14 | | | Same parameters as 02 (Reply) |

*Note: Assumes that the **MCC** and **MNC** of the CDMA **IMSI** are already programmed at manufacturing time.*

If the length of **MIN** is less than 10, zeros are padded by the modem.

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

10: Network Information

Introduction

This chapter provides reference for CnS messages that relate to information about the network and radio signal. It is further divided into two sections:

- [Radio](#)
- [Network service](#)

Radio summary

Table 11: Radio signal information

| Object | ID | Description | S | G | N |
|---------------------------------|--------|--|---|---|---|
| RSSI | 0x1001 | Received Signal Strength Indicator | X | ✓ | ✓ |
| Channel Number | 0x1004 | Current active channel number | X | ✓ | ✓ |
| Channel State | 0x1005 | Current channel acquisition state | X | ✓ | ✓ |
| Ec/Io Values | 0x100A | Ec/Io in dBm | X | ✓ | X |
| RSSI and Ec/Io for CDMA and HDR | 0x1065 | RSSI and Ec/Io values for CDMA and HDR service | X | ✓ | ✓ |
| HDR RSSI | 0x1067 | HDR RSSI value in positive dBm units | X | ✓ | ✓ |
| Frame Error Rate | 0x4005 | Report the frame error rate | X | ✓ | ✓ |
| Current Band Class | 0x4008 | Band of the current radio tuning (cellular or PCS) | X | ✓ | ✓ |
| HDR Sector | 0x401E | HDR sector information | X | X | ✓ |
| HDR Channel | 0x4020 | HDR channel | X | X | ✓ |
| HDR RF Statistics | 0x4026 | HDR RF related statistics | X | X | ✓ |

Network summary




Table 12: Network information

| Object | ID | Description | S | G | N |
|---------------------------------------|--------|--|---|---|---|
| Service Indication | 0x1006 | Type of network service that is currently available | X | ✓ | ✓ |
| Roaming Status | 0x1007 | Roaming state | X | ✓ | ✓ |
| Protocol Revision Number | 0x101C | Protocol revision (PREV) of the current network service | X | ✓ | ✓ |
| Current CDMA System Time | 0x1032 | Local time from the CDMA network | X | ✓ | ✓ |
| HDR Miscellaneous Notification Status | 0x1066 | Enable or disable a block of HDR Network information notifications | X | ✓ | X |
| HDR Roam Status | 0x1068 | Current roam status for HDR service | X | ✓ | ✓ |
| HDR Service State | 0x1069 | Status of the HDR service | X | ✓ | ✓ |
| HDR Hybrid Preference | 0x106A | Hybrid mode preference | ✓ | ✓ | ✓ |
| HDR Idle Digital Mode | 0x106B | Indicates whether a change in idle digital mode has occurred | X | X | ✓ |
| System Selection | 0x1070 | System selection | ✓ | X | X |
| SID Value | 0x4001 | Current network SID | X | ✓ | ✓ |
| NID Value | 0x4002 | Current network NID | X | ✓ | ✓ |
| Base Station Information | 0x4003 | Location of the serving base station | X | ✓ | ✓ |
| HDR Protocol Revision Number | 0x401D | Minimum and maximum HDR Protocol revisions supported by the current network. | X | ✓ | ✓ |

Radio reference

RSSI

Returns the current Received Signal Strength Indication (RSSI) as a positive dBm value.

| | | | | |
|---------------------------|---|---|--|-------------|
| Object ID | 0x1001 | RSSI | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | RSSI |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The modem issues the RSSI notification at intervals of one (1) second during normal operations.

CDMA does not have a formal RSSI value. The value reported is calculated by adding the received signal strength and the [Ec/Io](#) values. This applies a signal-to-noise metric to the received strength to give a better indication of signal quality.

The modem applies an IIR filter (averaging the last five readings) before delivering the value to the application. This removes jitter from the reading to permit smooth display in a user interface.




Interpreting the value as a five bar metric— common on mobile devices—is up to you. Values used by the software are shown in the following table.

Table 13: Signal strength interpretations

| Bars | Sierra Wireless Watcher™ | Sprint Connection Manager SM |
|------|-------------------------------|---|
| 1 | < -90 (as long as in service) | < -105 |
| 2 | -90 to -86 | -105 to -96 |
| 3 | -85 to -81 | -95 to -86 |
| 4 | -80 to -76 | -85 to -76 |
| 5 | >= -75 | >= -75 |

Channel Number

Returns the current 1X active channel number, or zero if digital service is not available.

| | | | | |
|-----------------------------------|---|---|--|--------------------|
| Object ID | 0x1004 | Channel Number | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Channel Number |

Notifications are triggered with any channel change, including channel scanning.

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Channel State

Returns the current 1X channel acquisition state for the modem.

| | | | | |
|-----------------------------------|---------------|---------------|-----------------|---|
| Object ID | 0x1005 | Channel State | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Unsigned 16-bit integer in one of three states: <ul style="list-style-type: none"> • 0x0000: Channel not acquired • 0x0001: Channel acquired • 0x0005: Scanning for channel |

Notifications are not based on channel changes; a wider view of channel acquisition is taken. This uses the CDMA definitions.

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Ec/lo Values

Returns Ec/lo in 0.5 dBm units.

| | | | | |
|-------------------|---------------|---------------|-----------------|--|
| Object ID | 0x100A | ECIO | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Ec/lo value as unsigned 16-bit integer (representing negative 0.5 dBm units) |

The values are always taken to be negative. The sign is not included, so the reply is an “absolute value”.

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

RSSI and Ec/lo for CDMA and HDR

Provides the [RSSI](#) and [Ec/lo](#) values for both [CDMA](#) and [HDR](#) services.

| | | | | |
|-----------------------------------|---------------|---------------------------------|-----------------|---|
| Object ID | 0x1065 | RSSI and Ec/lo for CDMA and HDR | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 8 | 0 | UINT16 | CDMA RSSI |
| | | 2 | UINT16 | CDMA Ec/lo value as unsigned 16-bit integer (representing negative 0.5 dBm units) |
| | | 4 | UINT16 | HDR RSSI |
| | | 6 | UINT16 | HDR Ec/lo value as unsigned 16-bit integer (representing negative 0.5 dBm units) |

Version information:




| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The RSSI values reported are calculated by adding the received signal strength and the Ec/lo values. This applies a signal-to-noise metric to the received strength to give a better indication of signal quality.

The modem applies an IIR filter (averaging the last five readings) before delivering the value to the application. This removes jitter from the reading to permit smooth display in a user interface.

HDR RSSI

Provides the [RSSI](#) (in positive [dBm](#) units) for [HDR](#) service. Notification of this data is enabled via [HDR Miscellaneous Notification Status](#) (page 154).

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x1067 | HDR RSSI | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | RSSI (in positive dBm units) |




Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Frame Error Rate

Note: This is available only during IS-95 data calls.

Returns the [frame](#) count and number of bad frames to calculate the frame error rate.

| | | | | |
|-----------------------------------|---|---|--|--------------------|
| Object ID | 0x4005 | Tech FER | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 8 | 0 | UINT32 | Total frames |
| | | 4 | UINT32 | Bad frames |




Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

Current Band Class

Returns the current band class of the modem.

To set the band preference, see [System Selection](#) (page 159).

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x4008 | Tech Bandclass | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Current tuning band: <ul style="list-style-type: none"> • 0x0000: Cellular band • 0x0001: PCS band |

Notifications trigger on [SID/NID](#) network updates.

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Sector

Indicates that the current [HDR](#) sector information has changed. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).

| | | | | |
|--------------------|---------------|-----------------|-----------------|--|
| Object ID | 0x401E | Tech HDR Sector | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 22 | 0 | CHAR | Sector ID, string of 16 characters (not null-terminated) |
| | | 16 | UINT16 | Subnet mask |
| | | 18 | UINT16 | Color code |
| | | 20 | UINT16 | Pn offset |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Channel

Indicates that the current [HDR](#) channel has changed. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).

| | | | | |
|--------------------|---------------|------------------|-----------------|--|
| Object ID | 0x4020 | Tech HDR Channel | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 2 | 0 | UINT16 | HDR channel. 0xFFFF indicates no service. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR RF Statistics

Indicates that the current [HDR RF](#) related information has changed. This is sent once every second when the object notification flag is enabled.

| | | | | |
|--------------------|---------------|------------------------|-----------------|---|
| Object ID | 0x4026 | Tech HDR RF Statistics | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 14 | 0 | INT16 | rx_agc0—HDR receiver gain control for Antenna 0, main path, range is +/-128 |
| | | 2 | INT16 | rx_agc—HDR receiver gain control for Antenna 1, diversity path, range is +/-128 |
| | | 4 | INT16 | tx_total—HDR TX Total Power in dbm |
| | | 6 | INT16 | tx_adj—HDR TX gain adjust in dbm |
| | | 8 | UINT16 | c2i—HDR C/I carrier-to-interference ratio |
| | | 10 | UINT16 | HDR Ec/lo (in 0.5 dBm units) |
| | | 12 | UINT16 | HDR RSSI |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |




Network reference

Service Indication

Indicates which type of network service is currently available to the modem.

To determine which service indicator to display in the GUI, see page 190.

To set the preferences used for system selection, see [System Selection](#) (page 159).

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x1006 | Service Indication | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Unsigned 16-bit integer in one of three states: <ul style="list-style-type: none"> • 0x0000: No service • 0x0002: Digital CDMA service • 0x0003: GPS service |

Version information:




| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The modem issues the Service Indication notification at intervals of one (1) second during normal operations if the new value is different from the one last reported. Otherwise, notification is suppressed.

Roaming Status

Indicates whether [roaming](#) is available, and if so, whether [SIDs](#) are guaranteed valid.

To set the roaming preference, see [System Selection](#) (page 159).

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x1007 | Roaming Status | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Unsigned 16-bit integer: <ul style="list-style-type: none"> • 0x0000: Not roaming • 0x0001: Roaming with guaranteed valid SIDs • 0x0002: Roaming without guaranteed SIDs • > 0x0002: Roaming; the roaming information is defined by the ERI file. To obtain the ERI banner, icon state, and icon image, parse the carrier's ERI file. |

Version information:




| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Notifications are triggered whenever a new [SID](#) is acquired.

Protocol Revision Number

Returns the [protocol](#) revision (P_REV) number reported by the current base station with which the modem is communicating.

To determine which service indicator to display in the GUI, see page [190](#).

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x101C | PREV | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Protocol revision: <ul style="list-style-type: none"> • 0x0001: JSTD-008 (PCS) • 0x0002: IS-95 • 0x0003: IS-95A (cellular) • 0x0004: Minimum requirements for IS-95B • 0x0005: Full requirements for IS-95B • 0x0006: CDMA 1X Rev. 0 • 0x0007: CDMA 1X Rev. 1 |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Notifications are triggered by any update of the PREV on the network.

Current CDMA System Time

Returns the system time in the current [CDMA](#) network.

| | | | | |
|-----------------------------------|---------------|---------------|-----------------|---|
| Object ID | 0x1032 | System Time | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 14 | 0 | UINT16 | Year (for example, 2005) |
| | | 2 | UINT16 | Month (for example, Dec = 12) |
| | | 4 | UINT16 | Day (for example, 31 st = 31) |
| | | 6 | UINT16 | Day of the week (for example, Monday = 1 through Sunday = 7) |
| | | 8 | UINT16 | Hour on a 24-hour clock (for example, 10 pm = 22) |
| | | 10 | UINT16 | Minutes |
| | | 12 | UINT16 | Seconds |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The time zone is local, as derived from [UTC](#) with an offset determined by the network.

This value can help travelers to set their computer system clock to local time.

HDR Miscellaneous Notification Status

Enables or disables notification for the following message objects, based on the bit flags:

- [HDR RSSI](#) (page 146)
- [HDR Roam Status](#) (page 155)
- [HDR Service State](#) (page 156)
- [HDR Hybrid Preference](#) (page 156)
- [HDR Idle Digital Mode](#) (page 158)
- [HDR DDTM Preference](#) (page 169)

This object does not issue notification [packets](#) (0x07) itself. Rather, it controls the enabling and disabling of a group of object notifications.

| | | | | |
|-------------------|------------------|---|---------------------|--|
| Object ID | 0x1066 | HDR Notification Status | | |
| Operations | X Set | <input checked="" type="checkbox"/> Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Bit Map for HDR Miscellaneous Notifications : Bit 0: HDR RSSI (page 145) Bit 1: HDR Roam Status (page 155) Bit 2: HDR Service State (page 156) Bit 3: HDR Hybrid Preference (page 157) Bit 4: HDR Idle Digital Mode (page 158) Bit 5: HDR DDTM Preference (page 169) Bit 6 thru Bit 15: Reserved for future use. For each bit: 0 = notification disabled; 1 = notification enabled. |




Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

HDR Roam Status

Indicates the current roam status for HDR service. Notification of this data is enabled via [HDR Miscellaneous Notification Status](#) (page 154).

To set the roaming preference, see [System Selection](#) (page 159).

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x1068 | HDR Roaming Status | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Roaming status <ul style="list-style-type: none"> • 0x0000: No Roaming • 0x0001: Roaming with guaranteed valid SIDs • 0x0002: Roaming without guaranteed SIDs • > 0x0002: Roaming; the roaming information is defined by the ERI file. To obtain the ERI banner, icon state, and icon image, parse the carrier's ERI file. |

Version information:




| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

HDR Service State

Indicates the current service status for HDR service. Notification of this data is enabled via [HDR Miscellaneous Notification Status](#) (page 154).

To determine which service indicator to display in the GUI, see page 190.

To set the preferences used for system selection, see [System Selection](#) (page 159).

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x1069 | HDR Service State | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | HDR status <ul style="list-style-type: none"> • 0x0000: No service • 0x0002: Digital service (1xEV-DO Rev. 0) • 0x0004: Digital service (1xEV-DO Rev. A) |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

HDR Hybrid Preference

Indicates the hybrid mode preference. Notification of this data is enabled [HDR Miscellaneous Notification Status](#) (page 154).

To determine which service indicator to display in the GUI, see page [190](#).

| | | | | |
|-----------------------------------|---------------|-----------------------|-------------|---|
| Object ID | 0x106A | HDR Hybrid Preference | | |
| Operations | ✓ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 03 (Set) | 2 | 0 | UINT16 | Hybrid mode preference |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | <ul style="list-style-type: none"> • 0x0000: Hybrid mode off • 0x0001: Hybrid mode on |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

HDR Idle Digital Mode

Indicates whether a change in idle digital mode has occurred. Notification of this data is enabled via [HDR Miscellaneous Notification Status](#) (page 154).

| | | | | |
|--------------------|---------------|--|-------------|---|
| Object ID | 0x106B | HDR Previous and Current Idle Digital Mode | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4 | 0 | UINT16 | Previous mode • 0x0000 : No service • 0x0001 : AMPS • 0x0002 : CDMA • 0x0003 : GSM • 0x0004 : HDR • 0x0005 : WCDMA • 0x0006 : GPS |
| | | 2 | UINT16 | Current mode Supported values are the same as for "Previous mode" (above). |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

System Selection

Set the preferences used for system selection.

| | | | | |
|------------------------------|---------------|------------------|-------------|--|
| Object ID | 0x1070 | System Selection | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) 04 (Ack) | 14 | 0 | UINT16 | Mode preference <ul style="list-style-type: none"> • 0x0000: Analog only • 0x0001: Digital only • 0x0002: Automatic • 0x0003: E911 • 0x0004: PCS CDMA only • 0x0005: Cell CDMA only • 0x0009: CDMA only • 0x000A: HDR only • 0x000B: CDMA AMPS only • 0x000C: GPS only • 0x000D: GSM only • 0x000E: WCDMA only • 0x000F: Persistent mode • 0x0010: No change |
| | | 2 | UINT16 | Mode in effect until: <ul style="list-style-type: none"> • 0x0000: Permanent mode change • 0x0001: Until power cycle of the modem • 0x0002: Until end of next call or power cycle • 0x0003: Until end of next call, specified timeout (see the next field) or power cycle |
| | | 4 | UINT32 | Mode duration, in seconds |
| | | 8 | UINT32 | Band preference: <ul style="list-style-type: none"> • 0x00000000: None • 0x00000001: Band Class 0, A-side systems only • 0x00000002: Band Class 0, B-side systems only • 0x00000003: Band Class 0 • 0x00000004: Band Class 1 • 0x00000010: Band Class 3 |




| | | | | |
|--|--|----|--------|--|
| | | | | <ul style="list-style-type: none"> • 0x00000020: Band Class 4 • 0x00000040: Band Class 5 • 0x00000080: Band Class 6 • 0x0000FFFF: Any • 0x00010000: No change |
| | | 12 | UINT16 | Roaming preference: <ul style="list-style-type: none"> • 0x0001: Home • 0x0003: Acquire only systems for which the roam indicator is either solid or off • 0x00FF: Any • 0x0100: No change |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

SID Value

Returns the current System Identifier (SID) of the network providing service.

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x4001 | Tech SID | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | SID from 1–32767, or 0 for any (wildcard) System ID |

Version information:

| | |
|---------------------------------------|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

Notifications are triggered by any update of the SID on the network.

NID Value

Returns the current Network Identifier ([NID](#)) of the station providing service.

| | | | | |
|-----------------------------------|---------------|---------------|-----------------|--|
| Object ID | 0x4002 | Tech NID | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | NID from 0–65534, or 65535 for any (wildcard) Network ID |

Notifications are triggered by any update of the [NID](#) on the network.

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

Base Station Information

Returns the location information (in geographical seconds) of the network base station with which the modem is communicating.

| | | | | |
|-----------------------------------|---------------|------------------------------|-----------------|--------------------|
| Object ID | 0x4003 | Tech BS Info | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 8 | 0 | INT32 | Longitude |
| | | 4 | INT32 | Latitude |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |




The following algorithm converts either value to degrees, minutes, and seconds:

```
Value ~longitude/latitude.
int32 gSec;          /* geographical seconds */
                    /* Calculate BS value in degrees/min/secs */
gSec = value / 4;
bs->degrees = gSec / 3600;
bs->minutes = ( gSec - bs->degrees * 3600 ) / 60;
bs->seconds=gSec - bs->degrees * 3600 - bs->minutes * 60;
```

HDR Protocol Revision Number

Indicates the minimum and maximum [HDR](#) Protocol revisions supported by the current network.

Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x401D | Tech HDR Minimum and Maximum PREV | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 4 | 0 | UINT16 | Minimum HDR Protocol revision supported by the current network. |
| | | 2 | UINT16 | Maximum HDR Protocol revision supported by the current network. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

11: Data Services

Introduction

This chapter provides reference for [CnS](#) messages that are related to data connection management and services.

There are a variety of additional messages listed in the chapter "[Voice Call Management](#)", under "[Call management summary](#)" (page 192), that have some application in data calls. Those objects are also listed in the summary below but the details are found in "[Voice Call Management](#)".

Data summary

Table 14: Data call management

| Object | ID | Description | S | G | N |
|--|--------|---|---|---|---|
| Get/Set Answer Mode | 0x1014 | Control how the modem answers incoming calls | ✓ | ✓ | ✗ |
| Mobile IP Error Code | 0x1056 | Error code of a Mobile IP connection | ✗ | ✓ | ✗ |
| HDR DDTM Preference | 0x106C | Preference for data dedicated transmission mode | ✓ | ✓ | ✓ |
| Close HDR Session | 0x106D | Close HDR session | ✓ | ✗ | ✗ |
| Call Notification Status | 0x3000 | Enable or disable a block of call management notifications. (See " Voice Call Management ", starting on page 192) | ✗ | ✓ | ✗ |
| Bytes Sent and Received | 0x3001 | Amount of traffic on the current data connection | ✗ | ✓ | ✓ |
| Call Duration | 0x3003 | Duration of the current call in seconds. (See " Voice Call Management ", starting on page 192) | ✗ | ✓ | ✗ |
| Incoming Call | 0x3004 | Notify of an incoming call and its type. (See " Voice Call Management ", starting on page 192) | ✗ | ✗ | ✓ |
| Caller ID Information Received | 0x3006 | Basic caller ID information for the current call. (See " Voice Call Management ", starting on page 192) | ✗ | ✓ | ✓ |

| Object | ID | Description | S | G | N |
|---|--------|--|---|---|---|
| Extended Caller ID Information | 0x3009 | Extended caller ID information for the current call. (See "Voice Call Management", starting on page 192) | X | ✓ | ✓ |
| Call Connected | 0x300A | Notify of a connected call. (See "Voice Call Management", starting on page 192) | X | X | ✓ |
| Connection State | 0x300B | Current status of the connection. (See "Voice Call Management", starting on page 192) | X | ✓ | X |
| Call Disconnected | 0x300C | Notify of a disconnected call. (See "Voice Call Management", starting on page 192) | X | X | ✓ |
| Connection Error | 0x300E | Notify of a connection error. (See "Voice Call Management", starting on page 192) | X | X | ✓ |
| Call Connecting | 0x3011 | Notify of an attempt to originate a call (See "Voice Call Management", starting on page 192) | X | X | ✓ |
| Dormant Packet Call | 0x3012 | Notify of entering dormancy during a packet connection | X | X | ✓ |
| KB Sent and Received | 0x3015 | Cumulative KB counter | ✓ | ✓ | X |
| HDR Tech Notification Status | 0x401C | Enable or disable a block of HDR related notifications | X | ✓ | X |
| HDR State | 0x401F | HDR AT and session state | X | ✓ | ✓ |
| HDR Session Info | 0x4021 | HDR session information | X | ✓ | ✓ |
| HDR Authentication Status | 0x4022 | HDR Authentication State | X | ✓ | ✓ |
| HDR DRC Value | 0x4023 | HDR DRC value (data rate control) | X | ✓ | ✓ |
| HDR DRC Cover | 0x4024 | Get HDR DRC cover | X | ✓ | ✓ |
| HDR RRI | 0x4025 | Get HDR RRI (reverse rate indicator) | X | ✓ | ✓ |
| HDR Single-user Forward statistics | 0x4027 | Single-user Forward Traffic Channel statistics | ✓ | X | ✓ |
| HDR Single-User Forward Traffic Channel DRC packet termination slot count | 0x4028 | Forward Traffic Channel packet termination slot count for the first ten supported DRCs | X | X | ✓ |

| Object | ID | Description | S | G | N |
|---|--------|---|---|---|---|
| HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count | 0x4029 | Forward Traffic Channel and Control Channel packet termination slot count for DRCs 10 through 14, and Control Channel packet termination slot count for both supported DRCs | X | X | ✓ |
| HDR Multi-user Forward statistics | 0x402A | Multi-user Forward Traffic Channel statistics | ✓ | X | ✓ |
| Current personality data | 0x402B | Current personality index and negotiated protocol subtypes for this personality | X | ✓ | ✓ |
| Negotiated protocols and application subtypes | 0x402C | Current personality, number of stored personalities, negotiated protocol subtypes for all protocols in all negotiated personalities, and negotiated application subtypes for all four streams | X | ✓ | X |
| Set Dormant | 0x500B | Directs the modem to go dormant | ✓ | X | X |

Data reference

Get/Set Answer Mode

Gets or sets whether the modem answers incoming calls as data, modem, fax, or voice, and whether that setting applies to all calls or only the next one (after which it returns to voice mode).

| | | | | |
|-------------------|---------------|---------------|-------------|--|
| Object ID | 0x1014 | Answer State | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Unsigned 16-bit integer in one of six answer states: <ul style="list-style-type: none"> • 0x0000: all calls as voice • 0x0001: as fax (next call only) • 0x0002: as fax (all calls) • 0x0003: as data (next call only) • 0x0004: as data (all calls) • 0x0005: disable auto-answer |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

*Note: "Next call only" is a setting that holds for **up to ten minutes** or until a call arrives, whichever comes first. If no call is received within the ten-minute interval, the modem reverts to answering all calls as voice.*

Mobile IP Error Code

Returns the network [registration](#) code for Mobile IP connections.

| | | | | |
|-------------------|---------------|----------------|-----------------|---|
| Object ID | 0x1056 | MIP Error Code | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | MIP network registration error code. (If DMU is present, the DMU error code will be present after the MIP error code) |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Returns 0xFFFF if the last call was not a Mobile IP connection.

The error codes (in decimal) are listed in the following table. For additional information, consult the referenced specification [RFC](#).

Table 15: [MIP](#) error codes

| Code | Meaning | RFC |
|---|---|------|
| 0 | Registration accepted | 2002 |
| 1 | Registration accepted, but simultaneous mobility bindings unsupported | 2002 |
| Registration denied by foreign agent | | |
| 64 | Reason unspecified | 2002 |
| 65 | Administratively prohibited | 2002 |
| 66 | Insufficient resources | 2002 |
| 67 | Mobile node failed authentication | 2002 |
| 68 | Home agent failed authentication | 2002 |
| 69 | Requested Lifetime too long | 2002 |
| 70 | Poorly formed request | 2002 |

| Code | Meaning | RFC |
|--|--|-------|
| 71 | Poorly formed reply | 2002 |
| 72 | Requested encapsulation unavailable | 2002 |
| 73 | Reserved and unavailable | 2002 |
| 74 | Requested reverse tunnel unavailable | 2344 |
| 75 | Reverse tunnel is mandatory and "T" bit not set | 2344 |
| 76 | Mobile node too distant | 2344 |
| 77 | Invalid "care-of" address | 2002b |
| 78 | Registration timeout | 2002b |
| 80 | Home network unreachable (ICMP error received) | 2002 |
| 81 | Home agent host unreachable (ICMP error received) | 2002 |
| 82 | Home agent port unreachable (ICMP error received) | 2002 |
| 88 | Home agent unreachable (other ICMP error received) | 2002 |
| 104 | Unknown challenge | 3012 |
| 105 | Missing challenge | 3012 |
| 106 | Stale challenge | 3012 |
| Registration denied by the home agent | | |
| 128 | Reason unspecified | 2002 |
| 129 | Administratively prohibited | 2002 |
| 130 | Insufficient resources | 2002 |
| 131 | Mobile node failed authentication | 2002 |
| 132 | Foreign agent failed authentication | 2002 |
| 133 | Registration identification mismatch | 2002 |
| 134 | Poorly formed request | 2002 |
| 135 | Too many simultaneous mobility bindings | 2002 |
| 136 | Unknown home agent address | 2002 |
| 137 | Requested reverse tunnel unavailable | 2344 |

| Code | Meaning | RFC |
|------|---|------|
| 138 | Reverse tunnel is mandatory and "T" bit not set | 2344 |
| 139 | Requested encapsulation unavailable | 2344 |

HDR DDTM Preference

Indicates preference for data dedicated transmission mode. Notification of this data is enabled via [HDR Miscellaneous Notification Status](#) (page 154).

| Object ID | 0x106C | Data Dedicated Transmission Mode for Hybrid | | |
|-------------|---|---|--|---|
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | HDR DDTM Preference <ul style="list-style-type: none"> • 0x0000: Prefer data dedicated mode off • 0x0001: Prefer data dedicated mode on |
| 03 (Set) | 0 | | | None |
| 07 (Notify) | 2 | 0 | UINT16 | HDR DDTM Preference <ul style="list-style-type: none"> • 0x0000: Prefer data dedicated mode off • 0x0001: Prefer data dedicated mode on |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Close HDR Session

Commands the modem to close the active HDR session.

| | | | | |
|-------------------|---------------|-------------------|-------------|--------------------|
| Object ID | 0x106D | Close HDR Session | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Bytes Sent and Received

Counts the total number of bytes that have been transmitted and received while a data call is active. The counters are reset after the call ends.

| | | | | |
|-----------------------------------|---------------|-------------------|-------------|--------------------------------------|
| Object ID | 0x3001 | Call Byte Counter | | |
| Operations | ✗ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 8 | 0 | UINT32 | TxCounter : bytes transmitted |
| | | 4 | UINT32 | RxCounter : bytes received |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The modem issues the notification, updating the counter values, at intervals of one (1) second during normal operations.

For cumulative figures on data sent and received, use [KB Sent and Received](#) (page 172).

Dormant Packet Call

Notification that the modem has gone [dormant](#) during a packet connection.

| | | | | |
|--------------------|---------------|---------------|-----------------|--|
| Object ID | 0x3012 | Call Dormant | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4 | 0 | UINT16 | Call State: See Table 21 on page 202. |
| | | 2 | UINT16 | End Reason: See Table 22 on page 204. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled using the [Call Notification Status](#) message (page 196).

Not all Call State or End Reason values are valid for packet calls.

The notification is given when a packet call **enters** dormancy. When the call leaves dormancy, this notification is **not** given. Instead the modem issues either:

- [Call Connecting](#) (page 210) and [Call Connected](#) (page 201) for dormant calls that are restoring on the existing PPP connection.
- [Incoming Call](#) (page 197) and [Call Connected](#) (page 201) for dormant calls that are reinitiating PPP.

KB Sent and Received

Get or reset the cumulative KB counter.

| | | | | |
|-------------------|---------------|----------------------|-------------|---------------------------------|
| Object ID | 0x3015 | KB Sent and Received | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 8 | 0 | UINT32 | KB transmitted |
| | | 4 | UINT32 | KB received |
| 03 (Set) | 0 | | | Reset the cumulative KB counter |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

For information on data sent and received in the current data call, use [Bytes Sent and Received](#) (page 170).

HDR Tech Notification Status

Enables or disables notification for the following message objects, based on the bit flags.

- [HDR Protocol Revision Number](#) (page 162)
- [HDR Sector](#) (page 148)
- [HDR AT and Session State](#) (page 174)
- [HDR Channel](#) (page 148)
- [HDR Session Info](#) (page 175)
- [HDR Authentication Status](#) (page 176)
- [HDR DRC Value](#) (page 176)
- [HDR DRC Cover](#) (page 177)
- [HDR RRI](#) (page 178)
- [HDR RF Statistics](#) (page 149)

- [HDR Single-user Forward statistics](#) (page 179)
- [HDR Single-User Forward Traffic Channel DRC packet termination slot count](#) (page 180), and [HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count](#) (page 182)
- [HDR Multi-user Forward statistics](#) (page 183)
- [Current personality data](#) (page 186)

This object does not issue notifications (0x07) itself. Rather, it controls the enabling and disabling of a group of object notifications.

| | | | | |
|-------------------|---------------|------------------------------|-----------------|---|
| Object ID | 0x401C | Tech HDR Notification Status | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Bit Map for Tech HDR Notifications: Bit 0: HDR Protocol Revision Number (page 162) Bit 1: HDR Sector (page 148) Bit 2: HDR AT and Session State (page 172) Bit 3: HDR State (page 148) Bit 4: HDR Session Info (page 175) Bit 5: HDR Authentication Status (page 176) Bit 6: HDR DRC Value (page 176) Bit 7: HDR DRC Cover (page 177) Bit 8: HDR RRI (page 178) Bit 9: HDR RF Statistics (page 149) Bit 10: HDR Single-user Forward statistics (page 179) Bit 11: HDR Single-User Forward Traffic Channel DRC packet termination slot count (page 180), and HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count (page 182) Bit 12: HDR Multi-user Forward statistics (page 183) Bit 13: Current personality data (page 186) Bit 14 thru Bit 15: Reserved for future use. |




| | | | | |
|--|--|--|--|---|
| | | | | For each bit: 0 = notification disabled; 1 = notification enabled. |
|--|--|--|--|---|

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR State

Indicates that the current AT and HDR session states have changed. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x401F | Tech HDR State | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 4 | 0 | UINT16 | Access Terminal state <ul style="list-style-type: none"> • 0x0000: Inactive • 0x0001: Acquisition • 0x0002: Sync • 0x0003: Idle • 0x0004: Access • 0x0005: Connected |
| | | 2 | UINT16 | HDR session state <ul style="list-style-type: none"> • 0x0000: Close • 0x0001: Address Management Protocol set up • 0x0002: AT initiated • 0x0003: AN initiated • 0x0004: Open |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Session Info

Indicates that the current [HDR](#) session parameters have changed. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).




| | | | | |
|-----------------------------------|---------------|------------------|-----------------|--|
| Object ID | 0x4021 | Tech HDR Session | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 40 | 0 | CHAR | UATI string as 16 ASCII characters (not null-terminated) |
| | | 16 | UINT16 | Color code |
| | | 18 | UINT32 | RATI |
| | | 22 | UINT16 | Session duration |
| | | 24 | UINT32 | Session start (UINT32– high word of qword, UINT32– low word of qword) |
| | | 32 | UINT32 | Session end (UINT32– high word of qword, UINT32– low word of qword) |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Authentication Status

Indicates authentication result on current HDR network. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).




| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x4022 | Tech HDR Authentication Status | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | HDR Authentication Status <ul style="list-style-type: none"> • 0x0000: Not authenticated • 0x0001: Authenticated • 0x0002: Authentication failed • 0x00FF: Authentication disabled |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR DRC Value

Indicates that the current HDR DRC (data rate control) value has changed. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).




| | | | | |
|-----------------------------------|---|---|--|-------------------------------|
| Object ID | 0x4023 | Tech HDR DRC | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | HDR DRC value |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR DRC Cover

Indicates that the current [HDR DRC](#) (data rate control) cover has changed. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).




| | | | | |
|-----------------------------------|---|---|--|-------------------------------|
| Object ID | 0x4024 | Tech HDR DRC Cover | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | HDR DRC cover |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR RRI

Indicates that the current [HDR RRI](#) (reverse rate indicator) has changed. Notification of this data is enabled via [HDR Tech Notification Status](#) (page 172).

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x4025 | Tech HDR RRI | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | HDR RRI <ul style="list-style-type: none"> • 0x00: Pilot Only • 0x01: 9.6 Kbps • 0x02: 19.2 Kbps • 0x03: 38.4 Kbps • 0x04: 76.8 Kbps • 0x05: 153.6 Kbps |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Single-user Forward statistics

Returns the Single-user [Forward Traffic Channel](#) statistics.

To reduce the traffic, it is recommended that the host enable this notification upon entering engineering menu, and disable it when closing the menu.

| | | | | |
|--------------------|---------------|--------------------------------|-------------|--|
| Object ID | 0x4027 | Single-user Forward statistics | | |
| Operations | ✓ Set | ✗ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 2 | 0 | UINT16 | <p>Object version.</p> <p>Resets the data reported in HDR Single-user Forward statistics (page 179), HDR Single-User Forward Traffic Channel DRC packet termination slot count (page 180), and HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count (page 182).</p> |
| 07 (Notify) | 164 | 0 | UINT16 | Object version |
| | | 2 | UINT32 | <p>Good Traffic Channel CRC count.</p> <p>Number of good packet count for each of the 15 supported Traffic Channel rates.</p> |
| | | 62 | UINT32 | <p>Bad Traffic Channel CRC count.</p> <p>Number of bad packet count for each of the 15 supported Traffic Channel rates.</p> |
| | | 122 | UINT32 | <p>Good Control Channel CRC count.</p> <p>Number of good packet count for each of the 2 supported Control Channel rates.</p> |
| | | 130 | UINT32 | <p>Bad Control Channel CRC count.</p> <p>Number of bad packet count for each of the 2 supported Control Channel rates.</p> |
| | | 138 | UINT16 | Traffic Channel packet error rate (%) |
| | | 140 | UINT32 | <p>TC throughput when served.</p> <p>Traffic Channel throughput in Kbps when served at requested DRC</p> |

| | | | |
|--|-----|--------|--|
| | 144 | UINT32 | TC Instantaneous throughput. Traffic Channel throughput in Kbps |
| | 148 | UINT32 | TC average instantaneous throughput. Average (1sec) Traffic Channel throughput in Kbps |
| | 152 | UINT32 | CC throughput when served. Control Channel throughput in Kbps when served at requested DRC |
| | 156 | UINT32 | CC Instantaneous throughput. Control Channel throughput in Kbps |
| | 160 | UINT32 | CC average instantaneous throughput. Average (1sec) Control Channel throughput in Kbps |

Version information:

| | |
|---------------------------------------|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Single-User Forward Traffic Channel DRC packet termination slot count

Returns the [Forward Traffic Channel packet](#) termination slot count for the first ten supported **DRCs**.

Data in this object is reset by a successful Set operation on [HDR Single-user Forward statistics](#) (page 179).

See also [HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count](#) (page 182).

To reduce the traffic, it is recommended that the host enable this notification upon entering engineering menu, and disable it when closing the menu.

| | | | | |
|--------------------|---------------|---|-----------------|--|
| Object ID | 0x4028 | Single-user Forward Traffic Channel DRC packet termination slot count | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 226 | 0 | UINT16 | Object version |
| | | 2 | UINT32 | TC DRC0 . Early termination slot count. Number of Traffic Channel DRC0 packets that terminated at slot 1-16. |
| | | 66 | UINT32 | TC DRC1 . Early termination slot count. Number of Traffic Channel DRC1 packets that terminated at slot 1-16. |
| | | 130 | UINT32 | TC DRC2 . Early termination slot count. Number of Traffic Channel DRC2 packets that terminated at slot 1-8. |
| | | 162 | UINT32 | TC DRC3 . Early termination slot count. Number of Traffic Channel DRC3 packets that terminated at slot 1-4. |
| | | 178 | UINT32 | TC DRC4 . Early termination slot count. Number of Traffic Channel DRC4 packets that terminated at slot 1-2. |
| | | 186 | UINT32 | TC DRC5 . Early termination slot count. Number of Traffic Channel DRC5 packets that terminated at slot 1-4. |
| | | 202 | UINT32 | TC DRC6 . Early termination slot count. Number of Traffic Channel DRC6 packets that terminated at slot 1. |
| | | 206 | UINT32 | TC DRC7 . Early termination slot count. Number of Traffic Channel DRC7 packets that terminated at slot 1-2. |
| | | 214 | UINT32 | TC DRC8 . Early termination slot count. Number of Traffic Channel DRC8 packets that terminated at slot 1-2. |
| 222 | UINT32 | TC DRC9 . Early termination slot count. Number of Traffic Channel DRC8 packets that terminated at slot 1. | | |

Version information:

| | |
|---------------------------------------|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Single-User Forward Traffic Channel and Control Channel DRC packet termination slot count

Returns the [Forward Traffic Channel](#) and Control Channel [packet](#) termination slot count for [DRCs](#) 10 through 14, and Control Channel packet termination slot count for both supported DRCs.

To reduce the traffic, it is recommended that the host enable this notification upon entering engineering menu, and disable it when closing the menu.

Notification for this object is enabled automatically when notification for [HDR Single-User Forward Traffic Channel DRC packet termination slot count](#) (page 180) is enabled (via bit 11 of [HDR Tech Notification Status](#), page 172). Note that [DRC](#) data is split up into two objects, due to CnS packet length constraints.

Data in this object is reset by a successful Set operation on [HDR Single-user Forward statistics](#) (page 179).

| | | | | |
|--------------------|---|---|--|---|
| Object ID | 0x4029 | Single-user Forward Traffic Channel and Control Channel DRC packet termination slot count | | |
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 126 | 0 | UINT16 | Object version |
| | | 2 | UINT32 | TC DRC10 . Early termination slot count. Number of Traffic Channel DRC10 packets that terminated at slot 1-2. |
| | | 10 | UINT32 | TC DRC11 . Early termination slot count. Number of Traffic Channel DRC11 packets that terminated at slot 1. |
| | | 14 | UINT32 | TC DRC12 . Early termination slot count. Number of Traffic Channel DRC12 packets that terminated at slot 1. |

| | | | |
|--|----|--------|--|
| | 18 | UINT32 | TC DRC13 . Early termination slot count. Number of Traffic Channel DRC13 packets that terminated at slot 1-2. |
| | 26 | UINT32 | TC DRC14 . Early termination slot count. Number of Traffic Channel DRC14 packets that terminated at slot 1. |
| | 30 | UINT32 | CC DRC1 . Early termination slot count. Number of Control Channel DRC1 packets that terminated at slot 1-16. |
| | 94 | UINT32 | CC DRC2 . Early termination slot count. Number of Control Channel DRC2 packets that terminated at slot 1-8. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

HDR Multi-user Forward statistics

Returns the Multi-user **Forward Traffic Channel** statistics.

To reduce the traffic, it is recommended that the host enable this notification upon entering engineering menu, and disable it when closing the menu.

| | | | | |
|--------------------|---|---|--|--|
| Object ID | 0x402A | Multi-user Forward statistics | | |
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 2 | 0 | UINT16 | Object version . Resets the data reported by this object. |
| 07 (Notify) | 172 | 0 | UINT16 | Object version |
| | | 2 | UINT32 | DRC3_128 Good packet count. Number of good DRC3 [128,4,256] short MU packets |

| | | |
|----|--------|--|
| 6 | UINT32 | DRC3_256 Good packet count. Number of good DRC3 [256,4,256] short MU packets |
| 10 | UINT32 | DRC3_512 Good packet count. Number of good DRC3 [512,4,256] short MU packets |
| 14 | UINT32 | DRC3_1024 Good packet count. Number of good DRC3 [1024,4,256] MU packets |
| 18 | UINT32 | DRC5_2048 Good packet count. Number of good DRC5 [2048,4,128] MU packets |
| 22 | UINT32 | DRC8_3072 Good packet count. Number of good DRC8 [3072,2,64] MU packets |
| 26 | UINT32 | DRC10_4096 Good packet count. Number of good DRC10 [4096,2,64] MU packets |
| 30 | UINT32 | DRC13_5120 Good packet count. Number of good DRC13 [5120,2,64] MU packets |
| 34 | UINT32 | DRC3 Bad packet count. Number of bad DRC3 [1024,4,256] MU packets |
| 38 | UINT32 | DRC5 Bad packet count. Number of bad DRC5 [2048,4,128] MU packets |
| 42 | UINT32 | DRC8 Bad packet count. Number of bad DRC8 [3072,2,64] MU packets |
| 46 | UINT32 | DRC10 Bad packet count. Number of bad DRC10 [4096,2,64] MU packets |
| 50 | UINT32 | DRC13 Bad packet count. Number of bad DRC13 [5120,2,64] MU packets |

| | | |
|-----|--------|---|
| 54 | UINT32 | DRC3 Termination Slot Count. Number of DRC3 [128,4,256] packets terminated at slot 1-4 |
| 70 | UINT32 | DRC3 Termination Slot Count. Number of DRC3 [256,4,256] packets terminated at slot 1-4 |
| 86 | UINT32 | DRC3 Termination Slot Count. Number of DRC3 [512,4,256] packets terminated at slot 1-4 |
| 102 | UINT32 | DRC3 Termination Slot Count. Number of DRC3 [1024,4,256] packets terminated at slot 1-4 |
| 118 | UINT32 | DRC5 Termination Slot Count. Number of DRC5 [2048,4,128] packets terminated at slot 1-4 |
| 134 | UINT32 | DRC8 Termination Slot Count. Number of DRC8 [3072,2,64] packets terminated at slot 1-2 |
| 142 | UINT32 | DRC10 Termination Slot Count. Number of DRC10 [4096,2,64] packets terminated at slot 1-2 |
| 150 | UINT32 | DRC13 Termination Slot Count. Number of DRC13 [5120,2,64] packets terminated at slot 1-2 |
| 158 | UINT16 | Traffic Channel packet error rate (%) |
| 160 | UINT32 | TC throughput when served. Traffic Channel throughput in Kbps when served at requested DRC |
| 164 | UINT32 | TC Instantaneous throughput. Traffic Channel throughput in Kbps |
| 168 | UINT32 | TC average instantaneous throughput. Average (1sec) Traffic Channel throughput in Kbps |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |



Current personality data

Returns the current personality index and negotiated [protocol](#) subtypes for this personality.

During negotiation of an HDR session, session configuration parameters are negotiated and stored on the modem as a “personality”. Multiple personalities are negotiated and stored (up to four) on the modem, to be used with [1xEV-DO Rev. 0](#) or [Rev. A](#) compliant networks, but only one personality is in use at any time.

When coverage changes from Rev. 0 to Rev. A, and vice versa, the personality in use is switched, and the modem updates the host with new service state information.

To reduce the traffic, it is recommended that the host enable this notification upon entering engineering menu, and disable it when closing the menu.

| | | | | | |
|-----------------------------------|---------------|---|-----------------|--------------------------------|--|
| Object ID | 0x402B | Personality data | | | |
| Operations | X Set | ✓ Get | ✓ Notify | | |
| Parameters | Len | Offset | Type | Description | |
| 01 (Get) | 2 | 0 | UINT16 | Object version | |
| 02 (Reply) 07 (Notify) | 28 | 0 | UINT16 | Object version | |
| | | 2 | UINT16 | Currently active personality | |
| | | Array elements of Current Personality Protocol Subtypes (UINT16[12]): | | | |
| | | 4 | UINT16 | Physical Layer Protocol | |
| | | 6 | UINT16 | Control Channel MAC Protocol | |
| | | 8 | UINT16 | Access Channel MAC Protocol | |

| | | | | |
|--|--|----|--------|---|
| | | 10 | UINT16 | Forward Traffic Channel MAC Protocol |
| | | 12 | UINT16 | Reverse Traffic Channel MAC Protocol |
| | | 14 | UINT16 | Key Exchange Protocol |
| | | 16 | UINT16 | Authentication Protocol |
| | | 18 | UINT16 | Encryption Protocol |
| | | 20 | UINT16 | Security Protocol |
| | | 22 | UINT16 | Idle State Protocol |
| | | 24 | UINT16 | Generic Multimode Capability Discovery Protocol |
| | | 26 | UINT16 | Generic Virtual Stream Protocol |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

Negotiated protocols and application subtypes

Returns the current personality, number of stored personalities, negotiated [protocol](#) subtypes for all protocols in all negotiated personalities, and negotiated application subtypes for all four streams.

| | | | | |
|-------------------|---------------|---|-----------------|--------------------------------|
| Object ID | 0x402C | Negotiated protocols and application subtypes | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 2 | 0 | UINT16 | Object version |
| 02 (Reply) | 110 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Currently active personality |

| | | |
|--|--------|---|
| 4 | UINT16 | Number of stored personalities |
| Array elements of Personality1 Protocol Subtypes (UINT16[12]): | | |
| 6 | UINT16 | Physical Layer Protocol |
| 8 | UINT16 | Control Channel MAC Protocol |
| 10 | UINT16 | Access Channel MAC Protocol |
| 12 | UINT16 | Forward Traffic Channel MAC Protocol |
| 14 | UINT16 | Reverse Traffic Channel MAC Protocol |
| 16 | UINT16 | Key Exchange Protocol |
| 18 | UINT16 | Authentication Protocol |
| 20 | UINT16 | Encryption Protocol |
| 22 | UINT16 | Security Protocol |
| 24 | UINT16 | Idle State Protocol |
| 26 | UINT16 | Generic Multimode Capability Discovery Protocol |
| 28 | UINT16 | Generic Virtual Stream Protocol |
| 30 | UINT16 | Personality2 Protocol Subtypes (UINT16[12]). Structure same as for Personality1 Protocol Subtypes , on page 188. |
| 54 | UINT16 | Personality3 Protocol Subtypes (UINT16[12]). Structure same as for Personality1 Protocol Subtypes , on page 188. |
| 78 | UINT16 | Personality4 Protocol Subtypes (UINT16[12]). Structure same as for Personality1 Protocol Subtypes , on page 188. |

| | | | | |
|--|--|-----|--------|--|
| | | 102 | UINT16 | Application subtypes per stream (UINT16[4]) <ul style="list-style-type: none"> • 0x0000: Default signaling application • 0x0001: Default packet application for AN • 0x0002: Default packet application for SN • 0x0003: Default test application (IS-890) • 0x000A: Rel. A test application (IS-890A) • 0x0004: Multiflow packet application for AN • 0x0005: Multiflow packet application for SN • 0x0006: 3G 1X circuit services notification application • 0xFFFFE: Any application non IS-856 value • 0xFFFF: Stream not used |
|--|--|-----|--------|--|

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA Tech object set Release | 1 |

Set Dormant

Commands the modem to go into dormancy.

| | | | | |
|-------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x500B | Set Dormant | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | 0 | | None |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Determining which service indicator to display

To determine which service indicator to display in the GUI, several CnS objects must be used, as shown in the following table.

Table 16: Determining which service indicator to display

| HDR Hybrid Preference (0x106A) Hybrid mode: | Service Indication (0x1006) | HDR Service State (0x1069) | Protocol Revision Number (0x101C) | Service indicator to display |
|---|------------------------------------|------------------------------------|--|---|
| On | No service | No service | Don't care | No service |
| On | Digital | Digital (1xEV-DO Rev. 0 or Rev. A) | Not yet received | 1xEV-DO, and wait for Protocol Revision Number (0x101C) for CDMA |
| On | Digital | No service | 1 or 3 | IS95-A only |
| On | Digital | No service | 4 | IS95-B only |
| On | Digital | No service | 6 | 1X only |
| On | Digital | No service | Not yet received | Invalid – Wait for Protocol Revision Number (0x101C) |
| On | Digital | Digital (1xEV-DO Rev. 0 or Rev. A) | 1 or 3 | 1xEV-DO and IS95-A |
| On | Digital | Digital (1xEV-DO Rev. 0 or Rev. A) | 4 | 1xEV-DO and IS95-B |
| On | Digital | Digital (1xEV-DO Rev. 0 or Rev. A) | 6 | 1xEV-DO and 1X |
| On | No service | Digital (1xEV-DO Rev. 0 or Rev. A) | Don't care | 1xEV-DO only |
| Off | No service | No service | Don't care | No service |

| HDR Hybrid Preference (0x106A) Hybrid mode: | Service Indication (0x1006) | HDR Service State (0x1069) | Protocol Revision Number (0x101C) | Service indicator to display |
|---|------------------------------------|------------------------------------|--|---|
| Off | Digital | No service | Not yet received | Invalid – Wait for Protocol Revision Number (0x101C) |
| Off | Digital | No service | 1 or 3 | IS95-A only |
| Off | Digital | No service | 4 | IS95-B only |
| Off | Digital | No service | 6 | 1X only |
| Off | No service | Digital (1xEV-DO Rev. 0 or Rev. A) | Don't care | 1xEV-DO only |

12: Voice Call Management

Introduction

This chapter provides reference for [CnS](#) messages that are related to voice call and connection management. It is further divided into three subsections:

- [Call management summary](#)
- [Headset controls](#) (page 193)
- [Dialing \(DTMF\) controls](#) (page 194)

Note: The MC5728V product supports voice. The AirCard402/ USB 598 products do not support voice.

Call management summary

Table 17: Voice call management

| Object | ID | Description | S | G | N |
|--|--------|---|---|---|---|
| Get/Set E911 Status | 0x1045 | Read (or clear) the E911 status | ✓ | ✓ | ✓ |
| Call Notification Status | 0x3000 | Enable or disable a block of call management notifications | ✗ | ✓ | ✓ |
| Call Duration | 0x3003 | Duration of the current call in seconds | ✗ | ✓ | ✗ |
| Incoming Call | 0x3004 | Notify of an incoming call and its type | ✗ | ✗ | ✓ |
| Answer Incoming Call | 0x3005 | Answer an incoming voice call | ✓ | ✗ | ✗ |
| Caller ID Information Received | 0x3006 | Basic caller ID information for the current call | ✗ | ✓ | ✓ |
| Originate Call | 0x3007 | Connect a voice call, including E911 and *2 (OTASP) calls | ✓ | ✗ | ✗ |
| Extended Caller ID Information | 0x3009 | Report extended caller ID information for the current call | ✗ | ✓ | ✓ |
| Call Connected | 0x300A | Notify of a connected call | ✗ | ✗ | ✓ |
| Connection State | 0x300B | Current status of the connection | ✗ | ✓ | ✗ |

| Object | ID | Description | S | G | N |
|----------------------|--------|---|---|---|---|
| Call Disconnected | 0x300C | Notify of a disconnected call | X | X | ✓ |
| Disconnect All Calls | 0x300D | Release the traffic channel (end a voice call) | ✓ | X | X |
| Connection Error | 0x300E | Notify of a connection error | X | X | ✓ |
| Send Flash | 0x300F | Send a specified flash message | ✓ | X | X |
| Call Connecting | 0x3011 | Notify of an attempt to originate a call | X | X | ✓ |
| Dormant Packet Call | 0x3012 | Notify of entering dormancy during a packet connection. See " Data Services ", starting on page 163. | X | X | ✓ |
| Call Privacy | 0x3013 | Privacy status of the current call | X | ✓ | ✓ |
| OTASP State | 0x3014 | Report OTASP progress | X | X | ✓ |

Headset controls

Table 18: Headset controls

| Object | ID | Description | S | G | N |
|---------------------------------|--------|--|---|---|---|
| Get/Set Speaker Volume | 0x1026 | Control the speaker (headset) volume | ✓ | ✓ | X |
| Mute/Unmute Speaker | 0x1027 | Mute the speaker (headset) | ✓ | ✓ | X |
| Mute/Unmute Mic | 0x1029 | Mute the microphone (headset) | ✓ | ✓ | X |
| Get/Set Tone Level | 0x102A | Control the volume of DTMF tones | ✓ | ✓ | X |
| Mute/Unmute DTMF Tones | 0x102D | Mute the playback of DTMF tones | ✓ | ✓ | X |
| Get/Set Echo Cancellation Level | 0x1035 | Control the audio echo cancellation environment | ✓ | ✓ | X |
| Enable/Disable TTY Option | 0x1046 | Control use of a TTY device | ✓ | ✓ | X |
| Sidetone Gain Level | 0x5009 | Read or set the sidetone gain level | ✓ | ✓ | X |

Dialing (DTMF) controls

Table 19: DTMF controls

| Object | ID | Description | S | G | N |
|------------------------------------|--------|--|---|---|---|
| Play Specified DTMF Tone | 0x102E | Play a DTMF tone both in the headset and over the air | ✓ | ✗ | ✗ |
| Get/Set Default DTMF Tone Duration | 0x102F | Control the duration of DTMF tone playback | ✓ | ✓ | ✗ |
| Send Out Pressed Key | 0x1030 | Play a DTMF tone | ✓ | ✗ | ✗ |
| Set All Keys Up | 0x1031 | Stop playing continuous DTMF tones | ✓ | ✗ | ✗ |
| Generate DTMF Burst | 0x1034 | Send a series of DTMF tones to the network | ✓ | ✗ | ✗ |
| DTMF Playback Mode | 0x1042 | Set the method of tone playback as continuous or burst | ✓ | ✓ | ✗ |

Voice management reference

Get/Set E911 Status

Returns or clears the current [E911](#) mode.

| | | | | |
|-------------------|---------------|---------------|--|---|
| Object ID | 0x1045 | E911 Mode | | |
| Operations | ✓ Set | ✓ Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | E911 state: <ul style="list-style-type: none"> • 0x0000: Set E911 Off • 0x0001: Set E911 On |
| 03 (Set) | 2 | 0 | UINT16 | 0x0000 : Clear E911 mode (off) |
| 04 (Ack) | 2 | 0 | UINT16 | E911 state: <ul style="list-style-type: none"> • 0x0000: Set E911 Off • 0x0001: Set E911 On |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The notification form of this item reports whenever the state changes. E911 mode is entered when an E911 call is made. Leaving E911 mode is determined by requirements of the specific service provider.

Call Notification Status

Enables or disables notification for the following message objects:

- [Call Connecting](#) (page 210)
- [Call Connected](#) (page 201)
- [Incoming Call](#) (page 197)
- [Call Disconnected](#) (page 204)
- [Dormant Packet Call](#) (page 171)
- [Connection Error](#) (page 207)
- [OTASP State](#) (page 212)

| | | | | |
|-------------------|---------------|--------------------------|-----------------|--|
| Object ID | 0x3000 | Call Notification Status | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | One of two states: <ul style="list-style-type: none"> • 0x0000: Notification Off • 0x0001: Notification On |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This object does not issue notifications (0x07) itself. It controls enabling and disabling a group of other object notifications.

Call Duration

Returns the duration of the current (or last) call, in seconds.

| | | | | |
|-------------------|---------------|---------------|-----------------|---------------------------------|
| Object ID | 0x3003 | Call Duration | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT32 | Duration of the call in seconds |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

Note: *This timer is not to be considered a reflection of the service provider's billing time.*

Technically, this timer measures the duration the [traffic channel](#) is assigned.

The timer starts at call initiation, not when the other party answers. It stops when the traffic channel is released. The timer value persists and can be read after the call ends. It resets to zero the next time the traffic channel is assigned.

Since this times the traffic channel, it is possible to use it with a data call; however, the timer stops whenever a [1X/1xEV-DO](#) data call goes [dormant](#) (the traffic channel is released), and restarts from zero when the dormant connection is restored.

See also "[Cumulative Call Duration](#)" on page 93.

Incoming Call

Notification of an incoming call and its type.

| | | | | |
|--------------------|---------------|---------------|---|--|
| Object ID | 0x3004 | Call Incoming | | |
| Operations | X Set | X Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 2 | 0 | UINT16 | Type of incoming call. See Table 20 on page 198. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled using the [Call Notification Status](#) message (page 196).

Call types are noted in the following table (not all types are used for incoming calls).

Table 20: Call types

| Value | Meaning |
|---------------|--------------------------------|
| 0x0000 | Voice |
| 0x0001 | Asynchronous Data |
| 0x0002 | SMS |
| 0x0003 | Position Determination |
| 0x0004 | Test |
| 0x0005 | OTAPA |
| 0x0006 | Standard OTASP |
| 0x0007 | Non-standard OTASP |
| 0x0008 | E911 |
| 0x0100 | Call Waiting |
| 0x0101 | Packet Data |
| 0x0201 | Fax |
| 0xFFFF | No Call Type |

By the time notification is received of the call type, it is too late to change the answer mode used by the modem.

Answer Incoming Call

Answers an incoming **voice** call.

| | | | | |
|-------------------|---------------|-----------------|-------------|--------------------|
| Object ID | 0x3005 | Call Answer Cmd | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

Incoming data or fax calls are answered on the AT command interface.

Caller ID Information Received

Notification of Caller ID information from an incoming call.

| | | | | |
|-----------------------------------|---------------|----------------|--------------------------|---|
| Object ID | 0x3006 | Call Caller ID | | |
| Operations | ✗ Set | ✓ Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 0–32 | 0 | CHAR | A string of up to 32 characters representing the incoming Caller ID information, or null if none. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

Enabling or disabling this notification also enables or disables the notification [Extended Caller ID Information](#) (page 200).

Originate Call

Connect a voice call, including [E911](#) and *2 ([OTASP](#)) calls.

| | | | | |
|-------------------|---------------|--------------------|-------------|--|
| Object ID | 0x3007 | Call Originate Cmd | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 3–34 | 0 | UINT16 | CallType : must be zero (0x0000) |
| | | 2 | CHAR | CallNumber : A non-zero terminated string representing the number to be called (up to 32 bytes) |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

All [CnS](#) originated calls must be type 0x0000 (voice). If an [E911](#) or [OTASP](#) (*2) call is being originated, the modem will report the change in call type when connecting. Other call types are used for incoming calls.

Extended Caller ID Information

Returns the Extended Caller ID information of an incoming call.

| | | | | |
|-----------------------------------|---------------|-------------------------|--|--|
| Object ID | 0x3009 | Call Extended Caller ID | | |
| Operations | ✗ Set | ✓ Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 0–64 | 0 | CHAR | A string of up to 64 characters representing the incoming Caller ID information, or null if none |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled and disabled via [Caller ID Information Received](#) (page 199).

Do **not** use this object to enable or disable the notification.

Call Connected

Notification of a successful call connection. The Call State is a bit-mapped indicator of all active connections.

| | | | | |
|--------------------|---------------|----------------|---|--|
| Object ID | 0x300A | Call Connected | | |
| Operations | X Set | X Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4 | 0 | UINT16 | Call State: See Table 21 on page 202. |
| | | 2 | UINT16 | Service Option: <ul style="list-style-type: none"> • 2: Loopback • 3: Enhanced Variable Rate Voice Service (8 kbps) • 4: Asynchronous Data Service (9.6 kbps) • 5: Group 3 Facsimile (9.6 kbps) • 6: Short Message Services (Rate Set 1) • 7: Packet Data Service: Internet or ISO Protocol Stack • 9: Mobile Station Loopback (13 kbps) • 12: Asynchronous Data Service (14.4 or 9.6 kbps) • 13: Group 3 Facsimile (14.4 or 9.6 kbps) • 14: Short Message Services (Rate Set 2) • 15: Packet Data Service: Internet or ISO Protocol Stack (14.4 kbps) • 17: High Rate Voice Service (13 kbps) • 18: Over-the-Air Parameter Administration (Rate Set 1) • 19: Over-the-Air Parameter |

| | | | |
|--|--|--|--|
| | | | <ul style="list-style-type: none"> Administration (Rate Set 2) • 32: Test Data Service Option (TDSO) • 33: 144 kbps Packet Data Service, Internet or ISO Protocol Stack • 35: IS-801 Position Determination (Rate Set 1) • 36: IS-801 Position Determination (Rate Set 2) • 54: 1xRTT Markov • 55: 1xRTT Loopback • 4100: Asynchronous Data Service, Revision 1 (9.6 or 14.4 kbps) • 4101: Group 3 Facsimile, Revision 1 (9.6 or 14.4 kbps) • 4103: Packet Data Service: Internet or ISO Protocol Stack, Revision 1 (9.6 or 14.4 kbps) • 32768: High Rate Voice Service (13 kbps) • 32798: Markov Calls (RS 1) • 32799: Markov Calls (RS 2) |
|--|--|--|--|

Version information:

| | |
|---------------------------------------|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled using the **Call Notification Status** message (page 196).

Call State is a bit-mapped indicator of all active calls.

Table 21: Call State

| Mask bit | Meaning |
|---------------|-----------------------|
| 0x0001 | CnS Asynchronous Data |
| 0x0002 | Voice Call |
| 0x0004 | CnS Packet |
| 0x0010 | AT Asynchronous Data |
| 0x0020 | AT Voice Call |
| 0x0040 | AT Packet |

| Mask bit | Meaning |
|----------|------------------------|
| 0x0080 | Fax |
| 0x0200 | SMS |
| 0x0400 | OTA |
| 0x0800 | Test |
| 0x1000 | Call Waiting |
| 0x2000 | Position Determination |
| 0x4000 | Dormant Packet |
| 0x8000 | Emergency Call |

Connection State

Returns the current status of the connection.

| Object ID | 0x300B | Call Connection State | | |
|------------|------------------|---|---------------------|--|
| Operations | X Set | <input checked="" type="checkbox"/> Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Call State: See Table 21 on page 202. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

Call Disconnected

Notification of a disconnected call, with a reason for the disconnection.

| | | | | |
|--------------------|---------------|-------------------|---------------------------------|---|
| Object ID | 0x300C | Call Disconnected | | |
| Operations | X Set | X Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4 | 0 | UINT16 | Call State: See Table 21 on page 202. |
| | | 2 | UINT16 | End Reason: See Table 22 on page 204 |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled using the [Call Notification Status](#) message (page 196).

The reasons for disconnection (or dormancy) are described in the following table.

Table 22: End Reason

| Value | Meaning |
|--------|---|
| 0x0000 | Modem offline |
| 0x0001 | Modem CDMA locked until power-cycle |
| 0x0002 | No service |
| 0x0003 | Abnormal call end |
| 0x0004 | Base-station intercept (orig. only) |
| 0x0005 | Base-station release (orig. only) |
| 0x0006 | Base-station release (no reason) |
| 0x0007 | Base-station release (SO reject) |

| Value | Meaning |
|--------|--|
| 0x0008 | Incoming call |
| 0x0009 | Base-station alert stop (incoming only) |
| 0x000A | Client ended call |
| 0x000B | Activation ended (OTASP only) |
| 0x000C | Obsolete |
| 0x000D | NDSS failure |
| 0x000E | Max access probes transmitted |
| 0x000F | Persistence test failure |
| 0x0011 | Access attempt already in progress |
| 0x0012 | Access failure for reason other than above |
| 0x0013 | Received retry order (orig. only) |
| 0x0014 | Failed to originate (modem locked) |
| 0x0015 | Not preferred by PRL |
| 0x0016 | GPS call ended for user call |
| 0x0017 | SMS call ended for user call |
| 0x0026 | Concurrent service not supported by BS |
| 0x0027 | No response from BS |
| 0x0028 | Rejected by BS |
| 0x0029 | Concurrent services not compatible |
| 0x002A | Access blocked by BS |

| Value | Meaning |
|--------|---|
| 0x002B | Already on traffic channel |
| 0x002C | Emergency call flashed |
| 0x002F | Data call ended for user call |
| 0x0096 | Conn deny – General or Busy error (HDR) |
| 0x0097 | Conn deny – Billing or Auth error (HDR) |
| 0x0098 | System Chg due to PRL or Redirect (HDR) |
| 0x0099 | Exit HDR due to PRL or redirect (HDR) |
| 0x009A | No session (HDR) |
| 0x009B | Failure to acquire collocated HDR for origination |
| 0x009C | Data call ended for user call (HDR) |
| 0x009D | Connection setup timeout |

Disconnect All Calls

Release the [traffic channel](#).

| | | | | |
|-------------------|---------------|---------------------|-------------|--------------------|
| Object ID | 0x300D | Call Disconnect Cmd | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

Version information:




| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This forces release of the traffic channel, effectively ending a voice call. However, it does not affect the logical ends of a PPP data connection. This is similar to forcing dormancy on a data connection, although this is not a recommended action for the mobile. The network should control dormancy.

This command should be used to end a voice call only. Data calls should be closed using the AT command interface to ensure a graceful protocol disconnection.

Connection Error

Notification of an error in the connection.

| | | | | |
|--------------------|---|---|--|--|
| Object ID | 0x300E | Call Error | | |
| Operations |  Set |  Get |  Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 6 | 0 | UINT16 | Call Command: which command yielded the error: <ul style="list-style-type: none"> 0x0000: Call origination command 0x0001: Call answer command 0x0002: Call end command 0x0003: Flash command 0x0004: Privacy Pref. Command 0x0005: Info command 0x0006: Hold command |
| | | 2 | UINT16 | Call Error: <ul style="list-style-type: none"> 0x0000: No errors found 0x0001: Invalid client ID - null or not initialized 0x0002: Bad call type 0x0003: Bad service type 0x0004: Bad service option 0x0005: Expecting a number to be provided via the number parameter 0x0006: Number length is > 0, but number buffer points to NULL 0x0007: Invalid digits found in number buffer 0x0008: Out of range value found in number length field 0x0009: Alpha length is >0, but alpha buffer points to NULL |

| | | | |
|--|---|--------|---|
| | | | <ul style="list-style-type: none"> • 0x000A: Out of range value found in alpha length field • 0x000B: Invalid OTASP activation code • 0x000C: Modem offline • 0x000D: CDMA locked • 0x000E: Client tried to send a flash while a call not supporting flashes was in progress • 0x000F: Dialed number is not allowed under the current origination restriction • 0x0010: Client tried to originate a non-E911 call while in E911 mode • 0x0011: Can't perform the command in question while modem is in use • 0x0012: Client originated with service type that is currently not supported by the modem/PRL • 0x0013: Client answered with inappropriate call type • 0x0014: The command in question is invalid in the current call state - for example, answering a call that was already answered, or ending a call that was already ended. • 0x0015: Command conflicts with current modem state • 0x0016: No valid service found for origination with current service type • 0x0017: Client is not allowed to answer the incoming call • 0x0018: Bad privacy setting • 0x0019: No available CM command buffers • 0x001A: Communication problems with MC • 0x001B: Unidentified problem • 0x001C: Invalid last active data network • 0x001D: No collocated HDR • 0x001E: UIM not present |
| | 4 | UINT16 | <p>Command Source: either CnS or AT command types:</p> <ul style="list-style-type: none"> • 0x0000: Non-CnS connection • 0x0001: CnS connection |

Version information:

| | |
|---------------------------------------|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled using the **Call Notification Status** message (page 196).

Send Flash

Send the specified type of flash message to the base station.

| | | | | |
|-------------------|---------------|----------------|-------------|--|
| Object ID | 0x300F | Call Flash Cmd | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 3–34 | 0 | UINT16 | Flash Context: flash types: <ul style="list-style-type: none"> • 0x0000: Manual flash with minimal error checking • 0x0001: Call waiting. Do not include a Call Number. • 0x0002: Three-way calling dial. Sends a phone number. • 0x0003: Three-way calling setup, connection, and disconnection. Do not include a Call Number. |
| | | 2 | CHAR | CallNumber: A non-zero terminated string representing the number to be called (up to 32 bytes) |
| 04 (Ack) | 0–64 | | CHAR | String value indicating any error, or no response if successful: <ul style="list-style-type: none"> • “Too many digits (X) in dial string; max = 64” • “Invalid Flash context value 0xHH” • “Invalid digits at digit X, value 0xHH” • “Invalid call state” • “Unexpected phone number” • “Missing third party phone number” |

Version information:

| | |
|---------------------------------------|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

Flash messages are used to enable or disable enhanced call features, such as three-way calling or answer hold. The flash context is used to specify the intent of a particular flash message that will be sent to the base station. Normally, the **firmware** performs basic error checking to ensure that the flash context is appropriate. When the flash context is zero, the flash message is sent with very limited error checking.

To manage a three-way call:

1. Use **Originate Call** (page 200) to connect to party 1.
2. Flash 0x0002 with number to hold party 1 and dial party 2.
3. Flash 0x0003 to attach party 1 to the call, creating the three-way call.
4. Flash 0x0003 to disconnect party 2, keeping party 1.
5. Use **Disconnect All Calls** (page 206) to disconnect party 1.

Call Connecting

Notification of an attempt to make a call.

| | | | | |
|--------------------|---------------|-----------------|---------------------------------|--|
| Object ID | 0x3011 | Call Connecting | | |
| Operations | X Set | X Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4–36 | 0 | UINT16 | Call Type: See Table 20 on page 198. |
| | | 2 | UINT16 | Command Source: <ul style="list-style-type: none"> • 0x0000: Non-CnS connection • 0x0001: CnS connection |
| | | 4 | CHAR | Call Number , a non-zero terminated string of maximum 32 characters indicating the number being called for the connection. |

Version information:




| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled using the [Call Notification Status](#) message (page 196).

This notification advises the application of calls being initiated by the modem. This is useful to track data or fax calls that are initiated on the AT command interface (source type 0x0000).

Call Privacy

Returns the privacy status of the current call.

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x3013 | Call Privacy | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 2 | 0 | UINT16 | Privacy state: <ul style="list-style-type: none"> • 0x0000: Privacy mode off • 0x0001: Privacy mode on |

Version information:




| | |
|--|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is triggered whenever a call connects to indicate privacy on the call, and also on any change in call privacy.

If it arrives without a call in place, the report is invalid.

OTASP State

Notifies the host of [OTASP](#) (Over The Air Service Provisioning) progress.

| | | | | |
|--------------------|---|---|--|---|
| Object ID | 0x3014 | Call OTASP State | | |
| Operations |  Set |  Get |  Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4 | 0 | UINT16 | OTASP State: See Table 23 on page 212. |
| | | 2 | UINT16 | OTASP Success: <ul style="list-style-type: none"> 0x0000: Failed 0x0001: Successful |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CALL object set Release | 1 |

This notification is enabled using the [Call Notification Status](#) message (page 196).

Table 23: OTASP State

| Value | Description |
|---------------|--|
| 0x0001 | SPL unlock - required to access certain parameters |
| 0x0002 | AKey exchange - Authentication key has been exchanged |
| 0x0003 | SSD update - Shared Secret Data has been updated |
| 0x0004 | NAM download - New NAM parameters have been downloaded |
| 0x0005 | MDM download - New MDM has been downloaded |
| 0x0006 | IMSI download - IMSI has been downloaded |
| 0x0007 | PRL download - PRL has been downloaded |
| 0x0008 | Commit - Newly downloaded parameters have been transferred to NV-RAM |

Headset control reference

Get/Set Speaker Volume

Reports or sets the current speaker volume, which is stored in non-volatile memory.

| | | | | |
|-------------------|---------------|----------------|-------------|---|
| Object ID | 0x1026 | Speaker Volume | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Volume setting: <ul style="list-style-type: none"> • 0x0000: No volume (speaker volume off) • 0x0001 to 0x0006: Minimum to maximum speaker volume |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Mute/Unmute Speaker

Gets or sets the mute status of the speaker.

| | | | | |
|-------------------|---------------|---------------|-------------|--|
| Object ID | 0x1027 | Speaker Mute | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Mute setting: <ul style="list-style-type: none"> • 0x0000: Unmute • 0x0001: Mute speaker |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Mute/Unmute Mic

Gets or sets the mute status of the microphone input codec, which is independent of the mic level (0x1028 - [Get/Set Mic Codec Gain \(Mic Level\)](#), page 104).

| | | | | |
|-------------------|---------------|---------------|-------------|--|
| Object ID | 0x1029 | Mic Mute | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Mute setting: <ul style="list-style-type: none"> • 0x0000: Unmute microphone • 0x0001: Mute microphone |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Get/Set Tone Level

Returns or sets the outgoing beep/tone level in the headset.

| Object ID | 0x102A | Tone Level | | |
|------------|--------|------------|----------|---|
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Current outgoing beep/tone level in the headset: <ul style="list-style-type: none"> • 0x0000: off (muted) • 0x0001: lowest, to • 0x0006: highest |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Mute/Unmute DTMF Tones

Sets the mute status of the outgoing [DTMF](#) tones from the modem.

| | | | | |
|-------------------|---------------|---------------|-------------|--|
| Object ID | 0x102D | DTMF Mute | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 03 (Set) | 2 | 0 | UINT16 | Mute setting: <ul style="list-style-type: none"> • 0x0000: Unmute tones • 0x0001: Mute tones |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

This mutes tones both in the headset and over the air.

Get/Set Echo Cancellation Level

Return or set the echo cancellation profile for the audio path during a voice call. The value is stored in non-volatile memory.

| | | | | |
|-------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x1035 | Echo Cancel | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |

| | | | | |
|-------------------|---|---|--------|---|
| 02 (Reply) | 2 | 0 | UINT16 | Environment profile: <ul style="list-style-type: none"> • 0x0000: Factory default profile (same as 0x0003 headset) • 0x0001: No echo cancellation • 0x0002: Handset • 0x0003: Headset • 0x0004: Acoustic (AEC) • 0x0005: Speakerphone |
| 03 (Set) | 2 | | | Same parameters as 02 (Reply) |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Enable/Disable TTY Option

Enables or disables the modem's TTY option.

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x1046 | TTY Option | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | TTY Mode: <ul style="list-style-type: none"> • 0x0000: TTY Disabled • 0x0001: TTY Enabled • 0x0002: TTY Enabled (Talk only) • 0x0003: TTY Enabled (Hear only) |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

When set to "**0x0001**: TTY Enabled", both the headset speaker and microphone are replaced by the TTY device.

Sidetone Gain Level

Set or read the sidetone gain level in non-volatile memory.

| | | | | |
|-------------------|---------------|---------------------|-------------|---|
| Object ID | 0x5009 | Sidetone Gain Level | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Sidetone gain level: <ul style="list-style-type: none"> • 0x0000: Sidetone is disabled (muted) • 0x0001: Lowest, to • 0x0006: Highest |
| 03 (Set) | 2 | 0 | UINT16 | |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The factory default setting is 0x0002.

Sidetone is the return of the microphone input to the speaker output as feedback to the user.

Dial control reference

Play Specified DTMF Tone

Generates the specified [DTMF](#) tone, whose duration is specified by the [Get/Set Default DTMF Tone Duration](#) (0x102F; page 220).

| Object ID | 0x102E | DTMF Key | | |
|------------|--------|----------|----------|---|
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 2 | 0 | UINT8 | 0x0000: Null filler |
| | | 1 | UINT8 | DTMF keypress: <ul style="list-style-type: none"> • 0x23: '#' key • 0x2A: '*' key • 0x30: '0' key • 0x31: '1' key • 0x32: '2' key • 0x33: '3' key • 0x34: '4' key • 0x35: '5' key • 0x36: '6' key • 0x37: '7' key • 0x38: '8' key • 0x39: '9' key |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

This plays the tone in the headset using timing set by [Get/Set Default DTMF Tone Duration](#) (0x102F; page 220), and over the air using [CDMA](#) commands. The duration of tones over the air is governed by network settings, not the modem.

Get/Set Default DTMF Tone Duration

Returns or sets the default duration for all **DTMF** (Touch-Tone™) tones played in the headset based on the DTMF mode setting of **DTMF Playback Mode** (page 224). These values are stored in non-volatile memory, and become the default.

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x102F | DTMF Duration | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 6 | 0 | UINT16 | DTMF keypress duration 100–3000 milliseconds inclusive, used for single-key DTMF tones. |
| | | 2 | UINT16 | DTMF burst-digit duration 95–350 milliseconds inclusive, used for tones sent in a DTMF burst. |
| | | 4 | UINT16 | DTMF burst-inter-digit duration 60–200 milliseconds inclusive, used for the tone-off spaces between tones sent in a DTMF burst. |
| 03 (Set) | 6 | | | Same parameters as 02 (Reply) |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Tones sent over the air are generated by the network based on in-band commands from the modem. The modem requests the timing set in this command, but there is no guarantee that the network will deliver that timing.

Send Out Pressed Key

Plays the specified [DTMF](#) tone.

| | | | | |
|-------------------|---------------|---------------|-------------|--|
| Object ID | 0x1030 | Key Pressed | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 2 | 0 | UINT16 | Keypress: <ul style="list-style-type: none"> • 0x23: '#' key • 0x2A: '*' key • 0x30: '0' key • 0x31: '1' key • 0x32: '2' key • 0x33: '3' key • 0x34: '4' key • 0x35: '5' key • 0x36: '6' key • 0x37: '7' key • 0x38: '8' key • 0x39: '9' key |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

This plays the tone in the headset. The duration of the tone is based on these setting of [DTMF Playback Mode](#) (page 224):

- If continuous mode, the tone continues until released by a call to [Set All Keys Up](#) (0x1031; page 222).
- If burst mode, the keypress duration set in [Get/Set Default DTMF Tone Duration](#) (0x102F; page 220).

This provided as a “comfort” tone for user feedback, not to send tones over the air to the network or [PSTN](#).

Set All Keys Up

Sends out a signal indicating that all **DTMF** or dialpad-equivalent keys have been released.

| | | | | |
|-------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x1031 | Key Released | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Generate DTMF Burst

Generates a burst of **DTMF** tones, specified in an ASCII string, maximum 32 characters.

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x1034 | DTMF Burst | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 1–32 | 0 | CHAR | String of maximum 32 ASCII characters of the following allowed values: <ul style="list-style-type: none"> • 0x23: '#' key • 0x2A: '*' key • 0x30: '0' key • 0x31: '1' key • 0x32: '2' key • 0x33: '3' key • 0x34: '4' key • 0x35: '5' key • 0x36: '6' key • 0x37: '7' key • 0x38: '8' key • 0x39: '9' key |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The tones to the network are issued using **CDMA** commands that include timing set using **Get/Set Default DTMF Tone Duration** (page 220); however the actual timing used by the network may vary. The user feedback in the headset is a single chirp of about 200 ms.

The string is not null-terminated. The length of the string is determined by the **CnS** length setting.

DTMF Playback Mode

Sets and reports whether **DTMF** tones are to played as continuous or burst.

| | | | | |
|-------------------|---------------|----------------------|-------------|--|
| Object ID | 0x1042 | Continuous DTMF Mode | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT8 | Null (0x00) |
| | | 1 | UINT8 | DTMF playback mode <ul style="list-style-type: none"> • 0x00: Continuous mode • 0x01: Burst mode |
| 03 (Set) | 2 | | | Same parameters as 02 (Reply) |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

This setting affects the behavior of **Send Out Pressed Key** (0x1030) on page 221.

Burst mode uses timings set in **Get/Set Default DTMF Tone Duration** (0x102F) on page 220.

Timing of tones over the air is governed by the network.

13: SMS Messages

Introduction

This chapter provides reference for [CnS](#) messages that are related to SMS messages. Before using these commands you should have a clear understanding of the flow of SMS messages through the modem. For a full discussion of SMS message management, consult “[Appendix A: SMS Overview](#)” on page 293.

This chapter includes descriptions of the data structures used within the CnS messages and then the CnS message reference itself.

Terminology

To try to maintain clarity, the following terminology must be precisely understood:

SMS message The entire CDMA SMS message composed of a [header](#) and body.

SMS header The [header](#) portion of a CDMA SMS message. For more information, see [SMS header](#) on page 228.

SMS body The text body portion of a CDMA SMS message

SMS packet The Parameter portion of a CnS message that carries some or all of an SMS message.

SMS packet header The first six bytes of an SMS [packet](#). It contains information about the packet type, segment management, and length of the SMS packet payload. For more information, see page [227](#).

SMS packet payload The portion of an SMS packet following the [SMS packet header](#). This can be either the [SMS header](#), the SMS body or a segment of it, or—for very short SMS messages—both the header and body.

Summary

Objects in this set are numbered in the range 0x2000–0x2FFF.

Table 24: SMS message handling objects

| Object | ID | Description | S | G | N |
|--|--------|--|---|---|---|
| SMS Receive Status (r1) | 0x2000 | Number of SMS messages received | ✗ | ✓ | ✓ |
| SMS Read (r1) | 0x2001 | Retrieve an SMS message | ✗ | ✓ | ✗ |
| SMS Delete (r1) | 0x2002 | Delete a received SMS message | ✓ | ✗ | ✗ |
| SMS Write | 0x2003 | Store an SMS message to the outbox | ✓ | ✗ | ✗ |
| SMS Send | 0x2004 | Send the contents of the SMS outbox | ✓ | ✗ | ✗ |
| SMS Send Status | 0x2005 | Notify of SMS Send Status | ✗ | ✗ | ✓ |
| SMS Voice Message Delete | 0x2006 | Delete the SMS message for voice message count | ✓ | ✗ | ✗ |
| SMS Full | 0x2007 | Notify of SMS storage full | ✗ | ✗ | ✓ |
| SMS Receive Status (r2) and Storage Capacity | 0x2008 | Number of SMS messages received, and maximum number of messages that the modem can store | ✗ | ✓ | ✓ |
| SMS Read (r2) | 0x2009 | Retrieve an SMS message | ✗ | ✓ | ✗ |
| SMS Delete (r2) | 0x200A | Delete a received SMS message | ✓ | ✗ | ✗ |

Data structures

SMS packet header

When reading or writing an SMS message, each CnS message's parameter begins with an SMS [packet header](#). This header conveys information about the SMS message being transferred, the type of SMS packet payload in this CnS message, and the presence of more SMS messages in the modem.

A CnS message used to send or receive an SMS message looks generally like this:

| Byte offset | Content |
|-------------|---|
| 0–1 | Object ID |
| 2 | Operation Type |
| 3 | Reserved |
| 4–7 | Application ID |
| 8 | Reserved |
| 9 | Length of parameter (0–246) |
| 10–15 | SMS packet header composed of: |
| 10–11 | Message ID (UINT) |
| 12 | Sequence Number of the segmented message (UINT) |
| 13 | Packet Type (UINT). One of: <ul style="list-style-type: none"> • 0 = Body segment with more to follow • 1 = First segment (message header) • 2 = Last body segment • 3 = Only body segment (not segmented) |
| 14 | Count of remaining SMS messages in the modem (excluding this one) (UINT) |
| 15 | Length of the payload in this segment. (UINT) |
| 16–Len | Payload based on Packet type |

For a discussion of the fields in the SMS packet header, see [“SMS packet header”](#) on page 299.

SMS header

The first (and maybe only) CnS message carrying an SMS message will have the SMS message [header](#) in the payload.

The SMS header structure is described in [Table 25](#). The offsets shown in that table start at zero, and begin in the CnS message at offset 16, at the start of the SMS [packet](#) payload portion.

Table 25: SMS header structure

| Byte offset | Type | Content | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|-----------------------|--|-----|-----------------|---|-----------------------|---|-----------|---|-----------|---|-----------|---|-----------|---|-----------|---|---------|---|-------------|---|-----------|---|----------|----|---------|----|--------------|----|--------------------|----|-------------------|----|----------|----|----------|----|--------------|
| 0–3 | UINT32 | <p>Parameter bit-mask to indicate the available parameters in the header. The bit for each parameter group will be set when its corresponding SMS parameters are available:</p> <table border="0"> <thead> <tr> <th>Bit</th> <th>Parameter group</th> </tr> </thead> <tbody> <tr><td>0</td><td>Bearer reply (TL ack)</td></tr> <tr><td>1</td><td>Valid ABS</td></tr> <tr><td>2</td><td>Valid REL</td></tr> <tr><td>3</td><td>Defer ABS</td></tr> <tr><td>4</td><td>Defer REL</td></tr> <tr><td>5</td><td>User resp</td></tr> <tr><td>6</td><td>Address</td></tr> <tr><td>7</td><td>Sub-address</td></tr> <tr><td>8</td><td>User data</td></tr> <tr><td>9</td><td>Priority</td></tr> <tr><td>10</td><td>Privacy</td></tr> <tr><td>11</td><td>Reply option</td></tr> <tr><td>12</td><td>Number of messages</td></tr> <tr><td>13</td><td>Alert on delivery</td></tr> <tr><td>14</td><td>Language</td></tr> <tr><td>15</td><td>Callback</td></tr> <tr><td>16</td><td>MC timestamp</td></tr> </tbody> </table> <p>(If no bits are set, no parameters are available.)</p> | Bit | Parameter group | 0 | Bearer reply (TL ack) | 1 | Valid ABS | 2 | Valid REL | 3 | Defer ABS | 4 | Defer REL | 5 | User resp | 6 | Address | 7 | Sub-address | 8 | User data | 9 | Priority | 10 | Privacy | 11 | Reply option | 12 | Number of messages | 13 | Alert on delivery | 14 | Language | 15 | Callback | 16 | MC timestamp |
| Bit | Parameter group | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | Bearer reply (TL ack) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | Valid ABS | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | Valid REL | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | Defer ABS | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | Defer REL | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | User resp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | Address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | Sub-address | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | User data | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 | Priority | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 10 | Privacy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 11 | Reply option | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 12 | Number of messages | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 13 | Alert on delivery | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 14 | Language | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15 | Callback | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 16 | MC timestamp | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | UINT8 | <p>Message Category:</p> <ul style="list-style-type: none"> • 0 = IS-637 SMS Point-to-Point Msg • 1 = IS-637 SMS Broadcast Msg • 2 = IS-637 SMS Acknowledge Msg | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| Byte offset | Type | Content |
|-------------|--------|--|
| 5 | UINT8 | <p>Message Teleservice (different for read and write)</p> <p>Received SMS (read):</p> <ul style="list-style-type: none"> • 0 = IS-95A Cellular Paging • 1 = IS-95A Cellular Messaging • 2 = IS-95A Voice Mail Notification • 3 = IS-91 Extended Protocol - CLI Order • 4 = IS-91 Extended Protocol - Voice Mail and Embedded Message Waiting Indicator • 5 = IS-91 Extended Protocol - Short Message • 6 = IS-95A Alert With Info <p>Outgoing SMS (write)</p> <ul style="list-style-type: none"> • 0 = CPT_95 - Page • 1 = CMT_95 - Short Message • 2 = VMN_95 - Voice Mail Notification • 5 = CMT_91 - Embedded IS-91 SMS |
| 6-7 | UINT16 | <p>Coding of Service Category (News, Restaurant, and so on). For more details, see <i>IS-627</i>, section 3.4.3.2.</p> |
| 8 | UINT8 | <p>TL Acknowledge Request Indicates if the client needs to confirm whether the message is received successfully or not.</p> |
| 9 | UINT8 | <p>Message ID Type</p> <ul style="list-style-type: none"> • 0 = NOT USED • 1 = Deliver (Regular MT SMS) • 2 = Submit (Regular MO SMS) • 3 = Cancellation (MO Only. Cancel previous SMS delivery) • 4 = Delivery Ack. (MT Only) • 5 = User Acknowledgement (Either Direction) |
| 10 | UINT8 | Address Digit Mode |
| 11 | UINT8 | Address number mode |
| 12 | UINT8 | Address number type |
| 13 | UINT8 | Address number plan |
| 14 | UINT8 | Length of Address |

| Byte offset | Type | Content |
|-------------|------------|---|
| 15–64 | String[50] | Destination address (for outgoing SMS) (write) or Origination address (for received SMS) (read) |
| 65 | UINT8 | Sub address type |
| 66 | UINT8 | Sub Address Odd byte |
| 67 | UINT8 | Length of the sub-address |
| 68-117 | String[50] | Destination sub-address (for outgoing SMS) (write) or Origination sub-address (for received SMS) (read) |
| 118 | UINT8 | Length of the Callback number |
| 119–150 | String[32] | Call Back number string (32-character maximum). If this is {blank} on outgoing SMS (write), the modem fills it with the phone number of the current account. |
| 151 | UINT8 | Message Priority <ul style="list-style-type: none"> • 0 = Normal • 1 = Interactive • 2 = Urgent • 3 = Emergency |
| 152 | UINT8 | Message Privacy Level <ul style="list-style-type: none"> • 0 = Not restricted (Level 0) • 1 = Restricted (Level 1) • 2 = Confidential (Level 2) • 3 = Secret (Level 3) |
| 153 | UINT8 | Message Language <ul style="list-style-type: none"> • 0 = Unspecified • 1 = English • 2 = French • 3 = Spanish • 4 = Japanese • 5 = Korean • 6 = Chinese • 7 = Hebrew <p>For outgoing SMS (write), a blank entry is defaulted to English (1) by the modem.</p> |




| Byte offset | Type | Content |
|-------------|-----------|---|
| 154 | UINT8 | User Acknowledge Required Specify whether user acknowledgement is required. True/False |
| 155 | UINT8 | DAK Required Specify whether delivery acknowledge is required. |
| 156 | UINT8 | User Response Code |
| 157–158 | String[2] | Number of Voice message |
| 159 | UINT8 | Alert on Delivery |
| 160 | UINT8 | Data Encoding |
| 161 | UINT8 | Time Stamp - Year (presented in BCD format, that is, 2002 is presented as 0x02) (Note: values from 96–99 are interpreted as 1996–1999) |
| 162 | UINT8 | Time Stamp - Month - presented in BCD format. Thus, December will be given as 0x12 |
| 163 | UINT8 | Time Stamp - Day - presented in BCD format. Thus, 30th will be given as 0x30 |
| 164 | UINT8 | Time Stamp - Hour - presented in BCD format. Thus, 9pm is presented as 0x0x21 (the time stamp uses a 24-hour clock). |
| 165 | UINT8 | Time Stamp - Minute - presented in BCD format. Thus, thirty minutes is presented as 0x30 |
| 166 | UINT8 | Time Stamp - Second - presented in BCD format. Thus, twenty-eight seconds is presented as 0x28 |
| 167 | UINT8 | Absolute Validity - Year |
| 168 | UINT8 | Absolute Validity - Month |
| 169 | UINT8 | Absolute Validity - Day |
| 170 | UINT8 | Absolute Validity - Hour |
| 171 | UINT8 | Absolute Validity - Minute |
| 172 | UINT8 | Absolute Validity - Second |

| Byte offset | Type | Content |
|-------------|-------|--|
| 173 | UINT8 | Relative Validity Time of an SMS transmission. For more details, see <i>TIA/EIA 637-A</i> , section 4.5.6. |
| 174 | UINT8 | Absolute Delivery Deferral Time - Year |
| 175 | UINT8 | Absolute Delivery Deferral Time - Month |
| 176 | UINT8 | Absolute Delivery Deferral Time - Day |
| 177 | UINT8 | Absolute Delivery Deferral Time - Hour |
| 178 | UINT8 | Absolute Delivery Deferral Time - Minute |
| 179 | UINT8 | Absolute Delivery Deferral Time - Second |
| 180 | UINT8 | Relative Delivery Deferral Time of an SMS transmission. For more details, see <i>IS-627</i> , section 4.5.8. |
| 181 | UINT8 | Message Display Mode: <ul style="list-style-type: none"> • 0 = Immediate Display • 1 = Radio default setting • 2 = User Invoke |

Reference

SMS Receive Status (r1)

Report the number of SMS messages stored in the modem.

| Object ID | 0x2000 | SMS Received Msg Status (r1) | | |
|------------|---|---|--|-----------------------------|
| Operations |  Set |  Get |  Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 6 | 0 | UINT16 | Urgent message count |

| | | | |
|--------------------|---|--------|-----------------------------|
| 07 (Notify) | 2 | UINT16 | Normal message count |
| | 4 | UINT16 | Voice message count |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

This message can be used to query the modem (Get Request) for the current SMS message counts, and to receive notifications whenever a new SMS message arrives.

The parameter field contains three counters—one for each of the message queues in the modem.

SMS Read (r1)

Retrieve the next pending SMS message. The process involves multiple steps using the same [CnS](#) object ID but with different parameter contents. For a full discussion of SMS message management, see "[Appendix A: SMS Overview](#)" on page 293.

| | | | | |
|----------------------------------|------------------|-----------------------|---------------------|-----------------------------------|
| Object ID | 0x2001 | SMS Retrieve SMS (r1) | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) 1st | 0 | | | None |
| 02 (Reply) 1st | 188 | 0 | Struct | SMS packet header |
| | | 6 | Struct | SMS header |
| 01 (Get) | 2 | 0 | UINT16 | Acknowledge index (see text) |
| 02 (Reply) | 7–246 | 0 | Struct | SMS packet header |
| | | 6 | BYTE | SMS body segment (1–240 bytes) |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

The first instance of the Get Request to read an SMS message does not use a parameter. Subsequent calls to retrieve the segments of the message pass a sequence number as an acknowledge index. To fetch the first body segment (following receipt of the [SMS header](#)), the acknowledge index is 0. Subsequent calls to retrieve body segments increment the index.

The parameter is only used to make sure that no parts of the message get lost; it cannot be used to request a retransmission of a certain segment. If an error is detected during message retrieval, the host application has to go through the whole message list (or what's left after deleting several messages) and then retrieve the initially corrupted message again.

SMS messages can be segmented as discussed in “[Appendix A: SMS Overview](#)” on page 293. To read an SMS message can require a sequence of [SMS Read \(r1\)](#) exchanges. See “[Reading incoming SMS messages](#)” on page 300.

After verifying, via [SMS Receive Status \(r1\)](#) (page 232), that there is a message to read, you make the first Get request to this service with the parameter (acknowledge index) set to 0. The modem responds with either a type 1 or type 3 [packet](#). If you receive a type 3 response, the entire SMS message is in that response. If you get a type 1 packet instead, it contains only the SMS message header; the body will follow in subsequent exchanges. For a description of the SMS message header, see page 228.

Where the SMS message has been segmented—a type 1 response to the first Get request—you must continue to issue Get requests, incrementing the acknowledge index each time. The modem responds to each request with the next segment of the SMS message body. This exchange continues until the host receives a response with packet type 2, indicating the last segment.

The host must re-assemble the SMS message from the segments. When all segments are received, the host must use [SMS Delete \(r1\)](#) (page 235) to remove the message from the modem's queue. If this is not done, the next call to read an SMS message retrieves the same message over again.

Messages can only be retrieved sequentially. Retrieval order is **urgent** messages first, then normal messages, then **voice** mail notification. Within each group, messages are sent to the host in “last-in first-out” (LIFO) order; the most recently received message is sent first, then progressively older messages. Use the time stamp in the [SMS header](#) to determine real-time message sequence.

When retrieving voice mail notification messages, the SMS message body will contain something like “Press SEND to check mailbox”. Your application must check the Message Teleservice field in the [SMS header](#) to determine that this is a voice mail indication.

SMS Delete (r1)

Delete a retrieved (via [SMS Read \(r1\)](#)) SMS message.

| | | | | |
|-------------------|------------|-----------------|-------------|----------------------|
| Object ID | 0x2002 | SMS Delete (r1) | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 2 | 0 | UINT16 | Message ID to delete |
| 04 (Ack) | 2 | 0 | UINT16 | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

This command does not delete the “voice” SMS message, used to convey the number of voice messages. If it becomes necessary to delete that message, use [SMS Voice Message Delete](#) (0x2006; page [240](#)).

SMS Write

Writes an SMS message to the modem's outbox.

| | | | | |
|--------------------------------|---------------|---------------|-------------|-----------------------------------|
| Object ID | 0x2003 | SMS Store SMS | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) 1st | 188 | 0 | Struct | SMS packet header |
| | | 6 | Struct | SMS header |
| 04 (Ack) 1st | 0 | | | None |
| 03 (Set) | 7–246 | 0 | Struct | SMS packet header |
| | | 6 | BYTE | SMS body segment (1–240 bytes) |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

Note: *Once written to the outbox, there is no way to recall or edit the SMS message.*

The modem's outbox supports up to ten SMS messages. Attempts to write an eleventh message result in the error return code. Use [SMS Send](#) (page 237) to transmit the messages in the outbox to the network. Once sent, the messages are automatically deleted from the outbox, freeing additional space for additional messages.

A counter controlling access to the outbox is incremented at the end of each SMS write operation. If the counter has reached ten, store operations are not permitted. The same counter is decremented at the end of the send and delete operation for each message.

SMS Send

Instructs the modem to send the SMS messages in the outbox.

| | | | | |
|-------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x2004 | SMS Send SMS | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

When the host has written all the outgoing SMS messages (via [SMS Write](#); page [236](#)) to the modem's outbox, you now send this request to tell the modem to transmit the SMS messages to the network.

As each message in the outbox is sent, it is deleted from the outbox and the modem can send a notification via [SMS Send Status](#) (page [238](#)).

A counter used to control access to the outbox is decremented at the end of the send and delete operation for each message. When the counter reaches zero, the send process initiated by this CnS item stops. As long as the counter is non-zero, the modem continues to send and delete messages.

SMS Send Status

A notification message to indicate the success or failure to send an SMS message to the [CDMA](#) network.

| | | | | |
|--------------------|---------------|-----------------|---------------------------------|---|
| Object ID | 0x2005 | SMS Send Status | | |
| Operations | X Set | X Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 6 | 0 | UINT16 | Msg ID: assigned by the host |
| | | 2 | UINT16 | Error Class: <ul style="list-style-type: none"> • 0 = No Error • 1 = Reserved (Ignore) • 2 = Temporary • 3 = Permanent |
| | | 4 | UINT16 | Cause Code: IS-41 Standard SMS Cause Code (From the CDMA Base Station) or Internal code as described in the text that follows. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

The host receives a notification of the send status of each outgoing SMS message in the following conditions:

- Message was sent successfully
- IS-41 Status Code (Cause Code < 0xFF) is received
- An Internal SMS Status code, which is either a temporary (Error Class = 2) or a permanent (Error Class = 3) error.

A successfully sent message reports an Error Class of **0** and a Cause Code of **0x8000**. Only after successful transmission should the host delete the original SMS message from its own storage.

If the CDMA base station reports an error, it is reported in the **Cause Code**. For details of base station error values, consult IS-41.

If the modem experienced an error, it returns a **Cause Code** from the following:

Table 26: SMS send status Cause Codes

| Code | Meaning |
|---------------|---|
| 0x4000 | SMS connection was rejected by the network |
| 0x4001 | SMS connection failed |
| 0x8000 | No error |
| 0x8001 | Intermediate status for a message requesting a TL acknowledge |
| 0x8002 | Out of memory buffer |
| 0x8003 | Message is too large to be sent over access |
| 0x8004 | Message is too large to be sent over DTC |
| 0x8005 | Lower layer is not in a ready state to send the message |
| 0x8006 | Modem is not allowed to send the message |
| 0x8007 | Cannot send message in analog mode |
| 0x8008 | Cannot send broadcast |
| 0x8009 | Invalid transaction ID |
| 0x800A | Message not sent |
| 0x800B | Message being sent |

The notification form of this item will trigger on any internal fault related to SMS transmission, and on any network status reports of the SMS message.

If any form of error is reported, the message must be stored and sent again by the host. The modem does not perform automatic retries.

SMS Voice Message Delete

Delete the SMS message used to indicate the voice message count.

| | | | | |
|-------------------|---------------|--------------------------------|-------------|--------------------|
| Object ID | 0x2006 | SMS Voice Message Count Delete | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 04 (Ack) | 0 | | | None |

Version information:




| | |
|---|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

The modem normally maintains a single SMS message containing the count of voice messages being held at the network. If this gets out of sync with the actual number of messages, the modem's SMS voice message can be deleted with this message. Upon next receipt of a voice mail notification from the network, the modem will again match the network's count.

SMS Full

A notification message to indicate that the SMS storage is full.

If the host is not available when mobile-terminated SMS messages are received, the messages are stored in the modem's NV. This message is triggered when the maximum number of messages is stored in NV. (The modem can store up to 98 text messages (urgent or normal) and one voice message.)




| | | | | |
|--------------------|---|---|--|--------------------|
| Object ID | 0x2007 | SMS Full | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 1 |

SMS Receive Status (r2) and Storage Capacity

Report the number of SMS messages stored in the modem and the maximum number of messages that the modem can store.

| | | | | |
|-----------------------------------|---|---|--|--------------------------------|
| Object ID | 0x2008 | SMS Received Msg Status (r2) and Storage Capacity | | |
| Operations |  Set |  Get |  Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 10 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Urgent message count |
| | | 4 | UINT16 | Normal message count |
| | | 6 | UINT16 | Voice message count |

| | | | | |
|--|--|---|--------|---|
| | | 8 | UINT16 | Maximum number of messages that the NV can store. |
|--|--|---|--------|---|

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 2 |




This message can be used to:

- Query the modem (Get Request) for the current SMS message counts and the maximum number of messages that the modem can store
- Receive notifications whenever:
 - A new SMS message arrives
 - The SMS message lists are rearranged (when a message is moved to the read list [as a result of [SMS Read \(r2\)](#)] or deleted [as a result of [SMS Delete \(r2\)](#)]).

The three counters in the parameter field are for the three message queues in the modem.

SMS Read (r2)

Retrieve the next pending SMS message. The process involves multiple steps using the same CnS object ID but with different parameter contents. For a full discussion of SMS message management, see “[Appendix A: SMS Overview](#)” on page 293.

| Object ID | 0x2009 | SMS Retrieve SMS (r2) | | |
|--------------------------|---|---|--|--|
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) 1 st | 6 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Message category: <ul style="list-style-type: none"> • 1 = New urgent messages • 2 = New normal messages • 3 = Old (read) messages • 4 = Voicemail |
| | | 4 | UINT16 | Message index: index in the “Message category” list (see above). Starts at 1. |

| | | | | |
|----------------------------------|--------|----|--------|--|
| 02 (Reply) 1st | 192 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Result: <ul style="list-style-type: none"> • 0x00: Success; the remaining fields of payload have valid data. • 0x01: Fail: General error • 0x02: Fail: Specified message list is empty • 0x03: Fail: Invalid message list (out of range) • 0x04: Fail: Message index out of range |
| | | 4 | Struct | SMS packet header |
| | | 10 | Struct | SMS header |
| 01 (Get) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Acknowledge index (see text) |
| 02 (Reply) | 11–250 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Result: <ul style="list-style-type: none"> • 0x00: Success; the remaining fields of payload have valid data. • 0x01: General error (most likely due to error in the first Get Request packet, so subsequent request can't be processed) • 0x05: Invalid segment index (out of order, or out of bounds) • 0x06: Fail: Voicemail can't be read |
| | | 4 | Struct | SMS packet header |
| | | 10 | BYTE | SMS body segment (1–235 bytes) |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 2 |

The first instance of the Get Request to read an SMS message uses two parameters: the message category (for example, urgent messages), and a parameter of the message index; the index starts at 1. Subsequent calls to

retrieve the segments of the message pass a sequence number as an acknowledge index. To fetch the first body segment (following receipt of the [SMS header](#)), the acknowledge index is 0. Subsequent calls to retrieve body segments increment the index.

The acknowledge index is used only to make sure that no parts of the message get lost; it cannot be used to request a retransmission of a certain segment. If an error is detected during message retrieval, the host application must start retrieving the same message from the beginning.

SMS messages can be segmented as discussed in “[Appendix A: SMS Overview](#)” on page 293. To read an SMS message can require a sequence of [SMS Read \(r2\)](#) exchanges. See “[Reading incoming SMS messages](#)” on page 300.

After verifying, via [SMS Receive Status \(r2\) and Storage Capacity](#) (page 241), that there is a message to read, you make the first Get request to this service with the parameter (acknowledge index) set to 0. The modem responds with either a type 1 or type 3 [packet](#). If you receive a type 3 response, the entire SMS message is in that response. If you get a type 1 packet instead, it contains only the SMS message header; the body will follow in subsequent exchanges. For a description of the SMS message header, see page 228.

Where the SMS message has been segmented—a type 1 response to the first Get request—you must continue to issue Get requests, incrementing the acknowledge index each time; the host is responsible for retrieving all segments of the message through multiple Get Requests. The modem responds to each request with the next segment of the SMS message body. This exchange continues until the host receives a response with packet type 2, indicating the last segment.

The host must re-assemble the SMS message from the segments.

The host must retrieve one message completely before requesting the read of another message; otherwise, the modem returns an error.

Once the message is completely read from the new urgent or new normal list, it is automatically moved to the read list; thus, the index of messages in the new, unread lists changes. For example, there are 3 new normal messages, and the host requests to read message index 2 from this list. The modem returns the message stored in index 2, and then moves it to the read list. This causes the new normal message index 3 to be adjusted to index 2.

It is recommended that the host always use message index 1 when retrieving new messages, until the modem returns an error. This ensures that all messages are read, even if a new message comes in the middle of the retrieval.

Reading messages out of sequence makes the most practical sense when retrieving messages from the read list.

To determine real-time message sequence (i.e. the order in which the modem received the SMS messages), use the time stamp in the [SMS header](#).

When retrieving voice mail notification messages, the SMS message body will contain something like “Press SEND to check mailbox”. Your application must check the Message Teleservice field in the SMS header to determine that this is a voice mail indication.

The count (number of voice messages) can be retrieved, using [SMS Receive Status \(r2\) and Storage Capacity](#) (page 241). The message body of the voicemail message cannot be retrieved.

SMS Delete (r2)

Delete a retrieved (via [SMS Read \(r2\)](#)) SMS message.

| | | | | |
|-------------------|------------|-----------------|-------------|---|
| Object ID | 0x200A | SMS Delete (r2) | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 6 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Message category: <ul style="list-style-type: none"> 1 = New urgent messages 2 = New normal messages 3 = Old (read) messages 4 = Voicemail |
| | | 4 | UINT16 | Message to delete. <ul style="list-style-type: none"> 0xFFFF = Delete all messages in the specified SMS category Other value: Message index to delete (index in the "Message category" list [see above]). Starts at 1. <p>If Message category = 4, the voicemail message is not deleted; rather, the voicemail count is reset to 1.</p> |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Result: <ul style="list-style-type: none"> 0x00: Success (message was deleted successfully, or the list is already empty list, or empty message). 0x01: General failure 0x02: Specified message list is empty 0x03: Specified message list is invalid 0x04: Specified message ID is invalid |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in CDMA SMS object set Release | 2 |

This command does not delete the “voice” SMS message, used to convey the number of voice messages. If it becomes necessary to delete that message, use **SMS Voice Message Delete** (0x2006; page 240).

14: IOTA Messages

Introduction

This chapter describes [CnS](#) messages that are related to IOTA. IOTA (Internet Over The Air), supported by some service providers, is an automated feature to perform account setup for you by making a connection to the [CDMA](#) network and using a secure Internet connection to download account parameters to your modem.

IOTA messages summary

Table 27: IOTA messages

| Object | ID | Description | S | G | N |
|------------------------------------|--------|---|---|---|---|
| Start IOTA Session | 0x2100 | Initiate an IOTA session | ✓ | ✗ | ✗ |
| Stop IOTA Session | 0x2101 | Abort an IOTA session | ✓ | ✗ | ✗ |
| IOTA Status | 0x2102 | Obtain information about the current/most recent IOTA session | ✗ | ✗ | ✓ |
| IOTA WAP Push | 0x2103 | Obtain the IOTA URL from the WAP push | ✗ | ✗ | ✓ |

IOTA reference

Start IOTA Session

Initiates the CIIP (client initiated initial provisioning) embedded [IOTA](#) session. Upon this request, the modem starts the provisioning process.

This object replaces object ID 0x7000.

| | | | | |
|-------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x2100 | IOTA Start | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Stop IOTA Session

Requests that the current embedded [IOTA](#) session be stopped. If no IOTA session is in progress, this message is ignored by the modem.

This object replaces object ID 0x7001.

| | | | | |
|-------------------|---------------|---------------|-------------|--------------------|
| Object ID | 0x2101 | IOTA Stop | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

IOTA Status

Enables the embedded [IOTA](#) session status notifications for embedded IOTA.

This object replaces object ID 0x7002.

| | | | | |
|--------------------|---------------|---------------|---------------------------------|---|
| Object ID | 0x2102 | IOTA Status | | |
| Operations | X Set | X Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 8 | 0 | UINT16 | Event 0x0000 : Session started 0x0001 : Session ended |
| | | 2 | UINT16 | Event status 0x0000 : Success 0x0001 : Failure |
| | | 4 | UINT16 | Reason of session failure 0x0000 : Success 0x0001 : Error: Disconnect 0x0002 : Error: Not allowed 0x0003 : Error: Already active 0x0004 : Client abort 0x0005 : Error: HTTP 0x0006 : Error: MIME 0x0007 : Error: XML 0x0008 : Error: MMC 0x0009 : Error: Not trusted 0x000A : Session timeout 0x000B : Error: Unknown 0x000C : Error: Internal 0x000D : Error: No Service 0x000E : Error: Active data call 0x000F : Error: Active voice call 0x0010 : Error: No digital service |
| | | 6 | UINT16 | Extended reason of session failure If Reason of session failure = 0x0005: refer to HTTP error codes |

| | | | |
|--|--|--|---|
| | | | <p>If Reason of session failure = 0x0008:</p> <p>0x0000: Normal Release (MMC “disconnect”)</p> <p>0x0001: MMC: Invalid valueref</p> <p>0x0002: MMC: CID missing</p> <p>0x0003: MMC : Download failed</p> <p>0x0004: reserved</p> <p>0x0005: MMC: Invalid content</p> <p>0x0006: MMC: User denied</p> <p>0x0007: MMC: Bad value</p> <p>0x0008: MMC: Timeout</p> <p>0x0009: MMC: New doc arrived</p> <p>0x000A: MMC: Delete failed</p> <p>0x000B: MMC: Bad data</p> <p>0x000C: MMC: Display disabled</p> <p>0x000D: MMC: Write only</p> <p>0x000E: MMC: Abort</p> <p>0x000F: reserved</p> <p>0x0010: MMC: Bad object</p> <p>0x0011: MMC: Unknown error</p> <p>0x0012: MMC RESULT FILE LOCKED</p> <p>0x0013: MMC RESULT FILE OPEN</p> <p>0x0014: MMC RESULT CHECK FAILED</p> <p>0x0015: MMC RESULT INVALID SIZE</p> <p>0x0016: MMC RESULT INVALID OFFSET</p> <p>0x0017: MMC RESULT DEVICE BUSY</p> <p>0x0018: Reserved</p> |
|--|--|--|---|

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

IOTA WAP Push

Requests that the modem notifies the host when a URL is received in a [WAP](#) push.

This object replaces object ID 0x7003.

| | | | | |
|--------------------|---------------|---------------|---|---|
| Object ID | 0x2103 | IOTA WAP Push | | |
| Operations | X Set | X Get | ✓ Notify (high priority) | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 128 | 0 | UINT8 | URL (null-terminated, up to 128 characters) |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

15: Location Based Services

Introduction

Location Based Services (LBS) is a feature supported by some carriers/networks. The CDMA network is used to assist the modem to acquire a location fix.

For message flow diagrams, see Figure 5 “LBS single fix session” (page 313) and Figure 6 “LBS tracking session” (page 314).

For more information on the implementation of LBS in Sierra Wireless products, see document 2130907.

Types of LBS clients

Registered clients are clients that are interested in receiving notification about LBS events.

A requesting (or active) client is a client that has initiated the current LBS session or event. A requesting client is typically also a registered client.

LBS messages summary

Table 28: LBS messages

| Object | ID | Description | S | G | N |
|------------------------------|--------|--|---|---|---|
| PD download | 0x0F01 | Perform a download of the ephemeris and almanac data | ✓ | ✗ | ✗ |
| PD get position | 0x0F02 | Initiate a location fix | ✓ | ✗ | ✗ |
| LBS status | 0x0F03 | Report the current status of the LBS | ✗ | ✓ | ✓ |
| PD track | 0x0F04 | Start a tracking session related to LBS NV items | ✓ | ✗ | ✗ |
| PD end session | 0x0F05 | Stop the current PD session | ✓ | ✗ | ✗ |
| PA parameter settings | 0x0F06 | Report the modem's default parameter settings used during LBS sessions | ✗ | ✓ | ✗ |
| Location notification status | 0x0F07 | Enables/disables, or reports the current notification state of a group of notifications related to LBS location session data | ✓ | ✓ | ✗ |

| Object | ID | Description | S | G | N |
|-------------------------------|--------|--|---|---|---|
| Download notification status | 0x0F08 | Enables/disables, or reports the current notification state of a group of notifications related to LBS data download | ✓ | ✓ | ✗ |
| Parameter notification status | 0x0F09 | Enables/disables, or reports the current notification state of a group of notifications related to LBS NV items | ✓ | ✓ | ✗ |
| Position Fix Data | 0x0F0A | Notifies the registered host clients of the results of the position fix | ✗ | ✓ | ✓ |
| Position Fix Session Done | 0x0F0B | Notifies the registered host clients that a position determination fix session has completed | ✗ | ✗ | ✓ |
| Position Fix Session End | 0x0F0C | Notifies the host clients if the position determination fix session has ended in error. | ✗ | ✗ | ✓ |
| Position Fix Session Begin | 0x0F0D | Notifies the registered host clients that a position determination fix session has begun | ✗ | ✗ | ✓ |
| PD Update Failed | 0x0F0E | Notifies the registered host clients that a position determination session has failed | ✗ | ✗ | ✓ |
| Download Data Session Begin | 0x0F0F | Notifies the registered host clients that a data download session of ephemeris and almanac data has begun | ✗ | ✗ | ✓ |
| Download Data | 0x0F10 | Notifies the registered host clients that ephemeris and almanac data has been downloaded | ✗ | ✗ | ✓ |
| Download Data Session Done | 0x0F11 | Notifies the registered host clients that a position determination download session has completed | ✗ | ✗ | ✓ |
| Download Data Session End | 0x0F12 | Notifies the registered host clients if a position determination download data session has ended in error | ✗ | ✗ | ✓ |
| TCP/IP Address | 0x0F13 | Default value for the TCP/IP Address used in LBS sessions | ✓ | ✓ | ✓ |
| GPS Lock | 0x0F14 | Default value for the GPS Lock used in LBS sessions | ✓ | ✓ | ✓ |
| Transport Mechanism | 0x0F15 | Default value for the mechanism used to transport LBS messages | ✓ | ✓ | ✓ |
| Port ID | 0x0F16 | Default value for the Port ID used in LBS sessions | ✓ | ✓ | ✓ |

| Object | ID | Description | S | G | N |
|----------------------------|--------|--|---|---|---|
| Privacy Level | 0x0F17 | Default value for the privacy level used for LBS sessions | ✓ | ✓ | ✓ |
| Network Access Permissions | 0x0F18 | Default value for the Network Access permissions used during LBS sessions | ✓ | ✓ | ✓ |
| Base Station information | 0x0F19 | Information about the Base Station currently being accessed. | ✗ | ✓ | ✓ |
| Satellite Information | 0x0F1F | Satellites visible during the latest position fix session | ✗ | ✓ | ✓ |
| Clear GPS Assistance Data | 0x0F20 | Clear various location assistance parameters (to simulate a cold start) | ✓ | ✗ | ✗ |
| Get/set NV Item (page 133) | 0x1033 | GPS Coarse Location Message (CLM), QUALCOMM GPS Smart Mode, NMEA output message type, LBS Fix type | ✓ | ✗ | ✗ |

LBS reference

Location notification status

Enables/disables, or reports the current notification state of the following group of CnS notifications related to LBS location session data:

- **Position Fix Session Begin** (page 274)
- **Position Fix Data** (page 267)
- **Position Fix Session End** (page 272)
- **Position Fix Session Done** (page 271)
- **PD Update Failed** (page 274)

| | | | | |
|-------------------|--|------------------------------|-------------|---|
| Object ID | 0x0F07 | Location Notification Status | | |
| Operations | <input checked="" type="checkbox"/> Set <input checked="" type="checkbox"/> Get <input checked="" type="checkbox"/> Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT16 | Object version |
| 03 (Set) | | 2 | UINT16 | Location Notification Status: <ul style="list-style-type: none"> • 0: Notification Off • 1: Notification On |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Download notification status

Enables/disables, or reports the current notification state of the following group of CnS notifications related to [LBS](#) data download:

- [Download Data Session Begin](#) (page 275)
- [Download Data](#) (page 275)
- [Download Data Session Done](#) (page 277)
- [Download Data Session End](#) (page 277)

| | | | | |
|--------------------------------|--|------------------------------|-------------|---|
| Object ID | 0x0F08 | Download Notification Status | | |
| Operations | <input checked="" type="checkbox"/> Set <input checked="" type="checkbox"/> Get <input checked="" type="checkbox"/> Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 03 (Set) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Download Notification Status: <ul style="list-style-type: none"> • 0: Notifications Off • 1: Notifications On |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Parameter notification status

Enables/disables, or reports the current notification state of the following group of CnS notifications related to [LBS NV](#) items:

- [TCP/IP Address](#) (page 278)
- [GPS Lock](#) (page 279)
- [Transport Mechanism](#) (page 280)
- [Port ID](#) (page 281)
- [Privacy Level](#) (page 282)
- [Network Access Permissions](#) (page 284)

| | | | | |
|--------------------------------|--|-------------------------------|-------------|--|
| Object ID | 0x0F09 | Parameter Notification Status | | |
| Operations | <input checked="" type="checkbox"/> Set <input checked="" type="checkbox"/> Get <input checked="" type="checkbox"/> Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 03 (Set) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Parameter Notification Status: <ul style="list-style-type: none"> • 0: Notifications Off • 1: Notifications On |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

PD download

Requests the modem to perform a download of the ephemeris and almanac data. The host uses Download Option to indicate whether the data download should occur just once or periodically, in an effort to keep the data “warm”. If the application requesting this data will need the data over a period of time, the application should indicate that the download should be periodic (Download Option = 1).

| | | | | |
|-------------------|--|---------------|-------------|---|
| Object ID | 0x0F01 | PD Download | | |
| Operations | <input checked="" type="checkbox"/> Set <input checked="" type="checkbox"/> Get <input checked="" type="checkbox"/> Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 6 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Download Option: <ul style="list-style-type: none"> • 1: Periodic Download • 2: Single Download |

| | | | | |
|-----------------|---|---|--------|---|
| | | 4 | UINT16 | Download Duration: Session duration in minutes. Used only with Periodic Download option. |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PD Error Type: See Table 29 . |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Table 29: PD Error Type values

| Code | Meaning |
|--------|-------------------------|
| 0x0000 | No error |
| 0x0001 | Internal modem error |
| 0x0002 | Bad Service type |
| 0x0003 | Bad Session type |
| 0x0004 | Invalid privacy |
| 0x0005 | Invalid data download |
| 0x0006 | Invalid network access |
| 0x0007 | Invalid operation mode |
| 0x0008 | Invalid number of fixes |
| 0x0009 | Invalid server info |
| 0x000A | Invalid timeout |
| 0x000B | Invalid QOS parameter |

| Code | Meaning |
|--------|--|
| 0x000C | No session active |
| 0x000D | Session already active |
| 0x000E | Session busy |
| 0x000F | Phone is offline |
| 0x0010 | CDMA lock error |
| 0x0011 | GPS lock error |
| 0x0012 | Invalid state |
| 0x0013 | Connection failure |
| 0x0014 | No buffers available |
| 0x0015 | Searcher error |
| 0x0016 | Cannot report now |
| 0x0017 | Resource contention |
| 0x0018 | Mode not supported |
| 0x0019 | Authentication failed |
| 0x001A | Other error |
| 0x001B | Fix rate for tracking (time between fixes) too large |

PD get position

Requests the modem to initiate a location fix. If the modem is unable to initiate the location fix, an error code is returned. If the modem is able to initiate the location fix, then the response with NO ERR as error code is returned to release the client.

Subsequent notification messages inform the host of the progress and termination of the location fix.

For the position fix, the host can choose between the methods shown in Table 30. See also [Get/set NV Item](#) (page 133).

For message flow diagrams, see Figure 5 “[LBS single fix session](#)” (page 313) and Figure 6 “[LBS tracking session](#)” (page 314).

Table 30: Location fix methods

| Mode selection | Description |
|-----------------------------------|---|
| Standalone Only | Mobile does everything involved in computing its position. No PDE is required. |
| MS -Assisted Only | Mobile provides information to the PDE so that the PDE can compute the mobile's position. |
| MS -Based Only | Mobile gets assistance from the PDE , but the mobile computes position. |
| Optimized for Speed | MS -Based or MS -Assisted, depending on Performance and Accuracy settings (page 261). |
| Optimized for Accuracy | MS -Assisted preferred; if MS -Assisted fails, then MS -Based |
| Optimized for Data | MS -Based preferred; if MS -Based fails, then MS -Assisted |




| | | | | |
|-------------------|---------------|-----------------|-------------|---|
| Object ID | 0x0F02 | PD Get Position | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 10 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Fix Type (described in Table 30 on page 260): <ul style="list-style-type: none"> 1: StandAlone 2: MS-Based Only 3: MS-Assisted Only 4: Optimized for Speed 5: Optimized for Accuracy 6: Optimized for Data |
| | | 4 | UINT16 | Performance: Time allowed for Satellite acquisition (in seconds). |
| | | 6 | UINT32 | Accuracy: Accuracy limit preferred, in meters. |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PD Error Type: See Table 29 on page 258. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

LBS status

Requests the modem to provide the current status of the [LBS](#).

| | | | | |
|-----------------------------------|---|---|--|---|
| Object ID | 0x0F03 | LBS Status | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 82 | 0 | UINT16 | Object version |
| | | 2 | CHAR | Last Fix Timestamp: 16 Character string in the format YYYYMMDDdHHMMSS<null> where d = day of week, 0 (Mon) – 6 (Sun) |
| | | 18 | UINT16 | Last Fix Status: <ul style="list-style-type: none"> • 0: None • 1: Active • 2: Success • 3: Failed |
| | | 20 | UINT16 | Last Fix Error: See Table 32 on page 272. |
| | | 22 | CHAR | Last Download Timestamp: Same as Last Fix Timestamp above. |
| | | 38 | UINT16 | Last Download Status: Same as Last Fix Status above. |
| | | 40 | UINT16 | Last Download Error: See Table 32 on page 272. |
| | | 42 | CHAR | Last Fix Session Timestamp: Same as Last Fix Timestamp above. |
| | | 58 | UINT16 | Last Fix Session Status: Same as Last Fix Status above. |
| 60 | UINT16 | Last Fix Session Error: See Table 32 on page 272. | | |

| | | | | |
|--|--|----|--------|--|
| | | 62 | CHAR | Last Download Session Timestamp: Same as Last Fix Timestamp above. |
| | | 78 | UINT16 | Last Download Session Status: Same as Last Fix Status above. |
| | | 80 | UINT16 | Last Download Session Error: See Table 32 on page 272. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

PD track

Requests the modem to start a tracking session related to [LBS NV](#) items. This command should be used when an application requires multiple location fixes over a small period of time (for example, a fix between every 1 to 240 seconds). The host can specify the fix type to be used for the tracking session (see [Table 30](#) on page 260). Using the [Fix Count](#) and [Fix Rate](#) parameters, the application may specify how many fixes should be done and at what rate they should occur. ([Fix Rate](#) is measured as the number of seconds between the start of one fix to the time that the subsequent fix is triggered.)

Due to the potential need for first updating the ephemeris, almanac and/or location data, the “time to first fix” may require more time than the subsequent fixes. Tracking applications need to consider this possible initial delay and may want to behave “proactively” by initiating a Data Download session ([PD download](#); page 257) and/or a single location fix ([PD get position](#); page 260) as part of their startup sequence. As a guideline: almanac data is valid for 3 to 4 days; ephemeris data is valid for 30 to 120 minutes; coarse location data is valid for 4 minutes.

If the modem is unable to initiate the tracking session, an error code is returned. If the modem is able to initiate the tracking session, then the NO ERR error code is returned to release the client. Subsequent notification messages inform the host of the tracking session progress and termination.

| | | | | |
|-------------------|--------------------------------|---------------|-------------|---|
| Object ID | 0x0F04 | PD Track | | |
| Operations | ✓ Set ✗ Get ✗ Notify | | | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 16 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Fix Type (described in Table 30 on page 260): <ul style="list-style-type: none"> • 1: StandAlone • 2: MS-Based Only • 3: MS-Assisted Only • 4: Optimized for Speed • 5: Optimized for Accuracy • 6: Optimized for Data |
| | | 4 | UINT16 | Performance: Time allowed for Satellite acquisition (in seconds). |
| | | 6 | UINT32 | Accuracy: Accuracy limit preferred (in meters). |
| | | 10 | UINT16 | Fix Count: Number of fixes to obtain. Between 1 and 999. For continuous mode, use 1000. |
| | | 12 | UINT32 | Fix Rate: Frequency of Fixes (in seconds). The number of seconds between the start of one fix to the time that the subsequent fix is triggered. |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PD Error Type: See Table 29 on page 258. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

PD end session

Requests that the current session of the type specified in [Session Type](#) be stopped. One of each type of session may be active simultaneously.

| | | | | |
|-------------------|---------------|----------------|-------------|--|
| Object ID | 0x0F05 | PD End Session | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Session Type: <ul style="list-style-type: none"> 0: Location Fix Session 1: Download Data Session |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PD Error Type: See Table 29 on page 258. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

PA parameter settings

Requests the modem's default parameter settings used during [LBS](#) sessions. The entire set of position determination parameters is returned to the host ([IP Address](#), [Port ID](#), [GPS Lock](#), [Transport Mechanism](#), [Privacy](#), [Network Access](#)).

| Object ID | 0x0F06 | PA Parameter Settings | | |
|------------|---|---|--|--|
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 18 | 0 | UINT16 | Object version |
| | | 2 | UINT32 | TCP/IP Address: TCP/IP Address of PDE . Used when Transport Mechanism is TCP/IP. |
| | | 6 | UINT32 | Port ID: Port ID of the PDE . Used when Transport Mechanism is TCP/IP. |
| | | 10 | UINT16 | GPS Lock: <ul style="list-style-type: none"> • 0: No GPS Lock Active • 1: MI Sessions not allowed • 2: MT Sessions not allowed • 3: E911 Only Allowed |
| | | 12 | UINT16 | Transport Mechanism: <ul style="list-style-type: none"> • 0: TCP/IP • 1: DBM |
| | | 14 | UINT16 | Privacy: <ul style="list-style-type: none"> • 0: Low • 1: Medium • 2: High |
| | | 16 | UINT16 | Network Access: <ul style="list-style-type: none"> • 0: Network Only • 1: No Network Access Allowed • 2: Network with Demodulation • 3: Network without Demodulation |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Position Fix Data

Notifies the registered host clients of the results of the position fix. If the host performs a Get on this object when there is no position fix to report, the modem responds with a generic "CnS Unavailable" error message.

For message flow diagrams, see Figure 5 "[LBS single fix session](#)" (page 313) and Figure 6 "[LBS tracking session](#)" (page 314).

| | | | | |
|-----------------------------------|---------------|-------------------|-----------------|--|
| Object ID | 0x0F0A | Position Fix Data | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 45 | 0 | UINT16 | Object version |
| | | 2 | INT32 | Latitude 25 bit field. Formatted as 2's complement number in units of $180/2^{25}$ degrees between +90 and -90 degrees. |
| | | 6 | INT32 | Longitude 26 bit field. Formatted as 2's complement number in units of $360/2^{26}$ degrees between +180 and -180 degrees. |
| | | 10 | UINT32 | Timestamp Timestamp of the location response (CDMA System Time) |
| | | 14 | UINT16 | Location Uncertainty Angle Formatted in integer steps of 5.625 degrees. |

| | | | | |
|--|--|----|--------|--|
| | | 16 | UINT16 | Location Uncertainty A Formatted as index into position uncertainty lookup table (Table 31 on page 269). |
| | | 18 | UINT16 | Location Uncertainty Position Formatted as index into position uncertainty lookup table (Table 31 on page 269). |
| | | 20 | UINT16 | Fix Type 0: 2-D Fix 1: 3-D Fix |
| | | 22 | UINT16 | Height Included 0: Height information not valid 1: Height information valid |
| | | 24 | INT16 | Height, in meters. If Height Included =1, this field is valid. |
| | | 26 | UINT16 | Location Uncertainty Vertical If Height Included =1, this field is valid. Formatted as index into position uncertainty lookup table (Table 31 on page 269). |
| | | 28 | UINT16 | Velocity Included 0: Velocity information not valid 1: Velocity information valid |
| | | 30 | UINT16 | Heading If Velocity Included = 1, this field is valid. Formatted in units of $360/2^{10}$ degrees. |
| | | 32 | UINT16 | Velocity Horizontal If Velocity Included = 1, this field is valid. Formatted in integer steps of 0.25 meters per second. |
| | | 34 | INT16 | Velocity Vertical If Velocity Included = 1 and Fix Type = 1, this field is valid. Formatted in integer steps of 0.5 meters per second. |

| | | | | |
|--|--|----|-------|---|
| | | 36 | INT16 | Uncertainty A Formatted in integer steps of 0.25 meters. |
| | | 38 | INT16 | Uncertainty P Formatted in integer steps of 0.25 meters. |
| | | 40 | INT16 | Uncertainty V If Height Included = 1, this field is valid. Formatted in integer steps of 0.25 meters. |
| | | 42 | INT16 | HEPE Formatted in integer steps of 0.1 meters. |
| | | 44 | UINT8 | Number of satellites used in position fix (maximum 12) |

Version information:

| | |
|--|---|
| Object version | 2 |
| Added in LBS CnS object set Release | 1 |

Table 31: Position Uncertainty

| Position Uncertainty index | Standard deviation for Position Uncertainty (meters) |
|----------------------------|--|
| 0x0000 | 0.5 |
| 0x0001 | 0.75 |
| 0x0002 | 1 |
| 0x0003 | 1.5 |
| 0x0004 | 2 |
| 0x0005 | 3 |
| 0x0006 | 4 |
| 0x0007 | 6 |
| 0x0008 | 8 |

| Position Uncertainty index | Standard deviation for Position Uncertainty (meters) |
|----------------------------|--|
| 0x0009 | 12 |
| 0x000A | 16 |
| 0x000B | 24 |
| 0x000C | 32 |
| 0x000D | 48 |
| 0x000E | 64 |
| 0x000F | 96 |
| 0x0010 | 128 |
| 0x0011 | 192 |
| 0x0012 | 256 |
| 0x0013 | 384 |
| 0x0014 | 512 |
| 0x0015 | 768 |
| 0x0016 | 1,024 |
| 0x0017 | 1,536 |
| 0x0018 | 2,048 |
| 0x0019 | 3,072 |
| 0x001A | 4,096 |
| 0x001B | 6,144 |
| 0x001C | 8,192 |

| Position Uncertainty index | Standard deviation for Position Uncertainty (meters) |
|----------------------------|--|
| 0x001D | 12,288 |
| 0x001E | >12,288 |
| 0x001F | Not computable |

Position Fix Session Done

Notifies the [registered](#) host clients that a position determination fix session has completed.

For message flow diagrams, see Figure 5 “[LBS single fix session](#)” (page 313) and Figure 6 “[LBS tracking session](#)” (page 314).

| | | | | |
|--------------------|---------------|---------------------------|-----------------|--------------------|
| Object ID | 0x0F0B | Position Fix Session Done | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Position Fix Session End

Notifies the host clients if the position determination fix session has ended in error.

| | | | | |
|--------------------|---------------|--------------------------|-----------------|---|
| Object ID | 0x0F0C | Position Fix Session End | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | End Status: See Table 32 that follows. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Table 32: End Status values

| Code | Meaning |
|--------|---------------------------|
| 0x0000 | Offline |
| 0x0001 | No service |
| 0x0002 | No connection |
| 0x0003 | No data |
| 0x0004 | Session busy |
| 0x0005 | CDMA lock |
| 0x0006 | GPS lock |
| 0x0007 | Connection failed |
| 0x0008 | Error state |

| Code | Meaning |
|--------|--|
| 0x0009 | Client ended |
| 0x000A | UI ended |
| 0x000B | Network ended |
| 0x000C | Timeout |
| 0x000D | Privacy level |
| 0x000E | Network access error |
| 0x000F | Fix error |
| 0x0010 | PDE rejected |
| 0x0011 | Traffic channel exited |
| 0x0012 | E911 |
| 0x0013 | Server error |
| 0x0014 | Stale BS information |
| 0x0015 | Resource contention |
| 0x0016 | Authentication parameter failed |
| 0x0017 | Authentication failed local |
| 0x0018 | Authentication failed network |
| 0x0400 | Fix error; fall back to Smart Mode Stand Alone GPS fix |
| 0x0401 | Fix error; fall back to Smart Mode MS-base GPS fix |

Position Fix Session Begin

Notifies the [registered](#) host clients that a position determination fix session has begun.

For message flow diagrams, see Figure 5 “[LBS single fix session](#)” (page 313) and Figure 6 “[LBS tracking session](#)” (page 314).

| | | | | |
|--------------------|---------------|----------------------------|-----------------|--------------------|
| Object ID | 0x0F0D | Position Fix Session Begin | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 0 | | | None |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

PD Update Failed

Notifies the [registered](#) host clients that a position determination session has failed.

| | | | | |
|--------------------|---------------|------------------|-----------------|--------------------|
| Object ID | 0x0F0E | PD Update Failed | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 0 | | | None |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Download Data Session Begin

Notifies the [registered](#) host clients that a data download session of ephemeris and almanac data has begun.

For message flow diagrams, see Figure 5 “[LBS single fix session](#)” (page 313) and Figure 6 “[LBS tracking session](#)” (page 314).

| | | | | |
|--------------------|---------------|-----------------------------|-----------------|--------------------|
| Object ID | 0x0F0F | Download Data Session Begin | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Download Data

Notifies the [registered](#) host clients that ephemeris and almanac data has been downloaded.

For message flow diagrams, see Figure 5 “[LBS single fix session](#)” (page 313) and Figure 6 “[LBS tracking session](#)” (page 314).

| | | | | |
|--------------------|---------------|---------------|-----------------|--|
| Object ID | 0x0F10 | Download Data | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 24 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | Number of Valid Entry pairs Indicates how many entries (below) contain valid data. |

| | | |
|----|--------|--|
| 4 | UINT16 | Entry 1 Data Type: <ul style="list-style-type: none"> • 1: Location Download • 2: AA Download • 4: SA Download • 8: Epemeris Download • 16: Almanac Download |
| 6 | UINT16 | Entry 1 Source: <ul style="list-style-type: none"> • 0: Mobile • 1: PDE |
| 8 | UINT16 | Entry 2 Data Type: Same as Entry 1 Data Type above. |
| 10 | UINT16 | Entry 2 Source: Same as Entry 1 Source above. |
| 12 | UINT16 | Entry 3 Data Type: Same as Entry 1 Data Type above. |
| 14 | UINT16 | Entry 3 Source: Same as Entry 1 Source above. |
| 16 | UINT16 | Entry 4 Data Type: Same as Entry 1 Data Type above. |
| 18 | UINT16 | Entry 4 Source: Same as Entry 1 Source above. |
| 20 | UINT16 | Entry 5 Data Type: Same as Entry 1 Data Type above |
| 22 | UINT16 | Entry 5 Source: Same as Entry 1 Source above. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Download Data Session Done

Notifies the [registered](#) host clients that a position determination download session has completed.

For message flow diagrams, see Figure 5 “[LBS single fix session](#)” (page 313) and Figure 6 “[LBS tracking session](#)” (page 314).

| | | | | |
|--------------------|---------------|----------------------------|-----------------|--------------------|
| Object ID | 0x0F11 | Download Data Session Done | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 0 | | | None |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Download Data Session End

Notifies the [registered](#) host clients if a position determination download data session has ended in error.

| | | | | |
|--------------------|---------------|---------------------------|-----------------|--|
| Object ID | 0x0F12 | Download Data Session End | | |
| Operations | X Set | X Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 07 (Notify) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | End Status: See Table 32 on page 272. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

TCP/IP Address

Contains the default value for the TCP/IP Address used in [LBS](#) sessions.

| | | | | |
|--------------------|---------------|----------------|-------------|---|
| Object ID | 0x0F13 | TCP/IP Address | | |
| Operations | ✓ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 6 | 0 | UINT16 | Object version |
| 03 (Set) | | 2 | UINT32 | TCP/IP Address: TCP/IP Address to connect to for LBS activity when using TCP/IP transport mechanism. |
| 07 (Notify) | | | | |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PA Error Type: See Table 33 (below). |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Table 33: PA Error Type values

| Code | Meaning |
|--------|---|
| 0x0000 | No error |
| 0x0001 | Internal error |
| 0x0002 | Invalid client ID |
| 0x0003 | Error in parameter to set |
| 0x0004 | Error in lock type |
| 0x0005 | Phone is in offline state |

| Code | Meaning |
|--------|--|
| 0x0006 | A command to set parameter is already active |
| 0x0007 | Incorrect application-specific information |
| 0x0008 | Unknown error other than above |

GPS Lock

Contains the default value for the [GPS](#) Lock used in [LBS](#) sessions.

| Object ID | 0x0F14 | GPS Lock | | |
|-------------|---|---|--|--|
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT16 | Object version |
| 03 (Set) | | 2 | UINT16 | GPS Lock: <ul style="list-style-type: none"> 0: No GPS Lock Active 1: MI Sessions not allowed 2: MT Sessions not allowed 3: E911 Only Allowed |
| 07 (Notify) | | | | |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PA Error Type: See Table 33 on page 278. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Transport Mechanism

Contains the default value for the mechanism used to transport [LBS](#) messages.

| | | | | |
|--------------------|---------------|---------------------|-------------|--|
| Object ID | 0x0F15 | Transport Mechanism | | |
| Operations | ✓ Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT16 | Object version |
| 03 (Set) | | 2 | UINT16 | Transport Mechanism: <ul style="list-style-type: none"> • 0: TCP/IP • 1: DBM (Data burst) |
| 07 (Notify) | | | | |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PA Error Type: See Table 33 on page 278. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Port ID

Contains the default value for the Port ID used in [LBS](#) sessions.

| | | | | |
|--------------------|---|---|--|---|
| Object ID | 0x0F16 | Port ID | | |
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 6 | 0 | UINT16 | Object version |
| 03 (Set) | | 2 | UINT32 | Port ID: Port ID to connect to for LBS activity when using TCP/IP transport mechanism. |
| 07 (Notify) | | | | |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PA Error Type: See Table 33 on page 278. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Privacy Level

Contains the default value for the privacy level used for [LBS](#) sessions.

| | | | | |
|---------------------------------------|---|---|--|--|
| Object ID | 0x0F17 | Privacy Level | | |
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT16 | Object version |
| 03 (Set) 07 (Notify) | | 2 | UINT16 | Privacy: <ul style="list-style-type: none"> • 0: Low • 1: Medium • 2: High |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PA Error Type: See Table 33 on page 278. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Base Station information

Contains information about the Base Station currently being accessed.

Note: Depending on the firmware of your Sierra Wireless product, the base station information may or may not be accessible.

| | | | | |
|---|---------------|---|-----------------|--|
| Object ID | 0x0F19 | Base Station Information | | |
| Operations | X Set | ✓ Get | ✓ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 26 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | BS info stale: <ul style="list-style-type: none"> 0: Info is not stale 1: Info is stale |
| | | 4 | UINT16 | BS ID: Base Station ID |
| | | 6 | UINT16 | SID: Base Station SID |
| | | 8 | UINT16 | NID: Base Station NID |
| | | 10 | UINT32 | BS info time: Time at which the System Parameter Message was received from the Base Station. This is the high 32-bits (time is stored in the modem as two 32-bit fields). |
| | | 14 | UINT32 | BS info time: Time at which the System Parameter Message was received from the Base Station. This is the low 32-bits (time is stored in the modem as two 32-bit fields). |
| | | 18 | INT32 | BS latitude: Base Station Latitude |
| 22 | INT32 | BS longitude: Base Station Longitude | | |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Network Access Permissions

Contains the default value for the Network Access permissions used during [LBS](#) sessions.




| | | | | |
|---------------------------------------|---|---|--|---|
| Object ID | 0x0F18 | Network Access Permissions | | |
| Operations | <input checked="" type="checkbox"/> Set | <input checked="" type="checkbox"/> Get | <input checked="" type="checkbox"/> Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 4 | 0 | UINT16 | Object version |
| 03 (Set) 07 (Notify) | | 2 | UINT16 | Network Access: <ul style="list-style-type: none"> 0: Network Only 1: No Network Access 2: Network with Demodulation 3: Network without Demodulation |
| 04 (Ack) | 4 | 0 | UINT16 | Object version |
| | | 2 | UINT16 | PA Error Type: See Table 33 on page 278. |

Version information:

| | |
|--|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Satellite Information

Information about satellites visible during the latest position fix session and satellites used in the latest position fix.

| | | | | |
|-----------------------------------|---|---|--|--|
| Object ID | 0x0F1F | Satellite Information | | |
| Operations |  Set |  Get |  Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) 07 (Notify) | 63 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Number of Satellites in View (SV). Maximum 12. For each SV, its number, elevation, azimuth, and SNR are reported (see below). |
| | | 3 | UINT8 | 1st SV - Satellite Number <ul style="list-style-type: none"> Bits 0–6: Satellite number (0–32) Bit 7: Used in latest fix: 0: This satellite wasn't used. 1: This satellite was used. |
| | | 4 | UINT8 | 1st SV - Elevation Satellite elevation, in degrees (0-90) |
| | | 5 | UINT16 | 1st SV - Azimuth Satellite azimuth, in degrees (0-360) |
| | | 7 | UINT8 | 1st SV – SNR Satellite Signal/Noise Ratio in dB (0-99) |
| | | 8 | UINT8 | 2nd SV - Satellite Number <ul style="list-style-type: none"> Bits 0–6: Satellite number (0–32) Bit 7: Used in latest fix: 0: This satellite wasn't used. 1: This satellite was used. |
| | | | | |

| | | | | |
|--|--|----|--------|---|
| | | 58 | UINT8 | 12th SV - Satellite Number <ul style="list-style-type: none"> • Bits 0–6: Satellite number (0–32) • Bit 7: Used in latest fix: 0: This satellite wasn't used. 1: This satellite was used. |
| | | 59 | UINT8 | 12th SV - Elevation Satellite elevation, in degrees (0-90) |
| | | 60 | UINT16 | 12th SV - Azimuth Satellite azimuth, in degrees (0-360) |
| | | 63 | UINT8 | 12th SV – SNR Satellite Signal/Noise Ratio in dB (0-99) |

Version information:

| | |
|-------------------------------------|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

Clear GPS Assistance Data

Clear various location assistance parameters (to simulate a cold start).

| | | | | |
|-------------------|---------------|---------------------------|-------------|---|
| Object ID | 0x0F20 | Clear GPS Assistance Data | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 9 | 0 | UINT16 | Object version |
| | | 2 | UINT8 | Delete Ephemeris Data: <ul style="list-style-type: none"> • 0: Do nothing • 1: Delete the satellite ephemeris data |
| | | 3 | UINT8 | Delete Almanac Data: <ul style="list-style-type: none"> • 0: Do nothing • 1: Delete the satellite almanac data |

| | | | | |
|-----------------|---|---|-------|---|
| | | 4 | UINT8 | Delete Position Data: <ul style="list-style-type: none"> 0: Do nothing 1: Delete the position and time information |
| | | 5 | UINT8 | Clear Standalone Self Learning Data Base: <ul style="list-style-type: none"> 0: Do nothing 1: Clear the Standalone Self Learning data base |
| | | 6 | UINT8 | Clear Standalone Serving System Data Base: <ul style="list-style-type: none"> 0: Do nothing 1: Clear the Standalone Serving System data base |
| | | 7 | UINT8 | Clear Standalone Time Reference: <ul style="list-style-type: none"> 0: Do nothing 1: Clear the Standalone Time Reference |
| | | 8 | UINT8 | Reset MSB Throttling Parameters: <ul style="list-style-type: none"> 0: Do nothing 1: Reset the MS-based throttling parameters |
| 04 (Ack) | 0 | | | None |

Version information:

| | |
|-------------------------------------|---|
| Object version | 1 |
| Added in LBS CnS object set Release | 1 |

16: Additional Features

Introduction

This chapter provides reference for CnS messages that handle additional features of the modem.

Feature summary

Table 34: Additional feature messages

| Object | ID | Description | S | G | N |
|-----------------------------------|--------|---|---|---|---|
| Modem Lock Status | 0x1018 | Report whether the modem is PIN security locked | X | ✓ | X |
| Lock/Unlock Modem | 0x1019 | Lock or unlock the modem with a 4-digit lock code | ✓ | X | X |
| Set Locking PIN | 0x101A | Set or change the security PIN | ✓ | X | X |
| Enable/Disable Power-Up Auto-Lock | 0x1043 | Control the use of PIN security lock on power up | ✓ | ✓ | X |
| Scratchpad | 0x5000 | Read or write to the scratchpad | ✓ | ✓ | X |

Reference

Modem Lock Status

Reports whether the modem is PIN security locked.

| | | | | |
|-------------------|---------------|---------------|-----------------|---|
| Object ID | 0x1018 | Lock Status | | |
| Operations | X Set | ✓ Get | X Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Status in one of two states: <ul style="list-style-type: none"> • 0x0000: Unlocked • Any other value indicates that the modem is locked. |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

When this reports a modem lock, the application is expected to enforce limited usability. The modem enforces the following:

- AT command interface is disabled except the unlock command
- [CnS](#) interface remains enabled
- For products that support voice [MC5728V MiniCard]:
 - Outgoing dialing (voice, fax, data) is limited to [E911](#) service
 - Incoming voice calls do not provide caller ID

Lock/Unlock Modem

Locks or unlocks the modem with a 4-digit lock code.

| | | | | |
|-------------------|---------------|---------------|-------------|--|
| Object ID | 0x1019 | Lock | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 4 | 0 | CHAR | String of 4 ASCII characters |
| 04 (Ack) | 2 | 0 | UINT16 | Lock state: <ul style="list-style-type: none"> • 0x0000: Modem is unlocked • 0x0001: Modem is locked |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The updated lock state is returned in the acknowledgement.

Set Locking PIN

Changes the personal identification number (PIN) for locking the modem. Requires the previous PIN in to permit the change.

| | | | | |
|-------------------|---------------|-----------------|-------------|--|
| Object ID | 0x101A | Change Lock PIN | | |
| Operations | ✓ Set | ✗ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 03 (Set) | 8 | 0 | CHAR | Current passcode: string of 4 ASCII characters as a password. |
| | | 4 | CHAR | New passcode: string of 4 ASCII characters to set as new PIN. |
| 04 (Ack) | 2 | 0 | UINT16 | Password state: <ul style="list-style-type: none"> • 0x0000: Password rejected • 0x0001: Password accepted |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Enable/Disable Power-Up Auto-Lock

Sets whether the modem locks on power-up, or gets the current power-up lock status.

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x1043 | Auto Lock | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 2 | 0 | UINT16 | Auto lock state: <ul style="list-style-type: none"> • 0x0000: Disable power-up lock • Any other value enables power-up lock. |
| 03 (Set) | 2 | 0 | UINT16 | |

Version information:

| | |
|---|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

Scratchpad

Read or write to the modem's non-volatile scratchpad.

| | | | | |
|-------------------|---------------|---------------|-------------|---|
| Object ID | 0x5000 | Scratchpad | | |
| Operations | ✓ Set | ✓ Get | ✗ Notify | |
| Parameters | Len | Offset | Type | Description |
| 01 (Get) | 0 | | | None |
| 02 (Reply) | 0–20 | 0 | CHAR | Scratchpad contents (not null-terminated) |
| 03 (Set) | 0–20 | 0 | CHAR | Scratchpad setting (not null-terminated) |
| 04 (Ack) | 0–20 | 0 | CHAR | Scratchpad contents echoed back (not null-terminated) |

Version information:

| | |
|--------------------------------------|---|
| Object version | 1 |
| Added in CDMA CnS object set Release | 1 |

The modem has a 20-byte scratchpad space in non-volatile memory. Any string can be stored in the scratchpad. This can be used to store OEM configuration notes or revision tags, or any other data you choose.

The length of the string can be from 0–20 characters. It does not need to be null-terminated.

Any Set command erases the previous contents of the scratchpad and replaces it with the new string.

17: Appendix A: SMS Overview

SMS overview

The modem supports both receiving and sending Short Message Service messages.

To make use of these features, you should be familiar with IS-637, the standard for CDMA Short Message Service. There are many parameters in the [headers](#) of SMS messages that require an understanding of the standard to use successfully.

Incoming messages

Messages arriving from the network (mobile terminated) are stored in the modem. They are organized into three types:

- Urgent
- Normal (text)
- Voice

The modem maintains three separate LIFO queues, one for each type. It can buffer up to 98 text messages (urgent or normal) and one voice message. The single SMS voice message is overwritten within the modem by subsequent incoming SMS voice messages since this type of message only conveys the current count of recorded voice messages being held by the network.

If an SMS text message arrives while the modem has no capacity to store it, the message is rejected. The handling at that point is up to the network; some hold for re-transmission later, others may discard the message, still others may take other actions.

If SMS release 1 (“r1”) objects are used

Note: For information on r1 and r2 objects, see page 60.

When messages are retrieved from the modem (via [SMS Read \(r1\)](#); page 233), they are retrieved on a “last-in first-out” (LIFO) basis from each queue in the order given above. That is, the modem checks the “urgent” queue first. If it is empty, it checks the “normal” queue, and lastly the “voice” queue. When a message is encountered, the most recent message in the queue is read first, then the next message, and so on, until the oldest message in the queue is read, emptying that queue.

If a new message arrives into an empty queue of higher priority during the process of reading through the buffered messages, it will be delivered before returning to the lower priority queue.

Messages remain in the queue until deleted (via [SMS Delete \(r1\)](#); page 235). That means you must expressly delete each message after successfully

transferring it from the modem to the host. This is done to ensure that the message is not lost before the host has retrieved and stored it.

An attempt to read the next SMS message **without** deleting the previously read message will result in the modem returning the same (already read) message again (unless a higher priority message has arrived in the interim).

If SMS release 2 (“r2”) objects are used

Note: For information on r1 and r2 objects, see page 60.

The host application specifies, via **SMS Read (r2)** (page 242), the message queue and the message to read.

Once the message is read, it is placed in the old (read) message list, and this list’s message count, and the indexes in the remaining (unread) messages of the queue, are adjusted accordingly.

Voice mail notification

Service providers usually issue an SMS message to notify the mobile of a change in the number of voice mail messages held by the network. As new voice mail messages are left, the mobile receives an SMS message indicating the number of voice mail messages being held. This is the information held in the single message in the **Voice** queue.

The modem, as shipped from the factory, does not contain any SMS messages. After receiving the first voice mail notification message, the modem opens the one message in the voice queue and stores the count of messages being held. As the network sends more notifications, the content of this one message is updated.

The usual command to delete SMS messages from the SMS message queue does **not** remove the voice mail notification message. Even when the number of voice mail messages held by the network is zero, there will always be this one SMS Voice queue message; the content of the message indicates there are zero voice messages pending.

If it becomes necessary to delete the voice mail message, use the separate CnS message **SMS Voice Message Delete** (page 240).

Outgoing messages

Note: *The outbox contents are lost when power is removed from the modem.*

The modem maintains an outbox of SMS messages for transmission. You first write messages (up to ten) to the outbox, then send the outbox contents to the network as a separate step.

This batching capability is provided for two reasons:

- The host application may allow the user to send a single message to a list of recipients. The host must store one copy of the message at the modem for each recipient in the list. Storing up to ten copies at a time allows the modem to save time when actually transmitting to the network.
- The modem must establish a [traffic channel](#) connection to send SMS messages. Doing it once to send a batch is more efficient than setting up the connection for each individual message.

Sending SMS messages involves five steps:

1. Compose and store the message on the host.
2. Write the message to the modem's outbox ([SMS Write](#); page 236). (Repeat steps 1 and 2, up to ten messages.)

Note: *There is no mechanism to query the number of messages currently in the outbox.*

3. Send the outbox contents to the network ([SMS Send](#); page 237).
4. Get notification of the transmission status (successful or not) of each message sent ([SMS Send Status](#); page 238).
5. When successfully sent, optionally delete the message at the host. If a message reports an error, the message can be re-sent (step 2).

Note: *Once written to the outbox, there is no way to recall or delete the SMS message. The only alternative is to power-cycle the modem. The outbox contents are lost when power is removed.*

First, compose the message and store it at the host, giving the message a unique ID number for tracking within the application and modem. Storing it at the host is needed to recover from transmission errors.

The second step is to write the outgoing message to an outbox buffer in the modem (via [SMS Write](#); page 236). The modem can store up to ten SMS messages in the outbox. Any attempt to write an eleventh message results in the [CnS](#) error response.

When you have written all desired messages, you use [SMS Send](#) to have the modem transmit the message(s) to the network.

As each message is sent, the network acknowledges it to the modem with either a success or error code. The modem can notify the host (via [SMS Send Status](#); page 238) as each message is acknowledged by the network. This will report if any error occurs in the transmission process.

When a message has been acknowledged by the network (successfully or with errors), it is automatically deleted from the modem's outbox. The host should track the status of outgoing messages using its message ID. Only after a message is successfully sent should it be deleted from the host.

If an SMS message reports an error in the notification, the host can re-write the message from its own storage, back to the outbox of the modem for re-transmission.

Once the transmission operation has begun, the host should not allow any further writes to the outbox until acknowledgements are received (via [SMS Send Status](#) notification; page 238) for all messages previously in the outbox. The host should hold any re-transmissions until all messages in the outbox have been processed and acknowledged. If you do not, messages arriving in the outbox during a transmission operation may or may not be included in the batch transfer depending on timing that cannot be verified. There is no way for the host to know whether or not the newly written message has been sent and is awaiting an acknowledgement.

Segmentation

CDMA SMS messages can have a maximum body size that is greater than the CnS packet size limit. Therefore, SMS messages are segmented when being transmitted between the host and the modem.

CnS employs a special “SMS packet structure” to read received SMS messages from the modem or to write outgoing SMS messages to the modem. These SMS packets form the object parameter within regular CnS packets using the [SMS Write](#) and [SMS Read \(r1\)](#) or [SMS Read \(r2\)](#) commands.

Terminology

To maintain clarity, the following terminology must be precisely understood:

SMS message The entire CDMA SMS message composed of a [header](#) and body.

SMS header The header portion of a CDMA SMS message. For more information, see [SMS header](#) on page 228.

SMS body The text body portion of a CDMA SMS message

SMS packet The Parameter portion of a CnS message that carries some or all of an SMS message.

SMS packet header The first six bytes of an SMS packet. It contains information about the packet type, segment management, and length of the SMS packet payload. For more information, see page 227.

SMS packet payload The portion of an SMS packet following the [SMS packet header](#). This can be either the [SMS header](#), the SMS body or a segment of it, or—for very short SMS messages—both the header and body.

SMS packet types

To distinguish the type of SMS packet payload being carried in a **CnS** message, there is a packet type element in the **SMS packet header**:

- 0 = Intermediate packet. This contains a segment of the SMS body. At least one more SMS body segment will follow.
- 1 = First packet. This indicates that the SMS packet payload of this CnS message contains the **SMS header**. The SMS body will follow in subsequent CnS messages.
- 2 = Last packet. The SMS packet payload contains the last of the SMS body segments. This indicates that there are no more segments to the current SMS message.
- 3 = Not segmented. The SMS packet payload contains the entire SMS body. This is equivalent to packet type 2 without the intervening packet type 0.

All SMS messages begin with packet type **1**. The payload carries only the SMS message's header portion. The body of the SMS message will follow in subsequent CnS messages.

Long SMS message bodies (more than 240 bytes [when **SMS Read (r1)** is used] or 235 bytes [when **SMS Read (r2)** is used]) are segmented into intermediate packet types (**0**). As long as you get this type, you will know that more CnS messages will follow with additional segments of the SMS message body.

The last segment of the SMS message body arrives with a packet type of **2** (or **3** if the body was not segmented). This indicates the end of the SMS message body (1–240 bytes [when **SMS Read (r1)** is used] or 1–235 bytes [when **SMS Read (r2)** is used]).

Managing segmentation

The need to allow segmentation requires the host to manage a sequence of Request and Response exchanges with the modem to transfer and assemble the full SMS message. There is a mechanism within the SMS packet header to control the exchange.

The **SMS packet header** includes a message ID, assigned by the host application for outgoing messages and by the network for incoming ones. Along with the ID is a sequence number, starting at zero for the packet with the **SMS header**, then incrementing to 1 for the first body segment and so on through any remaining body segments.

The packet type included in this header indicates if more packets are to be expected or if the transaction sequence is complete.

Read segment acknowledgement

When the host is requesting to retrieve an SMS message from the modem, it passes an acknowledge index number to track the segments.

The first instance of the Get Request to read an SMS message does not use the parameter. Subsequent calls to retrieve the segments of the message pass a

sequence number as an acknowledge index. To fetch the first body segment (following receipt of the [SMS header](#)), the acknowledge index is 0. Subsequent calls to retrieve body segments increment the index.

The parameter is only used to make sure that no parts of the message get lost; it cannot be used to request a retransmission of a certain segment. If an error is detected during message retrieval, the host application has to do the following:

- If using [SMS Read \(r1\)](#): Go through the whole message list (or what's left after deleting several messages) and then retrieve the initially corrupted message again.
- If using [SMS Read \(r2\)](#): Start retrieving the same message from the beginning.

CnS SMS packet structure

The SMS packet header precedes the SMS packet payload in order to maintain the sequential order of these fragmented SMS packets, and identify the payload.

A [CnS](#) message used to send or receive an SMS message looks generally like this:

| Byte offset | Content |
|-------------|---|
| 0–1 | Object ID |
| 2 | Operation Type |
| 3 | Reserved |
| 4–7 | Application ID |
| 8 | Reserved |
| 9 | Length of parameter (0–246) |
| 10–15 | SMS packet header, composed of: |
| 10–11 | Message ID (UINT) |
| 12 | Sequence Number of the segmented message (UINT) |
| 13 | Packet Type (UINT). One of: <ul style="list-style-type: none"> • 0 = Body segment with more to follow • 1 = First segment (message header) • 2 = Last body segment • 3 = Only body segment (not segmented) |

| Byte offset | Content |
|-------------|---|
| 14 | Count of remaining SMS messages in the modem (excluding this one) (UINT) |
| 15 | Length of the payload in this segment (UINT) |
| 16–Len | Payload based on Packet Type |

SMS packet header

All SMS message [packets](#) use the SMS packet [header](#) to identify the contents of the packet's payload, its sequence, and indicate whether or not subsequent segments are forthcoming.

The first field contains the **Message ID**. This is one of two parts to re-assembling a segmented message. The second field provides the sequence number. These two fields together allow you to reassemble a segmented SMS message.

The **Message ID** for outgoing messages is user-defined by the host application. The onus is on the host application to ensure these numbers are unique. This value does not persist in the CDMA network, which assigns an ID to the message when it is received at the base station for delivery.

For incoming messages, the **Message ID** is the value set by the network.

Whether sending or receiving, the **Message ID** is your mechanism to match messages in the modem with messages in the host.

The third field in the SMS packet header—**Packet Type**—identifies what type of payload is in this [CnS](#) message, and whether or not additional message segments are still to come.

The **Count** field indicates the current total count (all three message queues) of additional SMS messages in the modem. This count **does not** include the current message being transferred. If a new message arrives during the reading process, this count is incremented from one packet to the next in a segmented series. The field is only meaningful for packet types 2 and 3 (final packets).

When sending an SMS message, the host sets the Count field to zero; it is ignored by the modem.

The length field of the packet header is the length of the SMS packet payload portion only.

SMS packet payload

All SMS transactions begin a packet type 1, indicating the [SMS header](#). For details of the structure of the data in this header, see "[SMS header](#)" on page 228. The header is 182 bytes long.

If the body is 240 bytes or less (when [SMS Read \(r1\)](#) is used) or 235 bytes or less (when [SMS Read \(r2\)](#) is used), the second packet is type 3—the body is not segmented.

For message bodies longer than 240 bytes (when [SMS Read \(r1\)](#) is used) or longer than 235 bytes (when [SMS Read \(r2\)](#) is used), segmentation is needed, so the second packet type is 0. This indicates that more body packets follow. When the remaining message body fits in a packet, it will be type 3: the last body segment.

Where SMS messages are segmented, the host must manage the assembly (of incoming) or the disassembly (of outgoing) messages.

Receiving SMS messages

This section covers the basic CnS message exchanges to handle incoming SMS messages. The following cases are covered:

- Poll for received SMS messages
- [Notification of incoming message](#)
- [Retrieval of received messages](#) and deletion of them after reading

Poll for received SMS messages

Note: Notification is a more efficient approach to detecting received messages.

The host can, at any time, query the modem for any pending SMS messages. This is typically only done on application start-up. The notification mechanism is the recommended method during normal operation.

The host issues a Get Request (0x01) [SMS Receive Status \(r1\)](#) message (page 232) or [SMS Receive Status \(r2\) and Storage Capacity](#) message (page 241). The modem responds with the number of SMS messages in each of the three queues: urgent, normal text, and voice.

If there are messages in the modem, the host can then read as many as desired.

An alternate to polling for the status of the received SMS messages, the host can use a notification method as described next.

Notification of incoming SMS messages

To have the host receive a notification whenever an SMS message arrives, the host can enable notifications (0x05) of the [SMS Receive Status \(r1\)](#) or [SMS Receive Status \(r2\) and Storage Capacity](#) message object. Whenever an SMS message arrives, the modem issues a notification (0x07) [SMS Receive Status \(r1\)](#) or [SMS Receive Status \(r2\) and Storage Capacity](#) with the SMS message queue counters.

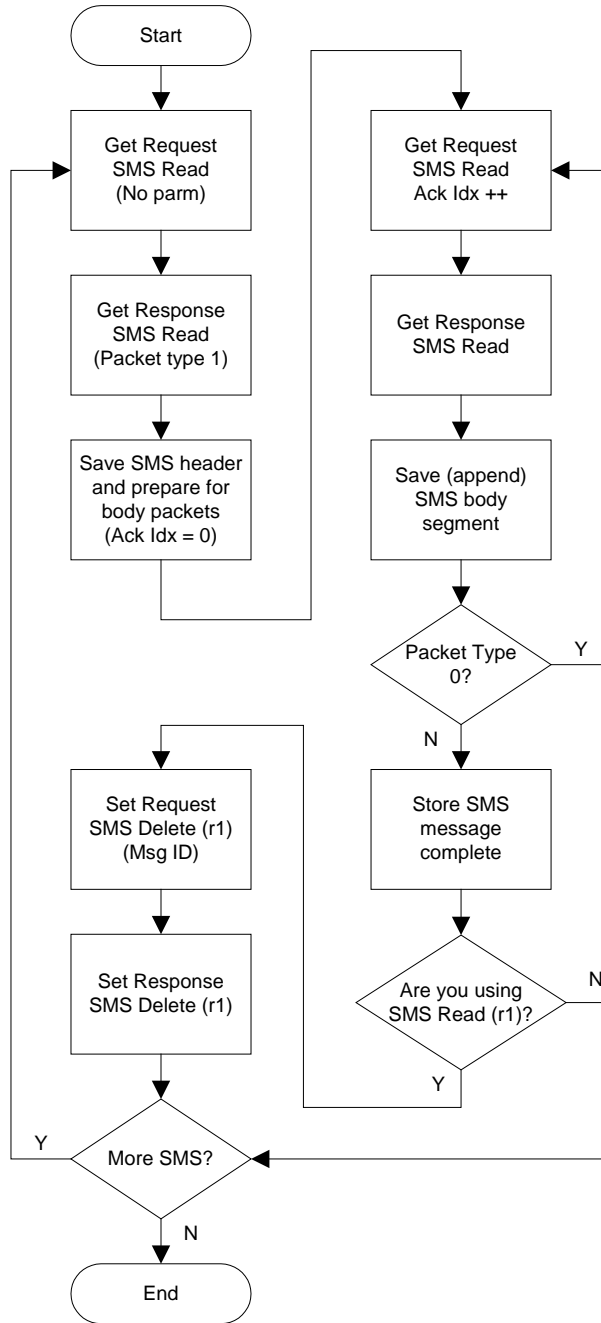
Using notifications allows you to inform the user immediately upon the arrival of an SMS message. This is much less demanding on the processor than using a polling approach.

Reading incoming SMS messages

When the host has determined, either by poll or notification, that there is an SMS message in the modem, it can read any pending messages using the algorithm in [Figure 1](#) on page 301.

SMS messages can be segmented as discussed earlier (“Segmentation”, page 296). To read an SMS message can require a sequence of **SMS Read (r1)** exchanges or **SMS Read (r2)** exchanges.

Figure 1: SMS retrieve and delete flowchart



After verifying (via the **SMS Receive Status (r1)** message [page 232] or **SMS Receive Status (r2) and Storage Capacity** message [page 241]) that there is a message to read, you make the first Get request to **SMS Read (r1)** [without a parameter] or **SMS Read (r2)**. The modem responds with a type 1 packet, the **SMS header**. Store the header and set the acknowledge index to 0.

Read the message body by incrementing the acknowledge index and requesting another read using **SMS Read (r1)** [this time with the parameter] or **SMS Read (r2)**. The modem responds with either a type 0 or type 3 packet. If you receive a type 3 response, the entire SMS message body is in that response.

Where the SMS message has been segmented—a type 0 response to the second Get request—you must continue to issue Get requests, incrementing the acknowledge index each time. The modem responds to each request with the next segment of the SMS message body. This exchange continues until the host receives a response with packet type 2, indicating the last segment.

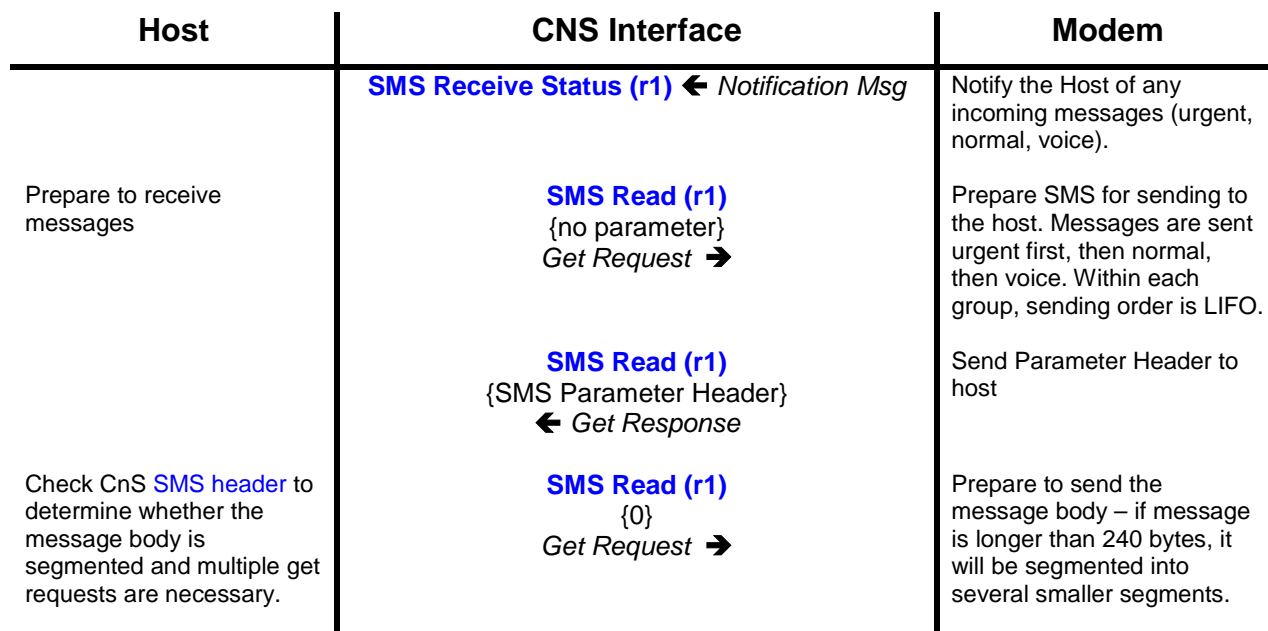
The host must re-assemble the SMS message from the segments and store it. When all segments are received:

- If r1 objects (**SMS Read (r1)**) were used, the host must use **SMS Delete (r1)** (page 235) to remove the message from the modem’s queue.
- If r2 objects (**SMS Read (r2)**) were used, the message is automatically moved to the read list, so there’s no need to immediately use **SMS Delete (r2)** (page 245) to remove the message.

If the last **SMS Read (r1)** or **SMS Read (r2)** message reported more SMS messages in the modem (the **Count** field of the SMS packet header), you should continue to read SMS messages from the modem by restarting the process.

If you’re using r1 objects, see Figure 2 below. If you’re using r2 objects, see Figure 3 (page 303) and Figure 4 (page 304).

Figure 2: SMS retrieve and delete: CnS message flow (for r1 objects)



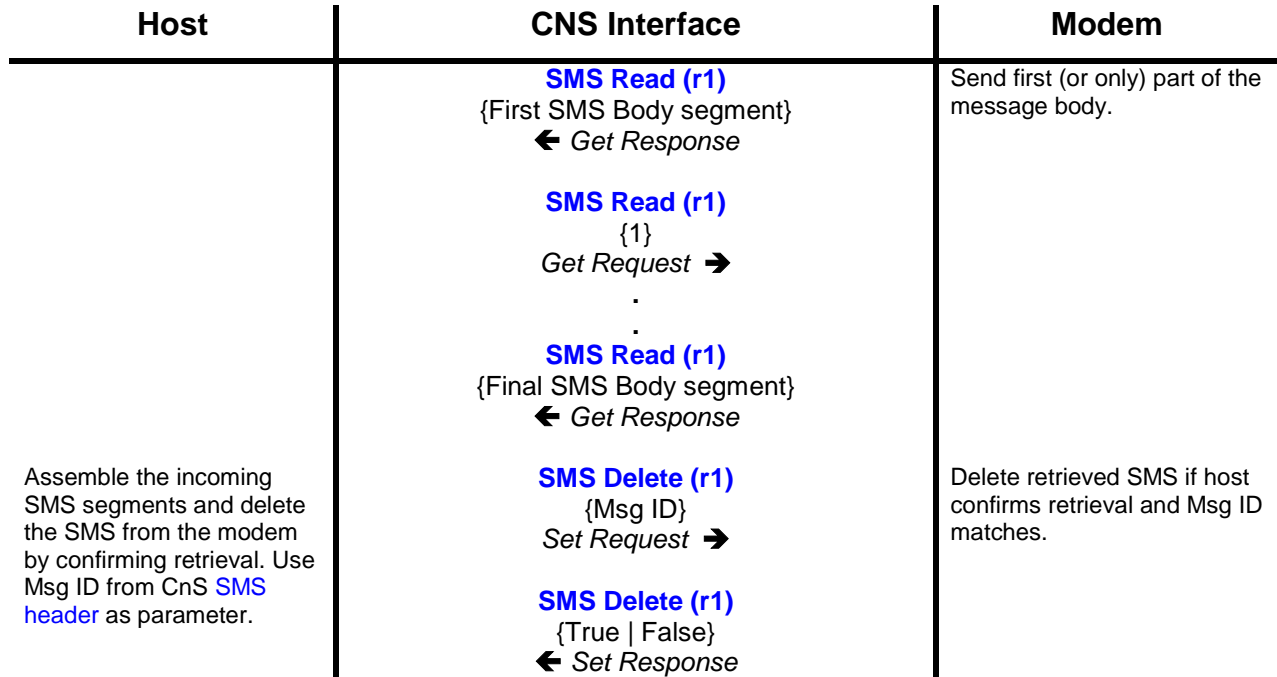
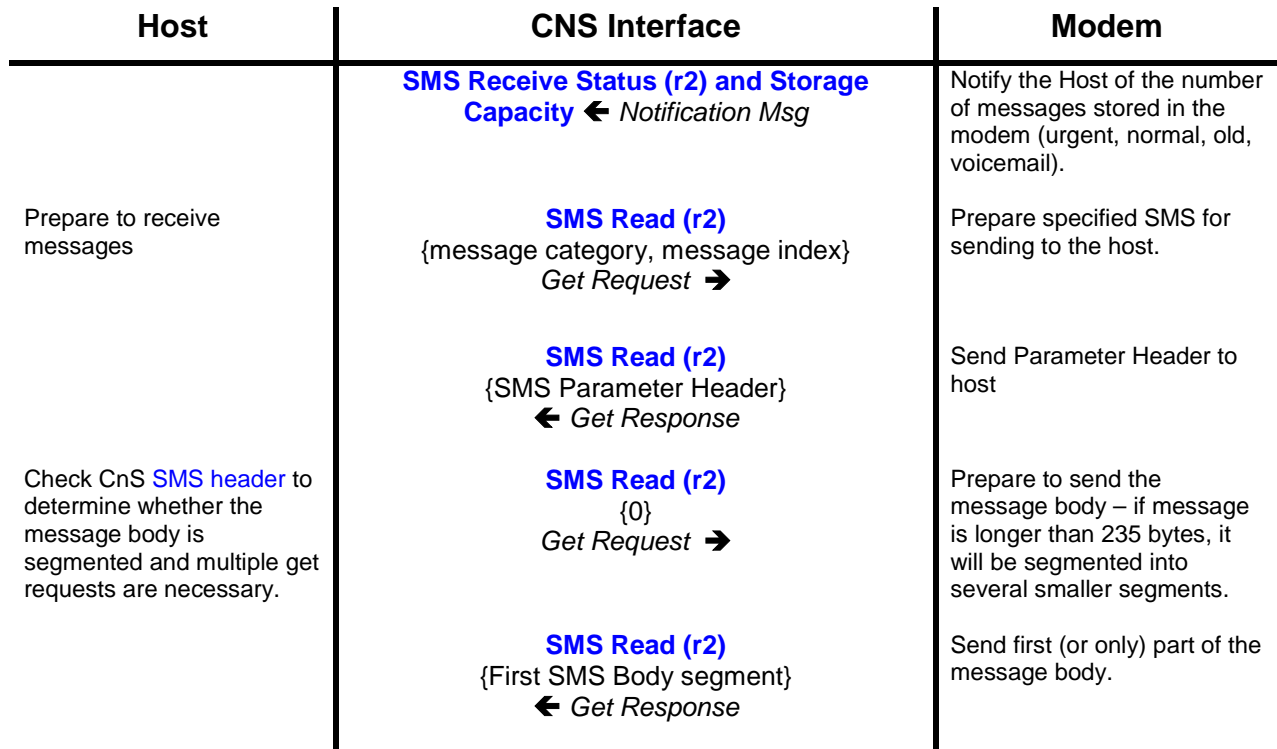


Figure 3: SMS retrieve: CnS message flow (for r2 objects)



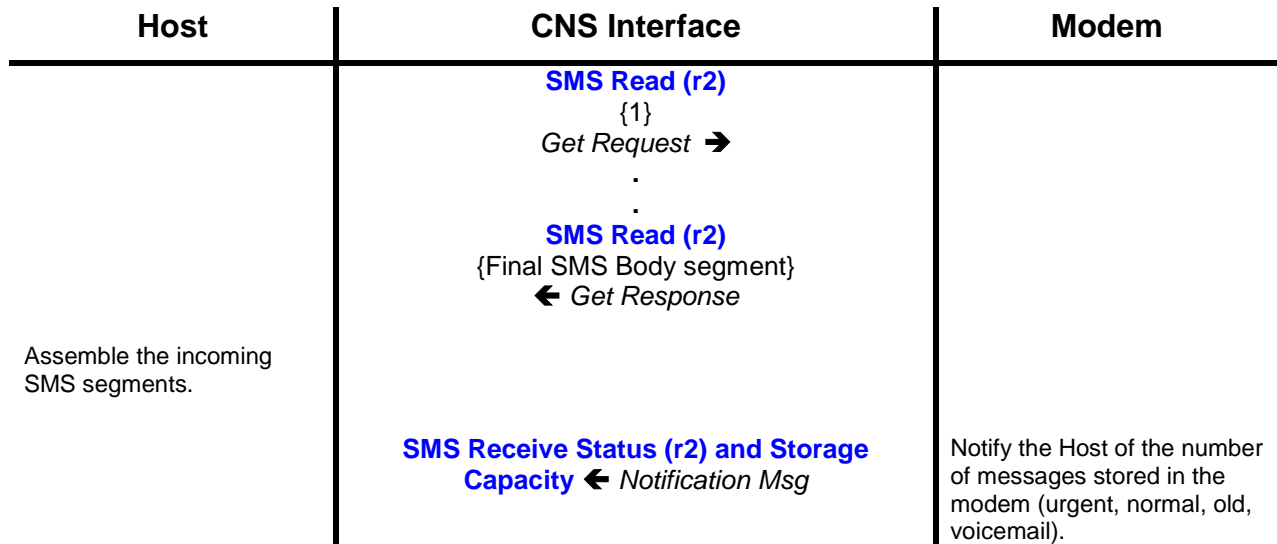
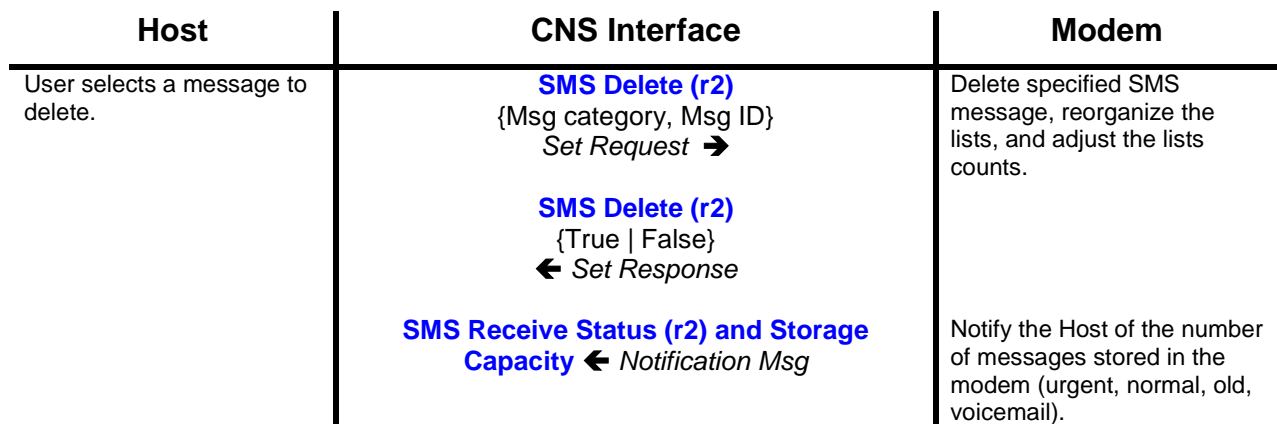


Figure 4: SMS delete: CnS message flow (for r2 objects)



Sample SMS read and delete sequence (for r1 objects)

Note: If you're using r2 objects, see [Sample SMS read sequence \(for r2 objects\)](#) on page 307 and [Sample SMS delete sequence \(for r2 objects\)](#) on page 310.

A sample exchange (using r1 SMS objects) would look like this:

- Host to modem - **SMS Read (r1)** Get Request

20 01 01 00 0A 0B 0C 0D 00 00

20 01 = **SMS Read (r1)**

01 = Get Request (from host)

00 Reserved
 0A 0B 0C 0D = Application parameter
 00 Reserved
 00 = Length

2. Modem to host - **SMS Read (r1)** Get Response (shaded section is the **SMS packet header**)

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|-----|----|
| 20 | 01 | 02 | 00 | 0A | 0B | 0C | 0D | 00 | BC |
| 1A | 2B | 00 | 01 | 02 | B6 | hh | hh | ... | hh |

02 = Get Response (from modem)
 BC = Length (188 = 6 hdr + 182 payload)
 1A 2B = Message ID
 00 = Sequence Number
 01 = Packet Type (first)
 02 = Count (remaining SMS messages)
 B6 = Length of payload (182 bytes)
 hh...hh = Payload—SMS message header

3. Host to modem - **SMS Read (r1)** Get Request

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|
| 20 | 01 | 01 | 00 | 0A | 0B | 0C | 0D | 00 | 02 |
| 00 | 01 | | | | | | | | |

02 = Length
 00 01 = Acknowledge index

4. Modem to host - **SMS Read (r1)** Get Response

| | | | | | | | | | |
|----|----|----|----|----|----|----|----|-----|----|
| 20 | 01 | 02 | 00 | 0A | 0B | 0C | 0D | 00 | F6 |
| 1A | 2B | 01 | 00 | 02 | F0 | bb | bb | ... | bb |

02 = Get Response (from modem)

- F6** = Length (246 CnS max)
- 1A 2B** = Message ID
- 01** = Sequence Number
- 00** = Packet Type (intermediate)
- 02** = Count (remaining SMS messages)
- F0** = Length of payload (240 bytes)
- bb...bb** = Payload—SMS message body segment

5. Host to modem - **SMS Read (r1)** Get Request

20 01 01 00 0A 0B 0C 0D 00 02
00 02

- 02** = Length
- 00 02** = Acknowledge index (incremented)

6. Modem to host - **SMS Read (r1)** Get Response

20 01 02 00 0A 0B 0C 0D 00 21
1A 2B 02 02 03 1B bb bb ... bb

- 02** = Get Response (from modem)
- 21** = Length (33)
- 1A 2B** = Message ID
- 02** = Sequence Number
- 02** = Packet Type (last)
- 03** = Count (new message arrived)
- 1B** = Length of payload (27 bytes)
- bb...bb** = Payload—SMS message body segment (last 11 bytes)

7. Following successful re-assembly and storage at the host, the host sends to the modem:

20 02 03 00 0A 0B 0C 0D 00 02
 1A 2B

20 02 = SMS Delete (r1)
 03 = Set Request (from host)
 00 Reserved
 0A 0B 0C 0D = Application parameter
 00 Reserved
 02 = Length
 1A 2B = Message ID to delete

8. Modem to host - **SMS Read (r1)** Set Response

20 02 04 00 0A 0B 0C 0D 00 02
 00 01

04 = Set Response (from modem)
 02 = Length
 00 01 = Deletion acknowledge = True

9. While the Count returned in step 6 (the last segment of the message) is greater than zero, there are more SMS messages in the modem; you should return to step 1.

Sample SMS read sequence (for r2 objects)

Note: If you're using r1 objects, see [Sample SMS read and delete sequence \(for r1 objects\)](#) on page 304.

A sample exchange (using r2 SMS objects) would look like this:

1. Host to modem - **SMS Read (r2)** Get Request (first)

20 09 01 00 0A 0B 0C 0D 00 06
 00 01 00 02 00 01

20 09 = SMS Read (r2)

- 01** = Get Request (from host)
- 00 Reserved
- 0A 0B 0C 0D = Application parameter
- 00 06 = Length
- 00 01 Object version
- 00 02 = New normal message category
- 00 01 = Message index

2. Modem to host - **SMS Read (r2)** Get Response (first) (shaded section is the **SMS packet header**)

```

20 09 02 00 0A 0B 0C 0D 00 C0
00 01 00 00 1A 2B 00 01 02 B6
hh hh ... hh
    
```

- 02** = Get Response (from modem)
- 00 C0 = Length
- 00 01 Object version
- 00 00 Operation Result - success
- 1A 2B** = Message ID
- 00** = Sequence Number
- 01 = Packet Type (first)
- 02** = Count (remaining SMS messages)
- B6** = Length of payload (182 bytes)
- hh...hh = Payload—SMS message header

3. Host to modem - **SMS Read (r2)** Get Request

```

20 09 01 00 0A 0B 0C 0D 00 04
00 01 00 00
    
```

01 = Get Request (from host)
00 04 = Length
00 01 = Object version
00 01 = Acknowledge index

4. Modem to host - **SMS Read (r2)** Get Response

| | | | | | | | | | |
|----|----|-----------|----|-----------|-----------|-----------|----|-----------|-----------|
| 20 | 09 | 02 | 00 | 0A | 0B | 0C | 0D | 00 | F5 |
| 00 | 01 | 00 | 00 | 1A | 2B | 01 | 00 | 02 | EB |
| bb | bb | ... | bb | | | | | | |

02 = Get Response (from modem)
00 F5 = Length (245 **CnS** max)
00 01 = Object version
00 00 = Operation Result - success
1A 2B = Message ID
01 = Sequence Number
00 = Packet Type (intermediate)
02 = Count (remaining SMS messages)
EB = Length of payload (235 bytes)
bb...bb = Payload—SMS message body segment

5. Host to modem - **SMS Read (r2)** Get Request

| | | | | | | | | | |
|----|----|-----------|-----------|----|----|----|----|----|----|
| 20 | 09 | 01 | 00 | 0A | 0B | 0C | 0D | 00 | 04 |
| 00 | 01 | 00 | 01 | | | | | | |

01 = Get Request (from host)
00 04 = Length
00 01 = Object version
00 01 = Acknowledge index (incremented)

6. Modem to host - **SMS Read (r2)** Get Response

```

20 09 02 00 0A 0B 0C 0D 00 F5
00 01 00 00 1A 2B 02 02 03 1B
bb bb ... bb
    
```

- 02** = Get Response (from modem)
- 00 F5** = Length (245 CnS max)
- 00 01** Object version
- 00 00** Operation Result - success
- 1A 2B** = Message ID
- 02** = Sequence Number
- 02** = Packet Type (last)
- 03** = Count (new message arrived)
- 1B** = Length of payload (27 bytes)
- bb...bb** = Payload—SMS message body segment (last 11 bytes)

Sample SMS delete sequence (for r2 objects)

Note: If you're using r1 objects, see [Sample SMS read and delete sequence \(for r1 objects\)](#) on page 304.

1. Host to modem - **SMS Delete (r2)** Set Request

```

20 0A 03 00 0A 0B 0C 0D 00 06
00 01 00 03 00 01
    
```

- 20 0A** = **SMS Delete (r2)**
- 03** = Set Request (from host)
- 00** Reserved
- 0A 0B 0C 0D** = Application parameter

| | |
|-------|------------------------|
| 00 06 | = Length |
| 00 01 | Object version |
| 00 03 | Old message category |
| 00 01 | = Message ID to delete |

2. Modem to host - **SMS Delete (r2)** Set Response

20 0A 04 00 0A 0B 0C 0D 00 04
00 01 00 00

| | |
|-------|-----------------------------|
| 04 | = Set Response (from modem) |
| 00 04 | = Length |
| 00 01 | Object version |
| 00 00 | = Operation result: success |

Sending SMS messages

This section covers the basic **CnS** message exchanges to handle outgoing SMS messages. The following cases are covered:

- Writing an outgoing message to the outbox
- Sending the SMS outbox
- Confirming transmission of outgoing messages

Writing outgoing SMS messages

The first step in sending an SMS message is to compose and store the message on the host. When the message has been stored at the host, it can be written to the modem's outbox.

Note: *There is no way to query the modem for the number of messages currently in the outbox.*

The outbox can hold up to ten messages. An attempt to write an eleventh message results in the error return code.

Allowing for segmented SMS messages means that the process for writing a message to the modem's outbox works very much like the process used to read messages. For a description, see "[Reading incoming SMS messages](#)" on page 300.

You make the first Set request to **SMS Write** with the **SMS packet header** indicating the header packet. The modem acknowledges without a parameter.

The sequence numbering is initiated by the host, but each Set Response from the modem does not include parameters.

When the modem receives either a type **2** (last) or type **3** (not segmented) packet, it re-assembles the SMS message and stores it in the outbox.

The host must manage its own count of how many messages have been placed in the outbox.

When you have no more messages to place in the outbox, or the outbox is full, you must use **SMS Send** (page 237) to have the modem transmit the messages to the network.

Sending SMS outbox

After writing one or more messages (up to ten) to the modem's outbox, you use **SMS Send** to transmit them to the network.

The modem acknowledges the Set Request of **SMS Send** with a Set Response. At that point, the modem makes the **traffic channel** connection and begins sending the messages.

As each message is sent, it is deleted from the outbox. For each message sent, a notification message (if enabled) is sent to the host. This notification indicates the success or failure of the transmission, as described in the next section.

Confirming SMS transmission

As outgoing SMS messages are sent to the network, the modem can notify the host using the **SMS Send Status CnS** message. The notification must be enabled by the host.

The notification includes the **Message ID** as assigned by the user when written to the outbox, and two status values: **Error Class** and **Cause Code**.

If the message was successfully sent, the **Error Class** is zero and the **Cause Code** is 0x8000.

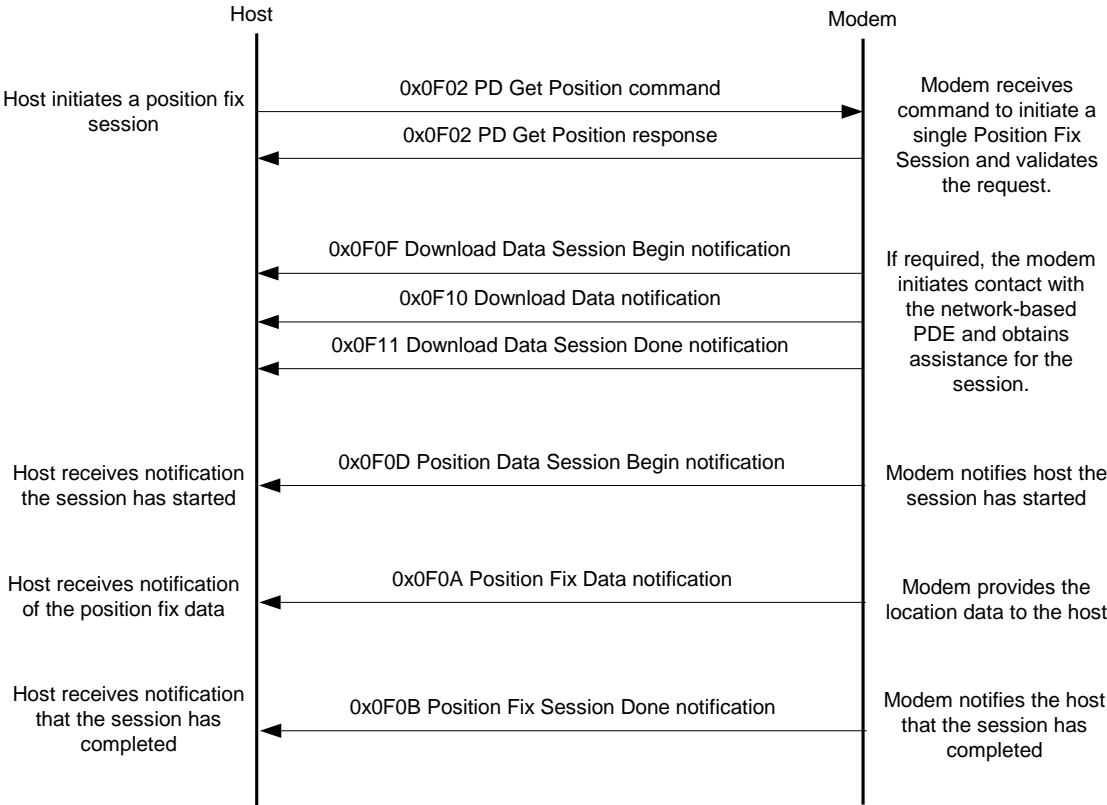
In the event of an error, the **Error Class** indicates if it is temporary (2) or permanent (3). In all cases, the message has been lost, and has to be re-sent from the host by writing it to the outbox again (**SMS Write**) and re-issuing it (**SMS Send**; page 237).

The **Cause Code** conveys the nature of the error that occurred. If the error was at the CDMA network, the IS-41 error code is given. If the error originated in the modem, the value is in the range 0x8001–0x80FF. The interpretation of these values is described in the reference for **SMS Send Status** on page 238.

18: Appendix B: LBS Message Flow

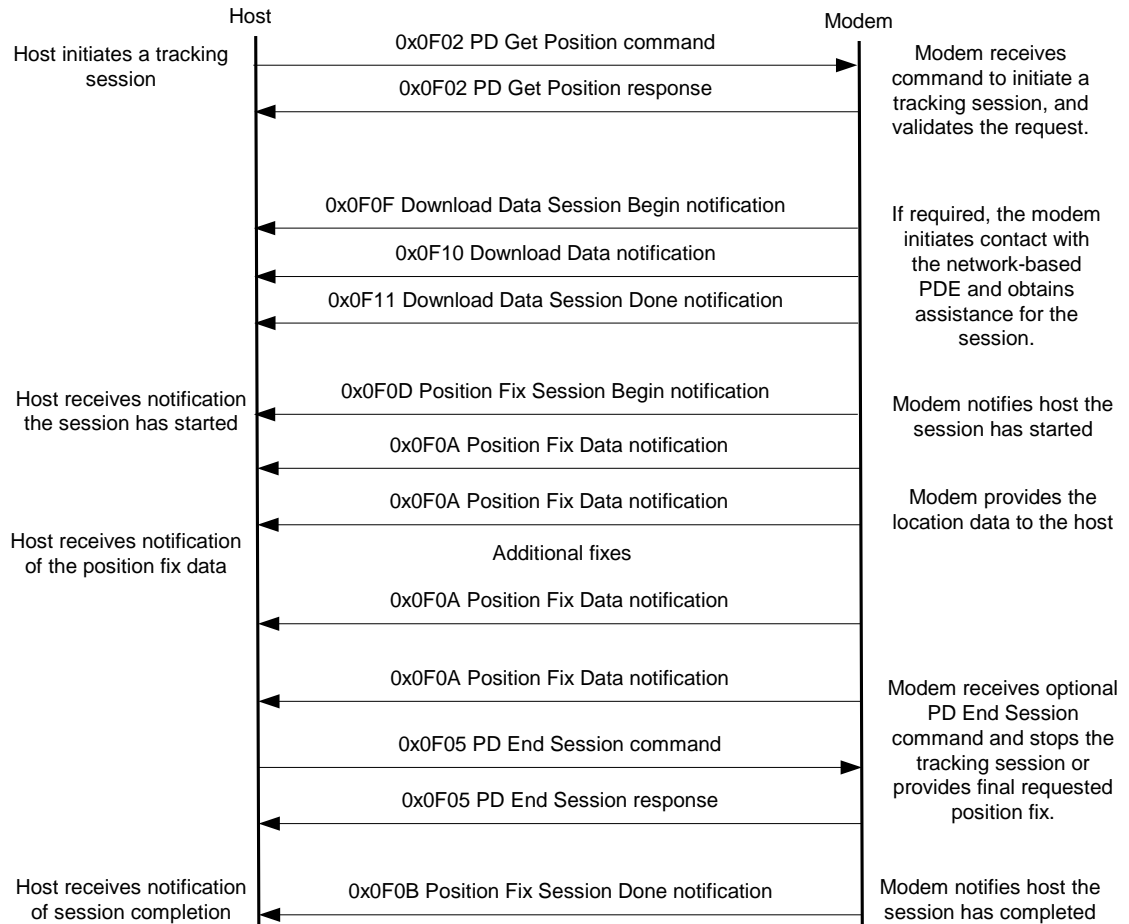
Mobile Originated Single Position Fix Session

Figure 5: LBS single fix session



Mobile Originated Position Tracking Session

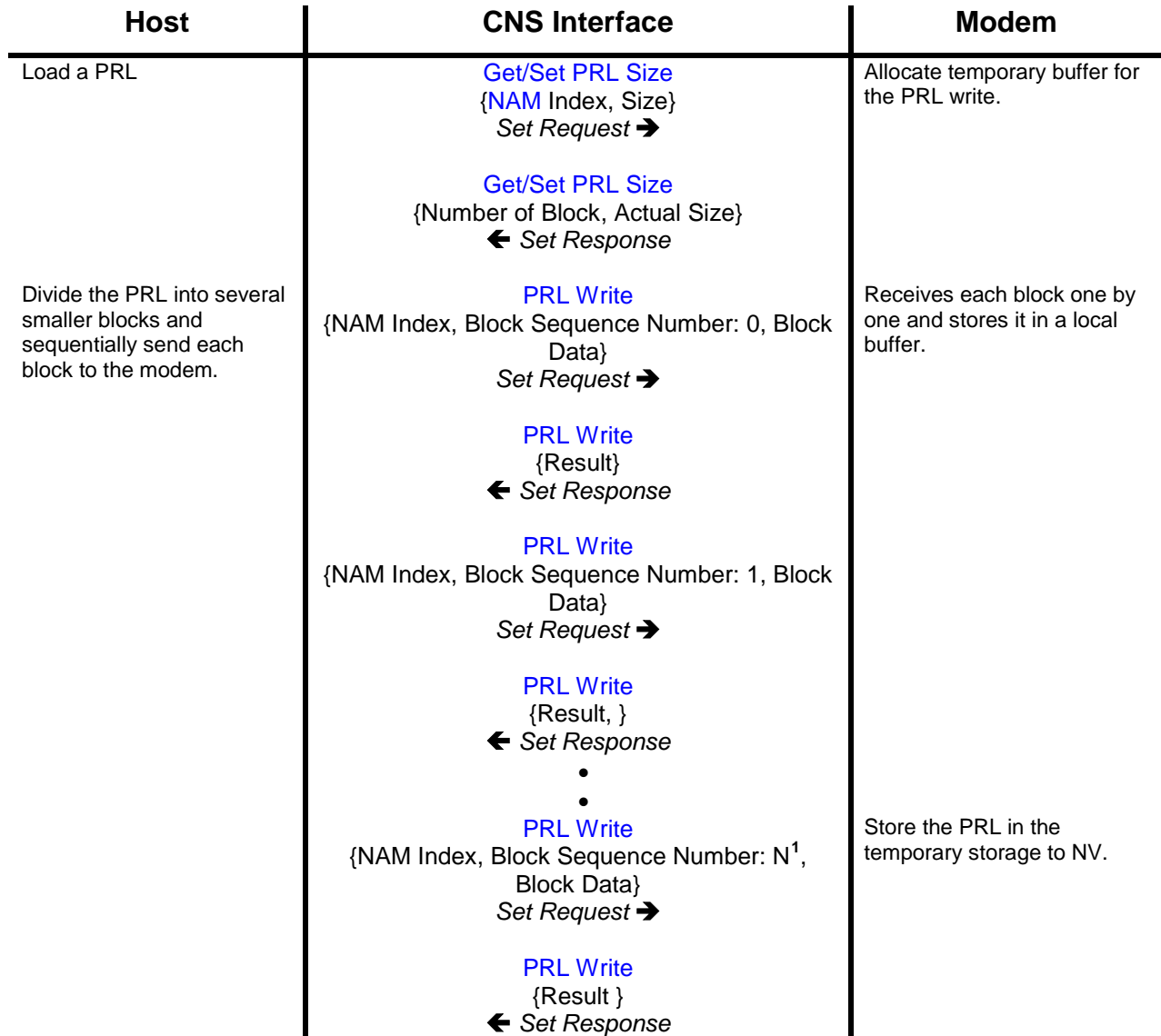
Figure 6: LBS tracking session



19: Appendix C: PRL Message Flow

Writing a PRL

Figure 7: Writing a PRL



¹ Total number of blocks

20: Appendix D: Objects That Have Been Replaced, and May Be Removed From Future Releases

The following table lists objects that have been replaced with other objects, and may no longer be supported in future firmware releases for new Sierra Wireless products (that is, new wireless modems not covered by this document).

Table 35: Objects that have been replaced, and may be removed from future releases

| Object ID | Object Name | Replaced in: | Replaced with (Object ID, name) |
|-----------|---------------------------------------|--|---|
| 0x0005 | Object Version | CWE CnS object set Release 2 | 0x0006—Object Version Expanded (see the next row in this table) |
| 0x0006 | Object Version Expanded | CWE CnS object set Release 3 | 0x000C— Supported Object Releases (page 58) |
| 0x100F | Reset Modem (page 92) | | 0x0010— Reset Device (page 63) |

21: Appendix E: ASCII Table

| Char | Dec | Hex | Char | Dec | Hex | Char | Dec | Hex | Char | Dec | Hex |
|-------------|-----|-----|--------------|-----|-----|----------|-----|-----|------------|-----|-----|
| NUL | 0 | 00 | SP | 32 | 20 | @ | 64 | 40 | ' | 96 | 60 |
| SOH | 1 | 01 | ! | 33 | 21 | A | 65 | 41 | a | 97 | 61 |
| STX | 2 | 02 | " | 34 | 22 | B | 66 | 42 | b | 98 | 62 |
| ETX | 3 | 03 | # | 35 | 23 | C | 67 | 43 | c | 99 | 63 |
| EOT | 4 | 04 | \$ | 36 | 24 | D | 68 | 44 | d | 100 | 64 |
| ENQ | 5 | 05 | % | 37 | 25 | E | 69 | 45 | e | 101 | 65 |
| ACK | 6 | 06 | & | 38 | 26 | F | 70 | 46 | f | 102 | 66 |
| BEL | 7 | 07 | ' | 39 | 27 | G | 71 | 47 | g | 103 | 67 |
| BS | 8 | 08 | (| 40 | 28 | H | 72 | 48 | h | 104 | 68 |
| HT | 9 | 09 |) | 41 | 29 | I | 73 | 49 | i | 105 | 69 |
| LF | 10 | 0A | * | 42 | 2A | J | 74 | 4A | j | 106 | 6A |
| VT | 11 | 0B | + | 43 | 2B | K | 75 | 4B | k | 107 | 6B |
| FF | 12 | 0C | , | 44 | 2C | L | 76 | 4C | l | 108 | 6C |
| CR | 13 | 0D | - | 45 | 2D | M | 77 | 4D | m | 109 | 6D |
| SO | 14 | 0E | . | 46 | 2E | N | 78 | 4E | n | 110 | 6E |
| SI | 15 | 0F | / | 47 | 2F | O | 79 | 4F | o | 111 | 6F |
| DLE | 16 | 10 | 0 | 48 | 30 | P | 80 | 50 | p | 112 | 70 |
| XON | 17 | 11 | 1 | 49 | 31 | Q | 81 | 51 | q | 113 | 71 |
| DC2 | 18 | 12 | 2 | 50 | 32 | R | 82 | 52 | r | 114 | 72 |
| XOFF | 19 | 13 | 3 | 51 | 33 | S | 83 | 53 | s | 115 | 73 |
| DC4 | 20 | 14 | 4 | 52 | 34 | T | 84 | 54 | t | 116 | 74 |
| NAK | 21 | 15 | 5 | 53 | 35 | U | 85 | 55 | u | 117 | 75 |
| SYN | 22 | 16 | 6 | 54 | 36 | V | 86 | 56 | v | 118 | 76 |
| ETB | 23 | 17 | 7 | 55 | 37 | W | 87 | 57 | w | 119 | 77 |
| CAN | 24 | 18 | 8 | 56 | 38 | X | 88 | 58 | x | 120 | 78 |
| EM | 25 | 19 | 9 | 57 | 39 | Y | 89 | 59 | y | 121 | 79 |
| SUB | 26 | 1A | : | 58 | 3A | Z | 90 | 5A | z | 122 | 7A |
| ESC | 27 | 1B | ; | 59 | 3B | [| 91 | 5B | { | 123 | 7B |
| FS | 28 | 1C | < | 60 | 3C | \ | 92 | 5C | | 124 | 7C |
| GS | 29 | 1D | = | 61 | 3D |] | 93 | 5D | } | 125 | 7D |
| RS | 30 | 1E | > | 62 | 3E | ^ | 94 | 5E | ~ | 126 | 7E |
| US | 31 | 1F | ? | 63 | 3F | _ | 95 | 5F | DEL | 127 | 7F |

22: Appendix F: Glossary

Table 36: Acronyms

| Acronym/term | Description |
|--------------|---|
| 1X | One Times Radio Transmission Technology (the “one times” refers to the frequency spectrum). Also known as 1xRTT. Supports Internet connections with data rates up to 153 Kbps. Actual speed depends on the network conditions. Compare to 1xEV-DO . |
| 1xEV-DO | <p>A high-speed standard for cellular packet data communications.</p> <p>Rev. A supports Internet connections with data rates up to 3.1 Mbps (downlink from the network) and 1.8 Mbps (uplink to the network).</p> <p>Rev. 0 supports Internet connections with data rates up to 2.4 Mbps (downlink from the network) and 153 Kbps (uplink to the network).</p> <p>Average data rates are roughly: for Rev. A: 450-800 Kbps (downlink from the network) and 300-400 Kbps (uplink to the network); for Rev. 0: 400-700 Kbps (downlink from the network) and 40-80 Kbps (uplink to the network). Actual speed depends on the network conditions. Compare to 1X.</p> |
| AMPS | Advanced Mobile Phone Service—original name given to the analog voice cellular telephone system. The standard for cellular telephone service in North and South America, plus some Pacific Rim countries. |
| AN | Access network |
| BS | <p>Base station.</p> <ul style="list-style-type: none"> • A station in the Domestic Public Cellular Radio Telecommunications Service that provides the means for mobile stations (for example, cell phones, radio modems) to access network services using radio. • A cell, a sector within a cell, a Mobile Switching Centre (MSC), an Interworking Function (IWF), or another part of the cellular system located at a cellular radio tower. |
| CC | Control Channel |
| CDMA | Code Division Multiple Access—A wideband spread spectrum technique used in digital cellular, personal communications services, and other wireless networks. Wide channels (1.25 MHz) are obtained through spread spectrum transmissions, thus allowing many active users to share the same channel. Each user is assigned a unique digital code, which differentiates the individual conversations on the same channel. |
| CI | Client-Initiated |

| Acronym/term | Description |
|--------------|---|
| CIDC | Client-Initiated Device Configuration |
| CIFUMO | Client-Initiated Firmware Update Management Object |
| CIPRL | Client-Initiated PRL Update |
| CnS | Control and Status (language)—a proprietary protocol for managing the control and status of the modem. |
| CWE | Core Wireless Engine |
| dBm | A logarithmic (base 10) measure of relative power (dB for decibels); relative to milliwatts (m). A dBm value will be 30 units (1000 times) larger (less negative) than a dBW value, because of the difference in scale (milliwatts vs. watts). |
| dBW | A logarithmic (base 10) measure of power (dB for decibels); relative to one watt. |
| DDTM | Data Dedicated Transmission Mode |
| DD | Download Descriptor |
| DM | Device Management |
| dormant | <p>The network switches a 3G (1X or 1xEV-DO) high-speed data connection into a dormant mode if there is no traffic on the connection for some time. This allows the modem, if it supports voice, to make and receive voice calls while the data connection is idle.</p> <p>When you resume data traffic, the high-speed data connection becomes active.</p> |
| DRC | Data Rate Control |
| DTMF | Dual Tone Multi-Frequency—the tone generation system used by standard telephone networks. Each key is allocated a row and column tone; both are played at once to create the DTMF signal. |
| Ec/Io | A dimensionless ratio of the average power of a channel, typically the pilot channel, to the total signal power. |
| EIA | Electronic Industries Alliance – a standards body. See www.eia.org . |

| Acronym/term | Description |
|-----------------------------------|--|
| E911 | Enhanced 911—The capability to automatically identify, to emergency dispatchers, the location from which a wireless call is being made. Phase I requires carriers to provide the telephone number of a mobile making a 911 call and the location of the cell site or base station handling the call. Phase II requires carriers to deliver more specific latitude and longitude location information, known as Automatic Location Identification (ALI). This typically involves GPS receivers at the mobile. |
| ERI | Enhanced Roaming Indicator |
| ESN | Electronic Serial Number—the unique serial number assigned to the modem for cellular network use. |
| FER | Frame Error Rate |
| firmware | Software stored in ROM or EEPROM; essential programs that remain even when the system is turned off. Firmware is easier to change than hardware, but more permanent than software stored on disk. |
| Forward Supplemental Code Channel | A Supplemental Code Channel that operates in the forward direction; from base station to mobile. |
| Forward Traffic Channel | One Forward Fundamental Code Channel and optionally up to seven Forward Supplemental Code Channels used to transport data blocks from the base station to the mobile station. |
| Forward Fundamental Code Channel | A Fundamental Code Channel that operates in the forward direction; from base station to mobile. |
| FOTA | Firmware Update Over The Air—a feature included in OMA Device Management (OMA-DM). |
| frame | A block of data formed from binary codes to be sent with a header and error detection information over a communications channel. |
| Fundamental Code Channel | A portion of a Forward Traffic Channel or Reverse Traffic Channel that is always present. It carries primary and secondary traffic, signaling, and power control information. |
| FUMO | Firmware Update Management Object |
| GPS | Global Positioning System—a system that uses a series of 24 geosynchronous satellites to provide navigational data. |
| HDR | High Data Rate—a QUALCOMM wireless technology that provides high-speed, high-capacity IP packet data services. Also known as 1xEV-DO . |

| Acronym/term | Description |
|--------------|---|
| header | Control information that ensures the datagram is sent to the correct address. |
| HFA | Hands Free Activation |
| HIP | Host Interface Protocol—a proprietary protocol for managing the control and status of the modem. Intended to carry a variety of other protocol packets across a single link layer. CnS is one of the types of packets that can be carried inside HIP. |
| ICMP | Internet Control Message Protocol—a protocol included in the network layer for CDMA analog fax services. |
| IMSI | International Mobile Station Identity |
| IOTA | Internet Over The Air—an automated feature, supported by some service providers, to perform account setup for you by making a connection to the CDMA network and using a secure Internet connection to download account parameters to your modem. |
| LBS | Location-Based Services |
| MCC | Mobile Country Code—a three digit number uniquely identifying a given country |
| MDN | Mobile Directory Number |
| MIN | Mobile Identification Number—a number that identifies a specific mobile unit within a wireless carrier's network. |
| MI | Mobile-Initiated. Compare to MT . |
| MIP | Mobile IP |
| MNC | Mobile Network Code—a two or three digit number used to uniquely identify a network within a specified country |
| MO | Mobile-Originated—(SMS) sent from the wireless device. Compare to MT . |
| MS | Mobile Station—a wireless terminal, such as a cell phone, pager, or radio modem, that is considered a station in the Domestic Public Cellular Radio Telecommunications Service. |
| MSL | Master Subsidy Lock—a password that allows a user to configure parameters that are normally not accessible. See also OTSL . |
| MT | Mobile-Terminated—(for example, SMS) received by the wireless device. Compare to MO . |

| Acronym/term | Description |
|--------------|--|
| MU | Multi-user |
| NAM | Number Assignment Module—a CDMA account definition that includes a phone number and other unique unit and network identifiers. |
| NDIS | Network Driver Interface Specification—a programming interface specification for connecting network interface cards in Windows. |
| NDSS | Network Directed System Selection—a feature that allows the Mobile Station to automatically register with a preferred system while roaming. |
| NI | Network-Initiated |
| NIA | Network-Initiated Alert |
| NID | Network Identification—a number that uniquely identifies a network. |
| NIDC | Network-Initiated Device Configuration |
| NIFUMO | Network-Initiated Firmware Update Management Object |
| NIPRL | Network-Initiated PRL Update |
| notification | A mechanism to send unsolicited data from either side (host; modem) of the interface; used when no reply or return data is required from the receiver (or conversations do not require stop-and-wait flow control) |
| NV | Non-Volatile (memory)—Random Access Memory that remembers its contents even if the power is removed. |
| NV-RAM | Non-Volatile RAM—Random Access Memory that remembers its contents even if the power is removed. |
| OEM | Original Equipment Manufacturer—a company that manufactures a product and sells it to a reseller. |
| offline | The device is disconnected from any remote terminal or without an active network session. |
| OMA-DM | Open Mobile Alliance - Device Management. A device management (DM) protocol specified by the Open Mobile Alliance (OMA) Device Management Working Group and the Data Synchronization (DS) Working Group. |
| OTAPA | Over-The-Air Parameter Administration Service—a feature that allows you to make updates to previously provisioned phones, without subscriber interaction or a voice connection. |

| Acronym/term | Description |
|-------------------------|---|
| OTASP | Over-the-Air Service Provisioning |
| OTSL | One Time Subsidy Lock—a password that allows a user to configure parameters that are normally not accessible. The OTSL is in effect for only one session. See also MSL . |
| packet | A short fixed-length block of data including a header that is transmitted as a unit in a communications network. |
| PCS | Personal Communications Services—a cellular communication infrastructure that uses a different frequency range than AMPS . |
| PDE | Position Determination Entity—the device that the mobile communicates with for assistance in acquiring a GPS location fix. |
| port | <ul style="list-style-type: none"> • A standard piece of the TCP address structure. The port serves as an extension of the IP address to permit a single host (one IP address) to provide multiple servers (applications), each defined by its unique port number. The port number is used to direct TCP traffic to the correct application. • The socket on the computer that allows external devices such as a modem or printer to be attached. |
| PPP | Point-to-Point Protocol—an alternative communications protocol used between computers, or between computers and routers on the Internet. PPP is an enhanced SLIP. |
| PRI | Product Release Instructions—a file that contains the settings used to configure wireless products for a particular service provider, customer, or purpose. |
| PRL | Preferred Roaming List—an account configuration item set by the user's service provider. It controls the radio channels/network carrier used by the modem. |
| protocol | A defined procedure for determining the proper way of exchanging information on a network. |
| PSTN | Public-Switched Telephone Network—the public telephone system. |
| RATI | Random Access Terminal Identifier |
| registered (LBS client) | A client that is interested in receiving notification about LBS events. |
| registration | The process of an ESN signing on to an available base station. Network registration must be performed before communication across the network can begin. Registering on the network is distinct from opening a communication session. The registration process involves an exchange of identification, authentication, and encryption keys. |

| Acronym/term | Description |
|-----------------------------------|---|
| requesting (LBS client) | A client that has initiated the current LBS session or event. A requesting client is typically also a registered client. |
| Reverse Fundamental Code Channel | A Fundamental Code Channel that operates in the reverse direction; from mobile to base station. |
| Reverse Supplemental Code Channel | A Supplemental Code Channel that operates in the reverse direction; from mobile to base station. |
| Reverse Traffic Channel | One Reverse Fundamental Code Channel and optionally up to seven Reverse Supplemental Code Channels used to transport data blocks from the mobile station to the base station. |
| RF | Radio Frequency |
| RFC | Request for Comments—the generic name that refers to a series of specifications developed by the Internet Engineering Task Force (IETF). The specifications include surveys, measurements, proposed and accepted TCP/IP protocol standards. |
| RI | Ring Indicator |
| roaming | A cellular subscriber is in an area where service is obtained from a cellular service provider that is not the subscriber's provider. This may be subject to roaming charges. |
| RRI | Reverse Rate Indicator. Used to indicate whether or not the Data Channel is being transmitted on the Reverse Traffic Channel and, if it is being transmitted, its data rate. |
| RSSI | Received Signal Strength Indicator—the signal power level at the antenna of the modem, usually measured in dBm . |
| SCR | Secure Challenge Response |
| SID | System ID—identifies your home network area and is used in conjunction with your phone number to determine if you are "home" or " roaming ". |
| SO | Status Output |
| SN | Serving network |
| Supplemental Code Channel | An optional portion of a Forward or Reverse Traffic Channel . It carries either primary or secondary traffic |
| SV | Satellite(s) in View |

| Acronym/term | Description |
|--------------|---|
| SWOC | SoftWare On Card—a feature that provides automatic installation of software directly from the modem, without the need for an installation CD or access to the Internet. Also known as TRU-Install [®] . |
| TC | traffic channel—a Forward or Reverse Fundamental Code Channel used to send user data and signaling traffic between the base station and the mobile station. It has up to seven optional Supplemental Code Channels . |
| TTY | TeleTYpe—a device that allows people who are deaf, hard of hearing, or speech-impaired to use the telephone to communicate. |
| UATI | Unicast Access Terminal Identifier |
| UTC | Universal Time Coordinated—UTC replaces Greenwich Mean Time as the basis for standard time throughout the world. UTC, which uses atomic measurements rather than the earth's rotation, is the equivalent of mean solar time at the prime meridian (0° longitude). |
| WAP | Wireless Application Protocol—a set of protocols used to efficiently provide Internet information over wireless communication networks to microbrowser-based devices, such as “smart phones.” |

23: Index

For an index by object name (Object ID), see page [342](#).

General index

\$

\$GPGGA, 135
 \$GPGSA, 135
 \$GPGSV, 135
 \$GPGVTG, 135
 \$GPRMC, 135

1

1xEV-DO. See HDR

3

3-way call, 210

A

account management messages, 127
 acquisition state of channel, 144
 acronyms, 21, 318
 Action Code, 99
 activation
 date, 137
 IOTA, 247
 IOTA Feature object, 108
 OTASP
 state, 212
 profile, select, 133
 status, 128
 terminate session, 131
 validation challenge, MSL/OTSL
 step 1, 129

step 2, 130

Activation Status object, 128
 active channel number, 143
 active NAM
 change, 132
 get, 132
 Active NAM object, 132
 additional feature messages, 288
 Agent, Remote Diagnostics, 88
 Alert
 Network-Initiated, 119
 Alert on Delivery, 231
 all keys up, 222
 almanac data
 request a download, 257
 validity period, 263
 almanac data, delete, 286
 answer hold, 210
 answer mode for incoming calls, 166
 Application ID, 33
 application subtypes, 187
 ASCII table, 317
 audio circuit mode, 107
 Audio Mode object, 107
 Audio Profile object, 78
 authentication status, HDR, 176
 Auto Lock object, 291
 auto-answer, disable, 166

Autoconnect object, 106
autoconnect, firmware, 106
automatically lock modem on power-up, 291
azimuth, satellite, 285

B

back up NVRAM, 57
Band Class, 67, 147
band preference, 159
band, radio, current, 67
base station
 location information, 161, 283
 protocol revision reported by, 152
Base Station Information object, 283
Baud Rate object, 106
baud rates supported, 107
beep level in headset, 215
Boot Build Date, 62
boot image, 84
boot loader, 84
boot loader build date, 62
Boot Version, 49
build date, boot loader, 62
burst mode, DTMF, 224
bytes sent and received
 cumulative, 172
 current data call, 170

C

c2i (carrier-to-interference ratio), 149
call
 summary tables
 data call management, 163
 voice call management, 192
 answer, 199
 connection state, 203
 disconnect, 206

dormant, 171
duration
 cumulative, 93
 current call, 197
 incoming, 197
 types, 198
Call Answer Cmd, 199
Call Back number, 230
Call Byte Counter, 170
Call Connected, 201
Call Connecting, 210
Call Connection State, 203
Call Disconnect Cmd, 206
Call Disconnected, 204
Call Dormant, 171
Call Error object, 207
Call Extended Caller ID, 200
Call Flash Cmd, 209
Call Incoming, 197
Call Notification Status, 196
Call Originate Cmd, 200
Call OTASP State, 212
Call Privacy, 211
Call State, 202
Caller ID, 199
Cancel DM Session, 113
car kit, 78, 79, 80
car kit, set audio mode, 107
carrier-to-interference ratio (c2i), 149
Cause Codes (SMS send), 239
CDMA system time, 153
change history, 4
Change Lock PIN, 290
channel
 acquisition state, 144
 HDR, 148

- number, 143
 - Channel Number object, 143
 - Channel State object, 144
 - checkmarks, 34
 - CIIP (Client Initiated Initial Provisioning), 247
 - Clear GPS Assistance Data, 286
 - Client Initiated Initial Provisioning (CIIP), 247
 - clients, LBS, types of, 252
 - CLM (Coarse Location Message), 135
 - Close HDR Session object, 170
 - CnS
 - definition, 18, 319
 - overview, 29
 - list of all objects, 36
 - format, 32
 - notification, 27
 - object releases supported by firmware, 58
 - Coarse Location Message (CLM), 135
 - codec gain
 - microphone, 104
 - speaker, 105
 - color code, 148
 - comfort tone, 221
 - Commit Type, 131
 - communication methodology, 23
 - configuration messages, 103
 - connect automatically - firmware, 106
 - connection
 - error, 207
 - state, 203
 - contact information, 3
 - Continuous DTMF Mode, 224
 - conventions, 21
 - conversations, 29
 - Core Wireless Engine, 23
 - cover, HDR DDC, 177
 - coverage restored, wake host on, 101
 - crosses, 34
 - cumulative bytes sent and received, 172
 - cumulative call duration, 93
 - current band class, 147
 - current CDMA system time, 153
 - current personality data, 186
 - Current Radio Band, 67
 - current temperature, 52
 - CWE (Core Wireless Engine), 23
- ## D
- DAK Required, 231
 - data. *See also* call.
 - answer calls as, 166
 - dedicated mode preference, 169
 - data call management, 163
 - data dedicated transmission mode (DDTM)
 - preference, HDR, 169
 - Data Dedicated Transmission Mode for Hybrid, 169
 - data rate
 - secondary port, 106
 - supported rates, 107
 - data rate control (DRC) cover
 - HDR, 177
 - data rate control (DRC) value
 - HDR, 176
 - data statistics, reset, 68
 - data statistics, retrieve, 69
 - date
 - activation, 137
 - firmware, 49
 - DDTM preference, HDR, 169
 - definitions, 318
 - delete
 - GPS Assistance Data, 286

- SMS (r1), 235
- SMS (r2), 245
- SMS voice message, 240
- device characteristics, 45
- device management. *See* DM (Device Management)
- device time, 81
- Device Time object, 81
- Diagnostics Agent, Remote, 88
- dialpad keys released, 222
- Digital CDMA service, 150
- digital mode, idle, 158
- disable notifications
 - call, 196
 - HDR, 154
 - HDR Tech, 172
 - LBS, 254, 255, 256
- disconnect
 - all calls, 206
 - forced, 98
- disconnected call, 204
- DM (Device Management)
 - summary table, 109
 - Cancel DM Session, 113
 - DM Configuration, 110
 - DM Download — Installation Status, 124
 - DM Download Description, 123
 - DM Download Progress, 120
 - DM Session State, 114
 - Download Alert, 121
 - Download Alert, 121
 - download installation status, 124
 - download package, description, 123
 - download progress, 120
 - Network-Initiated Alert, 119
 - session state, 114
 - session, cancel, 113
 - session, start, 112
 - Start DM Session, 112
- DM Configuration, 110
- DM Download — Installation Status, 124
- DM Download Alert, 121
- DM Download Description, 123
- DM Download Progress, 120
- DM Session State, 114
- documents, reference, 20
- dormant
 - Call Dormant object, 171
 - reasons, 204
 - set modem to go, 189
- Download Alert, DM (Device Management), 121
- download almanac and ephemeris data, 257
- Download Data, 275
- Download Data Session Begin, 275
- Download Data Session Done, 277
- Download Data Session End, 277
- download ephemeris and almanac data, 257
- download installation status, DM (Device Management), 124
- Download Notification Status, 256
- download package, DM, description, 123
- download progress, DM (Device Management), 120
- DRC cover, HDR, 177
- DRC value, HDR, 176
- DTMF
 - summary table, 194
 - burst, generate, 223
 - DTMF Duration object, 220
 - DTMF Key object, 219
 - keys released, 222
 - mute/unmute, 216
 - play tone, 219, 221

- playback mode, 224
- send out pressed key, 221
- tone duration, 220

duration

- all calls, 93
- current call, 197
- DTMF tone, 220

E**E911**

- E911 Mode object, 195
- mode, get/set, 195
- originate a call, 200

- ear piece mute setting, 79

Ec/Io

- CDMA, 144
- CDMA and HDR, 145

- Echo Cancel, 216

- echo cancellation level, 216

- ECIO object, 144

- Electronic Serial Number (ESN), 91

- elevation, satellite, 285

- enable host wakeup, 100

- enable notifications

- call, 196
- HDR, 154
- HDR Tech, 172
- IOTA, 249
- LBS, 254, 255, 256

- encapsulation, 33

- end IOTA session, 248

- end LBS session, 265

- End Reason, 204

- Enhanced Roaming Indicator. See ERI

- ephemeris data

- request a download, 257
- validity period, 263

- ephemeris data, delete, 286

ERI

- banner text, 151
- create file, 94
- icon image, 151
- query file, 94
- write file, 95

- ERI File Name object, 94

- ERI Write object, 95

- escaping, 25

- ESN, 91

- EVDO. See HDR

- example - SMS read sequence

- r1 objects, 304
- r2 objects, 307

- Extended Caller ID, 200

F

- fax, answer calls as, 166

- features of modem, 23

- FER (Frame Error Rate), 146

- Fetch Device Data Statistics, 69

- file. See ERI

- firmware

- autoconnect, 106
- date, 49
- object releases supported, 58
- version
 - bootstrap/loader, 49
 - current document, 20
 - modem, 48

- Firmware Date object, 49

- Firmware Version object, 48

- first packet (SMS packet type), 297

- fix

- last, 262
- satellite information, 285

- type, 260
- Fix type, 136
- Flash Context, 209
- Flash Image Information, 83
- flash, send, 209
- flow control, temperature-imposed, 98
- foreign agent, 167
- Forward Traffic Channel
 - multi-user, 183
 - single-user, 182
- Frame Error Rate, 146
- framing, HIP packet, 25
- full audio, set audio mode, 107
- G**
- gain
 - microphone codec, 104
 - sidetone, 218
 - speaker codec, 105
- generate DTMF burst, 223
- glossary, 21, 318
- GPS. *See* LBS
- GPS Lock, 279
- GPS NMEA output message type, 135
- GPS service, 150
- GPS Smart Mode, 135
- H**
- HAC (Hearing Aid Compatibility), 78, 79, 80, 107
- handset, 78, 79, 80
- handset, set audio mode, 107
- hardware
 - summary, 45
 - reference, 47
 - temperature, 98
- Hardware Temperature object, 98
- hazards, 2
- HDR
 - authentication status, 176
 - change of state, 174
 - channel, 148
 - close session, 170
 - DDTM preference, 169
 - DRC cover, 177
 - DRC value, 176
 - hybrid mode preference, 157
 - notifications, enable or disable, 154, 172
 - RF statistics, 149
 - roaming status, 155
 - RRI (reverse rate indicator), 178
 - SCP configuration, 133
 - sector, 148
 - service state, 156
 - session info, 175
 - HDR Hybrid Preference, 157
 - HDR Idle Digital Mode, 158
 - HDR Miscellaneous Notification Status, 154
 - HDR Multi-user Forward statistics, 183
 - HDR Notification Status, 154
 - HDR Previous and Current Idle Digital Mode, 158
 - HDR Roaming Status, 155
 - HDR RSSI, 146
 - HDR Service State, 156
 - HDR Single-user Forward statistics, 179
 - HDR Single-user Forward Traffic Channel and Control Channel DRC packet termination slot count, 182
 - HDR Single-user Forward Traffic Channel DRC packet termination slot count, 181
 - HDR Tech Notification Status, 172
- header, SMS
 - structure, 228
- header, SMS packet

- structure, 226, 298
- heading, 268
- headset, 78, 79, 80
 - summary table, 193
 - audio mode, set, 107
 - play DTMF tone, 221
 - tone level, 215
- Headset Icon object, 63
- hear audio
 - audio mode, set, 107
- Hearing Aid Compatibility (HAC), 78, 79, 80, 107
- Heartbeat object, 47
- height, 268
- HEPE, 269
- high battery critical state, 53
- High Data Rate. *See* HDR
- high priority notifications, 30
- high temperature warning state, 52
- HIP
 - definition, 18
 - overview, 24
 - encapsulation, 33
 - interface rules, 26
 - message types, 26
 - packet, framing, 25
 - packet, structure, 24
- home agent, 168
- horizontal velocity, 268
- host
 - CnS notification, 28
 - power-save configurations, 30
 - wakeup, enable, 100
- hot notification – modem too hot, 99
- hybrid mode preference, HDR, 157
- I**
- ICMPv4 statistics
- get, 69
- reset, 68
- icon, headset, 63
- idle digital mode, 158
- IIR filter, 142, 145
- IMSI, 139
- incoming calls
 - answer mode, 166
 - notification, 197
 - voice, answer, 199
- incoming SMS messages, 293, 300
- indicator, service, 190
- integration support, 4
- interface
 - rules, HIP, 26
- Interface Set Lock, 66
- Interface Unlock object, 64
- intermediate packet (SMS packet type), 297
- IOTA
 - summary table, 247
 - features, enable/disable, 108
 - notifications, enable, 249
 - start session, 247
 - stop session, 248
 - WAP push, 251
- IOTA Feature object, 108
- IOTA Start, 247
- IOTA Status, 249
- IOTA Stop, 248
- IOTA WAP Push, 251
- IPR2, 106
- IPv4 statistics
 - get, 69
 - reset, 68
- K**
- KB sent and received

- cumulative, 172
- current data call, 170
- KB Sent and Received object, 172
- Key Pressed object, 221
- Key Released object, 222
- keys released, dialpad, 222
- L**
- language of SMS message, 230
- last fix, 262
- last packet (SMS packet type), 297
- latitude
 - base station, 161
 - position fix, 267
- LBS
 - summary table, 252
 - Base Station Information, 283
 - Clear GPS Assistance Data, 286
 - clients, types of, 252
 - Coarse Location Message (CLM), 135
 - Download Data, 275
 - Download Data Session Begin, 275
 - Download Data Session Done, 277
 - Download Data Session End, 277
 - Download Notification Status, 256
 - enable/disable notifications, 254, 255, 256
 - GPS Lock, 279
 - LBS Status, 262
 - Location Notification Status, 255
 - MS-Assisted Mode, 136
 - MS-Based Mode, 136
 - Network Access Permissions, 284
 - notifications, enable/disable, 254, 255, 256
 - Optimal Accuracy Mode, 136
 - Optimal Data Mode, 136
 - Optimal Speed Mode, 136
 - PA Parameter Settings, 266
 - Parameter Notification Status, 257
 - PD Download, 257
 - PD End Session, 265
 - PD Get Position, 261
 - PD Track, 264
 - PD Update Failed, 274
 - Port ID, 281
 - Position Fix Data, 267
 - Position Fix Session Begin, 274
 - Position Fix Session Done, 271
 - Position Fix Session End, 272
 - Privacy Level, 282
 - Satellite Information, 285
 - single fix session, 313
 - Standalone Mode, 136
 - TCP/IP Address, 278
 - tracking session, 314
 - Transport Mechanism, 280
- LBS Fix type, 136
- LBS Status object, 262
- Length, 33
- liability, 2
- limitation of liability, 2
- list of all message objects, 36
- loader version, 49
- location based services. *See* LBS
- location fix, initiate, 260
- location information, base station, 161
- Location Notification Status, 255
- Location Uncertainty A, 268
- Location Uncertainty Angle, 267
- Location Uncertainty Position, 268
- location uncertainty vertical, 268
- lock modem
 - immediately, 290
 - on power-up, 291

- Lock object, 290
- lock status, 289
- locking PIN, change, 290
- longitude
 - base station, 161
 - position fix, 267
- low battery warning state, 53
- low power mode reason, 50
- low priority notifications, 30
- low temperature critical state, 52
- M**
- maximum PREV revision supported, 162
- MCC, 139
- Mdm TooHot Indication, 99
- MDN (mobile directory number), 138
- Message Category, 228
- Message ID, 25
- Message ID Type, 229
- Message Language, 230
- message objects, list of all, 36
- Message Priority, 230
- Message Privacy Level, 230
- Message Teleservice, 229
- message types, 26
- messages, SMS. *See* SMS, *See* SMS
- messaging. *See* SMS
- Mic Level object, 104
- Mic Mute object, 214
- microphone
 - input codec, 104
 - mute/unmute, 214
- microphone mute setting, 79
- microSD card (Removable Media Configuration), 87
- MIN, 139
- minimum PREV revision supported, 162
- MIP Error Code, 167
- MNC, 139
- MO SMS messages, 295
- Mobile Country Code (MCC), 139
- Mobile Directory Num object, 138
- mobile directory number (MDN), 138
- Mobile IP registration code, 167
- Mobile MIN, get/set, 139
- mobile originated position tracking session, 314
- mobile originated single position fix session, 313
- model of modem, 54
- modem CnS notification, 27
- modem device characteristics, 45
- modem lock status, 289
- modem model, 54
- modem too hot notification, 99
- modem wakeup reason, 101
- MS-Assisted Mode, 136
- MS-Assisted Only location fix method, 260
- MS-Based Mode, 136
- MS-Based Only location fix method, 260
- MS-Based Smart Mode, 135
- MS-based throttling parameters, reset, 287
- MSIN, 139
- MSL validation challenge
 - step 1, 129
 - step 2, 130
- MT SMS messages, 293, 300
- Multi-user Forward statistics, 183
- mute
 - DTMF, 216
 - microphone, 214
 - speaker, 214
- mute setting, 79, 80
- N**
- NAM, active

- change, 132
- get, 132
- NAM, set index, 133
- NAND image partition table, 84
- NDIS driver, firmware autoconnect, 106
- negotiated protocols and application subtypes, 187
- network
 - summary table, 141
 - connect automatically - firmware, 106
 - identifier (NID), 133, 161, 283
 - registration code for Mobile IP connections, 167
 - service indication, 150
- Network Access Permissions, 284
- Network-Initiated Alert, 119
- new in this document, 4
- new SMS messages, 293, 300
- NID (Network Identifier), 133, 161, 283
- NMEA output message type, 135
- No service, 150
- normal urgency SMS messages
 - number of messages in modem, 233, 241
- not segmented (SMS packet type), 297
- notifications
 - overview, 27, 30
 - enable or disable
 - call, 196
 - HDR, 154
 - HDR Tech, 172
 - IOTA, 249
 - LBS, 254, 255, 256
 - host CnS, 28
 - LBS, 252
 - modem CnS, 27
 - new SMS, 294
 - sequence of messages, 31
- numeric object listing, 36
- NV Items object, 133
- NVRAM Backup/Restore object, 57
- NVRAM Update from File, 65
- O**
- Object ID, 32
- Object Releases, Supported, 58
- object set Release, 60
- objects listing
 - all objects, 36
 - potentially obsolete objects, 316
 - account management messages, 127
 - additional features, 288
 - configuration messages, 103
 - data call management, 163
 - device characteristics, 45
 - Device Management messages, 109
 - DTMF, 194
 - headset controls, 193
 - IOTA, 247
 - LBS, 252
 - network information, 141
 - radio signal information, 140
 - SMS, 226
 - voice call management, 192
- obsoleted objects, 316
- OEM Secondary Boot Loader, 84
- OEMSBL (OEM Secondary Boot Loader), 84
- OEMSBL (SWI Boot Loader), 84
- OMA-DM. *See* DM (Device Management)
- Operation Type, 32
- Operations entry in a table, 34
- Optimal Data Mode, 136
- Optimal Speed Mode, 136
- Optimized for Accuracy location fix method, 260
- Optimized for Data location fix method, 260

Optimized for Speed location fix method, 260

OTASP

originate a call, 200

state, 212

OTSL validation challenge

step 1, 129

step 2, 130

outbox

send SMS

object description, 237

steps involved, 295, 311

write SMS to, 236

outgoing SMS messages, 295

Over The Air Service Provisioning. *See* OTASP

P

P_REV. *See* PREV

PA Parameter Settings, 266

packet header, SMS

definition, 225

structure, 226, 298

packet payload, SMS

definition, 225

description, 299

packet, HIP, 24

packet, SMS

definition, 225

types, 297

Parameter entry in tables, 33

Parameter Notification Status, 257

password-protected items, unlock access to, 64

payload, SMS packet

definition, 225

description, 299

PBL (Primary Boot Loader), 84

PD Download, 257

PD End Session, 265

PD Get Position, 261

PD Track, 264

PD Update Failed, 274

PDSM. *See* LBS

personality

current, 186

negotiated application subtypes, 187

negotiated protocols, 187

Phone Number object, 128

PIN. *See* security

play DTMF tone, 219, 221

playback mode, DTMF, 224

Pn offset, 148

poll for received SMS messages, 300

Port ID, 281

port, secondary. *See* secondary port

Position Fix Data, 267

Position Fix Session Begin, 274

Position Fix Session Done, 271

Position Fix Session End, 272

position fix, number of satellites used in, 269

Position Uncertainty, 269

Power Down Modem object, 93

power mode, 51

power, modem, 50

power-save configurations, 30

power-up auto-lock modem, 291

PPP statistics

get, 69

reset, 68

Preferred Roaming List. *See* PRL

PREV

minimum and maximum revisions supported, 162

reported by base station, 152

PREV object, 152

- PRI
 - version information, 82
- PRI Information, 82
- PRI revision, 133
- PRI SKU revision, 135
- Primary Boot Loader (PBL), 84
- Privacy Level, 282
- privacy, call, 211
- PRL
 - size, get or set, 96
 - version, 92
 - write, 97
 - writing, 315
- PRL Size object, 96
- PRL Version, 92
- PRL Write object, 97
- profile for activation, select, 133
- profile, audio, 78
- Protocol Revision Number. *See* PREV
- Provisioning Date object, 137
- provisioning, OTASP state, 212

- Q**
- QUALCOMM GPS Smart Mode, 135

- R**
- radio band, current, 67
- Radio Power object, 50
- radio reference, 142
- radio signal information, 140
- Radio Supply Voltage object, 53
- Radio Temperature object, 52
- RATI, 175
- read segment acknowledgement (SMS), 297
- read SMS
 - r1, 233
 - r2, 242
- reason for last modem wakeup, 101
- receive status, SMS
 - r1, 232
 - r2, 241
- Received Signal Strength Indication
 - CDMA, 142
 - CDMA and HDR, 145
 - HDR only, 146
- reference documents, 20
- reference tables, 34
- registered (LBS) client, 252
- registration code for Mobile IP connections, 167
- Remote Diagnostics Agent, 88
- Removable Media Configuration, 87
- request and error, 31
- request and response, 29, 31
- requesting (LBS) client, 252
- requests, 26
- Reset Device Data Statistics, 68
- Reset Device object, 63
- Reset Modem object, 92
- responses, 26
- restore NVRAM, 57
- retrieve SMS
 - r1, 233
 - r2, 242
- revision history, 4
- RF statistics, HDR, 149
- RI signal, 101
- ring, wake host on, 101
- Roaming Indicator, Enhanced. *See* ERI
- roaming preference, 160
- roaming status
 - CDMA, 151
 - HDR, 155
- Roaming Status object, 151

RRI (reverse rate indicator), HDR, 178

RSSI

CDMA, 142

CDMA and HDR, 145

HDR only, 146

RSSI and Ec/Io for CDMA and HDR object, 145

RSSI object, 142

rx_agc0, 149

rx_agc1, 149

RxCounter, 170

S

safety and hazards, 2

sales desk, 3

sample SMS read sequence

r1 objects, 304

r2 objects, 307

satellite data, delete, 286

Satellite Information, 285

satellites used in position fix, number of, 269

SBL (Secondary Boot Loader), 84

SCD (Secure Challenge Digits), 129

SCP configuration, HDR, 133

SCR (Secure Challenge Response), 129

Scratchpad, 292

SD card (Removable Media Configuration), 87

Secondary Boot Loader (SBL), 84

secondary port, data rate, 106

sector, HDR, 148

Secure Challenge, 129

Secure Challenge Digits (SCD), 129

Secure Challenge Response (SCR), 129

Secure Commit Result, 131

Secure Commit Result Mask Bit, 131

Secure Commit Update, 131

Secure Rebuttal, 130

Secure Rebuttal Digits (SRD), 130

Secure Rebuttal Response (SRR), 130

security

auto-lock on power-up, 291

lock modem immediately, 290

locking PIN, change, 290

modem lock status, 289

PIN, change, 290

power-up auto-lock, 291

unlock modem, 290

segmentation of SMS messages, 296

Select NAM object, 132

send flash, 209

send out pressed key, 221

send SMS

object description, 237

status, 238

steps involved, 295, 311

send status, SMS, 238

sequences of messages, 30

serial number, electronic (ESN), 91

Service Indication, 150

service indicator, 190

Service Option, 201

service state, HDR, 156

session state

DM, 114

session, DM (Device Management)

cancel, 113

start, 112

state, 114

session, HDR

parameters, change of, 175

state, change of, 174

session, IOTA

start, 247

stop, 248

- session, LBS
 - end, 265
- Set Dormant, 189
- shaded entries, 34
- short text messaging. *See* SMS
- shut down modem, 50
- shutdown, forced, 98
- SID (System ID)
 - Activate Profile object, 133
 - base station, 283
 - Tech SID object, 160
- Sidetone Gain Level object, 218
- Sierra Wireless Modem Type, 54
- signal strength
 - CDMA, 142
 - CDMA and HDR, 145
 - HDR only, 146
 - interpretation, 143
- signal/noise ratio (SNR), satellite, 285
- single fix session, 313
- Single-user Forward statistics, 179
- Single-user Forward Traffic Channel and Control
 - Channel DRC packet termination slot count, 182
- Single-user Forward Traffic Channel DRC
 - packet termination slot count, 181
- SKU revision, PRI, 135
- Smart Mode, QUALCOMM GPS, 135
- SMS
 - overview, 225, 293
 - objects listing, 226
 - delete (r1), 235
 - delete (r2), 245
 - full, storage, 241
 - header, structure, 228
 - incoming messages, 293, 300
 - incoming, wake host on, 101
 - maximum number of messages modem can store, 241
 - number of messages in modem (r1), 232
 - number of messages in modem (r2), 241
 - outgoing messages, 295
 - packet
 - definition, 225
 - header, definition, 225
 - header, structure, 226, 298
 - payload, definition, 225
 - payload, description, 299
 - types, 297
 - read (r1), 233
 - read (r2), 242
 - receive status (r1), 232
 - receive status (r2), 241
 - retrieve (r1), 233
 - retrieve (r2), 242
 - segmentation of messages, 296
 - send
 - object description, 237
 - status, 238
 - steps involved, 295, 311
 - voice message, delete, 240
 - write, 236
 - SMS Delete (r1), 235
 - SMS Delete (r2), 245
 - SMS Full, 241
 - SMS Received Msg Status (r1), 232
 - SMS Received Msg Status (r2) and Storage Capacity, 241
 - SMS Retrieve SMS (r1), 233
 - SMS Retrieve SMS (r2), 242
 - SMS Send SMS, 237
 - SMS Send Status, 238
 - SMS Store SMS, 236
 - SMS Voice Message Count Delete, 240

- SMS Voice Message Delete, 240
 - SNR (signal/noise ratio), satellite, 285
 - Software on Card image, 84
 - speaker
 - codec gain, 105
 - level, 105
 - mute/unmute, 214
 - speaker phone, set audio mode, 107
 - volume, 213
 - Speaker Level object, 105
 - speaker phone, 78, 79, 80
 - Speaker Volume object, 213
 - SRD (Secure Rebuttal Digits), 130
 - SRR (Secure Rebuttal Response), 130
 - Standalone Mode, 136
 - Standalone Only location fix method, 260
 - Standalone Self Learning data base, clear, 287
 - Standalone Serving System data base, clear, 287
 - Standalone Smart Mode, 135
 - Standalone Time Reference, clear, 287
 - Start DM Session, 112
 - start IOTA session, 247
 - state, HDR, 174
 - statistics
 - bytes sent and received
 - cumulative, 172
 - current data call, 170
 - HDR RF, 149
 - reset, 68
 - retrieve, 69
 - statistics, data, reset, 68
 - status
 - LBS, 262
 - stop IOTA session, 248
 - store SMS, 236
 - subnet mask, 148
 - summary tables. See objects listing
 - supply voltage, 53
 - support, technical, 4
 - Supported Object Releases, 58
 - suspended host, wakeup
 - enable, 100
 - SWOC
 - definition, 325
 - SWOC Configuration, 89
 - System ID (SID)
 - Activate Profile object, 133
 - base station, 283
 - Tech SID object, 160
 - System Selection, 159
 - System Time object, 153
 - system time of modem, 81
 - system time, CDMA, 153
- ## T
- talk audio
 - audio mode, set, 107
 - TCP statistics
 - reset, 68, 69
 - TCP/IP Address used in LBS sessions, 278
 - Tech Bandclass, 147
 - Tech BS Info, 161
 - Tech FER, 146
 - Tech HDR Authentication Status, 176
 - Tech HDR Channel, 148
 - Tech HDR DRC, 176
 - Tech HDR DRC Cover, 177
 - Tech HDR Minimum and Maximum PREV, 162
 - Tech HDR Notification Status, 173
 - Tech HDR RF Statistics, 149
 - Tech HDR RRI, 178
 - Tech HDR Sector, 148

- Tech HDR Session, 175
 - Tech HDR State, 174
 - Tech NID, 161
 - Tech SID, 160
 - technical support, 4
 - telephone number
 - modem, 128
 - technical support, 4
 - temperature
 - hardware, 98
 - modem too hot notification, 99
 - radio, 52
 - threshold, 99
 - Temperature Threshold object, 99
 - Terminate Activation object, 131
 - terminology, 21
 - text messaging. *See* SMS
 - three-way call, 210
 - threshold, temperature, 99
 - throttling parameters, MS-based - reset, 287
 - time to first fix, 263
 - time, CDMA system, 153
 - time, device, 81
 - timestamp
 - position fix, 267
 - TL Acknowledge Request, 229
 - tone duration, DTMF, 220
 - Tone Level object, 215
 - tone level, headset, 215
 - tones. *See* DTMF
 - Total Call Duration object, 93
 - tracking session, initiate, 263
 - tracking session, LBS
 - diagram, 314
 - traffic - bytes sent and received
 - cumulative, 172
 - current data call, 170
 - traffic channel, release, 207
 - Transport Mechanism, 280
 - TRU-Install, 325
 - TTY, 78, 79, 80, 217
 - TTY Option object, 217
 - tx_adj, 149
 - tx_total, 149
 - TxCounter, 170
- ## U
- UATI, 175
 - UDP statistics
 - get, 69
 - reset, 68
 - Uncertainty A, 269
 - Uncertainty P, 269
 - Uncertainty V, 269
 - unlock keys, set, 66
 - unlock modem, 290
 - unlock password-protected items, 64
 - unmute
 - DTMF, 216
 - microphone, 214
 - speaker, 214
 - update NVRAM from file, 65
 - urgent SMS messages
 - number of messages in modem, 232, 241
 - USB descriptor table, 84
 - User Acknowledge Required, 231
- ## V
- validation challenge, MSL/OTSL
 - step 1, 129
 - step 2, 130
 - velocity
 - horizontal, 268

- vertical, 268
- version
 - firmware
 - bootstrap/loader, 49
 - modem, 48
 - object releases supported, 58
 - PRI, 82
 - PRL, 92
 - protocol revision
 - minimum and maximum revisions supported, 162
 - reported by base station, 152
 - Supported Object Releases object, 58
- vertical location uncertainty, 268
- vertical velocity, 268
- voice
 - answer calls as, 166
 - answer incoming call, 199
 - call management, 192
 - disconnect, 206
 - message (SMS), delete, 240
 - originate a call, 200
 - voice circuitry, set the mode, 78
 - voice mail notification of new SMS, 294
- voltage, supply, 53
- volume level, 79
- volume, speaker, 213

W

- wakeup
 - host, enable, 100
 - modem, reason, 101
- Wakeup Enable object, 100
- Wakeup Reason object, 101
- WAP push, IOTA, 251
- write
 - ERI file, 95
 - PRL, 97
 - SMS, 236

By object name

A

- Activation Status, 128
- Active NAM, 132
- Answer State, 166
- Audio Mode, 107
- Audio Profile, 78
- Auto Lock, 291
- Autoconnect, 106

B

- Base Station Information, 283
- Baud Rate, 106

- Boot Build Date, 62
- Boot Version, 49

C

- Call Answer Cmd, 199
- Call Byte Counter, 170
- Call Caller ID, 199
- Call Connected, 201
- Call Connecting, 210
- Call Connection State, 203
- Call Disconnect Cmd, 206
- Call Disconnected, 204

- Call Dormant, 171
 - Call Duration, 197
 - Call Error, 207
 - Call Extended Caller ID, 200
 - Call Flash Cmd, 209
 - Call Incoming, 197
 - Call Notification Status, 196
 - Call Originate Cmd, 200
 - Call OTASP State, 212
 - Call Privacy, 211
 - Cancel DM Session, 113
 - Change Lock PIN, 290
 - Channel Number, 143
 - Channel State, 144
 - Clear GPS Assistance Data, 286
 - Close HDR Session, 170
 - Continuous DTMF Mode, 224
 - current personality data, 186
 - Current Radio Band, 67
- D**
- Data Dedicated Transmission Mode for Hybrid, 169
 - Device Time, 81
 - DM Configuration, 110
 - DM Download — Installation Status, 124
 - DM Download Alert, 121
 - DM Download Description, 123
 - DM Download Progress, 120
 - DM Session State, 114
 - Download Data, 275
 - Download Data Session Begin, 275
 - Download Data Session Done, 277
 - Download Data Session End, 277
 - Download Notification Status, 256
 - DTMF Burst, 223
 - DTMF Duration, 220
 - DTMF Key, 219
 - DTMF Mute, 216
- E**
- E911 Mode, 195
 - Echo Cancel, 216
 - ECIO, 144
 - ERI File Name, 94
 - ERI Write, 95
 - ESN, 91
- F**
- Fetch Device Data Statistics, 69
 - Firmware Date, 49
 - Firmware Version, 48
 - Flash Image Information, 83
- G**
- GPS Lock, 279
- H**
- Hardware Temperature, 98
 - HDR Hybrid Preference, 157
 - HDR Multi-user Forward statistics, 183
 - HDR Notification Status, 154
 - HDR Previous and Current Idle Digital Mode, 158
 - HDR Roaming Status, 155
 - HDR RSSI, 146
 - HDR Service State, 156
 - HDR Single-user Forward Traffic Channel and Control Channel DRC packet termination slot count, 182
 - HDR Single-user Forward Traffic Channel DRC packet termination slot count, 181
 - Headset Icon, 63
 - Heartbeat, 47
- I**
- IMSI, 139

Interface Set Lock, 66
Interface Unlock, 64
IOTA Feature, 108
IOTA Start, 247
IOTA Status, 249
IOTA Stop, 248
IOTA WAP Push, 251

K

KB Sent and Received, 172
Key Pressed, 221
Key Released, 222

L

LBS Status, 262
Location Notification Status, 255
Lock, 290
Lock Status, 289

M

Mdm TooHot Indication, 99
Mic Level, 104
Mic Mute, 214
MIP Error Code, 167
Mobile Directory Num, 138

N

Negotiated protocols and application subtypes,
187
Network Access Permissions, 284
Network-Initiated Alert, 119
NV Items, 133
NVRAM Backup/Restore, 57
NVRAM Update from File, 65

P

PA Parameter Settings, 266
Parameter Notification Status, 257
PD Download, 257

PD End Session, 265
PD Get Position, 261
PD Track, 264
PD Update Failed, 274
Phone Number, 128
Port ID, 281
Position Fix Data, 267
Position Fix Session Begin, 274
Position Fix Session Done, 271
Position Fix Session End, 272
Power Down Modem, 93
PREV, 152
PRI Information, 82
Privacy Level, 282
PRL Size, 96
PRL Version, 92
PRL Write, 97
Provisioning Date, 137

R

Radio Power, 50
Radio Supply Voltage, 53
Radio Temperature, 52
Remote Diagnostics Agent, 88
Removable Media Configuration, 87
Reset Device, 63
Reset Device Data Statistics, 68
Reset Modem, 92
Roaming Status, 151
RSSI, 142
RSSI and Ec/Io for CDMA and HDR, 145

S

Satellite Information, 285
Scratchpad, 292
Secure Challenge, 129
Secure Commit Update, 131

Secure Rebuttal, 130
Select NAM, 132
Service Indication, 150
Set Dormant T, 189
Sidetone Gain Level, 218
Sierra Wireless Modem Type, 54
Single-user Forward statistics, 179
SMS Delete (r1), 235
SMS Delete (r2), 245
SMS Full, 241
SMS Received Msg Status (r1), 232
SMS Received Msg Status (r2) and Storage Capacity, 241
SMS Retrieve SMS (r1), 233
SMS Retrieve SMS (r2), 242
SMS Send SMS, 237
SMS Send Status, 238
SMS Store SMS, 236
SMS Voice Message Count Delete, 240
Speaker Level, 105
Speaker Mute, 214
Speaker Volume, 213
Start DM Session, 112
Supported Object Releases, 58
SWOC Configuration, 89
System Selection, 159
System Time, 153

T

TCP/IP Address, 278
Tech Bandclass, 147
Tech BS Info, 161
Tech FER, 146
Tech HDR Authentication Status, 176
Tech HDR Channel, 148
Tech HDR DRC, 176
Tech HDR DRC Cover, 177
Tech HDR Minimum and Maximum PREV, 162
Tech HDR Notification Status, 173
Tech HDR RF Statistics, 149
Tech HDR RRI, 178
Tech HDR Sector, 148
Tech HDR Session, 175
Tech HDR State, 174
Tech NID, 161
Tech SID, 160
Temperature Threshold, 99
Tone Level, 215
Total Call Duration, 93
Transport Mechanism, 280
TTY Option, 217

W

Wakeup Enable, 100
Wakeup Reason, 101