nRF91x1 Cellular AT Commands

Command Reference Guide

v1.0

4418_1541 v1.0 / 2023-11-27

NORDIC® SEMICONDUCTOR

Contents

Revision history . . . vi

1 Introduction . . . 7

2 AT command syntax . . . 8

2.1 Set command **<CMD>[=...]** . . . 8
2.2 Read command **<CMD>?** . . . 8
2.3 Test command **<CMD>=?** . . . 9
2.4 Response . . . 9

3 Modem firmware versions and hardware variants . . . 10

4 General . . . 11

4.1 Manufacturer identification +CGMI . . . 11
4.2 Model identification +CGMM . . . 11
4.3 Revision identification +CGMR . . . 12
4.4 Product serial number identification +CGSN . . . 13
4.5 Short software identification %SHORTSWVER . . . 14
4.6 Hardware identification %HWVERSION . . . 15
4.7 Modem build UUID %XMODEMUUID . . . 16
4.8 Set and read ODIS fields +ODIS . . . 16
4.9 ODIS notification +ODISNTF . . . 17
4.10 SiP 2DID %2DID . . . 18
4.11 Read device UUID %DEVICEUUID . . . 19

5 Mobile termination control and status commands . . . 20

5.1 Functional mode +CFUN . . . 20
5.2 List all available AT commands +CLAC . . . 23
5.3 Extended signal quality +CESQ . . . 24
5.4 Signal quality notification %CESQ . . . 26
5.5 Signal quality +CSQ . . . 28
5.6 SNR signal quality notification %XSNRSQ . . . 29
5.7 Device activity status +CPAS . . . 31
5.8 Indicator control +CIND . . . 32
5.9 IP address format +CGPIAF . . . 34
5.10 Current band %XCBAND . . . 35
5.11 Neighboring cells %NBRGRSRP . . . 36
5.12 Mode of operation (CS/PS) +CEMODE . . . 37
5.13 Band lock %XBANDLOCK . . . 38
5.14 Data profile %XDATAPRFL . . . 40
5.15 Connectivity statistics %XCONNSTAT . . . 42
5.16 Battery voltage %XVBAT . . . 43
5.17 Battery voltage low level notification %XVBATLVL . . . 44
5.18 Battery voltage low level %XVBATLOWLVL . . . 45
5.19 External power off warnings %XPOFWARN . . . 46
5.20 Customer production done %XPRODDONE . . . 47
5.21 Internal temperature notification %XTEMP . . . 48
5.22 High level for internal temperature %XTEMPHIGHLVL . . . 49
5.23 Clock +CCLK . . . 50

5.24 Proprietary clock %CCLK . . . 51
5.25 Modem trace activation %XMODEMTRACE . . . 52
5.26 Fallback to SMS only %XSMSFALLBACK . . . 54
5.27 System mode %XSYSTEMMODE . . . 54
5.28 PTW setting %XPTW . . . 57
5.29 Extra maximum TX power reduction %XEMPR . . . 61
5.30 Write content to file %XFILEWRITE . . . 62
5.31 Coverage enhancement mode notification +CEINFO . . . 63
5.32 Factory reset %XFACTORYRESET . . . 65
5.33 Modem sleep notification %XMODEMSLEEP . . . 66
5.34 Periodic TAU notification %XT3412 . . . 68
5.35 Country-specific search optimization %XCOUNTRYDATA . . . 69
5.36 Modem domain event notification %MDMEV . . . 71
5.37 Neighboring cell measurement and notification %NCELLMEAS . . . 73
5.38 Stop neighboring cell measurement %NCELLMEASSTOP . . . 79
5.39 Evaluating connection parameters %CONEVAL . . . 80
5.40 Disabling SMS services in NB-IoT %SMSDISABLE . . . 85

6 SiP pin configuration . . . 87

6.1 COEX0 pin control configuration %XCOEX0 . . . 87
6.2 MAGPIO configuration %XMAGPIO . . . 89
6.3 SiP-external MIPI RFFE device introduction %XMIPIRFFEDEV . . . 91
6.4 SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL . . . 93
6.5 Alternative configuration of SiP antenna switch %XANTCFG . . . 97
6.6 COEX2 pin timing configuration %XCOEX2 . . . 98

7 Packet domain commands . . . 100

7.1 Define PDP context +CGDCONT . . . 100
7.2 Packet domain event notification +CGEREP . . . 102
7.3 Packet domain event notification +CGEV . . . 103
7.4 Activate PDP context +CGACT . . . 107
7.5 Allocate new CID %XNEWCID . . . 109
7.6 Map CID to PDN ID %XGETPDNID . . . 110
7.7 QoS dynamic parameters +CGEQOSRDP . . . 110
7.8 Show PDP addresses +CGPADDR . . . 111
7.9 PDN connection dynamic parameters +CGCONTRDP . . . 112
7.10 PS attach or detach +CGATT . . . 114
7.11 Power preference indication for EPS +CEPPI . . . 115
7.12 Protocol configuration options notification %XPCO . . . 116
7.13 Usage of ePCO/PCO in PDN connection establishment %XEPCO . . . 117
7.14 APN class access %XAPNCLASS . . . 118
7.15 External IP stack IPv6 address resolution/refresh failure %XIPV6FAIL . . . 119
7.16 Define PDN connection authentication parameters +CGAUTH . . . 119
7.17 Signaling connection status notification +CSCON . . . 120
7.18 Use of APN %XAPNSTATUS . . . 122
7.19 PDN configuration %XPDNCFG . . . 123
7.20 APN rate control %APNRATECTRL . . . 124
7.21 Exceptional data state %EXCEPTIONALDATA . . . 128

8 Network service related commands . . . 130

8.1 PLMN selection +COPS . . . 130
8.2 Forced PLMN search %COPS . . . 132
8.3 Power saving mode setting +CPSMS . . . 133

8.4 eDRX setting +CEDRXS . . . 137
8.5 EDRX dynamic parameters +CEDRXRDP . . . 143
8.6 Operator name +COPN . . . 146
8.7 Network registration status notification +CEREG . . . 146
8.8 Operator name notification %XOPNAME . . . 150
8.9 Network time notification %XTIME . . . 151
8.10 Operator ID %XOPERID . . . 152
8.11 Modem parameters %XMONITOR . . . 153
8.12 Network time support %XNETTIME . . . 158
8.13 Weak cell detection %XDEEPSEARCH . . . 160
8.14 Mobile network operator %XOPCONF . . . 161
8.15 Release assistance indication %RAI . . . 163
8.16 Periodic cell search configuration %PERIODICSEARCHCONF . . . 164
8.17 Reduced mobility feature configuration %REDMOB . . . 169
8.18 Feature configuration %FEACONF . . . 170

9 Mobile termination errors . . . 174

9.1 Mobile termination error notification +CMEE . . . 174
9.2 Network error code notification +CNEC . . . 175
9.3 Extended error report +CEER . . . 177

10 SMS commands . . . 179

10.1 Message format +CMGF . . . 179
10.2 New message indications +CNMI . . . 180
10.3 Send message, PDU mode +CMGS . . . 182
10.4 Received SMS notification in PDU mode +CMT . . . 183
10.5 Delivery status notification in PDU mode +CDS . . . 183
10.6 New message ACK, PDU mode +CNMA . . . 184
10.7 New message ACK, text mode +CNMA . . . 185
10.8 Preferred message storage +CPMS . . . 186
10.9 Message service failure result code +CMS ERROR . . . 188
10.10 SMS service +CGSMS . . . 189
10.11 Short message memory available %XSMMA . . . 190

11 Production test commands . . . 192

11.1 RX test %XRFTEST . . . 192

12 Security commands . . . 194

12.1 PIN code +CPIN . . . 194
12.2 Remaining PIN retries +CPINR . . . 195
12.3 Facility lock +CLCK . . . 196
12.4 Change password +CPWD . . . 197
12.5 Personalization of modem %XUSIMLCK . . . 198
12.6 Authenticated access %XSUDO . . . 201
12.7 Public key storage management %XPMNG . . . 202
12.8 Credential storage management %CMNG . . . 203
12.9 Attestation token generation %ATTESTTOKEN . . . 207
12.10 Key generation %KEYGEN . . . 208
12.11 Key injection %KEYINJECT . . . 212
12.12 Creating JWT %JWT . . . 213
12.13 Certificate expiry information %CERTEXPIRY . . . 215
12.14 Claim token generation %CLAIMTOKEN . . . 216

13 UICC access commands . . . 218
13.1 UICC state notification %XSIM . . . 218
13.2 Request IMSI +CIMI . . . 220
13.3 Request ICCID %XICCID . . . 220
13.4 Subscriber number +CNUM . . . 221
13.5 Restricted SIM access +CRSM . . . 222
13.6 Generic SIM access +CSIM . . . 224
13.7 Open logical channel +CCHO . . . 225
13.8 Close logical channel +CCHC . . . 226
13.9 Generic UICC access through logical channel +CGLA . . . 227
13.10 USAT event notification %USATEV . . . 229
13.11 USAT REFRESH and BIP state information %USATINFO . . . 231
13.12 UICC suspend and deactivate control +SSRDA . . . 233
13.13 Selecting UICC slot %CSUS . . . 234

14 Authenticating AT command usage . . . 237

Glossary . . . 238

Legal notices . . . 246

Revision history

Date	Version	Description
November 2023	1.0	First release

1 Introduction

This document describes the *AT command*s used to control the modem subsystem and 3GPP release 14 *LTE-M* and *Narrowband Internet of Things (NB-IoT)* modem firmware in nRF9131 and nRF9161 (nRF91x1). The AT command API enables modem control for firmware running in the application core on nRF91x1.

The AT command API can also be exposed on one of the nRF91x1 serial interfaces by programming appropriate firmware in the application core. The nRF Connect SDK contains examples of such proxy firmware that can be run stand-alone or as part of other firmware functionality in the nRF91x1 application core. The stand-alone example is called at_client. This way, an external MCU or computer can get access to the modem API either exclusively or in addition to application firmware running on the nRF91x1 itself.

The AT commands described in this document apply to all versions of the nRF91x1 module hardware. If a command applies only to a specific version of the module hardware, it is mentioned in the command description. The module hardware version is printed on the module label.

2 AT command syntax

The AT Commands have standardized syntax rules.

Words enclosed in <angle brackets> are references to syntactical elements. Words enclosed in [square brackets] represent optional items which may be left out from the command line at the specified point. The brackets are not used when the words appear in the command line.

`<CR>`, `<LF>`, and terminating `NUL` are allowed in an AT command sent by an application, but are not mandatory when using an interface where both the command string and length of command string are provided.

All standard AT commands for controlling a phone or a modem or managing the SMS feature begin with a plus sign (+), whereas Nordic-proprietary commands begin with a percent sign (%).

A string type parameter input should be enclosed between quotation marks ("").

For more information, see 3GPP 27.007 AT command set for User Equipment (UE) and 3GPP 27.005 Use of Data Terminal Equipment - Data Circuit terminating Equipment (DTE - DCE) interface for Short Message Service (SMS) and Cell Broadcast Service (CBS).

2.1 Set command `<CMD>[=...]`

Set commands set values or perform actions.

Example:

```
AT+CMD=1
```

where

- AT is the command line prefix
- + is the prefix for extended commands
- CMD is the body of a basic command
- 1 is a subparameter (multiple subparameters are separated by commas)

2.2 Read command `<CMD>?`

Read commands check the current values of subparameters.

Example:

```
AT+CMD?
```

where

- AT is the command line prefix
- + is the prefix for extended commands
- CMD is the body of a basic command
- ? represents a read command

2.3 Test command <CMD>=?

Test commands test the existence of the command and provide information about the type of its subparameters. Some test commands have also other functionality, which is described in the command-specific chapters.

Example:

```
AT+CMD=?
```

where

- AT is the command line prefix
- + is the prefix for extended commands
- CMD is the body of a basic command
- =? represents a test command for checking possible subparameter values

2.4 Response

AT responds to all commands with a final response.

The response is one of the following:

```
OK<CR><LF>
ERROR<CR><LF>
+CME ERROR: <cause_value><CR><LF>
+CMS ERROR: <cause_value><CR><LF>
```

"CMS ERROR:" is used as an error response for SMS related commands specified in *3GPP 27.005*.

Some commands may also produce a varying number of information response lines before the final response. An information response can be received only when a command-specific response syntax is specified. An information response line usually starts with a prefix, which is the command entered:

```
+CMD: [...]<CR><LF>
```

Some commands may also produce notifications, which do not start with the command prefix:

```
AT+CGSN
490154203237518
OK
```

3 Modem firmware versions and hardware variants

This document describes AT commands used in all versions of the nRF9131 and nRF9161 modem firmware.

The modem firmware versions that support a command are marked in the command description with the following version tag: vx.x.x

If a parameter is not supported by all versions of the modem firmware that support the command, the modem firmware versions that support the parameter are marked after the parameter with the version tag.

The version tags are read as follows:

- If a command or parameter is marked v2.0.x , it is supported by modem firmware versions where the first two digits are 2 and 0.
- If a command is marked v2.0.x v2.1.x v2.2.x , it is supported by modem firmware versions where the first two digits are 2 and 0, 2 and 1, or 2 and 2.
- If a command or parameter is marked v2.1.x≥3 , it is supported by modem firmware versions where the first two digits are 2 and 1 and the third digit is greater than or equal to 3.
- If a command or parameter is marked v2.1.3 , it is supported only by modem firmware version 2.1.3.
- If a command or parameter is marked nRF9131 , it is supported only by nRF9131.
- If a command or parameter is marked nRF9161 , it is supported only by nRF9161.

If a hardware variant is not specified for a command or parameter, it is supported by both nRF9131 and nRF9161.

4 General

The general commands are for the identification of the device.

For reference, see *3GPP 27.007 Ch. 5*.

4.1 Manufacturer identification +CGMI

The **+CGMI** command requests manufacturer identification. v2.0.x

For reference, see *3GPP 27.007 Ch. 5.1*.

4.1.1 Set command

The set command requests manufacturer identification.

Syntax:

```
+CGMI
```

Response syntax:

```
<manufacturer>
```

The <manufacturer> parameter returns a string of up to 2048 characters followed by `<CR><LF>`.

The following command example reads the manufacturer ID:

```
AT+CGMI
Nordic Semiconductor ASA
OK
```

4.1.2 Read command

The read command is not supported.

4.1.3 Test command

The test command is not supported.

4.2 Model identification +CGMM

For reference, see *3GPP 27.007 Ch. 5.2*. v2.0.x

4.2.1 Set command

The set command requests *System in Package (SiP)* model identification.

Syntax:

```
+CGMM
```

Response syntax:

```
<model>
```

The <model> parameter returns a string of up to 2048 characters followed by `<CR><LF>OK`.

The following command example reads the model ID:

```
AT+CGMM
nRF9161-LACA
OK
```

4.2.2 Read command

The read command is not supported.

4.2.3 Test command

The test command is not supported.

4.3 Revision identification +CGMR

The **+CGMR** command requests modem firmware revision identification. v2.0.x

For reference, see *3GPP 27.007 Ch. 5.3*.

4.3.1 Set command

The set command requests revision identification.

Syntax:

```
+CGMR
```

Response syntax:

```
<revision>
```

The <revision> parameter returns a string of up to 2048 characters followed by `<CR><LF>OK`.

The following command example reads the revision ID:

```
AT+CGMR
mfw_nrf91x1_2.0.0
OK
```

4.3.2 Read command

The read command is not supported.

4.3.3 Test command

The test command is not supported.

4.4 Product serial number identification +CGSN

The **+CGSN** command requests product serial number identification. v2.0.x

For reference, see *3GPP 27.007 Ch. 5.4*.

4.4.1 Set command

The set command requests product serial number identification.

Syntax:

```
+CGSN[=<snt>]
```

The set command parameters and their defined values are the following:

<snt>

0 – Respond with <sn> (default)

1 – Respond with `+CGSN: <imei>`

2 – Respond with `+CGSN: <imeisv>`

3 – Respond with `+CGSN: <svn>`

<sn>

Information text determined by the manufacturer. Up to 2048 characters. *Electronic Serial Number (ESN)* returned if available. *International Mobile (Station) Equipment Identity (IMEI)* returned if ESN not available.

<imei>

String in decimal format indicating the IMEI. Composed of *Type Allocation Code (TAC)* (8 digits), *Serial Number (SNR)* (6 digits), and *Check Digit (CD)* (1 digit).

<imeisv>

String in decimal format indicating the *International Mobile (Station) Equipment Identity, Software Version (IMEISV)*. The 16 digits of IMEISV are composed of TAC (8 digits), SNR (6 digits), and *Software Version Number (SVN)* (2 digits).

<svn>

String in decimal format indicating the current SVN which is part of IMEISV.

Response syntax when `<snt>=0` (or omitted):

```
<sn>
```

Response syntax for other <snt> values:

```
+CGSN: <string>
```

where <string> can be <imei>, <imeisv>, or <svn>.

The following command example reads the serial number:

```
AT+CGSN
352656100367872
OK
```

The following command example reads the IMEI:

```
AT+CGSN=1
+CGSN: "352656100367872"
OK
```

The following command example reads the IMEI when it is not written to the device:

```
AT+CGSN
FFFFFFFFFFFFFFF
OK
```

4.4.2 Read command

The read command is not supported.

4.4.3 Test command

The test command returns a list of supported <snt> values.

Response syntax:

```
+CGSN: (list of supported <snt>s)
```

The test command parameter and its defined values are the following:

<snt>

0 – Respond with <sn> (default)

1 – Respond with `+CGSN: <imei>`

2 – Respond with `+CGSN: <imeisv>`

3 – Respond with `+CGSN: <svn>`

The following command example reads the supported types of product serial number identifications:

```
AT+CGSN=?
+CGSN: (0-3)
OK
```

4.5 Short software identification %SHORTSWVER

The proprietary **%SHORTSWVER** command requests short software identification. v2.0.x

4.5.1 Set command

The set command requests short software identification.

Syntax:

```
%SHORTSWVER
```

Response syntax:

```
%SHORTSWVER: <version_string>
```

The response parameter and its defined value are the following:

<version_string>

String without double quotes. Short software identification.

The following command example requests short software identification:

```
AT%SHORTSWVER
%SHORTSWVER: nrf91x1_2.0.0
OK
```

4.5.2 Read command

The read command is not supported.

4.5.3 Test command

The test command is not supported.

4.6 Hardware identification %HWVERSION

The proprietary **`%HWVERSION`** command requests hardware identification. v2.0.x

4.6.1 Set command

The set command requests hardware identification.

Syntax:

```
%HWVERSION
```

Response syntax:

```
%HWVERSION: <version_string>
```

The response parameter and its defined value are the following:

<version_string>

String without double quotes. Hardware version.

The following command example requests hardware identification:

```
AT%HWVERSION
%HWVERSION: nRF9161 LACA A0A
OK
```

4.6.2 Read command

The read command is not supported.

4.6.3 Test command

The test command is not supported.

4.7 Modem build UUID %XMODEMUUID

The proprietary **%XMODEMUUID** command requests the *UUID* of a modem build. v2.0.x

4.7.1 Set command

The set command requests the *UUID* of a modem build.

Syntax:

```
%XMODEMUUID
```

Response syntax:

```
%XMODEMUUID: <UUID>
```

The response parameter and its defined value are the following:

<UUID>

String without double quotes. UUID of the modem build.

The following command example requests the UUID of a modem build:

```
AT%XMODEMUUID
%XMODEMUUID: 25c95751-efa4-40d4-8b4a-1dcaab81fac9
OK
```

4.7.2 Read command

The read command is not supported.

4.7.3 Test command

The test command is not supported.

4.8 Set and read ODIS fields +ODIS

The **+ODIS** command sets and reads ODIS fields. v2.0.x

4.8.1 Set command

The set command sets ODIS fields. The command configuration is stored to *Non-volatile Memory (NVM)* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
+ODIS=<HDID>,<HDMAN>,<HDMOD>,<HDSW>
```

The set command parameters and their defined values are the following:

<HDID>

String in alphanumeric format. Host device ID.

<HDMAN>

String in alphanumeric format. Host device manufacturer.

<HDMOD>

String in alphanumeric format. Host device model.

<HDSW>

String in alphanumeric format. Host device software version.

The following command example sets host device ID to HDID01, host device manufacturer to HDMAN01, host device model to HDMOD01, and host device software version to HDSW01:

```
AT+ODIS="HDID01","HDMAN01","HDMOD01","HDSW01"
OK
```

4.8.2 Read command

The read command reads ODIS fields.

The response includes all values except the host device ID.

Response syntax:

```
+ODIS: <HDMAN>,<HDMOD>,<HDSW>
```

The following command example reads the current values:

```
AT+ODIS?
+ODIS: "HDMAN01","HDMOD01","HDSW01"
OK
```

4.8.3 Test command

The test command is not supported.

4.9 ODIS notification +ODISNTF

The **+ODISNTF** command subscribes unsolicited ODIS notifications. v2.0.x

4.9.1 Set command

The set command subscribes ODIS notifications.

The notifications are triggered by modifications to ODIS fields.

Syntax:

```
+ODISNTF=<reporting>
```

Notification syntax:

```
+ODISNTF: <HDID>,<HDMAN>,<HDMOD>,<HDSW>
```

The set command parameter and its defined values are the following:

<reporting>

0 – Unsubscribe unsolicited notifications

1 – Subscribe unsolicited notifications

The following command example subscribes ODIS notifications:

```
AT+ODISNTF=1
OK
```

The following is an example of an unsolicited ODIS notification:

```
+ODISNTF: "HDID01","HDMAN01","HDMOD01","HDSW01"
```

4.9.2 Read command

The read command is not supported.

4.9.3 Test command

The test command is not supported.

4.10 SiP 2DID %2DID

The proprietary **`%2DID`** command reads the 2DID of the nRF91x1 *SiP*. The 2DID corresponds to the two-dimensional barcode etched on top of the module. v2.0.x

4.10.1 Set command

The set command reads the 2DID (SERIAL_NUMBER_A) of the nRF91x1 *SiP*. If the 2DID has not been written to the device, the response is an empty string.

Syntax:

```
%2DID
```

The following command example reads the 2DID when a 2DID exists on the device:

```
AT%2DID
%2DID: "F1413BQF072303"
OK
```

The following command example reads the 2DID when a 2DID does not exist on the device:

```
AT%2DID
%2DID: ""
OK
```

4.10.2 Read command

The read command is not supported.

4.10.3 Test command

The test command is not supported.

4.11 Read device UUID %DEVICEUUID

The proprietary **%DEVICEUUID** command reads the device-specific *UUID*. v2.0.x

4.11.1 Set command

The set command reads the 36-character hardware *UUID* from the device.

Syntax:

```
%DEVICEUUID
```

The following command example reads the device UUID:

```
AT%DEVICEUUID
%DEVICEUUID: 504d5632-3739-42fb-80c8-1411df972166
OK
```

4.11.2 Read command

The read command is not supported.

4.11.3 Test command

The test command is not supported.

5 Mobile termination control and status commands

Mobile termination control and status commands are used for mobile-terminated power and indicator handling. Commands are listed for accessing *Subscriber Identity Module (SIM)/Universal Integrated Circuit Card (UICC)* database records.

5.1 Functional mode +CFUN

The **+CFUN** command sets and reads the modem functional mode. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.2*.

5.1.1 Set command

The set command sets the functional mode to Minimum functionality, Normal, or Offline mode (Flight mode). There is a specific mode for Flight mode with *UICC* on. It is also possible to activate or deactivate LTE or *Global Navigation Satellite System (GNSS)* separately.

%XSYSTEMMODE should be used for enabling system modes. It is possible to activate enabled modes. The response to activating a **+CFUN** mode is `ERROR` if **%XSYSTEMMODE** is not properly set.

The response to changing to Normal mode can be `ERROR` if UICC activation has failed.

Setting the device to Minimum functionality or Offline mode might take some time if signaling with the network is needed.

The execution of **+CFUN** is blocked if the required production calibrations have not been executed. When this happens, the modem responds with the appropriate +CME ERROR code and returns the `%MDMEV: RF CALIBRATION NOT DONE` modem event. nRF9131

Syntax:

```
+CFUN=<fun>
```

+CME ERROR code

60 – System error. Hardware variant does not support the requested mode.

531 – Not allowed. AFC calibration (**%XRFTEST**) or TX self-calibration (**%TXSELFCAL**) or both have not been executed with *Production Test Image (PTI)* firmware. nRF9131

532 – Not allowed. Empty *IMEI* slot locking failed.

533 – Not allowed. Modem *Universal Asynchronous Receiver/Transmitter (UART)* locking failed.

The set command parameter and its defined values are the following:

<fun>

0 – Sets the device to minimum functionality. Disables both transmit and receive RF circuits and deactivates LTE and GNSS services.

> **Note:** A configuration made with an AT command can be stored to *NVM* using `+CFUN=0` if the option is mentioned in the description of the command. This is useful if the command relates to a permanent hardware configuration because it avoids the need to give the command every time in the beginning of the application.
>
> Active **`%XPOFWARN`** warning blocks the storing to NVM.
>
> `+CFUN=0` resets unsolicited notification requests.
>
> When setting the device to Minimum functionality mode, wait for `OK` to make sure that NVM has been updated.

> **CAUTION:** `+CFUN=0` causes writing to NVM. When using `+CFUN=0`, take NVM wear into account.

1 – Sets the device to full functionality. Active modes depend on the **`%XSYSTEMMODE`** setting.

2 – Sets the device to receive only functionality. Can be used, for example, to pre-evaluate connections with **`%CONEVAL`**.

4 – Sets the device to flight mode. Disables both transmit and receive RF circuits and deactivates LTE and GNSS services.

20 – Deactivates LTE without shutting down GNSS services.

21 – Activates LTE without changing GNSS.

30 – Deactivates GNSS without shutting down LTE services.

31 – Activates GNSS without changing LTE.

40 – Deactivates UICC.

41 – Activates UICC.

44 – Sets the device to flight mode without shutting down UICC.

If LTE is enabled with **`%XSYSTEMMODE`**, registration is performed to LTE network with `+CFUN=1` and `+CFUN=21`. If GNSS is enabled with **`%XSYSTEMMODE`**, GNSS is activated and GNSS services are available with `+CFUN=1`, `+CFUN=2`, and `+CFUN=31`.

> **Note:**
>
> - `+CFUN=41` is allowed only when *LTE-M* or *NB-IoT* is enabled by **`%XSYSTEMMODE`**.
> - UICC initialization is started in modes `+CFUN=1`, `+CFUN=21`, and `+CFUN=41`. **`%XSIM`** indications shall be followed for the UICC state.
> - Transition from modes `+CFUN=1` and `+CFUN=21` to mode `+CFUN=2` is not supported.
> - The response for `+CFUN=40` is `ERROR` if the modem has been activated with `+CFUN=1` or `+CFUN=21`.

The following command example activates the modem Normal mode:

```
AT+CFUN=1
OK
```

5.1.2 Read command

The read command reads the current functional mode.

Response syntax:

```
+CFUN: <fun>
```

The read response parameter and its defined value are the following:

<fun>

0 – Minimum functionality mode. RF circuits are disabled by deactivating LTE and *GNSS* services.

1 – Normal mode. The active mode is either LTE or GNSS, or both. Full functional mode. Active modes depend on **%XSYSTEMMODE** setting.

2 – Receive only mode in active system mode. Active modes depend on **%XSYSTEMMODE** setting.

4 – Flight mode. RF circuits are disabled by deactivating LTE and GNSS services.

21 – LTE is activated.

31 – GNSS is activated.

41 – UICC is activated.

The following command example reads the current functional mode:

```
AT+CFUN?
+CFUN: 1
OK
```

5.1.3 Test command

The test command lists supported functional modes.

Response syntax:

```
+CFUN: (list of supported <fun>s)
```

The response parameters and their defined values are the following:

<fun>

0 – Sets the device to minimum functionality. Disables both transmit and receive RF circuits and deactivates LTE and GNSS services.

> **Note:** A configuration made with an AT command can be stored to *NVM* using `+CFUN=0` if the option is mentioned in the description of the command. This is useful if the command relates to a permanent hardware configuration because it avoids the need to give the command every time in the beginning of the application.
>
> Active **`%XPOFWARN`** warning blocks the storing to NVM.
>
> `+CFUN=0` resets unsolicited notification requests.
>
> When setting the device to Minimum functionality mode, wait for `OK` to make sure that NVM has been updated.

> **CAUTION:** `+CFUN=0` causes writing to NVM. When using `+CFUN=0`, take NVM wear into account.

1 – Sets the device to full functionality. Active modes depend on the **`%XSYSTEMMODE`** setting.

2 – Sets the device to receive only functionality. Can be used, for example, to pre-evaluate connections with **`%CONEVAL`**.

4 – Sets the device to flight mode. Disables both transmit and receive RF circuits and deactivates LTE and GNSS services.

20 – Deactivates LTE without shutting down GNSS services.

21 – Activates LTE without changing GNSS.

30 – Deactivates GNSS without shutting down LTE services.

31 – Activates GNSS without changing LTE.

40 – Deactivates UICC.

41 – Activates UICC.

44 – Sets the device to flight mode without shutting down UICC.

The following command example returns the supported functional modes:

```
AT+CFUN=?
+CFUN: (0,1,4,20,21,30,31,40,41,44)
OK
```

5.2 List all available AT commands +CLAC

The **`+CLAC`** command returns a list of all available AT commands. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.37*.

5.2.1 Set command

The set command returns a list of all available AT commands.

Syntax:

```
+CLAC
```

Response syntax:

```
<AT Command1>[<CR><LF><AT Command2>[...]]
```

+CME ERROR code

518 – Not allowed in active state.

The following command example lists the supported AT commands:

```
AT+CLAC
AT+CFUN
AT+COPS
...
OK
```

5.2.2 Read command

The read command is not supported.

5.2.3 Test command

The test command is not supported.

5.3 Extended signal quality +CESQ

The **+CESQ** command returns received signal quality parameters. This command issues a valid response only when the modem is activated. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.69*.

> **Note:** When *NB-IoT* system mode is used and the device is in RRC CONNECTED state, old signal quality parameter values are reported. The values are recorded and reported from the previous RRC IDLE state.

5.3.1 Set command

The set command returns received signal quality parameters.

Syntax:

```
+CESQ
```

Response syntax:

```
+CESQ: <rxlev>,<ber>,<rscp>,<ecno>,<rsrq>,<rsrp>
```

The set command parameters and their defined values are the following:

<rxlev>

99 – Not known or not detectable

<ber>

99 – Not known or not detectable

<rscp>

255 – Not known or not detectable

<ecno>

255 – Not known or not detectable

<rsrq>

0 rsrq < −19.5 dB

1 – When −19.5 dB ≤ RSRQ < −19 dB

2 – When −19 dB ≤ RSRQ < −18.5 dB

...

32 – When −4 dB ≤ RSRQ < −3.5 dB

33 – When −3.5 dB ≤ RSRQ < −3 dB

34 – When −3 dB ≤ RSRQ

255 – Not known or not detectable

The index value of RSRQ can be converted to decibel with the following formula: Index x ½ – 19,5 = dB. For example, (32 x ½) – 19,5 = –3,5 dB.

<rsrp>

0 – RSRP < −140 dBm

1 – When −140 dBm ≤ RSRP < −139 dBm

2 – When −139 dBm ≤ RSRP < −138 dBm

...

95 – When −46 dBm ≤ RSRP < −45 dBm

96 – When −45 dBm ≤ RSRP < −44 dBm

97 – When −44 dBm ≤ RSRP

255 – Not known or not detectable

The index value of RSRP can be converted to decibel with the following formula: Index – 140 = dBm. For example, 95 – 140 = –45 dBm.

The following command example reads the current signal quality, mapped *Reference Signal Received Quality (RSRQ)* 31, and *Reference Signal Received Power (RSRP)* 62:

```
AT+CESQ
+CESQ: 99,99,255,255,31,62
OK
```

5.3.2 Read command

The read command is not supported.

5.3.3 Test command

The test command returns supported values as compound values.

Response syntax:

```
+CESQ: (list of supported <rxlev>s),(list of supported <ber>s),(list of supported <rscp>s),
(list of supported <ecno>s),(list of supported <rsrq>s),(list of supported <rsrp>s)
```

The following command example returns supported values as compound values:

```
AT+CESQ=?
+CESQ: (99),(99),(255),(255),(0-34,255),(0-97,255)
OK
```

5.4 Signal quality notification %CESQ

The proprietary **%CESQ** command subscribes notifications of changes in signal quality. v2.0.x

> **Note:** When *NB-IoT* system mode is used and the device is in RRC CONNECTED state, old signal quality parameter values are reported. The values are recorded and reported from the previous RRC IDLE state.

5.4.1 Set command

The set command subscribes notifications of changes in signal quality.

Syntax:

```
%CESQ=<n>
```

Notification syntax:

```
%CESQ: <rsrp>,<rsrp_threshold_index>,<rsrq>,<rsrq_treshold_index>
```

The set command parameters and their defined values are the following:

<n>

0 – Unsubscribe signal quality notifications

1 – Subscribe signal quality notifications

<rsrp>

0 – RSRP < −140 dBm

1 – When −140 dBm ≤ RSRP < −139 dBm

2 – When −139 dBm ≤ RSRP < −138 dBm

...

95 – When −46 dBm ≤ RSRP < −45 dBm

96 – When −45 dBm ≤ RSRP < −44 dBm

97 – When −44 dBm ≤ RSRP

255 – Not known or not detectable

The index value of RSRP can be converted to decibel with the following formula: Index – 140 = dBm. For example, 95 – 140 = –45 dBm.

<rsrp_threshold_index>

Index of *RSRP* threshold which is below measured RSRP value.

0 – RSRP is below the first threshold. 0–20 (–140 dBm to –121 dBm).

1 – RSRP is between the first and second threshold. 20–39 (–120 dBm to –101 dBm).

2 – RSPR is between the second and third threshold. 40–59 (–100 dBm to –81 dBm).

3 – RSRP is between the third and fourth threshold. 60–79 (–80 dBm to –61 dBm).

4 – RSRP is above the fourth threshold. 80–97 (–60 dBm to –44 dBm).

255 – Unknown.

With default thresholds 20, 40, 60, and 80, the measured value 70 leads to index 3.

<rsrq>

0 rsrq < −19.5 dB

1 – When −19.5 dB ≤ RSRQ < −19 dB

2 – When −19 dB ≤ RSRQ < −18.5 dB

...

32 – When −4 dB ≤ RSRQ < −3.5 dB

33 – When −3.5 dB ≤ RSRQ < −3 dB

34 – When −3 dB ≤ RSRQ

255 – Not known or not detectable

The index value of RSRQ can be converted to decibel with the following formula: Index x ½ – 19,5 = dB. For example, (32 x ½) – 19,5 = –3,5 dB.

<rsrq_threshold_index>

Index of RSRQ threshold which is below the measured RSRQ value.

0 – RSRQ is below the first threshold. 0–7 (–19.5 dB to –16.5 dB).

1 – RSRQ is between the first and second threshold. 7–13 (–16 dB to –13 dB).

2 – RSRQ is between the second and third threshold. 14–20 (–12.5 dB to –9.5 dB).

3 – RSRQ is between the third and fourth threshold. 21–27 (–9 dB to –6 dB).

4 – RSRQ is above the fourth threshold. 28–34 (–5.5 dB to –3 dB).

255 – Unknown

With the default thresholds 7, 14, 21, and 28, the measured value 17 leads to index 2.

The following command example subscribes *Evolved Universal Terrestrial Radio Access (E-UTRA)* signal quality notifications:

```
AT%CESQ=1
OK
```

The following notification example indicates a change in the measured average RSRP. The average RSRP is 62 and mapped to threshold 3, the measured RSRQ average has been 12 and mapped to threshold index 1.

```
%CESQ: 62,3,12,1
```

5.4.2 Read command

The read command is not supported.

5.4.3 Test command

The test command is not supported.

5.5 Signal quality +CSQ

The **+CSQ** command reads 2G and 3G signal quality. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.5*.

5.5.1 Set command

The set command is reads 2G and 3G signal quality.

Syntax:

```
+CSQ
```

Response syntax:

```
+CSQ: <rssi>,<ber>
```

Note: Not detectable, RAT not supported. Use **+CESQ** and **%CESQ** for *E-UTRA* signal quality.

The set command parameters and their value are the following:

<rssi>, <ber>

99 – Not detectable

The following command example reads signal quality:

```
AT+CSQ
+CSQ: 99,99
OK
```

5.5.2 Read command

The read command is not supported.

5.5.3 Test command

The test command lists supported signal quality values.

Response syntax:

```
+CSQ: (list of supported <rssi>s,(list of supported <ber>s)
```

The test command parameters and their defined value are the following:

<rssi>, <ber>

99 – Not detectable

The following command example lists the supported signal quality values:

```
AT+CSQ=?
+CSQ: (99),(99)
OK
```

5.6 SNR signal quality notification %XSNRSQ

The proprietary **%XSNRSQ** command subscribes notifications of changes in *Signal-to-Noise Ratio (SNR)* signal quality. v2.0.x

> **Note:** When *NB-IoT* system mode is used and the device is in RRC CONNECTED state, old signal quality parameter values are reported. The values are recorded and reported from the previous RRC IDLE state.

5.6.1 Set command

The set command subscribes notifications of changes in *SNR* signal quality.

Syntax:

```
%XSNRSQ=<n>
```

Notification syntax:

```
%XSNRSQ: <snr>,<threshold_index>,<srxlev>,<ce_level>
```

The parameters and their defined values are the following:

<n>

0 – Unsubscribe SNR signal quality notifications

1 – Subscribe SNR signal quality notifications

<snr>

0 – SNR < −24 dB

1 – When −24 dB ≤ SNR < −23 dB

2 – When −23 dB ≤ SNR < −22 dB

...

47 – When 22 dB ≤ SNR < 23 dB

48 – When 23 dB ≤ SNR < 24 dB

49 – When 24 dB ≤ SNR

127 – Not known or not detectable

The index value of SNR can be converted to decibel with the following formula: Index – 24 = dB. For example, 1 – 24 = –23 dB and 47 – 24 = 23 dB.

<threshold_index>

The index of the SNR threshold which is below the measured SNR value.

0 – SNR is below the first threshold

1 – SNR is between the first and second threshold

2 – SNR is between the second and third threshold

3 – SNR is between the third and fourth threshold

4 – SNR is above the fourth threshold

With default thresholds 16, 24, 32, and 40, the measured value 35 leads to index 3.

<srxlev>

0 – SRXLEV –127 or below –127

1 to 254 – SRXLEV –126 to 127

255 – SRXLEV above 127

32767 – Invalid or not know

<ce_level>

0 – CE Level 0. Normal coverage.

1 – CE Level 1. Enhanced coverage, according to LTE cell access criteria specified in *3GPP TS 36.304*.

255 – Invalid or not known

The following command example subscribes *E-UTRA* signal quality notifications:

```
AT%XSNRSQ=1
OK
```

The following notification example indicates that the measured average SNR has changed to 39 and is mapped to threshold 3:

```
%XSNRSQ: 39,3,130,1
```

5.6.2 Read command

The read command reads *SNR* signal quality.

Response syntax:

```
%XSNRSQ: <snr>,<srxlev>,<ce_level>
```

The read command parameters and their defined values are the following:

<snr>

0 – SNR < −24 dB

1 – When −24 dB ≤ SNR < −23 dB

2 – When −23 ≤ SNR < −22 dB

...

47 – When 22 ≤ SNR < 23 dB

48 – When 23 ≤ SNR < 24 dB

49– When 24 ≤ SNR

127 – Not known or not detectable

<srxlev>

0 – SRXLEV –127 or below –127

1 to 254 – SRXLEV –126 to 126

255 – SRXLEV 127 or above

32767 – Invalid or not know

<ce_level>

0 – CE Level 0. Normal coverage.

1 – CE Level 1. Enhanced coverage.

255 – Invalid or not known

The following command example reads SNR signal quality:

```
AT%XSNRSQ?
%XSNRSQ: 39,168,0
OK
```

5.6.3 Test command

The test command is not supported.

5.7 Device activity status +CPAS

The **+CPAS** command returns the device activity status. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.1*.

5.7.1 Set command

The set command returns the device activity status.

Syntax:

```
+CPAS
```

Response syntax:

```
+CPAS: <pas>
```

The set command parameter and its defined value are the following:

<pas>

Activity status.

0 – Ready (*Mobile Termination (MT)* allows commands from *Terminal Adapter (TA)/Terminal Equipment (TE)*)

The following command example checks the activity status:

```
AT+CPAS
+CPAS: 0
OK
```

5.7.2 Read command

The read command is not supported.

5.7.3 Test command

The test command is not supported.

5.8 Indicator control +CIND

The **+CIND** command sets indicator states. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.9*.

5.8.1 Set command

The set command sets indicator states.

Syntax:

```
+CIND=[<ind>[,<ind>[,...]]]
```

Response syntax:

```
+CIND: <descr>,<value>
```

The set command parameters and their defined values are the following:

<ind>

Integer. 0 – Off.

Other values are <descr>-specific.

"service": 1 – On

"roam": 1 – On

"message": 1 – On

<descr>

"service" – Service availability

"roam" – Roaming indicator

"message" – Message received

<value>

Integer. Values are <descr>-specific.

"service": 0 – Not registered, 1 – Registered

"roam": 0 – Not roaming, 1 – Roaming

"message": 1 – Message received

The following command example enables service and message indicators:

```
AT+CIND=1,0,1
OK
```

The following notification example indicates that the device is in service:

```
+CIND: "service",1
```

5.8.2 Read command

The read command returns indicator states.

Response syntax:

```
+CIND: <ind>[,<ind>[,...]]
```

The read command parameters and their defined values are the following:

<ind>

Integer. 0 – Off.

Other values are <descr>-specific.

"service": 1 – On

"roam": 1 – On

"message": 1 – On

<descr>

"service" – Service availability

"roam" – Roaming indicator

"message" – Message received

The following command example reads the indicator states:

```
AT+CIND?
+CIND: 1,0,1
OK
```

5.8.3 Test command

The test command returns supported indicator states.

Response syntax:

```
+CIND: (<descr>,(list of supported <ind>s))[,(<descr>,(list of supported <ind>s))[,...]]
```

The test command parameters and their defined values are the following:

<ind>

Integer. 0 – Off.

Other values are <descr>-specific.

"service": 1 – On

"roam": 1 – On

"message": 1 – On

<descr>

"service" – Service availability

"roam" – Roaming indicator

"message" – Message received

The following command example reads the indicator states:

```
AT+CIND=?
+CIND: ("service",(0,1)),("roam",(0,1)),("message",(0,1))
OK
```

5.9 IP address format +CGPIAF

The **+CGPIAF** command returns information about IPv6 address format. v2.0.x

For reference, see *3GPP 27.007 Ch 8.62*.

5.9.1 Set command

The set command is not supported.

5.9.2 Read command

The read command returns the IPv6 address format.

Response syntax:

```
+CGPIAF:
 <IPv6_AddressFormat>,<IPv6_SubnetNotation>,<IPv6_LeadingZeros>,<IPv6_CompressZeros>
```

The read command parameters and their defined values are the following:

<IPv6_AddressFormat>

1 – Use IPv6-like colon notation

<IPv6_SubnetNotation>

1 – Use / (forward slash) subnet prefix *Classless Inter-domain Routing (CIDR)* notation

<IPv6_LeadingZeros>

1 – Leading zeros are included

<IPv6_CompressZeros>

0 – No zero compression

The following command example reads the current IPv6 address format:

```
AT+CGPIAF?
+CGPIAF: 1,1,1,0
OK
```

5.9.3 Test command

The test command returns the supported IPv6 address formats.

Response syntax:

```
+CGPIAF: (list of supported <IPv6_AddressFormat>s),(list of supported
 <IPv6_SubnetNotation>s),(list of supported <IPv6_LeadingZeros>s),(list of supported
 <IPv6_CompressZeros>s)
```

The test command parameters and their defined values are the following:

<IPv6_AddressFormat>

1 – Use IPv6-like colon notation

<IPv6_SubnetNotation>

1 – Use / (forward slash) subnet prefix *CIDR* notation

<IPv6_LeadingZeros>

1 – Leading zeros are included

<IPv6_CompressZeros>

0 – No zero compression

The following command example reads the current IPv6 address format:

```
AT+CGPIAF=?
+CGPIAF: (1),(1),(1),(0)
OK
```

5.10 Current band %XCBAND

The proprietary **%XCBAND** command returns the current *E-UTRA* band. v2.0.x

5.10.1 Set command

The set command reads the current band. The command issues a valid response only when the modem is activated.

Syntax:

```
%XCBAND
```

Response syntax:

```
%XCBAND: <band>
```

The set command parameter and its defined values are the following:

<band>

Integer, 1–71. See *3GPP 36.101*.

0 when current band information not available

The following command example reads the current band:

```
AT%XCBAND
%XCBAND: 13
OK
```

5.10.2 Read command

The read command is not supported.

5.10.3 Test command

The test command returns a list of supported bands.

> **Note:** **%XBANDLOCK** usage has an impact on the list of supported bands.

Response syntax:

```
%XCBAND: (list of supported bands <band>)
```

The following command example returns a list of supported bands:

```
AT%XCBAND=?
%XCBAND: (1,2,3,4,12,13)
OK
```

5.11 Neighboring cells %NBRGRSRP

The proprietary **%NBRGRSRP** command reads measured *RSRP* values of neighboring cells. The command issues a valid response only when the modem is activated. v2.0.x

Neighboring cell measurements are valid and available only when neighbors are being monitored, which means that the strength and quality of the current cell signal do not meet the network configured level. For more information, see the requirements in *3GPP TS 36.304*.

To save energy, nRF91x1 does not search and measure neighboring cells for mobility purposes if the level and quality of the serving cell signal are above the thresholds defined by the network.

5.11.1 Set command

The set command reads measured *RSRP* values of neighboring cells.

Syntax:

```
%NBRGRSRP
```

Response syntax:

```
%NBRGRSRP: <phys_cellID>1,<EARFCN>1,<RSRP>1,<phys_cellID>2,
 <EARFCN>2,<RSRP>2,<phys_cellID>n, <EARFCN>n,<RSRP>n
```

The set command parameters and their defined values are the following:

<phys_cellID>

Integer. Physical cell ID.

<EARFCN>

Integer. *E-UTRA Absolute Radio Frequency Channel Number (EARFCN)* for a given cell where EARFCN is according to *3GPP TS 36.101*.

<rsrp>

0 – RSRP < −140 dBm

1 – When −140 dBm ≤ RSRP < −139 dBm

2 – When −139 dBm ≤ RSRP < −138 dBm

...

95 – When −46 dBm ≤ RSRP < −45 dBm

96 – When −45 dBm ≤ RSRP < −44 dBm

97 – When −44 dBm ≤ RSRP

255 – Not known or not detectable

The index value of RSRP can be converted to decibel with the following formula: Index – 140 = dBm. For example, 95 – 140 = –45 dBm.

5.11.2 Read command

The read command is not supported.

5.11.3 Test command

The test command is not supported.

5.12 Mode of operation (CS/PS) +CEMODE

The **+CEMODE** command sets the device mode of operation. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.28*.

5.12.1 Set command

The set command sets the *CS/PS Mode of Operation*. The **+CEMODE** command should be used only when the modem is not activated. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
+CEMODE=[<mode>]
```

The set command parameter and its defined values are the following:

<mode>

0 – PS mode 2 of operation

2 – CS/PS mode 2 of operation

The following command example sets the operating mode to PS mode 2:

```
AT+CEMODE=0
OK
```

5.12.2 Read command

The read command reads the current mode of operation.

Response syntax:

```
+CEMODE: <mode>
```

The read command parameter and its defined values are the following:

<mode>

0 – PS mode 2 of operation

2 – CS/PS mode 2 of operation

The following command example reads the current operating mode:

```
AT+CEMODE?
+CEMODE: 2
OK
```

5.12.3 Test command

The test command lists the supported modes of operation.

Response syntax:

```
+CEMODE: <mode>
```

The test command parameter and its defined values are the following:

<mode>

0 – PS mode 2 of operation

2 – CS/PS mode 2 of operation

The following command example returns the supported modes of operation:

```
AT+CEMODE=?
+CEMODE: (0,2)
OK
```

5.13 Band lock %XBANDLOCK

The proprietary **%XBANDLOCK** command sets locked bands. v2.0.x

Set the band lock before activating the modem with the **+CFUN** command.

5.13.1 Set command

The set command sets locked bands and bitmasks to limit supported bands.

Supported bands are masked with permanent and runtime masks. A logical `AND` operation is performed to **`%XBANDLOCK`** commands. If a permanent or runtime mask has been given, it is required that the second mask has at least one band in common with the first mask. Otherwise, the command returns `ERROR`. The command returns `ERROR` at an attempt to disable all supported bands.

Permanent mask is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

When permanently limiting the used bands with `%XBANDLOCK=3` and then performing TX self-calibration with `%TXSELFCAL=1`, execute `%XBANDLOCK=3` before `%TXSELFCAL=1`. nRF9131

> **Note:** Set band lock before activating modem with **`+CFUN`**.

Syntax:

```
%XBANDLOCK=<operation>[,<band_mask>]
```

+CME ERROR code

518 – Not allowed in active state.

The set command parameters and their defined values are the following:

<operation>

0 – Remove band locks.

1 – Set permanent band mask.

2 – Set runtime band mask.

3 – Set erase-protected band mask. Writing the mask is possible only in the *PTI* firmware. nRF9131

<band_mask>

String. Bit string where LSB is band 1. Leading zeros can be omitted. Maximum length 88 characters.

The following command example sets permanent band 4 lock:

```
AT%XBANDLOCK=1,"1000"
OK
```

The following command example sets runtime band 4 and 13 lock:

```
AT%XBANDLOCK=2,"1000000001000"
OK
```

5.13.2 Read command

The read command reads locked bands.

An empty string is returned if bandlock is not set.

Response syntax:

```
%XBANDLOCK: <permanent_lock>,<run_time_lock>,<erase_protected_lock>
```

The read response parameter and its defined value are the following:

<permanent_lock>

String. Bit string where LSB is band 1. Leading zeros can be omitted. Maximum length 88 characters.

<run_time_lock>

String. Bit string where LSB is band 1. Leading zeros can be omitted. Maximum length 88 characters.

<erase_protected_lock>

String. Bit string where LSB is band 1. Leading zeros can be omitted. Maximum length 88 characters.

The following command example reads the locked bands. No permanent lock, runtime lock for bands 13, 4, and 1. No erase-protected lock:

```
AT%XBANDLOCK?
%XBANDLOCK:
"",
"0000000000
```

An application accessing data on USIM or UICC through the AT command interface might lead to UICC activation if the UICC has been shut down to save power.

Note: Consider UICC NVM wear when setting power saving levels.

Syntax:

```
%XDATAPRFL=<power_level>
```

The set command parameter and its defined values are the following:

<power_level>

0 – Ultra-low power

1 – Low power

2 – Normal

3 – Performance

4 – High performance

The modem uses specific periodic search patterns for each power level. For details, see Periodic cell search configuration %PERIODICSEARCHCONF on page 164.

The following command example sets a low power level:

```
AT%XDATAPRFL=1
OK
```

5.14.2 Read command

The read command reads the application data profile.

Syntax:

```
%XDATAPRFL: <power_level>
```

The read command parameter and its defined values are the following:

<power_level>

0 – Ultra-low power

1 – Low power

2 – Normal

3 – Performance

4 – High performance

The following command example reads the power level:

```
AT%XDATAPRFL?
%XDATAPRFL: 2
OK
```

5.14.3 Test command

The test command is not supported.

5.15 Connectivity statistics %XCONNSTAT

The proprietary **%XCONNSTAT** command starts and stops the collecting of connectivity statistics. v2.0.x

> **Note:** Do not use **%XCONNSTAT** in applications which use the LwM2M carrier library. The application's use of **%XCONNSTAT** causes incorrect measurements in the operator's device management solution.

5.15.1 Set command

The set command starts and stops the collecting of connectivity statistics.

Syntax:

```
%XCONNSTAT=<command>
```

The set command parameter and its defined values are the following:

<command>

0 – Stop

1 – Start

The following command example makes the application start and stop connectivity statistics:

```
AT%XCONNSTAT=1
OK
AT%XCONNSTAT=0
OK
```

5.15.2 Read command

The read command reads the connectivity statistics.

Syntax:

```
%XCONNSTAT: <SMS_Tx>,<SMS_Rx>,<Data_Tx>,<Data_Rx>,<Packet_max>,<Packet_average>
```

The read command parameters and their defined values are the following:

<SMS_Tx>

Indicate the total number of SMSs successfully transmitted during the collection period

<SMS_Rx>

Indicate the total number of SMSs successfully received during the collection period

<Data_Tx>

Indicate the total amount of data (in kilobytes) transmitted during the collection period

<Data_Rx>

Indicate the total amount of data (in kilobytes) received during the collection period

<Packet_max>

The maximum packet size (in bytes) used during the collection period

<Packet_average>

The average packet size (in bytes) used during the collection period

The following command example makes the application read the connectivity statistics:

```
AT%XCONNSTAT?
%XCONNSTAT=2,3,45,60,708,650
OK
```

5.15.3 Test command

The test command is not supported.

5.16 Battery voltage %XVBAT

The proprietary **%XVBAT** command reads battery voltage. v2.0.x

When the modem is active (either LTE communication or *GNSS* receiver), the **%XVBAT** command returns the latest voltage measured automatically during modem wakeup or reception. The voltage measured during transmission is not reported. During modem inactivity, the modem measures battery voltage when the **%XVBAT** command is received.

> **Note:** Longer sleep modes, such as *eDRX* and *Power Saving Mode (PSM)*, are modem active time. Therefore, in those cases the **%XVBAT** value returned is from the time just before entering sleep or from previous GNSS reception during the eDRX/PSM gap.

5.16.1 Set command

The set command reads the battery voltage in mV.

Syntax:

```
%XVBAT
```

Response syntax:

```
+XVBAT: <vbat>
```

The response parameter is the following:

<vbat>

Integer. Battery voltage in mV, with a resolution of 4 mV.

The following command example reads the battery voltage and the response is for a successful case:

```
AT%XVBAT
%XVBAT: 3600
OK
```

5.16.2 Read command

The read command is not supported.

5.16.3 Test command

The test command is not supported.

5.17 Battery voltage low level notification %XVBATLVL

The proprietary **%XVBATLVL** command subscribes to unsolicited battery voltage low level notifications. v2.0.x

The notification is sent when the battery voltage level is under the currently set low level. The voltage is the latest voltage measured automatically during wakeup or reception.

The battery voltage low level is set using the **%XVBATLOWLVL** command.

5.17.1 Set command

The set command subscribes unsolicited notifications of battery voltage low level. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XVBATLVL=<n>
```

Notification syntax:

```
%XVBATLOWLVL: <battery_voltage>
```

The set command parameter and its defined values are the following:

<n>

0 – Unsubscribe unsolicited notifications of battery voltage low level

1 – Subscribe unsolicited notifications of battery voltage low level

The notification parameter and its defined values are the following:

<battery_voltage>

Integer, 3000–5000 mV.

0 – No valid battery voltage available

The following command example subscribes unsolicited notifications of battery voltage low level:

```
AT%XVBATLVL=1
OK
```

The following notification example indicates that the battery voltage level is under the currently set battery voltage low level when the level has been set to 3750 mV:

```
%XVBATLOWLVL: 3700
```

5.17.2 Read command

The read command is not supported.

5.17.3 Test command

The test command is not supported.

5.18 Battery voltage low level %XVBATLOWLVL

The proprietary **%XVBATLOWLVL** command sets the battery voltage low level for a modem. If notifications of battery voltage low level have been subscribed to, the modem sends clients a notification when the measured battery voltage is below the defined level. The modem reads sensors periodically in connected mode. The default period is 60 seconds. If the temperature or voltage gets close to the set threshold, a shorter period is used. v2.0.x

The notifications are subscribed using the **%XVBATLVL** command.

5.18.1 Set command

The set command sets the battery voltage low level for a modem. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XVBATLOWLVL=<battery_level>
```

The set command parameter and its defined value are the following:

<battery_level>

Integer, 3100–5000 mV. Factory default 3300 mV.

The following command example sets the battery voltage low level to 3500 mV:

```
AT%XVBATLOWLVL=3500
OK
```

5.18.2 Read command

The read command reads the battery voltage low level from a modem.

Response syntax:

```
%XVBATLOWLVL?
```

The following command example reads the current value of the battery voltage low level:

```
AT%XVBATLOWLVL?
%XVBATLOWLVL: 3500
OK
```

5.18.3 Test command

The test command is not supported.

5.19 External power off warnings %XPOFWARN

The proprietary **%XPOFWARN** command controls the power off warnings from the modem. v2.0.x

%XPOFWARN is based on voltage level detection without any delay. When this warning has been sent once, it needs to be enabled again by sending `AT%XPOFWARN=1,<voltage>`.

%XPOFWARN is enabled by default at the 3000 mV level. Active **%XPOFWARN** warning blocks the storing to NVM.

For a detailed description of the nRF91x1 power supply monitoring functionality, see Power supply monitoring in nRF9131 Product Specification and nRF9161 Product Specification.

> **CAUTION:** Configuring and enabling the external power failure warning described in External power failure warning in nRF9131 Product Specification and nRF9161 Product Specification is supported by modem firmware.

5.19.1 Set command

The set command configures the power off warnings from the modem. The warning is received as an unsolicited AT indication. A hardware indication event is also sent to the application domain.

When the power off warning is sent, the modem sets itself to Offline mode and sends `+CGEV: ME BATTERY LOW` and `%MDMEV: ME BATTERY LOW` in modem firmware that supports **%MDMEV**.

To receive unsolicited **+CGEV** and **%MDMEV** notifications, the application must first request and enable them.

The application is responsible for detecting possible increase in the battery voltage level and for restarting the LTE protocol activities. This can be detected by issuing the `%XVBAT` query. If the level is acceptable again (>3000 mV), the application can proceed with `+CFUN=1`. For more information, see nRF9131 Product Specification and nRF9161 Product Specification.

Syntax:

```
%XPOFWARN=<state>[,<voltage>]
```

The set command parameters and their defined values are the following:

<state>

0 – Disable power off warnings

1 – Enable power off warnings

<voltage>

The voltage level when the power off warning is sent. Mandatory when enabling power off warnings. Optional when disabling power off warnings.

30 – 3000 mV

31 – 3100 mV

32 – 3200 mV

33 – 3300 mV

The following command example enables the power off warning in 3000 mV:

```
AT%XPOFWARN=1,30
OK
```

5.19.2 Read command

The read command is not supported.

5.19.3 Test command

The test command is not supported.

5.20 Customer production done %XPRODDONE

The proprietary **%XPRODDONE** command is sent after customer production is complete. v2.0.x

5.20.1 Set command

The set command disables R&D features by closing the modem *UART* connection, which is meant only for nRF91x1 *SiP* manufacturing test purposes. The command allows to permanently disable the use of the **%XEMPR** command. The command also permanently enables the downgrade prevention of modem firmware.

Downgrade prevention means that it is not possible to update an older modem firmware version to a device. Downgrade prevention applies to both *Firmware-Over-The-Air (FOTA)* and wired firmware updates.

If the <value> parameter is not set or is 0, the **%XEMPR** command is permanently disabled after the **%XPRODDONE** command has been performed. To keep the command usable after **%XPRODDONE**, use `AT %XPRODDONE=1`. If the command has been disabled with **%XPRODDONE**, it cannot be re-enabled.

The **%XEMPR** TX power reduction feature can be used with the PTI modem firmware even if it has been disabled with **%XPRODDONE**.

CAUTION: Downgrade prevention cannot be disabled once it has been enabled.

CAUTION: If **%XPRODDONE** is set on a device manufactured with CIoT firmware, *DECT NR+* firmware cannot be updated on the device. If **%XPRODDONE** is set on a device manufactured with DECT NR+ firmware, CIoT firmware cannot be updated on the device.

Syntax:

```
%XPRODDONE=[<value>]
```

The set command parameter and its defined values are the following:

<value>

0 – Permanently disable **%XEMPR**.

1 – Leave **%XEMPR** usable.

The following command example sets the customer production to done and permanently disables the **%XEMPR** command:

```
AT%XPRODDONE
OK
```

or

```
AT%XPRODDONE=0
OK
```

The following command example sets customer production to done and leaves **%XEMPR** usable:

```
AT%XPRODDONE=1
OK
```

5.20.2 Read command

The read command is not supported.

5.20.3 Test command

The test command is not supported.

5.21 Internal temperature notification %XTEMP

The proprietary **%XTEMP** command subscribes unsolicited internal temperature notifications. The modem reads sensors periodically in connected mode. The default period is 60 seconds. If the temperature or voltage gets close to the critical temperature, a shorter period is used. v2.0.x

5.21.1 Set command

The set command subscribes unsolicited internal temperature notifications.

A notification is sent when the temperature is rising above a high or critical temperature level or cooling down from a critical or high temperature level.

Syntax:

```
%XTEMP=<n>
```

Notification syntax:

```
%XTEMP: <temperature_level>,<temperature>
```

The set command parameter and its defined values are the following:

<n>

0 – Unsubscribe unsolicited temperature indications

1 – Subscribe unsolicited temperature indications

The notification parameters and their defined values are the following:

<temperature_level>

1 – Normal temperature

2 – High temperature. Factory default 55. This can be changed with **%XTEMPHIGHLVL**.

3 – Critical temperature. TX/RX disabled. Factory default 90.

<temperature>

Integer. Celsius degrees between −40 and 125.

The following command example subscribes notifications:

```
AT%XTEMP=1
OK
```

The following example shows an unsolicited notification for an internal temperature level:

```
%XTEMP: 1,37
%XTEMP: 2,56
%XTEMP: 3,91
```

5.21.2 Read command

The read command reads the internal temperature level and the temperature.

Syntax:

```
%XTEMP?
```

Response syntax:

```
%XTEMP: <temperature>
```

The following command example reads the current modem temperature:

```
AT%XTEMP?
%XTEMP: 50
OK
```

5.21.3 Test command

The test command is not supported.

5.22 High level for internal temperature %XTEMPHIGHLVL

The proprietary **%XTEMPHIGHLVL** command sets the high level to internal temperature in the modem. v2.0.x

5.22.1 Set command

The set command sets the high internal temperature level for the notification in the **%XTEMP** command.

When the high temperature level is reached, data transmission should be controlled and minimized to prevent modem overheating. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XTEMPHIGHLVL=<temperature>
```

The set command parameter and its defined value are the following:

<temperature>

Integer.

Celsius degrees between 1 and 85. Factory default 55.

Celsius degrees between 1 and 90. Factory default 55.

The following command example sets the high temperature level:

```
AT%XTEMPHIGHLVL=60
OK
```

5.22.2 Read command

The read command reads the internal high temperature level of a modem.

When a high temperature level is reached, data transmission should be controlled and minimized to prevent modem overheating.

Syntax:

```
%XTEMPHIGHLVL?
```

The following command example reads the current internal high temperature level:

```
AT%XTEMPHIGHLVL?
%XTEMPHIGHLVL: 60
OK
```

5.22.3 Test command

The test command is not supported.

5.23 Clock +CCLK

The **+CCLK** command sets the clock of the device. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.15*.

5.23.1 Set command

The set command sets the real-time clock of the *User Equipment (UE)*.

Syntax:

```
+CCLK=<time>
```

The set command parameter and its defined value are the following:

<time>

String. Current time in the format "yy/MM/dd,hh:mm:ss±zz", where the characters, from left to right, indicate year, month, day, hour, minutes, seconds, and time zone. Time zone indicates the difference, expressed in quarters of an hour, between the local time and GMT (value range from −48 to +48).

The following command example sets the real-time clock:

```
AT+CCLK="18/12/06,22:10:00+08"
OK
```

5.23.2 Read command

The read command reads the real-time clock.

Response syntax:

```
+CCLK: <time>
```

If time is not received or set with the **+CCLK** command, the response is `ERROR`.

> **Note:** The device clock updates are based on network time when available. The time can be requested using the read command, but not all networks provide the information, nor can the highest accuracy requirements be guaranteed, either.

The read response parameters and their defined values are the following:

<time>

String. Current time in the format "yy/MM/dd,hh:mm:ss±zz", where the characters, from left to right, indicate year, month, day, hour, minutes, seconds, and time zone. Time zone indicates the difference, expressed in quarters of an hour, between the local time and GMT (value range from −48 to +48).

The following command example reads the real-time clock:

```
AT+CCLK?
+CCLK: "18/12/06,22:10:00+08"
OK
```

5.23.3 Test command

The test command is not supported.

5.24 Proprietary clock %CCLK

The proprietary **%CCLK** command sets the real-time clock of the device. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.15*.

5.24.1 Set command

The set command sets the current time and daylight saving time of the *UE*.

Syntax:

```
%CCLK=<time>,<daylight_saving_time>
```

The set command parameters and their defined values are the following:

<time>

String. Current time in the format "yy/MM/dd,hh:mm:ss±zz", where the characters, from left to right, indicate year, month, day, hour, minutes, seconds, and time zone. Time zone indicates the difference, expressed in quarters of an hour, between the local time and GMT (value range from −48 to +48 and 99 for "not set" or "unknown").

<daylight_saving_time>

0 – No adjustment of daylight saving time

1 – +1 h adjustment of daylight saving time

2 – +2 h adjustment of daylight saving time

The following command example sets the real-time clock:

```
AT%CCLK="02/05/07,14:08:17+00",2
OK
```

5.24.2 Read command

The read command reads the current time and daylight saving time.

Response syntax:

```
%CCLK: <time>[,<daylight_saving_time>]
```

If time is not received or set with the **%CCLK** command, the response is ERROR.

> **Note:** The device clock updates are based on network time when available. The time can be requested using the read command, but not all networks provide the information, nor can the highest accuracy requirements be guaranteed, either.

The read command parameters and their defined values are the following:

<time>

String. Current time in the format "yy/MM/dd,hh:mm:ss±zz", where the characters, from left to right, indicate year, month, day, hour, minutes, seconds, and time zone. Time zone indicates the difference, expressed in quarters of an hour, between the local time and GMT (value range from −48 to +48).

<daylight_saving_time>

Optional. Present if received from the network or if the user has set it in **%CCLK**.

0 – No adjustment of daylight saving time

1 – +1 h adjustment of daylight saving time

2 – +2 h adjustment of daylight saving time

The following command example reads the current date, time, and daylight saving time:

```
AT%CCLK?
%CCLK: "02/05/07,14:08:17+00",2
OK
```

5.24.3 Test command

The test command is not supported.

5.25 Modem trace activation %XMODEMTRACE

The proprietary **%XMODEMTRACE** command activates modem traces. The trace data is in binary format and can help the Nordic customer support to analyze and resolve issues. v2.0.x

Traces can be captured using Trace Collector in the nRF Connect toolset.

5.25.1 Set command

The set command activates and deactivates modem trace. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

Syntax:

```
%XMODEMTRACE=<oper>[,<set_id>[,<bitmap_id>,<bitmap>]]
```

Response syntax for Read trace bitmap:

```
%XMODEMTRACE: <bitmap>
```

The set command parameters and their defined values are the following:

<oper>

Operation

0 – Deactivate traces.

1 – Activate predefined trace set.

2 – Activate trace bitmap. To be used only on request by Nordic customer support.

3 – Read trace bitmap. To be used only on request by Nordic customer support.

<set_id>

Integer. Predefined trace set identifier.

1 – Coredump only.

2 – *Long-Term Evolution (LTE)*, *Internet Protocol (IP)*, *GNSS*, and coredump.

3 – Reserved for future use.

4 – IP only.

5 – LTE and IP.

<bitmap_id>

Integer. Trace bitmap identifier. Used only with the assistance of Nordic customer support.

<bitmap>

String in hexadecimal *International Reference Alphabet (IRA)* format. Used only with the assistance of Nordic customer support.

The following command example activates trace set 1 (Coredump only):

```
AT%XMODEMTRACE=1,1
OK
```

The following command example deactivates trace:

```
AT%XMODEMTRACE=0
OK
```

5.25.2 Read command

The read command is not supported.

5.25.3 Test command

The test command is not supported.

5.26 Fallback to SMS only %XSMSFALLBACK

The proprietary **%XSMSFALLBACK** command sets the SMS only fallback functionality. With SMS only fallback, *UE* immediately triggers a *Tracking Area Update (TAU)* request for SMS only when CS service registration fails with permanent cause. This ensures that SMS services are available as soon as possible after registration. SMS only and SMS only fallback are available only in *NB-IoT*. v2.0.x

5.26.1 Set command

The set command enables and disables immediate SMS only fallback in *NB-IoT* if CS services are permanently unavailable via combined procedures.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XSMSFALLBACK=<fallback_status>
```

The set command parameter and its defined values are the following:

<fallback_status>

0 – Fallback is not performed

1 – Fallback is performed

The following command example sets SMS fallback in NB-IoT:

```
AT%XSMSFALLBACK=1
OK
```

5.26.2 Read command

The read command is not supported.

5.26.3 Test command

The test command is not supported.

5.27 System mode %XSYSTEMMODE

The proprietary **%XSYSTEMMODE** command sets the modem system mode. v2.0.x

5.27.1 Set command

The set command sets the supported system modes of the modem. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

If the modem detects a conflicting band lock and operator restrictions in at least one system, this means that there are no usable bands in that system and the command returns `ERROR` and +CME ERROR code 522.

Note:

- Only one supported LTE mode is allowed at a time.
- Both the *LTE-M* and *NB-IoT* system modes can be configured and enabled.
- This command is allowed only before activating the modem using the `+CFUN=1` command. If the mode needs to be changed, enter Minimum functionality mode using the `+CFUN=0` command or set it to flight mode using the `+CFUN=4` command.
- Operator configurations can override values set by the user.

Syntax:

```
%XSYSTEMMODE=<LTE_M_support>,<NB_IoT_support>,<GNSS_support>,<LTE_preference>
```

+CME ERROR code

518 – Not allowed in active state.

522 – Band configuration not valid for selected mode.

The set command parameters and their defined values are the following:

<LTE_M_support>

0 – *LTE-M* not supported

1 – LTE-M supported

<NB_IoT_support>

0 – *NB-IoT* not supported

1 – NB-IoT supported

<GNSS_support>

0 – *GNSS* not supported

1 – GNSS supported

<LTE_preference>

0 – No preference. Initial system selection is based on history data and *Universal Subscriber Identity Module (USIM)*. If history data or USIM configuration are not available, LTE-M is prioritized in the initial system selection.

1 – LTE-M preferred.

2 – NB-IoT preferred.

3 – Network selection priorities override system priority, but if the same network or equal priority networks are found, LTE-M is preferred.

4 – Network selection priorities override system priority, but if the same network or equal priority networks are found, NB-IoT is preferred.

> **Note:** If <LTE_preference> is set to a non-zero value, <LTE_M_support> or <NB_IoT_support> or both must be set.

The following command example sets LTE-M and GNSS as the system modes. No preferred LTE mode set:

```
AT%XSYSTEMMODE=1,0,1,0
OK
```

5.27.2 Read command

The read command reads the supported modem system modes.

Response syntax:

```
%XSYSTEMMODE: <LTE_M_support>,<NB_IoT_support>,<GNSS_support>,<LTE_preference>
```

The read response parameters and their defined values are the following:

<LTE_M_support>

0 – *LTE-M* not supported

1 – LTE-M supported

<NB_IoT_support>

0 – *NB-IoT* not supported

1 – NB-IoT supported

<GNSS_support>

0 – *GNSS* not supported

1 – GNSS supported

<LTE_preference>

0 – No preference. Initial system selection is based on history data and *USIM*. If history data or USIM configuration are not available, LTE-M is prioritized in the initial system selection.

1 – LTE-M preferred.

2 – NB-IoT preferred.

3 – Network selection priorities override system priority, but if the same network or equal priority networks are found, LTE-M is preferred.

4 – Network selection priorities override system priority, but if the same network or equal priority networks are found, NB-IoT is preferred.

> **Note:** If <LTE_preference> is set to a non-zero value, <LTE_M_support> or <NB_IoT_support> or both must be set.

The following command example reads the supported system mode:

```
AT%XSYSTEMMODE?
%XSYSTEMMODE: 1,0,0,0
OK
```

5.27.3 Test command

The test command is not supported.

5.28 PTW setting %XPTW

The proprietary **%XPTW** command sets the *Paging Time Window (PTW)*. v2.0.x

5.28.1 Set command

The set command sets the requested *Paging Time Window (PTW)* parameters. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

> **Note:** Use the command with caution. The requested values must be compliant with the *eDRX* cycle values configured using the **+CEDRXS** command. The modem uses the configured value in eDRX cycle/PTW length negotiation with the network when eDRX is enabled using the **+CEDRXS** command.

When eDRX parameters are changed using the **+CEDRXS** command, the PTW value is set as default. The default values can vary between different modem firmware releases. If the application needs to use a certain value, it must be set. If other than the default PTW has to be used, the **%XPTW** command shall be sent after the **+CEDRXS** command.

Syntax:

```
%XPTW=<AcT-type>[,<Requested_ptw_value>]
```

The set command parameters and their defined values are the following:

<AcT-type>

4 – *Evolved Terrestrial Radio Access Network (E-UTRAN)* (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

<Requested_ptw_value>

String. Half a byte in 4-bit format. The PTW value refers to bits from 8 to 5 of octet 3 of the eDRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*). Optional. If not present, the value of the requested AcT-type is reset to the manufacturer-specific default.

LTE-M mode

Bit

4 3 2 1 – PTW length

0 0 0 0 – 1.28 s

0 0 0 1 – 2.56 s

0 0 1 0 – 3.84 s

0 0 1 1 – 5.12 s

0 1 0 0 – 6.4 s

0 1 0 1 – 7.68 s

0 1 1 0 – 8.96 s

0 1 1 1 – 10.24 s

1 0 0 0 – 11.52 s

1 0 0 1 – 12.8 s

1 0 1 0 – 14.08 s

1 0 1 1 – 15.36 s

1 1 0 0 – 16.64 s

1 1 0 1 – 17.92 s

1 1 1 0 – 19.20 s

1 1 1 1 – 20.48 s

NB-IoT mode

Bit

4 3 2 1 – PTW length

0 0 0 0 – 2.56 s

0 0 0 1 – 5.12 s

0 0 1 0 – 7.68 s

0 0 1 1 – 10.24 s

0 1 0 0 – 12.8 s

0 1 0 1 – 15.36 s

0 1 1 0 – 17.92 s

0 1 1 1 – 20.48 s

1 0 0 0 – 23.04 s

1 0 0 1 – 25.6 s

1 0 1 0 – 28.16 s

1 0 1 1 – 30.72 s

1 1 0 0 – 33.28 s

1 1 0 1 – 35.84 s

1 1 1 0 – 38.4 s

1 1 1 1 – 40.96 s

The following command example sets the requested PTW value:

```
AT%XPTW=4,"1000"
OK
```

5.28.2 Read command

The read command reads the requested *Paging Time Window (PTW)* parameters.

Response syntax:

```
%XPTW: <AcT-type>,<Requested_ptw_value>
```

The read response parameters and their defined values are the following:

<AcT-type>

4 – *E-UTRAN* (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

<Requested_ptw_value>

String. Half a byte in 4-bit format. The PTW value refers to bits from 8 to 5 of octet 3 of the *eDRX* parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

The following command example reads the requested PTW value(s):

```
AT%XPTW?
%XPTW: 4,"0110"
%XPTW: 5,"1110"
OK
```

Note:

- If the device supports many access technologies, each access technology is included in a separate line as illustrated in the example above.
- The negotiated PTW value can be checked with the **+CEDRXRDP** command.

5.28.3 Test command

The test command is not supported.

5.29 Extra maximum TX power reduction %XEMPR

The proprietary **%XEMPR** command allows to configure an extra reduction of 0.5 or 1 dB to the maximum transmission power on all or selected supported 3GPP bands separately in the *NB-IoT* and *LTE-M* modes.
v2.0.x

> **Note:**
> - The use of this command in the nRF91x1 modem firmware image can be permanently prevented with the **%XPRODDONE** command.
> - The **%XEMPR** TX power reduction feature can be used in the *PTI* modem firmware image even if the **%XEMPR** feature has been disabled with **%XPRODDONE**.

5.29.1 Set command

The set command sets the extra reduction to maximum TX power.

The command can be given separately to the *NB-IoT* and *LTE-M* modes. If a band is not mentioned in the command, the EMPR is 0 for that band. The command cannot be used to increase transmission power. **%XEMPR** should be given before the activation of the modem to be effective. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the +CFUN=0 command. The configuration can be stored to NVM with the **%XFSSYNC** command when the *PTI* modem firmware is used. If a valid EMPR configuration exists, the reduction is automatically applied to the applicable transmissions.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XEMPR=<system_mode>,<k>,<band0>,<pr0>,<band1>,<pr1>,…,<bandk-1>,<prk-1>
```

or

```
%XEMPR=<system_mode>,0,<pr_for_all_bands>
```

The set command parameters and their defined values are the following:

<system_mode>

0 – NB-IoT

1 – LTE-M

<k>

The number of bands to which EMPR is set. If <k> is 0, the next parameter <pr_for_all_bands> is applied to all supported 3GPP bands. The **%XEMPR** command supports listing the power reduction for all the supported bands of nRF91x1 in one command for both NB-IoT and LTE-M.

<bandn>

The number of the 3GPP band to which the following <prn> is applied.

<prn>

EMPR for <bandn>

0 – 0 dB

1 – Maximum power reduced 0.5 dB

2 – Maximum power reduced 1.0 dB

> 2 is not allowed

The following command example reduces the maximum TX power on all bands in the NB-IoT mode by 1 dB:

```
AT%XEMPR=0,0,2
OK
```

The following command example reduces the maximum TX power on three (`<k>=3`) bands in the LTE-M mode (`<system_mode>=1`). The maximum TX power is reduced by 1 dB on bands 5 and 8 and by 0.5 dB on band 13:

```
AT%XEMPR=1,3,5,2,8,2,13,1
OK
```

The following command example deletes the existing configuration by sending the command without any parameters:

```
AT%XEMPR
OK
```

5.29.2 Read command

The read command reads the currently active configuration.

Syntax:

```
%XEMPR?
```

The following command example reads the currently active configuration after both examples of the set command have been given:

```
AT%XEMPR?
%XEMPR:
0,0,2
1,3,5,2,8,2,13,1
OK
```

5.29.3 Test command

The test command is not supported.

5.30 Write content to file %XFILEWRITE

The proprietary **`%XFILEWRITE`** command writes given content to a file. v2.0.x

5.30.1 Set command

The set command requests to write content to a file.

Active **%XPOFWARN** warning prevents writing to NVM.

Syntax:

```
%XFILEWRITE=<file>,<content>[,<checksum>]
```

+CME ERROR code

518 – Not allowed in active state.

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<file>

1 – *GNSS* almanac.

Up-to-date content of the GNSS almanac file can be found in the GNSS sample applications of nRF Connect SDK.

<content>

Hexadecimal numbers containing two *IRA* characters per octet.

<checksum>

Checksum of the content calculated over the sha256 algorithm. Mandatory for GNSS almanac file.

The following command example requests to write the GNSS almanac to a file:

```
AT%XFILEWRITE=1,
 "f0ea0200312a080000000031...","bf38c845eab79f459f7b3ef4393f1a2860d309952832a0073b990f12a7274e64"
OK
```

5.30.2 Read command

The read command is not supported.

5.30.3 Test command

The test command is not supported.

5.31 Coverage enhancement mode notification +CEINFO

The **+CEINFO** command subscribes Coverage Enhancement (CE) notifications. v2.0.x

5.31.1 Set command

The set command subscribes Coverage Enhancement (CE) notifications and reads current parameters.

Syntax:

```
+CEINFO=<reporting>
```

Notification syntax:

```
+CEINFO: <reporting>,<CE_Enabled>,<UE_State>,<Downlink_Repetition_Factor>,
<Uplink_Repetition_Factor>,<RSRP>,<CINR>
```

<UE_State> is not between double quotation marks.

+CME ERROR code

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<reporting>

0 – Unsubscribe unsolicited notifications

1 – Subscribe unsolicited notifications

<CE_Enabled>

0 – Serving cell does not support CE mode A/B

1 – Serving cell supports CE mode A/B

<UE_State>

UE state at the time of the report.[1]

I – Idle

R – RACH

C – Connected

<Downlink_Repetition_Factor>

Downlink repetition factor.

If <UE_State> is Idle or RACH, it is set to mpdcch-NumRepetition according to the current radio condition (that is, *RSRP*) and prach-ParametersListCE-r13 in SIB2 if the access technology is *LTE-M*. It is set to npdcch-NumRepetitions according to the current radio condition and NPRACH-Parameters-NB-r13 in SIB2-NB if the access technology is *NB-IoT*.

If <UE_State> is Connected, it is set to mpdcch-NumRepetition for the radio bearer if the access technology is LTE-M. It is set to npdcch-NumRepetitions for the radio bearer if the access technology is *NB-IoT*.

<Uplink_Repetition_Factor>

Uplink repetition factor.

If <UE_State> is Idle, it is set to numRepetitionPerPreambleAttempt accordig to the current radio condition.

If <UE_State> is RACH, it is set to numRepetitionPerPreambleAttempt selected by UE.

If <UE_State> is Connected, it is set to repetition number for PUSCH if the access technology is LTE-M. It is set to repetition number for NPUSCH if the access technology is NB-IoT.

[1] The value corresponds to the last value that has been stored before the time of report. For example, when releasing an RRC connection, the transition from connection to idle state and the next <UE_State> being stored and reported can take several seconds.

<RSRP>

Current RSRP level at the time of the report.[1] Numerical range in dBm.

255 – Not known or not detectable.

<CINR>

Current CINR level at the time of the report[1]. Numerical range in dB.

127 – Not known or not detectable.

The following command example subscribes unsolicited CE notifications:

```
AT+CEINFO=1
OK
```

The following is an example of an unsolicited CE notification:

```
+CEINFO: 1,1,C,5,3,-50,10
```

5.31.2 Read command

The read command requests Coverage Enhancement (CE) mode information.

Response syntax:

```
+CEINFO: <Reporting>,<CE_Enabled>,<UE_State>,<Downlink_Repetition_Factor>,
<Uplink_Repetition_Factor>,<RSRP>,<CINR>
```

The following command example reads CE mode information:

```
AT+CEINFO?
+CEINFO: 1,1,C,5,3,-50,10
OK
```

5.31.3 Test command

The test command is not supported.

5.32 Factory reset %XFACTORYRESET

The proprietary **%XFACTORYRESET** command resets the modem to factory settings. This operation is allowed only when the modem is not activated. v2.0.x

5.32.1 Set command

The set command resets the modem to factory settings. *NVM* is updated with this command. `+CFUN=0` is not needed to trigger the update.

Active **%XPOFWARN** warning blocks the NVM factory reset.

Syntax:

```
%XFACTORYRESET=<reset_type>
```

+CME ERROR code

518 – Not allowed in active state.

528 – Not allowed when power off warning is active.

The set command parameter and its defined values are the following:

<reset_type>

0 – Reset all modem data to factory settings

1 – Reset user-configurable data to factory settings

The following command example resets all modem data to factory settings:

```
AT%XFACTORYRESET=0
```

The following command example resets user-configurable data to factory settings:

```
AT%XFACTORYRESET=1
```

5.32.2 Read command

The read command is not supported.

5.32.3 Test command

The test command is not supported.

5.33 Modem sleep notification %XMODEMSLEEP

The proprietary **%XMODEMSLEEP** command subscribes modem sleep notifications. Based on the **%XMODEMSLEEP** indications, the application can optimize its power efficiency by synchronizing its operations with the sleep and wake-up events of the modem and the radio activity and inactivity. v2.0.x

The application can use modem sleep notifications to get information about inactive periods in modem radio usage and to get a prewarning at a desired time before radio activity happens. The modem API works normally, and radio inactivity is interrupted if a command requires the use of the radio or the modem is requested to send data.

When the modem is inactive, certain hardware components can be deactivated, and hardware resources, such as the antenna or battery, can be used for other purposes. Before the predicted end of sleep or before requesting an operation that requires the use of the modem's radio resources, the application must activate the components that are needed for LTE connections if they have been switched off to save power during modem inactivity.

Some LTE connection use cases are not known on the application domain. For example, an operator applet on *UICC* can request the modem to set up a data connection. If the LTE connection use case is not known, the modem sends a sleep interrupted notification and has a hardcoded 500 ms delay before it proceeds to set up the connection. If an application has switched off some components, it must handle the notification promptly and switch on the needed components during the delay.

After subscribing to the sleep notifications, the first notification is sent when the modem goes to sleep, unless the modem is in a deactivated state, in which case the application is notified immediately.

> **Note:** There are no sleep notifications if only *GNSS* is activated with the **%XSYSTEMMODE** command.

For more information on the use of the **%XMODEMSLEEP** command, see Modem Sleep Notifications.

5.33.1 Set command

The set command subscribes and unsubscribes all types of modem sleep notifications. The notification returns the type of sleep. The application can use the warning time to trigger an action before modem wakeup. The <threshold> parameter is used to indicate the length of the sleep periods the application is interested in.

Syntax:

```
%XMODEMSLEEP=<n>[,<warning_time>,<threshold>]
```

The set command parameters and their defined values are the following:

<n>

0 – Unsubscribe modem sleep notifications

1 – Subscribe modem sleep notifications

<warning_time>

Advance warning time in ms. A notification is sent as a prewarning of modem wakeup.

Range 500–3600000 ms.

Value `0` can be used if advance warning is not needed.

The <warning_time> parameter is required when <n> is `1`.

<threshold>

The shortest sleep time indicated to application in ms.

Affects sleep start indications for types 1, 2, 3, 6, and 7.

Range 10240–3456000000 ms.

The <threshold> parameter is required when <n> is `1`.

Notification syntax:

```
%XMODEMSLEEP: <type>[,<time>]
```

The notification parameters and their defined values are the following:

<type>

Type of modem sleep

1 – *PSM*. Modem is in power saving mode.

2 – RF inactivity. Modem sleeps due to RF inactivity, including *eDRX*.

3 – Limited service. Modem sleeps due to being out of coverage.

4 – Flight mode. Modem has been set to flight mode.

5 – Sleep is interrupted. Indicated sleep is interrupted.

6 – Sleep is resumed. Indicated sleep has been resumed.

7 – Proprietary PSM.

<time>

Sleep time in ms. If the <time> parameter is not present, sleep time is considered infinite. If the <time> parameter is `0`, sleep mode is exited. The value might exceed the 32-bit maximum value.

<type> 4 is typically indicated as infinite because the expected sleep time is not known.

<type> 6 is not indicated if less than the <threshold> time is left before wakeup.

The following command example subscribes modem sleep notifications. The shortest indicated sleep time is 60 seconds, and a notification is sent 1 second before the end of sleep:

```
AT%XMODEMSLEEP=1,1000,60000
```

The following notification example indicates PSM duration of 320 hours:

```
%XMODEMSLEEP: 1,1152000000
```

5.33.2 Read command

The read command is not supported.

5.33.3 Test command

The test command is not supported.

5.34 Periodic TAU notification %XT3412

The proprietary **`%XT3412`** command subscribes periodic *TAU* notifications. Based on XT3412 indications, the application can optimize its power efficiency by synchronizing its data transmissions with periodic TAU procedures. v2.0.x

For more information on the use of the **`%XT3412`** command, see Modem Sleep Notifications.

5.34.1 Set command

The set command subscribes and unsubscribes periodic *TAU* notifications.

Syntax:

```
%XT3412=<n>[,<warning_time>,<threshold>]
```

The set command parameters and their defined values are the following:

<n>

0 – Unsubscribe periodic TAU notifications

1 – Subscribe periodic TAU notifications

<warning_time>

Advance warning time in ms.

Range 500–3600000 ms.

A notification is sent as a prewarning of periodic TAU.

0 – No prewarning

The <warning_time> parameter is required when <n> is 1.

<threshold>

Shortest periodic TAU time indicated to application in ms.

Range 10240–3456000000 ms.

The <threshold> parameter is required when <n> is `1`.

Notification syntax:

```
%XT3412: <time>
```

The notification parameter and its defined value are the following:

<time>

Remaining timer T3412 time in ms. If the <time> parameter is `0`, timer T3412 has expired. The value might exceed the 32-bit maximum value.

The following command example subscribes periodic TAU notifications. The shortest indicated period is 30 seconds, and a notification is sent 2 seconds before TAU:

```
AT%XT3412=1,2000,30000
```

The following notification example indicates T3412 duration of 20 minutes:

```
%XT3412: 1200000
```

5.34.2 Read command

The read command is not supported.

5.34.3 Test command

The test command is not supported.

5.35 Country-specific search optimization %XCOUNTRYDATA

The proprietary **`%XCOUNTRYDATA`** command can be used to configure country-specific search parameters. The list contains access technology, band, and *EARFCN* to be used. The maximum number of countries is 70. v2.0.x

The information inserted with the **`%XCOUNTRYDATA`** command can speed up network selection in a location that is new to the device. The modem uses the information first during network selection when moving to a new area, but performs normal full searches if they are needed to find service. This feature does not break the 3GPP network selection rules defined in *3GPP TS 23.122*. The network that is found based on this information must be possible to select according to the 3GPP network selection rules.

This operation is allowed only when the modem is not activated.

5.35.1 Set command

The set command sets country-specific data. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

Syntax:

```
%XCOUNTRYDATA=<oper>[,<country_1>][,<country_2>] ... [,<country_70>]
```

where <country_x> is used only for <oper> `1` and contains <act>,<mcc>,<band>,<earfcn>[,band_2,earfcn_2]..[,band_5,earfcn_5].

+CME ERROR code

513 – Not found.

518 – Not allowed in active state.

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<oper>

0 – Delete stored country data

1 – Store country data

<act>

4 – *E-UTRAN* (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

<mcc>

Integer. *Mobile Country Code (MCC)*s defined by ITU-T.

<band>

Integer, 1–71. See *3GPP 36.101*.

<earfcn>

Integer. *EARFCN* as defined in *3GPP TS 36.101*.

0 – EARFCN not specified

The following command example writes two country elements. In the first country element, <act> is *LTE-M*, <mcc> is 244, <band> is 3, and <earfcn> is 1207. In the second country element, <act> is *NB-IoT*, <mcc> is 212, <band> is 20, and <earfcn> is 6150, <band> is 13, without specific <earfcn>:

```
AT%XCOUNTRYDATA=1,”4,244,3,1207”, “5,242,20,6175,13,0”
OK
```

The following command example deletes existing country data from the modem:

```
AT%XCOUNTRYDATA=0
OK
```

5.35.2 Read command

The read command reads a list of countries that contains access technology, band, and *EARFCN*.

Syntax:

```
%XCOUNTRYDATA: [<country_1>][,<country_2>] ... [,<country_70>]
```

The following command example reads the country data:

```
AT%XCOUNTRYDATA?
%XCOUNTRYDATA: "4,244,3,1207", "5,242,20,6175,13,5180"
```

5.35.3 Test command

The test command is not supported.

5.36 Modem domain event notification %MDMEV

The proprietary **%MDMEV** command subscribes modem domain event notifications. v2.0.x

5.36.1 Set command

The set command subscribes modem domain event notifications. The unsolicited result code is `%MDMEV: XXX`.

Syntax:

```
%MDMEV=<n>
```

The set command parameter and its defined values are the following:

<n>

0 – Unsubscribe modem domain event notifications

1 – Subscribe modem domain events notifications

2 – Subscribe modem domain warning and informational events

The unsolicited notification syntax descriptions are as follows:

Mobile Equipment (ME) is overheated and therefore modem is deactivated:

```
%MDMEV: ME OVERHEATED
```

Battery voltage is low and therefore modem is deactivated:

```
%MDMEV: ME BATTERY LOW
```

Modem indicates status of ongoing light search:

```
%MDMEV: SEARCH STATUS 1
```

`SEARCH STATUS 1` indicates light search performed. This event gives the application a chance to stop the modem from using more power on searching networks from possibly weaker radio condition. Before sending this event, the modem searches the cells based on previous cell history, measures the radio conditions, and makes assumptions on where networks might be deployed. This event means that the modem has not found a network that it could select based on the 3GPP network selection rules

from those locations. It does not mean that there are no networks to be found in the area. The modem continues more thorough searches automatically after sending this status. The modem can be deactivated, for example, with `+CFUN=4`, to stop it from using more power on the search.

Modem indicates completed status of search:

```
%MDMEV: SEARCH STATUS 2
```

`SEARCH STATUS 2` indicates search performed. The modem has found a network that it can select according to the 3GPP network selection rules or all frequencies have been scanned and a suitable cell has not been found. In the latter case, the modem enters normal limited service state functionality and performs scans for service periodically.

Modem has detected a reset loop:

```
%MDMEV: RESET LOOP
```

`RESET LOOP` indicates that the modem restricts Attach attempts for the next 30 minutes. The timer does not run when the modem has no power or while it stays in the reset loop. The modem counts all the resets where the modem is not gracefully deinitialized with `+CFUN=0`.

If there are more than five resets, reset loop restriction is activated.

If there are more than seven resets, reset loop restriction is activated.

For more information on reset loop, see Modem Reset Loop Restriction.

Modem has detected that *IMEI* has not been written with the *PTI* firmware:

```
%MDMEV: NO IMEI
```

Mandatory AFC calibration (**`%XRFTEST`**) or TX self-calibration (**`%TXSELFCAL`**) or both have not been executed with the PTI firmware. :

```
%MDMEV: RF CALIBRATION NOT DONE
```

Modem indicates the selected CE level for RACH procedure:

```
%MDMEV: PRACH CE-LEVEL <ce-level>
```

The notification parameter and its defined values are the following:

<ce-level>

0 – CE Level 0. No repetitions or small number of repetitions.

1 – CE Level 1. Medium number of repetitions.

2 – CE Level 2. Large number of repetitions. *NB-IoT* only.

3 – CE Level 3. Very large number of repetitions (not supported).

Modem detects a conflicting band lock and operator restrictions in at least one system which indicates that there are no usable bands in that system:

```
%MDMEV: INVALID BAND CONFIGURATION <status_emtc> <status_nbiot>
```

The notification parameters and their defined values are the following:

<status_emtc>

0 – Usable LTE bands are available in the system.

1 – There are no usable LTE bands available in the system.

2 – System support is not configured.

<status_nbiot>

0 – Usable LTE bands are available in the system.

1 – There are no usable LTE bands available in the system.

2 – System support is not configured.

The following command example subscribes **%MDMEV** notifications:

```
AT%MDMEV=1
OK
```

5.36.2 Read command

The read command is not supported.

5.36.3 Test command

The test command is not supported.

5.37 Neighboring cell measurement and notification %NCELLMEAS

The proprietary **%NCELLMEAS** command starts cell search and measurements and reports the channel parameters with a result notification when the measurements are finished. The results can be used, for example, in cellular-based positioning services. v2.0.x

When using search types 0–2, the results contain parameters from the serving cell and optionally up to 17 neighboring cells. Neighboring cell measurements are available during a data or signaling RRC connection only when the network has configured measurement rules for the device and the configured criteria is met. If neighboring cell measurements have been active before the connection, measurement results from RRC IDLE state are available in RRC CONNECTED state for 10 s after the last measurement in RRC IDLE state has passed.

With search types 0–2, the measurements in RRC IDLE state are based on generic configurations from the network and radio conditions. Usually, measurements are not performed in RRC IDLE state during good radio conditions unless they are requested with the **%NCELLMEAS** command. If the network configures RRC CONNECTED state measurements, measurement results from RRC IDLE state are cleared before the 10 s has passed.

When using search types 3–5, Global Cell ID (GCI) can be used to acquire cell ID, *Public Land Mobile Network (PLMN)*, and *TAC* for the surrounding *Enhanced Machine Type Communication (eMTC)* or *NB-IoT* cells limited by the <gci_count> parameter. The GCI searches can require a substantial amount of time, especially with high <gci_count> values.

With all search types, search time can be limited by using the **%NCELLMEASSTOP** command to stop the measurements and return the results collected by the time when the measurements are stopped.

5.37.1 Set command

The set command starts neighboring cell measurements and reports the channel parameters.

Syntax:

```
%NCELLMEAS[=<search_type>][,<gci_count>]
```

Notification syntax for <search_type> 0–2:

```
%NCELLMEAS: <status>
[,<cell_id>,<plmn>,<tac>,<timing_advance>,<earfcn>,<phys_cell_id>,<rsrp>,<rsrq>,
<measurement_time>,]
[,<n_earfcn>1,<n_phys_cell_id>1,<n_rsrp>1,<n_rsrq>1,<time_diff>1]
[,<n_earfcn>2,<n_phys_cell_id>2,<n_rsrp>2,<n_rsrq>2,<time_diff>2]
...
[,<n_earfcn>17,<n_phys_cell_id>17,<n_rsrp>17,<n_rsrq>17,<time_diff>17]
[,<timing_advance_measurement_time>]
```

Notification syntax for <search_type> 3–5:

```
%NCELLMEAS: <status>
[,<cell_id>,<plmn>,<tac>,<timing_advance>,<timing_advance_measurement_time>,<earfcn>,
<phys_cell_id>,<rsrp>,<rsrq>,<measurement_time>,<serving>,<neighbor_count>
[,<n_earfcn>1,<n_phys_cell_id>1,<n_rsrp>1,<n_rsrq>1,<time_diff>1]
[,<n_earfcn>2,<n_phys_cell_id>2,<n_rsrp>2,<n_rsrq>2,<time_diff>2]...],
[<cell_id>,<plmn>,<tac>,<timing_advance>,<timing_advance_measurement_time>,<earfcn>,
<phys_cell_id>,<rsrp>,<rsrq>,<measurement_time>,<serving>,<neighbor_count>
[,<n_earfcn>1,<n_phys_cell_id>1,<n_rsrp>1,<n_rsrq>1,<time_diff>1]
[,<n_earfcn>2,<n_phys_cell_id>2,<n_rsrp>2,<n_rsrq>2,<time_diff>2]...]...
```

Note: The optional part is included in the response only when <status> is `0` (measurement successful).

The result notification for <search_type>s 3–5 contains cell ID, *PLMN*, and *TAC* for up to <gci_count> cells, and optionally a list of neighboring cell measurement results related to each cell. The <neighbor_count> indicates how many neighboring cells are present after each cell.

For <search_type>s 3–5, <neighbor_count> is always `0` for other cells except the serving cell, which means that the neighboring cell list is not available. If neighbor cell results for the serving cell are available when the *UE* is in RRC IDLE state, they are preserved for 10 s after entering RRC CONNECTED state unless they are overwritten by RRC CONNECTED state measurements. If there are no preserved RRC IDLE state measurements in RRC CONNECTED state, neighbor cell results are available only if the network has configured neighbor measurements for the UE. Valid *TA* information is available only for the serving cell. These parameters are included for all cells in the result format to keep the same structure for every cell to simplify response parsing.

If there is currently a serving cell, it is indicated by <serving> parameter value `1` and listed as the first cell in the result notification.

+CME ERROR code

517 – Modem not activated.

530 – Measurement ongoing. Wait for result notification or stop ongoing measurement by using **%NCELLMEASSTOP** before starting a new measurement.

The notification parameters and their defined values are the following:

<search_type>

0 – Modem searches RPLMN based on previous cell history.

1 – Modem starts with the same search method as in <search_type> `0`.

If a suitable cell is not found, the modem performs a light search where it continues the search by measuring the radio conditions and makes assumptions on where networks might be deployed. If RPLMN is not found based on previous cell history, the modem accepts any found PLMN.

2 – Modem uses otherwise the same search method as in <search_type> 1, except that it performs a complete search for all supported bands.

3 – GCI search. Modem searches *EARFCN*s based on previous cell history.

4 – GCI search. Modem starts with the same search method as in <search_type> 3. If less than <gci_count> cells are found, modem performs light search on bands that are valid for the area of the current ITU-T region.

5 – GCI search. Modem starts with the same search method as in <search_type> 3. If less than <gci_count> cells are found, modem performs complete search on all supported bands.

The modem uses the <search_type> parameter only when cell search is required to complete the requested measurement.

> **Note:** Supported bands might be limited by band lock or operator requirements for bands.

<gci_count>

Integer, 2 – 15.

Maximum number of cells to be searched. Mandatory for GCI search types 3–5. Ignored with search types 0–2.

<status>

0 – Measurement successful.

1 – Measurement failed.

2 – Measurement interrupted. Results can be incomplete.

<cell_id>

String in hexadecimal format. The cell ID the UE is camped on. 4-byte *E-UTRAN* cell ID.

<plmn>

String. *MCC* and *Mobile Network Code (MNC)* values.

<tac>

String in hexadecimal format. 2-byte *Tracking Area Code (TAC)*.

<timing_advance>

Integer. Timing advance value (Ts). Time units as specified in *3GPP TS 36.211*.

0–20512 – When timing advance is valid.

65535 – When timing advance is not valid.

> **Note:** Timing advance might be reported from past measurements. The client application is responsible for checking the <timing_advance_measurement_time> parameter and the serving cell's <measurement_time> parameter for the difference between the measurement times.

<earfcn>

Integer. EARFCN of the cell where the EARFCN is as defined in *3GPP TS 36.101*.

<measurement_time>

Measurement time of the cell in ms calculated from modem boot time.

Range 0–18 446 744 073 709 551 614 ms.

<n_earfcn>

Integer. *EARFCN* of the neighboring cell where the EARFCN is as defined in *3GPP TS 36.101*.

<phys_cell_id>

Integer. Physical cell ID of the cell.

<n_phys_cell_id>

Integer. Physical cell ID of the neighboring cell.

<rsrp>

RSRP of the cell.

–17 – When RSRP < –156 dBm.

–16 – When –156 ≤ RSRP < –155 dBm.

… … …

–3 – When –143 ≤ RSRP < –142 dBm.

–2 – When –142 ≤ RSRP < –141 dBm.

–1 – When –141 ≤ RSRP < –140 dBm.

0 – When RSRP < –140 dBm.

1 – When –140 ≤ RSRP < –139 dBm.

2 – When –139 ≤ RSRP < –138 dBm.

… … …

95 – When –46 ≤ RSRP < –45 dBm.

96 – When –45 ≤ RSRP < –44 dBm.

97 – When –44 ≤ RSRP dBm.

255 – Not known or not detectable

The index value of RSRP can be converted to decibel with the following formula: Index – 140 = dBm. For example, 95 – 140 = –45 dBm and –16 – 140 = –156 dBm.

<n_rsrp>

RSRP of the neighboring cell.

–17 – When RSRP < –156 dBm.

–16 – When –156 ≤ RSRP < –155 dBm.

… … …

–3 – When –143 ≤ RSRP < –142 dBm.

–2 – When –142 ≤ RSRP < –141 dBm.

–1 – When –141 ≤ RSRP < –140 dBm.

0 – When RSRP < –140 dBm.

1 – When –140 ≤ RSRP < –139 dBm.

2 – When –139 ≤ RSRP < –138 dBm.

… … …

95 – When –46 ≤ RSRP < –45 dBm.

96 – When –45 ≤ RSRP < –44 dBm.

97 – When –44 ≤ RSRP dBm.

255 – Not known or not detectable.

<rsrq>

RSRQ of the cell.

–30 – When RSRQ < –34 dB.

–29 – When –34 ≤ RSRQ < –33.5 dB.

… … …

–2 – When –20.5 ≤ RSRQ < –20 dB.

–1 – When –20 ≤ RSRQ < –19.5 dB.

0 – When RSRQ < –19.5 dB.

1 – When –19.5 ≤ RSRQ < –19 dB.

2 – When –19 ≤ RSRQ < –18.5 dB.

… … …

32 – When –4 ≤ RSRQ < –3.5 dB.

33 – When –3.5 ≤ RSRQ < –3 dB.

34 – When –3 ≤ RSRQ dB.

35 – When –3 ≤ RSRQ < –2.5 dB.

36 – When –2.5 ≤ RSRQ < –2 dB.

… … …

45 – When 2 ≤ RSRQ < 2.5 dB.

46 – When 2.5 ≤ RSRQ dB.

255 – Not known or not detectable.

The index value of RSRQ can be converted to decibel with the following formula: Index x ½ – 19,5 = dB. For example, (32 x ½) – 19,5 = –4 dB and (–29 x ½) – 19,5 = –34 dB.

<n_rsrq>

RSRQ of the neighboring cell.

–30 – When RSRQ < –34 dB.

–29 – When –34 ≤ RSRQ < –33.5 dB.

… … …

–2 – When –20.5 ≤ RSRQ < –20 dB.

–1 – When –20 ≤ RSRQ < –19.5 dB.

0 – When RSRQ < –19.5 dB.

1 – When –19.5 ≤ RSRQ < –19 dB.

2 – When –19 ≤ RSRQ < –18.5 dB.

… … …

32 – When –4 ≤ RSRQ < –3.5 dB.

33 – When –3.5 ≤ RSRQ < –3 dB.

34 – When –3 ≤ RSRQ dB.

35 – When –3 ≤ RSRQ < –2.5 dB.

36 – When –2.5 ≤ RSRQ < –2 dB.

… … …

45 – When 2 ≤ RSRQ < 2.5 dB.

46 – When 2.5 ≤ RSRQ dB.

255 – Not known or not detectable.

<time_diff>

Difference in ms between the latest performed serving cell measurement and neighboring cell measurement.

–99999 ms < time_diff < 99999 ms.

0 – Value not valid

<timing_advance_measurement_time>

0 – 18 446 744 073 709 551 614 ms.

Measurement time of timing advance in milliseconds calculated from modem boot time.

<serving>

0 – Current cell is not serving cell.

1 – Current cell is serving cell.

<neighbor_count>

Integer. Number of neighboring cell results after current cell. Can be 0.

The following command example starts neighboring cell measurement with default <search_type> 0:

```
AT%NCELLMEAS
OK
```

The following notification example indicates a finished measurement. The result contains the serving cell and two neighboring cells:

```
%NCELLMEAS: 0,"00011B07","26295","00B7",10512,2300,7,63,31,150344527,
 2300,8,60,29,0,2400,11,55,26,184,9034
```

The following command example starts a GCI search based on cell history with <search_type> 3 searching two cells:

```
AT%NCELLMEAS=3,2
OK
```

The following notification example indicates that the UE is in RRC CONNECTED state. The result notification contains serving cell with GCI information and measurement results of four neighboring cells:

```
%NCELLMEAS:  0,"00011B07","26295","00B7",10512,9034,2300,7,63,31,150344527,1,4,2300,
8,60,29,92,2300,9,59,28,100,2400,10,56,27,162,2400,11,55,26,184
```

The following notification example indicates that the UE is in RRC IDLE state. The result notification contains two cells with GCI information and no neighboring cell measurement results:

```
%NCELLMEAS:  0,"00011B07","26295","00B7",10512,9034,2300,7,63,31,150344527,
1,0,"00011B08","26295","00B7",65535,0,2300,9,62,30,150345527,0,0
```

5.37.2 Read command

The read command is not supported.

5.37.3 Test command

The test command is not supported.

5.38 Stop neighboring cell measurement %NCELLMEASSTOP

The proprietary **%NCELLMEASSTOP** command stops neighboring cell measurement. v2.0.x

5.38.1 Set command

The set command stops neighboring cell measurement if measuring has started, and causes the modem to send a response to the **%NCELLMEAS** command.

If the modem has already received the measurement results, the **%NCELLMEAS** response contains the normal response parameters. If the measurement is still ongoing and can be stopped, the response does not contain any measurement results.

After the **%NCELLMEAS** response, OK is sent as a response to stop the command.

If the measuring has not started, the response is OK.

Syntax:

```
%NCELLMEASSTOP
```

The following command example starts neighboring cell measurement and stops the measuring successfully:

```
AT%NCELLMEAS
OK
AT%NCELLMEASSTOP
%NCELLMEAS:  1
OK
```

The following command example starts neighboring cell measurement and tries to stop the measurement, but the measurement has been done before the stop command is given. The response can contain, for example, the serving cell and two neighboring cells:

```
AT%NCELLMEAS
OK
AT%NCELLMEASSTOP
%NCELLMEAS:0,"00011B07","26295","00B7",2300,7,63,31,2300,8,60,29,0,2400,11,55,26,0
OK
```

The following command example stops neighboring cell measurement when the neighboring cell measurement command **%NCELLMEAS** has been given but the modem has not started the measurement due to other prioritized activities:

```
AT%NCELLMEAS
OK
AT%NCELLMEASSTOP
OK
```

5.38.2 Read command

The read command is not supported.

5.38.3 Test command

The test command is not supported.

5.39 Evaluating connection parameters %CONEVAL

The proprietary **%CONEVAL** command requests an evaluation of the LTE radio signal state and an estimation of the energy efficiency in a cell that the modem would use for data transmission. The information can be used to determine the energy efficiency of data transmission before actual data transmission is started. The **%CONEVAL** command is the most useful for a static or slow-moving device. v2.0.x

The evaluation is based on a cell that would likely be used for RRC connection by the current state. The cell used for RRC connection can be different than the one that is used for the evaluation. Changing radio conditions and mobility procedures can affect the LTE cell selection.

The connection evaluation is possible in **+CFUN** modes 1, 2, and 21. The cell selection and evaluation are based on information in the latest detected *UICC* and collected cell history.

Note:

- The accuracy of the evaluation result can be affected by factors, such as network configuration, current modem and RRC state, signal quality, and movement of the device.
- Coverage Enhancement (CE) levels 0–2 are supported in NB-IoT. CE levels 0–1 (CE mode A) are supported in LTE-M. CE level 3 is not supported in the current modem firmware version.
- In **+CFUN** mode 2, evaluation is possible only if cell history is available.

For more information on the **%CONEVAL** command, see Energy Estimate.

5.39.1 Set command

The set command requests evaluation of connection parameters.

Syntax:

```
%CONEVAL
```

Response syntax:

```
%CONEVAL: <result>[,<rrc_state>,<energy_estimate>,<rsrp>,<rsrq>,<snr>,<cell_id>,<plmn>,
<phyc_cell_id>,<earfcn>,<band>,<tau_triggered>,<ce_level>,<tx_power>,<tx_repetitions>,
<rx_repetitions>,<dl-pathloss>]
```

The response parameters and their defined values are the following:

<result>

0 – Connection pre-evaluation successful

1 – Evaluation failed, no cell available

2 – Evaluation failed, *UICC* not available

3 – Evaluation failed, only barred cells available

4 – Evaluation failed, busy (for example, *GNSS* activity)

5 – Evaluation failed, aborted because of higher priority operation

6 – Evaluation failed, not registered

7 – Evaluation failed, unspecified

<rrc_state>

0 – RRC connection in RRC IDLE state during measurements

1 – RRC connection in RRC CONNECTED state during measurements

<energy_estimate>

Relative estimated energy consumption of data transmission compared to nominal consumption. A higher value means smaller energy consumption.

5 – Bad conditions. Difficulties in setting up connections. Maximum number of repetitions might be needed for data.

6 – Poor conditions. Setting up a connection might require retries and a higher number of repetitions for data.

7 – Normal conditions for cIoT device. No repetitions for data or only a few repetitions in the worst case.

8 – Good conditions. Possibly very good conditions for small amounts of data.

9 – Excellent conditions. Efficient data transfer estimated also for larger amounts of data.

<rsrp>

Current *RSRP* level at the time of the report

–17 – When RSRP < –156 dBm

–16 – When –156 ≤ RSRP < –155 dBm

… … …

–3 – When –143 ≤ RSRP < –142 dBm

–2 – When –142 ≤ RSRP < –141 dBm

–1 – When –141 ≤ RSRP < –140 dBm

0 – When RSRP < –140 dBm

1 – When –140 ≤ RSRP < –139 dBm

2 – When –139 ≤ RSRP < –138 dBm

… … …

95 – When –46 ≤ RSRP < –45 dBm

96 – When –45 ≤ RSRP < –44 dBm

97 – When –44 ≤ RSRP dBm

255 – Not known or not detectable

The index value of RSRP can be converted to decibel with the following formula: Index – 140 = dBm. For example, 95 – 140 = –45 dBm and –16 – 140 = –156 dBm.

<rsrq>

Current *RSRQ* level at the time of the report

–30 – When RSRQ < –34 dB

–29 – When –34 ≤ RSRQ < –33.5 dB

… … …

–2 – When –20.5 ≤ RSRQ < –20 dB

–1 – When –20 ≤ RSRQ < –19.5 dB

0 – When RSRQ < –19.5 dB

1 – When –19.5 ≤ RSRQ < –19 dB

2 – When –19 ≤ RSRQ < –18.5 dB

… … …

32 – When –4 ≤ RSRQ < –3.5 dB

33 – When –3.5 ≤ RSRQ < –3 dB

34 – When –3 ≤ RSRQ dB

35 – When –3 ≤ RSRQ < –2.5 dB

36 – When –2.5 ≤ RSRQ < –2 dB

… … …

45 – When 2 ≤ RSRQ < 2.5 dB

46 – When 2.5 ≤ RSRQ dB

255 – Not known or not detectable

The index value of RSRQ can be converted to decibel with the following formula: Index x ½ – 19,5 = dB. For example, (32 x ½) – 19,5 = –4 dB and (–29 x ½) – 19,5 = –34 dB.

<snr>

Current *SNR* level at the time of the report

0 SNR < -24 dB

1 – When –24 dB <= SNR < –23 dB

2 – When –23 dB <= SNR < –22 dB

: : : :

snr

47 – When 22 dB <= SNR < 23 dB

48 – When 23 dB <= SNR < 24 dB

49 – When 24 dB <= SNR

127 – Not known or not detectable

The index value of SNR can be converted to decibel with the following formula: Index – 24 = dB. For example, 1 – 24 = –23 dB and 47 – 24 = 23 dB.

<cell_id>

String in hexadecimal format. 4-byte *E-UTRAN* cell ID.

<plmn>

String. *MCC* and *MNC* values.

<phys_cell_id>

Integer, 0–503. Physical cell ID of evaluated LTE cell.

<earfcn>

Integer. EARFCN of a cell where the EARFCN is as defined in *3GPP TS 36.101*.

<band>

Integer, 1–88. See *3GPP 36.101*.

0 when current band information is not available.

<tau_triggered>

0 – *UE* is registered on the evaluated cell. Therefore, *TAU* is not needed before sending user data.

1 – *UE* is not registered on the evaluated cell. Therefore, the TAU procedure is initiated before sending user data.

255 – Not known.

<ce_level>

0 – CE Level 0. No repetitions or small number of repetitions.

1 – CE Level 1. Medium number of repetitions.

2 – CE Level 2. Large number of repetitions. *NB-IoT* only.

3 – CE Level 3. Very large number of repetitions (not supported).

255 – Invalid or not known.

If <rrc_state> is `0`, the CE level is estimated based on RSRP and network configuration.

If <rrc_state> is `1`, the CE level corresponds to the currently used CE level.

<tx_power>

Integer, –45 to +23 dBm.

The estimate of TX power depends on the RRC state in which the measurement is taken.

If <rrc_state> is `0`, the value corresponds to the estimated TX power level of the first preamble transmission in (N)PRACH.

If <rrc_state> is `1`, the value corresponds to the latest physical data channel (N)PUSCH transmission power level.

See *3GPP 36.101* and *3GPP 36.213*.

<tx_repetitions>

Integer, 1–2048.

The estimate of TX repetitions depends on the RRC state in which the measurement is taken.

If <rrc_state> is `0`, the value corresponds to the initial preamble repetition level in (N)PRACH based on <ce_level> and network configuration.

If <rrc_state> is `1`, the value corresponds to the latest physical data channel (N)PUSCH transmission repetition level.

See *3GPP TS 36.331* and *3GPP TS 36.213*.

<rx_repetitions>

Integer, 1–2048.

The estimate of RX repetitions depends on the RRC state in which the measurement is taken.

If <rrc_state> is `0`, Initial Random Access control channel (M/NPDCCH) reception repetition level based on <ce_level> and network configuration.

If <rrc_state> is `1`, the latest physical data channel (N)PDSCH reception repetition level.

See *3GPP TS 36.331* and *3GPP TS 36.213*.

<dl-pathloss>

Integer. Reduction in power density in dB.

The following command example requests the pre-evaluation of a connection:

```
AT%CONEVAL
%CONEVAL: 0,1,5,8,2,14,"011B0780”,"26295",7,1575,3,1,1,23,16,32,130
OK
```

5.39.2 Read command

The read command is not supported.

5.39.3 Test command

The test command is not supported.

5.40 Disabling SMS services in NB-IoT %SMSDISABLE

The proprietary **`%SMSDISABLE`** command disables SMS services in *NB-IoT*. v2.0.x

Before using the command, set the device to PS mode 2 with `+CEMODE=0`.

5.40.1 Set command

The set command disables SMS services in *NB-IoT*.

Syntax:

```
%SMSDISABLE=<sms_disable>
```

+CME ERROR code

518 – Not allowed in active state.

The set command parameter and its defined values are the following:

<sms_disable>

0 – Enable SMS services in NB-IoT.

1 – Disable SMS services in NB-IoT.

The following command example disables SMS services in NB-IoT:

```
AT%SMSDISABLE=1
OK
```

The following command example enables SMS services in NB-IoT:

```
AT%SMSDISABLE=0
OK
```

5.40.2 Read command

The read command reads the current value of the SMS disable setting.

Syntax:

```
%SMSDISABLE?
```

Response syntax:

```
%SMSDISABLE: <sms_disable>
```

The read response parameter and its defined values are the following:

<sms_disable>

0 – SMS services are enabled in NB-IoT.

1 – SMS services are disabled in NB-IoT.

The following command example reads the SMS disable setting:

```
AT%SMSDISABLE?
%SMSDISABLE: 0
OK
```

5.40.3 Test command

The test command is not supported.

6 SiP pin configuration

SiP pin configuration commands can be used to configure the behavior of selected pins of the nRF91x1 SiP. The pins that can currently be configured are **COEX0**, **MAGPIO[0:2]**, and **MIPI RFFE**.

For more information on the nRF91x1 SiP pins, see *Pin assignments* in nRF9131 Product Specification and nRF9161 Product Specification.

The control of these pins is tied to the modem operations, that is, the pins are controllable only when the modem is active. For example, if the modem goes to a long *PSM* sleep mode, the supply voltage for the pins is removed for power saving reasons and the pin state goes low until the modem wakes up again. The pin configuration can be made dependent on the modem's RF frequency. This means that instead of using the cell's static center frequency for decision-making, the dynamically changing center frequency of the current narrowband is used. Downlink or uplink direction does not affect the decision.

The commands in this section are intended to be given only once at boot or in device production if you want to store the configurations in *NVM*. This can be done with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

When the *PTI* modem firmware is used, the configuration can be stored in NVM with the **%XFSSYNC** command. After giving the commands, the modem software automatically toggles the pins, depending on RF frequency and modem state. In other words, the application does not need to send these commands during modem active usage.

6.1 COEX0 pin control configuration %XCOEX0

The proprietary **%XCOEX0** command writes the **COEX0** pin configuration to device RAM. v2.0.x

The **COEX0** pin can be configured to switch its state based on the modem's RF frequency to enable external *Low-Noise Amplifier (LNA)* in *GNSS* mode. The behavior is similar to the **%XMAGPIO** command except this command controls only one pin.

The **%XCOEX0** command needs to be sent before any modem activity occurs. Based on the given configuration, the modem applies the **COEX0** state corresponding to the RF frequency range automatically during runtime.

6.1.1 Set command

The set command writes the **COEX0** pin configuration to device RAM.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. The configuration can be stored to NVM with the **%XFSSYNC** command when the *PTI* modem firmware is used. The stored configuration is applied when the device is powered on. When RF is turned off, the given **COEX0** state is inverted.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XCOEX0=<count>,<state_0>,<freqlo_0>,<freqhi_0>,...
<state_count-1><freqlo_count-1><freqhi_count-1>
```

The set command parameters and their defined values are the following:

<count>

The number of frequency ranges. Valid values are 1, 2, 3, and 4.

<state_x>

The state of **COEX0** with the following frequency range. Valid values are 0 and 1.

<freqlo_x>

Low limit for the frequency range in MHz.

<freqhi_x>

High limit for the frequency range in MHz.

The following command example sets **COEX0** to 1 when *GNSS* is enabled (and 0 when GNSS is turned off). **COEX0** is not used with other frequencies (or LTE):

```
AT%XCOEX0=1,1,1570,1580
OK
```

The following command example sets **COEX0** to 1 when GNSS is enabled, or LTE frequency is 600 MHz to 800 MHz or 2000 MHz to 2180 MHz:

```
AT%XCOEX0=3,1,1570,1580,1,2000,2180,1,600,800
OK
```

If the command is given without any parameters, it deletes the previously written values:

```
AT%XCOEX0
OK
```

6.1.2 Read command

The read command returns the stored pin configuration.

Response syntax:

```
%XCOEX0: <count>,<state_0>,<freqlo_0>,<freqhi_0>,...
<state_count-1><freqlo_count-1><freqhi_count-1>
```

The read response parameters and their defined values are the following:

<count>

The number of frequency ranges. Valid values are 1, 2, 3, and 4.

<state_x>

The state of **COEX0** with the following frequency range. Valid values are 0 and 1.

<freqlo_x>

Low limit for the frequency range in MHz.

<freqhi_x>

High limit for the frequency range in MHz.

The following command example returns the stored configuration:

```
AT%XCOEX0?
AT%XCOEX0: 3,1,1570,1580,1,2000,2180,1,600,800
OK
```

6.1.3 Test command

The test command is not supported.

6.2 MAGPIO configuration %XMAGPIO

The proprietary **%XMAGPIO** command writes the MAGPIO configuration to device RAM. v2.0.x

The MAGPIO pins can be used, for example, to control an external antenna tuner, or any other *General-Purpose Input/Output (GPIO)*-controlled device, whose state depends on modem's RF frequency.
The **%XMAGPIO** command needs to be sent before any modem activity occurs. Based on the given configuration, the modem applies the MAGPIO state corresponding to the RF frequency range automatically during runtime.

6.2.1 Set command

The set command writes the MAGPIO configuration to device RAM.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. The configuration can be stored to NVM with the **%XFSSYNC** command when the *PTI* modem firmware is used. The stored configuration is applied when the device is powered on.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XMAGPIO=<gpio_0>,<gpio_1>,<gpio_2>,<num_ranges>,<state_0>,<flo_0>,<fhi_0><state_1>,<flo_1>,
<fhi_1>,...
```

A command without any parameters deletes the previously written values.

The set command parameters and their defined values are the following:

<gpio_x>

0 – MAGPIO_x is not used

1 – MAGPIO_x used

<num_ranges>

The number of frequency ranges, maximum value 12

<state_y>

Settings of the MAGPIO pins for the range x that follows

<flo_y>

Frequency range low value when the setting is active, in MHz

<fhi_y>

Frequency range high value when the setting is active, in MHz

The following table contains an example configuration for an antenna tuner:

	State	MAGPIO2	MAGPIO1	MAGPIO0	Low MHz	High MHz
Unused	0	0	0	0	-	-
LTE(746–803)	1	0	0	1	746	803
LTE(698–746)	2	0	1	0	698	746
LTE(1710–2200)	2	0	1	0	1710	2200
LTE(849–894)	3	0	1	1	849	894
LTE(894–960)	4	1	0	0	894	960
Unused	5	1	0	1	-	-
LTE(803–849)	6	1	1	0	803	849
GNSS	7	1	1	1	1574	1577

Table 1: Example configuration for an antenna tuner

The following command example writes seven ranges to device RAM:

```
AT%XMAGPIO=1,1,1,7,1,746,803,2,698,746,2,1710,2200,3,849,894,4,894,960,6,803,849,7,
1574,1577
OK
```

The following command example writes three ranges to device RAM:

```
AT%XMAGPIO=1,1,1,3,0,1574,1577,1,705,747,6,748,804
OK
```

The following command example deletes the previously written values:

```
AT%XMAGPIO
OK
```

6.2.2 Read command

The read command returns the stored MAGPIO configuration.

Response syntax:

```
%XMAGPIO:
 <gpio_0>,<gpio_1>,<gpio_2>,<num_ranges>,<state_0>,<flo_0>,<fhi_0><state_1>,<flo_1>,
<fhi_1>,...
```

The read response parameters and their defined values are the following:

<gpio_x>

0 – MAGPIO_x is not used

1 – MAGPIO_x used

<num_ranges>

The number of frequency ranges, maximum value 12

<state_y>

Settings of the MAGPIO pins for the range x that follows

<flo_y>

Frequency range low value when the setting is active, in MHz

<fhi_y>

Frequency range high value when the setting is active, in MHz

The following command example returns the stored configuration:

```
AT%XMAGPIO?
AT%XMAGPIO: 1,1,1,3,0,1574,1577,1,705,747,6,748,804
OK
```

6.2.3 Test command

The test command is not supported.

6.3 SiP-external MIPI RFFE device introduction %XMIPIRFFEDEV

The proprietary **%XMIPIRFFEDEV** command introduces the device and its static parameters to the nRF91x1 *SiP*. v2.0.x

nRF91x1 can be configured to control a SiP-external, *MIPI RF Front-End Control Interface (RFFE)*-controlled device to a limited extent. Antenna tuner is the primary use case.

After introducing the MIPI RFFE device, the configuration for the various control phases can be given using the **%XMIPIRFFECTRL** command.

The **%XMIPIRFFEDEV** command needs to be sent before any modem activity occurs.

6.3.1 Set command

The set command writes the **%XMIPIRFFEDEV** configuration to nRF91x1 RAM memory.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. The configuration can be stored to NVM with the **%XFSSYNC** command when the *PTI* modem firmware is used. The stored configuration is applied when any modem or *GNSS* RF activity occurs.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XMIPIRFFEDEV=<dev_id>,<def_usid>,<prod_id>,<man_id>,<pm_trig>
```

The set command parameters and their defined values are the following:

<dev_id>

Selectable identification number for the device. Non-zero. Valid range 1–255.
The given <dev_id> is used with the **%XMIPIRFFECTRL** and **%XMIPIRFFEDBG** commands.

<def_usid>

A 4-bit default *Unique Slave Identifier (USID)* for the *MIPI RFFE* device. Typically, 7 for antenna tuners (as suggested by MIPI).

<prod_id>

An 8-bit PRODUCT_ID of the MIPI RFFE device. Only used if automatic reprogramming of the USID is needed. EXT_PRODUCT_ID is not supported.

<man_id>

A 10-bit MANUFACTURER_ID of the MIPI RFFE device. Only used if automatic reprogramming of the USID is needed.

<pm_trig>

An 8-bit content for PM_TRIG (address 0x1C = 28 dec) register. This is for setting the default power and triggering mode. The setting of PM_TRIG can be also changed in the ON phase. See **%XMIPIRFFECTRL**.

All numbers should be given as decimals, that is, not as hexadecimals. Currently, nRF91x1 supports only one *SiP*-external MIPI RFFE-controlled device.

6.3.2 Read command

The read command returns the introductory information given for a device using the **%XMIPIRFFEDEV** command and the phase-specific configurations given in the **%XMIPIRFFECTRL** command. There is no dedicated read command for **%XMIPIRFFECTRL**.

Response syntax:

```
%XMIPIRFFEDEV?
%XMIPIRFFEDEV: <dev_id>,<def_usid>,<prod_id>,<man_id>,<pm_trig>
INIT:
ON:
OFF:
PWROFF:
OK
```

The read response parameters and their descriptions for the “%XMIPIRFFEDEV” row are as defined in Set command on page 91 if a valid **%XMIPIRFFEDEV** command has been given earlier. Otherwise, the row is empty. The phase-specific rows that follow (INIT, ON, OFF, PWROFF) contain the parameters given for that phase or they are empty.

In the following command example, the following commands have been given:

```
AT%XMIPIRFFEDEV=1,7,171,331,184
OK
AT%XMIPIRFFECTRL= 1,1,1,28,56,6,1,2,2,3,750,3,8,850,18,9,1000,20,12,1700,35,19,1900,37,
25,2200
OK
```

In the following command example, the read command returns:

```
AT%XMIPIRFFEDEV?
%XMIPIRFFEDEV: 1,7,171,331,184
INIT:
ON: 1,1,1,28,56,6,1,2,2,3,750,3,8,850,18,9,1000,20,12,1700,35,19,1900,37,25,2200
OFF:
PWROFF:
OK
```

6.3.3 Delete configuration

A *MIPI RFFE* device configuration and control phase information can be deleted from the nRF91x1 memory using this command.

Syntax:

```
%XMIPIRFFEDEV=<dev_id>
```

The following command deletes the device whose `<dev_id>=1` and all related phase controls that have been given using the **%XMIPIRFFECTRL** command:

```
AT%XMIPIRFFEDEV=1
 OK
```

> **CAUTION:** The combined load of *Printed Circuit Board (PCB)* routing, the input load of the MIPI RFFE-controlled device, and any parasitic load from application shall not exceed 15 pF at **SCLK** or **SDATA** pins. This load translates roughly to narrow transmission line length of less than 10 cm at the application board but it is dependent on the actual PCB design. A load higher than 15 pF at **SCLK** or **SDATA** pin increases the risk of unwanted behavior of the nRF91x1 *SiP* itself and of MIPI RFFE control.

6.4 SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL

The proprietary **%XMIPIRFFECTRL** command is used to configure the *SiP*-external *MIPI RFFE* device register accesses in different phases of modem operation. v2.0.x

The device must first be introduced using the **%XMIPIRFFEDEV** command.

MIPI RFFE devices contain an internal register map described in the datasheet of the device. To control the device, the registers in the device must be written with valid values. The **%XMIPIRFFECTRL** command allows configuring the nRF91x1 *SiP* to write the device's registers. The register addresses, the values, and timing (phase) can be configured as described below.

The external MIPI RFFE control in nRF91x1 supports configuring the RFFE device for four different phases of RF operation. The phases are initializing (INIT), start receiving or transmitting (ON), stop receiving or transmitting (OFF), and going to sleep (PWROFF).

The **%XMIPIRFFECTRL** command must be sent separately for each phase. It is not mandatory to configure all phases.

The **%XMIPIRFFECTRL** command must be sent before any modem activity occurs.

The **%XMIPIRFFEDBG** command can be used to debug MIPI RFFE configurations by reading or writing to device registers when using *PTI* image.

The phases are defined as follows:

INIT

Applied when RF is waking up. INIT is frequency-agnostic. Controls up to four MIPI RFFE device registers. The main purpose is to allow preparation or activation of the MIPI RFFE device if activation requires long settling.

ON

Applied when RF is starting for a specific frequency or when *LTE-M* frequency hopping is performed by the modem RF. Controls a maximum of two frequency-agnostic registers that can be used for device activation, for instance. This phase also controls a maximum of two registers whose value can be defined to depend on the RF frequency of the modem. The table for the frequency-dependent control can have a maximum of 64 frequencies.

OFF

Applied when RF is stopping. The configuration is frequency-agnostic. Controls up to four MIPI RFFE device registers.

PWROFF

Applied when RF is going to sleep. The configuration is frequency-agnostic. Controls up to four MIPI RFFE device registers. The main purpose is to deactivate the MIPI RFFE device.

6.4.1 Set command

The set command writes the **`%XMIPIRFFECTRL`** configuration to the nRF91x1 RAM memory.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. The configuration can be stored to NVM with the **`%XFSSYNC`** command when the *PTI* modem firmware is used.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

The command is given separately for each phase. It is not necessary to send the command for each phase, which means it is possible to configure only one phase.

Syntax:

```
%XMIPIRFFECTRL=<dev_id>,<phase#>,<variable_number_of_phase_specific_parameters>
```

The set command parameters and their defined values are the following:

<dev_id>

The identification number of the *MIPI RFFE* device given when it was introduced using the **`%XMIPIRFFEDEV`** command.

<phase#>

Number of the phase INIT = 0, ON = 1, OFF = 2, PWR_OFF = 3. All numbers must be given as decimals (hexadecimals are not allowed).

The following figure illustrates the RFFE device control in different phases:

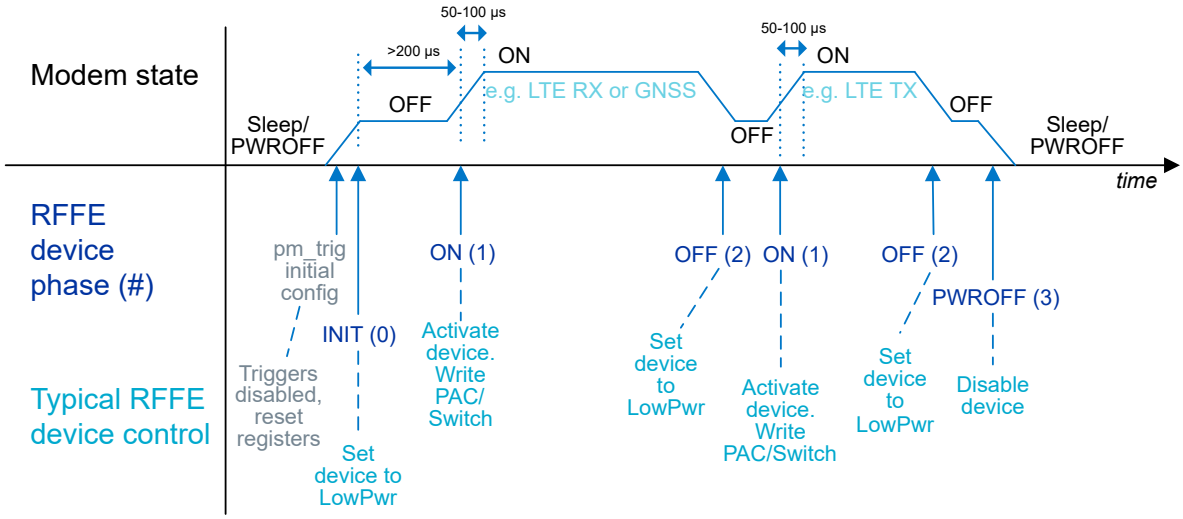

Figure 1: RFFE device control and timing in different phases

In the figure, PAC/Switch refers to a register in an example antenna tuner that controls the tunable capacitors and/or switches.

6.4.2 Phases INIT(0), OFF(2), and PWROFF(3)

Phases INIT(0), OFF(2), and PWROFF(3) are introduced here.

The syntax of each phase is the same except for the phase# parameter:

INIT (0)

```
%XMIPIRFFECTRL=<dev_id>,0,<n>,<address_0>,<data_0>,…,<address_n-1>,<data_n-1>
```

OFF (2)

```
%XMIPIRFFECTRL=<dev_id>,2,<n>,<address_0>,<data_0>,…,<address_n-1>,<data_n-1>
```

PWROFF (3)

```
%XMIPIRFFECTRL=<dev_id>,3,<n>,<address_0>,<data_0>,…,<address_n-1>,<data_n-1>
```

The parameters and their defined values are the following:

<n>

The number of address or data pairs. Valid values are 0, 1, 2, 3, 4. If the value is 0, all the following fields must be omitted.

<address_x>

The 8-bit address of the internal register in *MIPI RFFE* device. x = 0, ..., n−1.

<data_x>

The 8-bit data to be written to <address_x>. x = 0, ..., n−1.

6.4.3 Phase ON(1)

Phase ON(1) contains the most options for configuring the device.

Syntax:

```
%XMIPIRFFECTRL=<dev_id>,1,<n>,<act_addr_0>,<act_data_0>,<act_addr_n-1>,<act_data_n-1>,<k>,
<addr_0>,<addr_1>,<data_0_0>,<data_1_0>,<freq_0>,...,<data_0_k-1>,<data_1_k-1>,<freq_k-1>
```

The parameters and their descriptions are the following:

<n>

The number of activation register address-data pairs. Valid values are 0, 1, 2. If n = 0, act_addr_0/1 and act_data_0/1 must be omitted.

<act_addr_x>

8-bit address of the first register whose value is set to activate device. This is written each time RF starts. Must be given if <n> is greater than 0.

<act_data_x>

Optional 8-bit data for the register in <act_addr_x>.

Must be given if <n> is greater than 0.

<k>

The number of frequencies in the configuration. Valid values are 0–64. If k = 0, all the following fields must be omitted.

<addr_0>

The 8-bit address of the first register, whose value is changed on the basis of RF frequency.

<addr_1>

The 8-bit address of the other register, whose value is changed on the basis of RF frequency. If addr_1 == addr_0, then only <data_0_y> is written.

<data_0_y>

The 8-bit data for the register in <addr_0>, if frequency is smaller than or equal to <freq_y>.

<data_1_y>

The 8-bit data for the register in <addr_1>, if frequency is smaller than or equal to <freq_y>. data_1_y must be given (for example, as 0) even if addr_1 == addr_0.

<freq_y>

Integer. The frequency in MHz to which the current RF frequency is compared. If current RF frequency is smaller than or equal to <freq_y>, then <data_0_y> is written to <addr_0> and <data_1_y> is written to <addr_1>. If the RF frequency is greater than <freq_k−1> (the last given frequency), then neither <addr_0> nor <addr_1> is written.

6.4.4 Delete configuration

The `AT%XMIPIRFFEDEV= <dev_id >` command deletes all configurations for the *MIPI RFFE* device, including phase configurations. To delete the configuration of each phase individually, set <n> = 0 or/and <k> = 0 in the phase-specific command.

The following command example deletes only the ON phase configuration:

```
AT%XMIPIRFFECTRL=<dev_id>,1,0,0
```

The following command example deletes the PWROFF phase configuration:

```
AT%XMIPIRFFECTRL==<dev_id>,3,0
```

6.5 Alternative configuration of SiP antenna switch %XANTCFG

The proprietary **%XANTCFG** command configures the *SiP*-internal antenna switch to an alternative predefined position. v2.0.x

6.5.1 Set command

The set command configures the *SiP*-internal antenna switch to an alternative predefined position.

The **%XANTCFG** command supports one predefined setting. Therefore, in the *GNSS* mode the input signal to nRF91x1's **ANT** input is routed back out from the **AUX** output. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. The configuration can be stored to NVM with the **%XFSSYNC** command when the *PTI* modem firmware is used. After a valid configuration is set, nRF91x1 automatically controls the switch during GNSS reception. Adding further predefined settings requires a modification to modem firmware.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XANTCFG=<cfg>
```

The set command parameter and its defined values are the following:

<cfg>

0 – Reserved, no action

1 – **ANT** input directed to **AUX** output in the GNSS mode

2 – Reserved, no action

...

7 – Reserved, no action

The following command example directs **ANT** input to **AUX** output:

```
AT%XANTCFG=1
OK
```

The following command example deletes the previous configuration:

```
AT%XANTCFG
OK
```

6.5.2 Read command

The read command reads the currently active configuration.

Syntax:

```
%XANTCFG?
```

The following command example returns the currently active configuration:

```
AT%XANTCFG?
%XANTCFG: 1
OK
```

6.5.3 Test command

The test command is not supported.

6.6 COEX2 pin timing configuration %XCOEX2

The proprietary **%XCOEX2** command determines the timing for enabling the **COEX2** pin. v2.0.x

By default, the **COEX2** pin is enabled automatically when LTE/*GNSS* RF is used and disabled when RF is powered down. The default timing of the **COEX2** pin has approximately 1 ms anticipation to the first sample in the antenna.

In addition to the default mode, the **%XCOEX2** command has two alternative anticipation times of approximately 400 µs and 50 µs to 100 µs before the first sample in the antenna. The pin toggling can also be disabled.

The **COEX2** pin can also be set to toggle at only RF transmission or only RF reception.

The **%XCOEX2** command must be sent before any modem activity occurs. Based on the given configuration, the modem toggles the **COEX2** pin automatically during runtime.

6.6.1 Set command

The set command determines the timing for enabling the **COEX2** pin.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. The configuration can be stored to NVM with the **%XFSSYNC** command when the *PTI* modem firmware is used.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XCOEX2[=<ctrl>]
```

The set command parameter and its defined values are the following:

<ctrl>

Not set – Clears the current configuration and uses default values (1).

0 – **COEX2** pin toggling disabled, pin `OFF`.

1 – **COEX2** pin toggling enabled, default timing, 1 ms anticipation.

2 – **COEX2** pin toggling enabled, mid-timing, approximately 400 µs anticipation.

3 – **COEX2** pin toggling enabled, as late as possible timing, approximately 50 µs to 100 µs anticipation.

4 – **COEX2** pin toggling enabled, TX indicator. Pin `ON` when RF is transmitting.

5 – **COEX2** pin toggling enabled, RX indicator. Pin `ON` when RF is receiving.

The following command example sets the mid-timing of the **COEX2** pin:

```
AT%XCOEX2=2
OK
```

6.6.2 Read command

The read command reads the current configuration of the **COEX2** pin.

Syntax:

```
%XCOEX2?
```

Response syntax:

```
%XCOEX2: <ctrl>
```

The read command parameter and its defined values are the following:

<ctrl>

Not set – Clears the current configuration and uses default values (1).

0 – **COEX2** pin toggling disabled, pin `OFF`.

1 – **COEX2** pin toggling enabled, default timing, 1 ms anticipation.

2 – **COEX2** pin toggling enabled, mid-timing, approximately 400 µs anticipation.

3 – **COEX2** pin toggling enabled, as late as possible timing, approximately 50 µs to 100 µs anticipation.

4 – **COEX2** pin toggling enabled, TX indicator. Pin `ON` when RF is transmitting.

5 – **COEX2** pin toggling enabled, RX indicator. Pin `ON` when RF is receiving.

The following command example reads the current configuration of the **COEX2** pin.

```
AT%XCOEX2?
%XCOEX2: 2
OK
```

6.6.3 Test command

The test command is not supported.

7 Packet domain commands

Commands for the packet domain include commands that control packet-switched services.

7.1 Define PDP context +CGDCONT

The **+CGDCONT** command defines *Packet Data Protocol (PDP) Context*. v2.0.x

> **Note:** Operator configurations can override values set by the user.

For reference, see *3GPP 27.007 Ch. 10.1.1*.

7.1.1 Set command

The set command configures connection parameters.

Syntax:

```
+CGDCONT=<cid> [,<PDP_type> [,<APN> [,<PDP_addr> [,<d_comp> [,<h_comp> [,<IPv4AddrAlloc>
 [,<request_type> [,<P-CSCF_discovery> [,<IM_CN_Signalling_Flag_Ind> [,<NSLPI>
 [,<securePCO>]]]]]]]]]]]
```

> **Note:** `+CGDCONT=<cid>` causes the values for context number <cid> to become undefined.

A maximum of four contexts can be defined. In some network configurations, the maximum can be less than four.

The set command parameters and their defined values are the following:

<cid>

Integer, 0–10 (mandatory). Specifies a particular *Packet Data Protocol (PDP) Context* definition. The parameter is local to the device and is used in other PDP context-related commands.

<PDP_type>

String

IP – Internet Protocol

IPV6 – Internet Protocol version 6

IPV4V6 – Virtual type of dual IP stack

Non-IP – Transfer of non-IP data to external packet data network (see *3GPP TS 23.401 [82]*)

<APN>

String. *Access Point Name (APN)*.

<PDP_addr>

Ignored

<d_comp>

Ignored

<h_comp>

Ignored

<IPv4AdrAlloc>

0 – IPv4 address via *Non-access Stratum (NAS)* signaling (default)

1 – IPv4 address via *Dynamic Host Configuration Protocol (DHCP)*

<request_type>

Ignored

<P-CSCF_discovery>

Ignored

<IM_CN_SignallingFlag>

Ignored

<NSLPI>

0 – *Non-access Stratum (NAS) Signalling Low Priority Indication (NSLPI)* value from configuration is used (default)

1 – Value "Not configured" for NAS signaling low priority

<securePCO>

0 – Protected transmission of *Protocol Configuration Options (PCO)* is not requested (default)

1 – Protected transmission of PCO is requested

The following command example configures CID 1 to use IPv4 and access point "IOT_apn":

```
AT+CGDCONT=1,"IP","IOT_apn"
OK
```

7.1.2 Read command

The read command reads the list of defined contexts.

Response syntax:

```
+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>
```

The read command parameters and their defined values are the following:

<cid>

Integer, 0–10.

<PDP_type>

String

IP – Internet Protocol

IPV6 – Internet Protocol version 6

IPV4V6 – Virtual type of dual IP stack

Non-IP – Transfer of non-IP data to external packet data network (see *3GPP TS 23.401 [82]*)

<APN>

String. *APN*.

<PDP_addr>

String. IP address.

<d_comp>

0 – Compression not supported

<h_comp>

0 – Compression not supported

The following command example reads configured default bearers:

```
AT+CGDCONT?
+CGDCONT: 0,"IP","internet","10.0.1.1",0,0
+CGDCONT: 1,"IP","IOT_apn","10.0.1.2",0,0
OK
```

7.1.3 Test command

The test command is not supported.

7.2 Packet domain event notification +CGEREP

The **+CGEREP** command subscribes packet domain event notifications. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.19*.

7.2.1 Set command

The set command subscribes packet domain event notifications. The unsolicited result code is `+CGEV: XXX`.

For information on **+CGEV**, see Packet domain event notification +CGEV on page 103.

Syntax:

```
+CGEREP=[<mode>]
```

The set command parameter and its defined values are the following:

<mode>

0 – Unsubscribe unsolicited result codes

1 – Subscribe unsolicited result codes

The following command example subscribes CGEV notifications:

```
AT+CGEREP=1
OK
```

7.2.2 Read command

The read command reads the current mode and buffering settings.

Response syntax:

```
+CGEREP: <mode>,<bfr>
```

The read command parameters and their defined values are the following:

<mode>

0 – Unsubscribe unsolicited result codes

1 – Subscribe unsolicited result codes

<bfr>

0 – MT buffer of unsolicited result codes is cleared when <mode> 1 is entered.

The following command example reads the current mode:

```
AT+CGEREP?
+CGEREP: 1,0
OK
```

7.2.3 Test command

The test command reads supported modes and buffering settings.

Response syntax:

```
+CGEREP: (list of supported <mode>s),(list of supported <bfr>s)
```

The test command parameters and their defined values are the following:

<mode>

0 – Unsubscribe unsolicited result codes

1 – Subscribe unsolicited result codes

<bfr>

0 – MT buffer of unsolicited result codes is cleared when <mode> 1 is entered.

The following command example lists supported modes and buffering settings:

```
AT+CGEREP=?
+CGEREP: (0,1),(0)
OK
```

7.3 Packet domain event notification +CGEV

Unsolicited packet domain notifications are sent when the device is detached from the network or when a packet data connection is activated, deactivated, or modified. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.19*.

These notifications are subscribed using the **+CGEREP** command.

Syntax descriptions are as follows:

Network detach. All active contexts are deactivated. The deactivation of individual contexts is not reported separately:

```
+CGEV: NW DETACH
```

ME detach:

```
+CGEV: ME DETACH
```

ME overheated and flight mode enabled:

```
+CGEV: ME OVERHEATED
```

Battery voltage is low:

```
+CGEV: ME BATTERY LOW
```

The ME has activated a default bearer:

```
+CGEV: ME PDN ACT <cid>[,<reason>]
```

The network has activated a dedicated bearer:

```
+CGEV: NW ACT <p_cid>,<cid>,<event_type>
```

The network has deactivated a default bearer:

```
+CGEV: NW PDN DEACT <cid>
```

The *UE* has deactivated a default bearer:

```
+CGEV: ME PDN DEACT <cid>
```

The network has deactivated a dedicated bearer:

```
+CGEV: NW DEACT <p_cid>,<cid>,<event_type>
```

The UE has deactivated a dedicated bearer:

```
+CGEV: ME DEACT <p_cid>,<cid>,<event_type>
```

The network has modified a bearer:

```
+CGEV: NW MODIFY <cid>,<change_reason>,<event_type>
```

The UE has modified a bearer:

```
+CGEV: ME MODIFY <cid>,<change_reason>,<event_type>
```

IPv6 link is up for the default bearer:

```
+CGEV: IPV6 <cid>
```

IPv6 address resolution or refresh failure:

```
+CGEV: IPV6 FAIL <cid>
```

Requested procedure restricted:

```
+CGEV: RESTR <cause>,<validity>
```

Configuration of *APN* rate control:

```
+CGEV: APNRATECTRL CFG
 <cid>,<rate>,<time_window>,<exceptional_data_allowed>,<additional_rc_rate>,
<additional_rc_time_window>
```

Status of APN rate control:

```
+CGEV: APNRATECTRL STAT
 <cid>,<status>,<remaining_time>,<exceptional_data_allowed>,<additional_rc_status>,
<additional_rc_remaining_time>
```

Status indicating if exceptional data is allowed. Only changes in allowance are notified:

```
+CGEV: EXCE STATUS <exce_status>
```

<cid>

Integer, 0–10.

<reason>

0 – Only IPv4 allowed

1 – Only IPv6 allowed

2 – Only single access bearers allowed

3 – Only single access bearers allowed and context activation for a second address type bearer was not successful.

<change_reason>

Integer. A bitmap that indicates what kind of change has occurred. The <change_reason> value is determined by adding together all the applicable bits.

Bit 1 – TFT changed

Bit 2 – *Quality of Service (QoS)* changed

Bit 3 – WLAN offload changed

<cid_other>

Integer, 0–10. Indicates the context identifier allocated for an *MT*-initiated context of a second address type. This parameter is included only if the <reason> parameter indicates that only single address bearers are allowed.

<p_cid>

Integer, 0–10. Context identifier for an associated default context.

<event_type>

0 – Informational event

1 – Information request. Acknowledgement is required, and it can be either accept or reject.

<cause>

Restriction cause

1 – *Radio Policy Manager (RPM)*. Procedure restricted by RPM.

2 – Throttling. Procedure restricted by 3GPP or operator-specific throttling.

3 – Invalid configuration. Procedure restricted by invalid context configuration.

<validity>

Validity of restriction

1 – Permanent restriction. Enabling requires a power-off, *UICC* change, or a configuration change.

2 – Temporary restriction. Enabling requires, for example, back-off timer expiry.

<number_of_packets>

0–16 777 215 – Number of data packets that the *UE* is allowed to send during the time period indicated in <time_window>.

Values 0,0 of <number_of_packets>,<time_window> indicate that APN rate control is not configured.

<time_window>

60–604 800 – Time window in seconds (1 minute to 1 week).

<exceptional_data_allowed>

Indicates if exceptional data is allowed when APN rate control blocks the sending of data. Valid only if additional APN rate control is not configured (<additional_rc_status> is `2`).

0 – Not allowed.

1 – Allowed.

<additional_rc_rate>

0–65535 – Additional configured exceptional data rate. Number of additional exceptional data packets that the UE is allowed to send when <rate> is exceeded during the time period indicated in <additional_rc_time_window>.

Values `0,0` of <additional_rc_rate>,<additional_rc_time_window> indicate that additional APN rate control is not configured.

If additional APN rate control is configured, it controls the sending of exceptional data, and the value of<exceptional_data_allowed> is `0`.

<additional_rc_time_window>

60–604 800 – Additional time window in seconds (1 minute to 1 week).

The time window is used to calculate exceptional data packets configured at <additional_rc_rate>.

<status>

0 – OFF. APN rate control does not block uplink data.

1 – ON. APN rate control blocks uplink data.

2 – APN rate control is not configured for this CID.

<remaining_time>

0–604 800 – Remaining uplink data blocking time in seconds when status is `1`. Can be `0` if less than a full second remains.

<additional_rc_status>

0 – APN rate control is configured for exceptional data and currently not blocking exceptional uplink data.

1 – APN rate control is configured for exceptional data and currently blocking exceptional uplink data.

2 – APN rate control is not configured for exceptional data for this CID.

<additional_rc_remaining_time>

0–604 800 – Remaining exceptional uplink data blocking time in seconds when <additional_rc_status> is `1`.

Can be `0` if less than a full second remains, but rate control blocks exceptional data if <additional_status> is `1`. `0` when <additional_rc_status> is `0`.

<exce_status>

0 – Exceptional data not allowed.

1 – Exceptional data allowed.

The following notification example shows that an initial *Packet Data Network (PDN)* connection is activated:

```
+CGEV: ME PDN ACT 0
```

The following notification example shows that the device is detached from network:

```
+CGEV: ME DETACH
```

The following notification example shows a restriction caused by throttling with temporary validity.

```
+CGEV: RESTR 2,2
```

7.4 Activate PDP context +CGACT

The **+CGACT** command activates or deactivates a *PDN* connection. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.10*.

7.4.1 Set command

The set command activates or deactivates a *PDN* connection.

Note: Initial PDN connection (cid 0) could not be activated or deactivated.

First, the *Packet Data Protocol (PDP) Context* needs to be defined with the **+CGDCONT** command.

Syntax:

```
+CGACT=<state>,<cid>
```

The set command parameters and their defined values are the following:

<state>

0 – Deactivate

1 – Activate

<cid>

Integer, 0–10.

The following command example activates a bearer configured with CID 1:

```
AT+CGACT=1,1
OK
```

7.4.2 Read command

The read command reads a list of *PDN* connections and states.

Response syntax:

```
+CGACT: <cid>,<state>
```

The read command parameters and their defined values are the following:

<state>

0 – Deactivate

1 – Activate

<cid>

Integer, 0–10.

The following command example returns a list of connections with states:

```
AT+CGACT?
+CGACT: 0,1
+CGACT: 1,1
OK
```

7.4.3 Test command

The test command returns a list of supported states.

Response syntax:

```
+CGACT: (list of supported <state>s)
```

The test command parameter and its defined values are the following:

<state>

0 – Deactivate

1 – Activate

The following command example returns a list of supported states:

```
AT+CGACT=?
+CGACT: (0,1)
OK
```

7.5 Allocate new CID %XNEWCID

The proprietary **%XNEWCID** command allocates a new context identifier. Several context identifiers can share the same *PDN*. v2.0.x

7.5.1 Set command

The set command is not supported.

7.5.2 Read command

The read command allocates a new context identifier.

The command allocates a unique context identifier, which can be referenced with other commands like **+CGDCONT**. The allocated identifier can be deallocated with the **+CGDCONT** command by giving only the <cid> parameter.

The **%XNEWCID** command can be used instead of reading existing default context with `AT+CGDCONT?` and finding an unused <cid> value before configuring a new context.

A maximum of four context identifiers can be defined. In some network configurations, the maximum can be less than four. The read command allocates a new context identifier.

Response syntax:

```
%XNEWCID: <cid>
```

The read command parameter and its defined value are the following:

<cid>

Integer, 0–10.

The following command example requests the allocation of a new context identifier:

```
AT%XNEWCID?
%XNEWCID: 2
OK
```

7.5.3 Test command

The test command is not supported.

7.6 Map CID to PDN ID %XGETPDNID

The proprietary **%XGETPDNID** command maps the context identifier to *PDN* ID. Several context identifiers can share the same PDN. This command can be used only when the modem is activated. v2.0.x

7.6.1 Set command

The set command maps the context identifier to *PDN* ID.

PDN ID is used on a data path to select one of the existing connections for data transfer.

Syntax:

```
%XGETPDNID=<cid>
```

Response syntax:

```
%XGETPDNID: <pdn_id>
```

The set command parameters and their defined values are the following:

<cid>

Integer, 0–10.

<pdn_id>

Integer, 0–10.

The following command example maps the context identifier to PDN ID 0:

```
AT%XGETPDNID=0
%XGETPDNID: 1
OK
```

7.6.2 Read command

The read command is not supported.

7.6.3 Test command

The test command is not supported.

7.7 QoS dynamic parameters +CGEQOSRDP

The **+CGEQOSRDP** command reads dynamic *Evolved Packet System (EPS) QoS* parameters. The command issues a valid response only when the modem is activated. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.27*.

7.7.1 Set command

The set command reads dynamic *EPS QoS* parameters.

Syntax:

```
+CGEQOSRDP[=<cid>]
```

Response syntax:

```
[+CGEQOSRDP: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>][,<DL_AMBR>,<UL_AMBR>]]
```

The set command parameters and their defined values are the following:

<cid>

Integer, 0–10. If the parameter <cid> is omitted, the QoS parameters for all active *Packet Data Protocol (PDP) Contexts* are returned.

<QCI>

Integer. Specifies a class of EPS QoS (see *3GPP TS 23.203* and *3GPP TS 24.301*).

<DL_GBR>, <UL_GBR>, <DL_MBR>, <UL_MBR>

Not supported

<DL_AMBR>

Integer. Specifies downlink *APN* aggregate maximum bitrate. Value range 0–65280000 kbps.

<UL_AMBR>

Integer. Specifies uplink APN aggregate maximum bitrate. Value range 0–65280000 kbps.

The following command example returns a list of contexts with QoS parameters:

```
AT+CGEQOSRDP
+CGEQOSRDP: 0,0,,
+CGEQOSRDP: 1,2,,
+CGEQOSRDP: 2,4,,,1,65280000
OK
```

7.7.2 Read command

The read command is not supported.

7.7.3 Test command

The test command is not supported.

7.8 Show PDP addresses +CGPADDR

The **+CGPADDR** command returns a list of *Packet Data Protocol (PDP)* addresses for the specified context identifiers. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.14*.

7.8.1 Set command

The set command returns a list of *PDP* addresses for the specified context identifiers. This command issues a valid response only when the modem is activated.

Syntax:

```
+CGPADDR[=<cid>]
```

If <cid> is not present, all activated contexts are listed.

Response syntax:

```
[+CGPADDR: <cid>[,<PDP_addr_1>[,<PDP_addr_2>]]]
```

The set command parameters and their defined values are the following:

<cid>

Integer, 0–10.

<PDP_addr_1>

String. For IPv4 given as a dot-separated numeric (0–255) parameter. For IPv6 given as a colon-separated hexadecimal (0x0000–0xFFFF) parameter.

<PDP_addr_2>

String. Given as a colon-separated hexadecimal (0x0000–0xFFFF) parameter. Included when both IPv4 and IPv6 addresses are assigned.

The following command example returns the IP address for context 1:

```
AT+CGPADDR=1
+CGPADDR: 1,"10.0.0.130","1050:0000:0000:0000:0005:0600:300c:326b"
OK
```

7.8.2 Read command

The read command is not supported.

7.8.3 Test command

The test command returns a list of defined <cid> values.

Response syntax:

```
+CGPADDR: (list of defined <cid>s)
```

The test command parameter and its defined value are the following:

<cid>

Integer, 0–10.

The following command example returns a list of defined <cid> values:

```
AT+CGPADDR=?
+CGPADDR: (0,1)
OK
```

7.9 PDN connection dynamic parameters +CGCONTRDP

The **+CGCONTRDP** command returns information for an active *PDN* connection. This command issues a valid response only when the modem is activated. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.23*.

7.9.1 Set command

The set command returns information for an active *PDN* connection.

Syntax:

```
+CGCONTRDP=<cid>
```

Response syntax:

```
+CGCONTRDP: <cid>,<bearer_id>,<apn>[,<local_addr_and_subnet_mask>[,<gw_addr>[,
<DNS_prim_addr>[,<DNS_sec_addr>[,,,,,<IPv4_MTU]]]]]
```

The set command parameters and their defined values are the following:

<cid>

Integer, 0–10 (mandatory).

<bearer_id>

Integer. Not supported.

<apn>

String. A logical name for the network.

<local_addr_and_subnet_mask>

String. Not supported.

<gw_addr>

String. Not supported.

<DNS_prim_addr>, <DNS_sec_addr>

String. DNS server IP address.

IPv4_MTU

IPv4 *Maximum Transmission Unit (MTU)* size.

> **Note:** If the PDN connection has dual stack capabilities, at least one pair of lines with information is returned per <cid>: First one line with the IPv4 parameters followed by one line with the IPv6 parameters.

The following command example reads dynamic parameters for an initial PDN connection:

```
AT+CGCONTRDP=0
+CGCONTRDP: 0,,"internet","","","10.0.0.1","10.0.0.2",,,,,1028
OK
```

7.9.2 Read command

The read command is not supported.

7.9.3 Test command

The test command is not supported.

7.10 PS attach or detach +CGATT

The **+CGATT** command attaches the *MT* to or detaches the MT from the Packet Domain services. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.9*.

> **Note:** The **+CGATT** command is for testing purposes only.

7.10.1 Set command

The set command attaches the *UE* to or detaches the UE from the Packet Domain services.

> **Note:** The UE performs an attach automatically when activated. In normal operation there is no need to issue the **+CGATT** command.

Syntax:

```
+CGATT=<state>
```

+CME ERROR code

517 – Modem not activated.

The set command parameter and its defined values are the following:

<state>

0 – Detached

1 – Attached

The following command example performs an *EPS* attach:

```
AT+CGATT=1
OK
```

7.10.2 Read command

The read command reads the state.

Response syntax:

```
+CGATT: <state>
```

The response parameters and their defined values are the following:

<state>

0 – Detached

1 – Attached

The following command example reads the state in *EPS* attach state:

```
AT+CGATT?
+CGATT: 1
OK
```

7.10.3 Test command

The test command returns a list of supported states.

Response syntax:

```
+CGATT: (list of supported <state>s)
```

The test command parameter and its defined values are the following:

<state>

0 – Detached

1 – Attached

The following command example returns a list of supported states:

```
AT+CGATT=?
+CGATT: (0,1)
OK
```

7.11 Power preference indication for EPS +CEPPI

The **+CEPPI** command selects the power saving preference. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.38*.

7.11.1 Set command

The set command selects if the *UE* indicates to the network during radio connection that it prefers low power configuration.

Syntax:

```
+CEPPI=<power_preference>
```

The set command parameter and its defined values are the following:

<power_preference>

0 – Normal

1 – Low power consumption

The following command example selects the power saving preference:

```
AT+CEPPI=1
OK
```

7.11.2 Read command

The read command is not supported.

7.11.3 Test command

The test command lists the supported power preferences.

Syntax:

```
+CEPPI=(list of supported <power preference>s)
```

The test command parameter and its defined values are the following:

<power_preference>

0 – Normal

1 – Low power consumption

The following command example lists the supported power saving preferences:

```
AT+CEPPI=?
+CEPPI: (0,1)
```

7.12 Protocol configuration options notification %XPCO

The proprietary **%XPCO** command subscribes *PCO* notifications. v2.0.x

7.12.1 Set command

The set command subscribes *PCO* notifications.

Syntax:

```
%XPCO=<n>
```

Notification syntax:

```
%XPCO: <id>,<container_data>
```

The set command parameter and its defined values are the following:

<n>

0 – Unsubscribe PCO notifications

1 – Subscribe PCO notifications

The notification parameters and their defined values are the following:

<id>

PCO identifier in decimal format

<container_data>

String in hexadecimal *IRA* format. An empty container data string indicates that PCO container has not been received.

The following command example subscribes *E-UTRA* signal quality notifications:

```
AT%XPCO=1
OK
```

The following is an example of a PCO notification for a FF00h container:

```
%XPCO: 65280,"A1B1C1D1"
```

7.12.2 Read command

The read command is not supported.

7.12.3 Test command

The test command is not supported.

7.13 Usage of ePCO/PCO in PDN connection establishment %XEPCO

The proprietary **%XEPCO** command selects the usage of ePCO/*PCO* in *PDN* connection establishment.
v2.0.x

7.13.1 Set command

The set command selects ePCO/*PCO* usage. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XEPCO=<epco>
```

The set command parameter and its defined values are the following:

<epco>

0 – Use PCO

1 – Use ePCO

The following command example disables ePCO and selects PCO:

```
AT%XEPCO=0
OK
```

7.13.2 Read command

The read command reads the state of ePCO/*PCO* usage.

Response syntax:

```
%XEPCO=<epco>
```

The following command example reads the state of ePCO/PCO usage:

```
AT%XEPCO?
%XEPCO: 1
OK
```

7.13.3 Test command

The test command is not supported.

7.14 APN class access %XAPNCLASS

The proprietary **%XAPNCLASS** command reads *APN* class data. v2.0.x

7.14.1 Set command

The set command reads *APN* class data. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XAPNCLASS=<oper>,<class>[,<apn>]
```

Read response syntax:

```
%XAPNCLASS: <class>,<apn>,<addr_type>
```

The set command and response parameters and their defined values are the following:

<oper>

0 – Read

<class>

APN class

<apn>

APN name string

<addr_type>

String

IP – Internet Protocol

IPV6 – Internet Protocol version 6

IPV4V6 – Virtual type of dual IP stack

The following command example reads APN class 3:

```
AT%XAPNCLASS=0,3
%XAPNCLASS: 3,"VZWAPN","IPV4V6"
OK
```

7.14.2 Read command

The read command is not supported.

7.14.3 Test command

The test command is not supported.

7.15 External IP stack IPv6 address resolution/refresh failure %XIPV6FAIL

The proprietary **%XIPV6FAIL** indicates an external IP stack IPv6 address resolution or refresh failure. v2.0.x

7.15.1 Set command

The set command indicates the modem an external IP stack IPv6 address resolution or refresh failure.

Syntax:

```
%XIPV6FAIL=<cid>,<failure_type>
```

The set command parameters and their defined values are the following:

<cid>

Integer, 0-10.

<failure_type>

0 – IPv6 address refresh failure

1 – IPv6 address resolution failure

The following command example indicates the modem an IPv6 address resolution failure in the default context identifier 0:

```
AT%XIPV6FAIL=0,1
OK
```

7.15.2 Read command

The read command is not supported.

7.15.3 Test command

The test command is not supported.

7.16 Define PDN connection authentication parameters +CGAUTH

The **+CGAUTH** command specifies authentication parameters. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.31*.

7.16.1 Set command

The set command specifies authentication parameters for a *PDN* connection specified by the <cid> parameter.

Syntax:

```
+CGAUTH=<cid>[,<auth_prot>[,<userid>[,<password>]]]
```

The set command parameters and their defined values are the following:

<cid>

Integer, 0–10.

<auth_prot>

0 – None. Username and password are removed if they have been specified.

1 – PAP

2 – CHAP

<userid>

String

<password>

String

The following command example sets authentication parameters for CID=1 context:

```
AT+CGAUTH=1,1,"username","password"
OK
```

7.16.2 Read command

The read command is not supported.

7.16.3 Test command

The test command is not supported.

7.17 Signaling connection status notification +CSCON

The **+CSCON** command subscribes and configures unsolicited result code notifications. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.30*.

7.17.1 Set command

The set command subscribes and configures unsolicited result code notifications.

Syntax:

```
+CSCON=[<n>]
```

The set command parameter and its defined values are the following:

<n>

0 – Unsubscribe unsolicited indications

1 – Subscribe <mode>

2 – Subscribe <mode>[,<state>]

3 – Subscribe <mode>[,<state>[,<access>]]

Notification syntax:

```
+CSCON: <mode>[,<state>[,<access]]
```

The response parameters and their defined values are the following:

<mode>

0 – Idle

1 – Connected

<state>

7 – *E-UTRAN* connected

<access>

4 – Radio access of type E-UTRAN FDD

The following command example enables level 3 indications:

```
AT+CSCON=3
OK
```

The following is an example of a level-3-related unsolicited indication:

```
+CSCON: 1,7,4
```

7.17.2 Read command

The read command returns the current status of unsolicited result code presentation <n>.

The parameter <mode> is returned always when `<n>=0` or when `<n>=1`. The optional parameter <state> is returned when `<n>=2` and <access> when `<n>=3`.

Response syntax:

```
+CSCON: <n>,<mode>[,<state>[,<access]]
```

The read command parameters and their defined values are the following:

<n>

0 – Unsubscribe unsolicited indications

1 – Subscribe <mode>

2 – Subscribe <mode>[,<state>]

3 – Subscribe <mode>[,<state>[,<access>]]

<mode>

0 – Idle

1 – Connected

<state>

7 – *E-UTRAN* connected

<access>

4 – Radio access of type E-UTRAN FDD

When reading the current signaling connection status, the following response indicates that unsolicited indications are disabled, and the modem is an idle state:

```
AT+CSCON?
+CSCON: 0,0
OK
```

The following response indicates that unsolicited indications are enabled, the modem mode is 1, E-UTRAN is connected and the radio access type is E-UTRAN FDD:

```
AT+CSCON?
+CSCON: 3,1,7,4
OK
```

7.17.3 Test command

The test command returns a list of supported values of <n> as a compound value.

Response syntax:

```
+CSCON: (list of supported <n>s)
```

The test command parameter and its defined values are the following:

<n>

0 – Unsubscribe unsolicited indications

1 – Subscribe <mode>

2 – Subscribe <mode>[,<state>]

3 – Subscribe <mode>[,<state>[,<access>]]

The following command example returns the supported values:

```
AT+CSCON=?
+CSCON: (0,1,2,3)
OK
```

7.18 Use of APN %XAPNSTATUS

The proprietary **%XAPNSTATUS** command enables and disables the use of an *APN*. v2.0.x

7.18.1 Set command

The set command enables or disables the use of an *APN*. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XAPNSTATUS=<status>,<APN>
```

The set command parameters and their defined values are the following:

<status>

0 – Disable APN

1 – Enable APN

<APN>

String. APN.

The following command example disables the use of an Internet APN:

```
AT%XAPNSTATUS=0,"internet"
```

The following command example enables the use of an Internet APN:

```
AT%XAPNSTATUS=1,"internet"
```

7.18.2 Read command

The read command reads a list of disabled *APN*s. v1.2.x

Response syntax:

```
%XAPNSTATUS: [<apn_1>[,…<apn_n>]]
```

The read command parameter and its defined value are the following:

<apn_x>

String. APN.

The following command example reads a list of disabled APNs:

```
AT%XAPNSTATUS?
%XAPNSTATUS: ”APN_1”,”APN_2”
OK
```

7.18.3 Test command

The test command is not supported.

7.19 PDN configuration %XPDNCFG

The proprietary **%XPDNCFG** command sets and reads *PDN* configurations. v2.0.x

7.19.1 Set command

The set command sets the *PDN* configuration.

The initial PDN connection that has been activated during the Attach procedure is automatically reactivated if an always-on PDN has been configured with the **%XPDNCFG** command. The reactivation is triggered if deactivation occurs unexpectedly and is not requested by the client. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
%XPDNCFG=<lifetime>
```

The set command parameter and its defined values are the following:

<lifetime>

0 – Default PDN lifetime

1 – Always-on PDN. The initial PDN connection is automatically reactivated.

The following command example sets an always-on PDN for the initial PDN connection:

```
AT%XPDNCFG=1
OK
```

The following command example removes the setting of the always-on PDN of the initial PDN connection:

```
AT%XPDNCFG=0
OK
```

7.19.2 Read command

The read command reads the *PDN* configuration.

Response syntax:

```
%XPDNCFG?
```

The following command example reads the initial PDN configuration when an always-on PDN has been configured:

```
AT%XPDNCFG?
%XPDNCFG: 1
OK
```

7.19.3 Test command

The test command is not supported.

7.20 APN rate control %APNRATECTRL

The proprietary **%APNRATECTRL** command requests *APN* rate control information. v2.0.x

7.20.1 Set command

The set command requests the configuration or status information of the *APN* rate control.

The APN rate control configuration and status changes are available also as **+CGEV** notifications.

APN rate control is active only when the context is active. An indication of context deactivation, such as `+CGEV ME PDN DEACT <cid>`, received by a client also indicates that the APN rate control configuration is removed for the context identified by <cid>. If the context is activated again, a new APN rate control configuration is indicated to the client if it is configured by the network.

The network can assign a separate APN rate control for exceptional data called additional APN rate control.

Syntax:

```
%APNRATECTRL=<mode>[,<cid>]
```

Response syntax for status query:

```
%APNRATECTRL:
 <mode>,<cid>,<status>,<remaining_time>,<exceptional_data_allowed>,<additional_rc_status>,
<additional_rc_remaining_time>
```

Response syntax for configuration query:

```
%APNRATECTRL:
 <mode>,<cid>,<rate>,<time_window>,<exceptional_data_allowed>,<additional_rc_rate>,
<additional_rc_time_window>
```

The set command parameters and their defined values are the following:

<mode>

0 – Request APN rate control status for a CID.

1 – Request APN rate control configurations.

<cid>

Integer, 0–10.

Mandatory in mode 0. Not used in mode 1.

<status>

0 – APN rate control is configured and currently not blocking uplink data.

1 – APN rate control is configured and currently blocking uplink data.

2 – APN rate control is not configured for this CID.

<remaining_time>

0–604 800 – Remaining uplink data blocking time in seconds when status is `1`.

Can be `0` if less than a full second remains, but rate control blocks uplink data if <status> is `1`.

<exceptional_data_allowed>

Indicates if exceptional data is allowed when APN rate control blocks the sending of data. Valid only if additional APN rate control is not configured (<additional_rc_status> is `2`).

0 – Not allowed.

1 – Allowed.

<additional_rc_status>

0 – APN rate control is configured for exceptional data and currently not blocking exceptional uplink data.

1 – APN rate control is configured for exceptional data and currently blocking exceptional uplink data.

2 – APN rate control is not configured for exceptional data for this CID.

<additional_rc_remaining_time>

0–604 800 – Remaining exceptional uplink data blocking time in seconds when <additional_rc_status> is `1`.

Can be `0` if less than a full second remains, but rate control blocks exceptional data if <additional_status> is `1`. `0` when <additional_rc_status> is `0`.

<rate>

0–16 777 215 – Configured data rate. Number of data packets that the *UE* is allowed to send during the time period indicated in <time_window>.

Values `0,0` of <rate>,<time_window> mean that APN rate control is not configured.

<time_window>

60–604 800 – Time window in seconds (1 minute to 1 week).

<additional_rc_rate>

0–65535 – Additional configured exceptional data rate. Number of additional exceptional data packets that the UE is allowed to send when <rate> is exceeded during the time period indicated in <additional_rc_time_window>.

Values `0,0` of <additional_rc_rate>,<additional_rc_time_window> indicate that additional APN rate control is not configured.

If additional APN rate control is configured, it controls the sending of exceptional data, and the value of<exceptional_data_allowed> is `0`.

<additional_rc_time_window>

60–604 800 – Additional time window in seconds (1 minute to 1 week).

The time window is used to calculate exceptional data packets configured at <additional_rc_rate>.

The following command example requests the status of CID 1. APN rate control is configured and blocking uplink data with one minute remaining. Additional APN rate control is not configured for CID 1:

```
AT%APNRATECTRL=0,1
%APNRATECTRL: 0,1,1,60,0,2,0
OK
```

The following command example requests the status of CID 1. APN rate control is configured and blocking uplink data with one minute remaining. Additional APN rate control is blocking with two minutes remaining:

```
AT%APNRATECTRL=0,1
%APNRATECTRL: 0,1,1,60,0,1,120
OK
```

The following command example requests the status of CID 1. APN rate control is configured and blocking uplink data with one minute remaining. Exceptional data is allowed:

```
AT%APNRATECTRL=0,1
%APNRATECTRL: 0,1,1,60,1,2,0
OK
```

The following command example requests the status of CID 1. APN rate control is configured and not blocking uplink data, but if <rate> is exceeded before <additional_remaining_time> is elapsed, also exceptional data is blocked immediately:

```
AT%APNRATECTRL=0,1
%APNRATECTRL: 0,1,0,0,0,1,180
OK
```

The following command example requests the status of CID 2. APN rate control is configured and not blocking uplink data. Additional APN rate control is not configured for CID 2:

```
AT%APNRATECTRL=0,2
%APNRATECTRL: 0,2,0,0,0,2,0
OK
```

The following command example requests the status of CID 2. APN rate control and additional APN rate control are not configured:

```
AT%APNRATECTRL=0,2
%APNRATECTRL: 0,2,2,0,0,2,0
OK
```

The following command example requests APN rate control configurations. APN rate control is configured for CID 1 and 2. Additional APN rate control is not configured:

```
AT%APNRATECTRL=1
%APNRATECTRL: 1,1,10,180,0,0,0
%APNRATECTRL: 1,2,100,60,0,0,0
OK
```

The following command example requests APN rate control configurations. APN rate control is configured for CID 1 and 2. Exceptional data is allowed for CID1. Additional configuration for CID2:

```
AT%APNRATECTRL=1
%APNRATECTRL: 1,1,10,180,1,0,0
%APNRATECTRL: 1,2,100,60,0,400,120
OK
```

The following command example requests APN rate control configurations. No configurations are active:

```
AT%APNRATECTRL=1
%APNRATECTRL: 1,0,0,0,0,0,0
OK
```

7.20.2 Read command

The read command is not supported.

7.20.3 Test command

The test command is not supported.

7.21 Exceptional data state %EXCEPTIONALDATA

The proprietary **%EXCEPTIONALDATA** command sets or reads the state of *PDN* connection usage for exceptional data. v2.0.x

The command can be used to enable the sending of exceptional data in a PDN connection

Exceptional data can be used only in *NB-IoT*. The command has no effect in *LTE-M*.

To send exceptional data, the application must also set the exceptional data socket option to send exceptional data packets through the socket. The socket option is used to determine if the data packet has normal or exceptional priority.

If PDN connection usage allows the sending of exceptional data, data packets, such as Router Solicitation sent in IPv6 negotiation or DNS Query to resolve other end IP address, are handled as exceptional data

PDN connection usage for exceptional data is also used to determine if a possible activation of initial or additional PDN connection should establish RRC connection for exceptional data.

7.21.1 Set command

The set command sets or reads the state of *PDN* connection usage for exceptional data.

Syntax:

```
%EXCEPTIONALDATA=<cmd>,<cid>[,<state>]
```

Response syntax:

```
%EXCEPTIONALDATA: <cid>,<state>
```

+CME ERROR code

0 – Invalid parameters or invalid CID.

3 – Not allowed. Exceptional data not enabled in EFnasconfig in inserted UICC.

The set command parameters and their defined values are the following:

<cmd>

0 – Set exceptional data state.

1 – Read exceptional data state.

<cid>

Integer, 0–10.

<state>

Mandatory when <cmd> is `0`. Not allowed when <cmd> is `1`.

0 – PDN connection is used only for regular data (default).

1 – PDN connection can be used for exceptional data.

The following command example allows exceptional data for CID 0:

```
AT%ECEPTIONALDATA=0,0,1
OK
```

The following command example reads the exceptional data state of CID 0:

```
AT%EXCEPTIONALDATA=1,0
%EXCEPTIONALDATA: 0,0
OK
```

7.21.2 Read command

The read command is not supported.

7.21.3 Test command

The test command is not supported.

8 Network service related commands

For reference, see *3GPP 27.007 Ch. 7*.

8.1 PLMN selection +COPS

The **+COPS** command selects a *PLMN* automatically or manually, and reads and searches the current mobile network. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.3*.

8.1.1 Set command

The set command selects a mobile network automatically or manually. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
+COPS=<mode>[,<format>[,<oper>[,<AcT>]]]
```

The set command parameters and their defined values are the following:

<mode>

0 – Automatic network selection

1 – Manual network selection

3 – Set <format> of **+COPS** read command response

<format>

0 – Long alphanumeric <oper> format. Only for <mode> 3.

1 – Short alphanumeric <oper> format. Only for <mode> 3.

2 – Numeric <oper> format

<oper>

String of digits. *MCC* and *MNC* values.

<AcT>

7 – *E-UTRAN*

9 – E-UTRAN (NB-S1 mode)

For manual selection, only the numeric string format is supported and <oper> is mandatory.

The following command example selects the automatic network selection:

```
AT+COPS=0
OK
```

The following command manually selects network 24407:

```
AT+COPS=1,2,"24407"
OK
```

8.1.2 Read command

The read command reads the current mobile network.

Response syntax:

```
+COPS: <mode>[,<format>,<oper>,[AcT>]]
```

The read command parameters and their defined values are the following:

<mode>

0 – Automatic network selection

1 – Manual network selection

2 – Deregistered. Only for the read command.

<format>

0 – Long alphanumeric <oper> format

1 – Short alphanumeric <oper> format

2 – Numeric <oper> format

<oper>

String in double quotes. Operator name or *MCC* and *MNC* values.

<AcT>

7 – *E-UTRAN*

9 – E-UTRAN (NB-S1 mode)

The following command example reads the current selection mode and network:

```
AT+COPS?
+COPS: 0,2,"26201",7
OK
```

The following command example reads the current selection mode and network with the operator name in alphanumeric format:

```
AT+COPS?
+COPS: 0,0,"RADIOLINJA",7
OK
```

8.1.3 Test command

The test command searches the mobile network and returns a list of operators found. If the search is interrupted, the search returns existing results and the list may be incomplete.

Response syntax:

```
+COPS: [(<stat>,long alphanumeric <oper>,short alphanumeric <oper>,numeric <oper>[,<AcT>])]
```

+CME ERROR code

516 – Radio connection is active.

521 – *PLMN* search interrupted, partial results.

The test command parameters and their defined values are the following:

<oper>

String of digits. *MCC* and *MNC* values.

<stat>

0 – Unknown

1 – Available

2 – Current

3 – Forbidden

<AcT>

7 – *E-UTRAN*

9 – E-UTRAN (NB-S1 mode)

Note:

- The command fails if the device has an active radio connection. It returns `ERROR` or `+CME ERROR: 516`.
- The time needed to perform a network search depends on device configuration and network conditions.

The following command example is used for a manual network search:

```
AT+COPS=?
+COPS: (2,"","","26201",7),(1,"","","26202",7)
OK
```

8.2 Forced PLMN search %COPS

The proprietary **%COPS** command performs a forced *PLMN* search. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.3*.

8.2.1 Set command

The set command is not supported.

8.2.2 Read command

The read command is not supported.

8.2.3 Test command

The test command searches the *PLMN* and returns a list of operators found.

The command is similar to the **+COPS** command with the exception that the **%COPS** test command is considered a high priority search. This means that, for example, data transfer is suspended, pagings

lost, and registration is not maintained. In other words, the search is not delayed because of any other procedure.

Response syntax:

```
%COPS: [(<stat>,long alphanumeric <oper>,short alphanumeric <oper>,numeric <oper>[,<AcT>])]
```

The test command parameters and their defined values are the following:

<oper>

String of digits. *MCC* and *MNC* values.

<stat>

0 – Unknown

1 – Available

2 – Current

3 – Forbidden

<AcT>

7 – *E-UTRAN*

9 – E-UTRAN (NB-S1 mode)

The following command example is used for a manual network search:

```
AT%COPS=?
%COPS: (2,"","","26201",7),(1,"","","26202",7)
OK
```

8.3 Power saving mode setting +CPSMS

The **+CPSMS** command controls *PSM* settings. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.38*.

8.3.1 Set command

The set command sets the power saving mode. Sets activity timer and *PSM* period after *NAS* signaling connection release. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

Syntax:

```
+CPSMS=[<mode>[,<Requested_Periodic-RAU>,<Requested_GPRS-READY-timer>,
<Requested_Periodic-TAU-ext>[,<Requested_Active-Time>]]]
```

The command can be given as `+CPSMS=` (with all parameters omitted). In this form, the parameter <mode> is set to `0`, the use of PSM is disabled, and data for all parameters is set to the manufacturer-specific default values.

The set command parameters and their defined values are the following:

<mode>

0 – Disable power saving mode

1 – Enable power saving mode

<Requested_Periodic-RAU>

Ignored

<Requested_GPRS-READY-timer>

Ignored

<Requested_Periodic-TAU-ext>

String. 1 byte in 8-bit format.

Indicates the extended periodic TAU value (T3412_EXT extended value) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 3 IE in *3GPP TS 24.008* Table 10.5.163a/3GPP TS 24.008.

GPRS Timer 3 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the *General Packet Radio Services (GPRS)* timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 10 min

0 0 1 – Value is incremented in multiples of 1 h

0 1 0 – Value is incremented in multiples of 10 h

0 1 1 – Value is incremented in multiples of 2 s

1 0 0 – Value is incremented in multiples of 30 s

1 0 1 – Value is incremented in multiples of 1 min

1 1 0 – Value is incremented in multiples of 320 h

> **Note:** Interpreted as 1 hour if the request sent to the network is not integrity protected. After registration, check the value given by the network (see **+CEREG**). If 1 hour unit is given, disable and enable PSM using commands `+CPSMS=0` and `+CPSMS=1`.

1 1 1 – Value indicates that the timer is deactivated

> **Note:** If the *USIM* profile in use is a Verizon one, the minimum value for <Requested_Periodic-TAU-ext> is 190 minutes.

<Requested_Active-Time>

String. 1 byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Active-Time is set to the manufacturer-specific default. For the coding and value range, see the GPRS Timer 2 IE in *3GPP TS 24.008* Table 10.5.163/3GPP TS 24.008.

GPRS Timer 2 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the GPRS timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 2 s

0 0 1 – Value is incremented in multiples of 1 min

0 1 0 – Value is incremented in multiples of 6 min

1 1 1 – Value indicates that the timer is deactivated

The following command example enables power saving mode and set timer values. Set Periodic-TAU timer to 10 minutes and Active-Time to 1 minute:

```
AT+CPSMS=1,"","","10101010","00100001"
OK
```

The following command example disables power saving mode:

```
AT+CPSMS=0
OK
```

The following command example disables power saving mode and sets timer to default values:

```
AT+CPSMS=
OK
```

8.3.2 Read command

The read command reads the current *PSM* settings.

Response syntax:

```
+CPSMS: <mode>,[<Requested_Periodic-RAU>],[<Requested_GPRS-READY-timer>],
[<Requested_Periodic-TAU-ext>],[<Requested_Active-Time>]
```

The read command parameters and their defined values are the following:

<mode>

0 – Disable power saving mode

1 – Enable power saving mode

<Requested_Periodic-RAU>

Ignored

<Requested_GPRS-READY-timer>

Ignored

<Requested_Periodic-TAU-ext>

String. 1 byte in 8-bit format.

Indicates the extended periodic TAU value (T3412_EXT extended value) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 3 IE in *3GPP TS 24.008* Table 10.5.163a/3GPP TS 24.008.

GPRS Timer 3 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 10 min

0 0 1 – Value is incremented in multiples of 1 h

0 1 0 – Value is incremented in multiples of 10 h

0 1 1 – Value is incremented in multiples of 2 s

1 0 0 – Value is incremented in multiples of 30 s

1 0 1 – Value is incremented in multiples of 1 min

1 1 0 – Value is incremented in multiples of 320 h

> **Note:** Interpreted as 1 hour if the request sent to the network is not integrity protected. After registration, check the value given by the network (see **+CEREG**). If 1 hour unit is given, disable and enable PSM using commands `+CPSMS=0` and `+CPSMS=1`.

1 1 1 – Value indicates that the timer is deactivated

> **Note:** If the *USIM* profile in use is a Verizon one, the minimum value for <Requested_Periodic-TAU-ext> is 190 minutes.

<Requested_Active-Time>

String. 1 byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Active-Time is set to the manufacturer-specific default. For the coding and value range, see the GPRS Timer 2 IE in *3GPP TS 24.008* Table 10.5.163/3GPP TS 24.008.

GPRS Timer 2 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the GPRS timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 2 s

0 0 1 – Value is incremented in multiples of 1 min

0 1 0 – Value is incremented in multiples of 6 min

1 1 1 – Value indicates that the timer is deactivated

The following command example reads the current power saving mode settings:

```
AT+CPSMS?
+CPSMS: 1,,,"10101111","01101100"
OK
```

8.3.3 Test command

The test command is not supported.

8.4 eDRX setting +CEDRXS

The **+CEDRXS** command controls the setting of *eDRX* parameters. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.40*.

8.4.1 Set command

The set command sets the requested *eDRX* parameters. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

When an eDRX parameter is changed, the default *Paging Time Window (PTW)* is set. If the default PTW is not used, the **%XPTW** command must be sent after the **+CEDRXS** command.

To ensure a proper amount of paging occasions and time for reception with all possible DRX cycle lengths, it is recommended to set the PTW length to ≥5.12 s for *LTE-M* and to ≥10.24 s for *NB-IoT* when possible.

Note: Use a shorter PTW length than the eDRX length provided by the network.

[2] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as if the Extended DRX parameters IE were not included in the message by this version of the protocol.

[3] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as 0010 by this version of the protocol.

Syntax:

```
+CEDRXS=[<mode>,[,<AcT-type>[,<Requested_eDRX_value>]]]
```

Unsolicited result code syntax:

```
+CEDRXP: <AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>
 [,<Paging_time_window>]]]
```

The set command parameters and their defined values are the following:

<mode>

0 – Disable the use of eDRX

1 – Enable the use of eDRX

2 – Enable the use of eDRX and enable the unsolicited result code

3 – Disable the use of eDRX and discard all parameters for eDRX or, if available, reset to the manufacturer-specific default values

<ActT-type>

0 – Current cell is not using eDRX. Used only in the unsolicited result code.

4 – *E-UTRAN* (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

[4] The value is applicable only in NB-S1 mode. If received in WB-S1 mode it is interpreted as 1101 by this version of the protocol.

<Requested_eDRX_value>

String. Half a byte in 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*). Optional. If not present, the value of the requested eDRX period is set to the manufacturer-specific default.

Bit

4 3 2 1 – E-UTRAN eDRX cycle length duration

0 0 0 0 – 5.12 s[2]

0 0 0 1 – 10.24 s[2]

0 0 1 0 – 20.48 s

0 0 1 1 – 40.96 s

0 1 0 0 – 61.44 s[3]

0 1 0 1 – 81.92 s

0 1 1 0 – 102.4 s[3]

0 1 1 1 – 122.88 s[3]

1 0 0 0 – 143.36 s[3]

1 0 0 1 – 163.84 s

1 0 1 0 – 327.68 s

1 0 1 1 – 655,36 s

1 1 0 0 – 1310.72 s

1 1 0 1 – 2621.44 s

1 1 1 0 – 5242.88 s[4]

1 1 1 1 – 10485.76 s[4]

<NW-Provided_eDRX_value>

String. Half a byte in 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

Bit

4 3 2 1 – E-UTRAN eDRX cycle length duration

0 0 0 0 – 5.12 s[2]

0 0 0 1 – 10.24 s[2]

0 0 1 0 – 20.48 s

0 0 1 1 – 40.96 s

0 1 0 0 – 61.44 s[3]

0 1 0 1 – 81.92 s

0 1 1 0 – 102.4 s[3]

0 1 1 1 – 122.88 s[3]

1 0 0 0 – 143.36 s[3]

1 0 0 1 – 163.84 s

1 0 1 0 – 327.68 s

1 0 1 1 – 655,36 s

1 1 0 0 – 1310.72 s

1 1 0 1 – 2621.44 s

1 1 1 0 – 5242.88 s[4]

1 1 1 1 – 10485.76 s[4]

<Paging_time_window>

String. Half a byte in 4-bit format. The paging time window refers to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

LTE-M mode

Bit

4 3 2 1 – PTW length

0 0 0 0 – 1.28 s

0 0 0 1 – 2.56 s

0 0 1 0 – 3.84 s

0 0 1 1 – 5.12 s

0 1 0 0 – 6.4 s

0 1 0 1 – 7.68 s

0 1 1 0 – 8.96 s

0 1 1 1 – 10.24 s

1 0 0 0 – 11.52 s

1 0 0 1 – 12.8 s

1 0 1 0 – 14.08 s

1 0 1 1 – 15.36 s

1 1 0 0 – 16.64 s

1 1 0 1 – 17.92 s

1 1 1 0 – 19.20 s

1 1 1 1 – 20.48 s

NB-IoT mode

Bit

4 3 2 1 – PTW length

0 0 0 0 – 2.56 s

0 0 0 1 – 5.12 s

0 0 1 0 – 7.68 s

0 0 1 1 – 10.24 s

0 1 0 0 – 12.8 s

0 1 0 1 – 15.36 s

0 1 1 0 – 17.92 s

0 1 1 1 – 20.48 s

1 0 0 0 – 23.04 s

1 0 0 1 – 25.6 s

1 0 1 0 – 28.16 s

1 0 1 1 – 30.72 s

1 1 0 0 – 33.28 s

1 1 0 1 – 35.84 s

1 1 1 0 – 38.4 s

1 1 1 1 – 40.96 s

The following command example enables eDRX and sets the requested eDRX value:

```
AT+CEDRXS=1,4,"1000"
OK
```

The unsolicited notification when <mode> 2 is used:

```
+CEDRXP: 4,"1000","0101","1011"
OK
```

8.4.2 Read command

The read command is used to read the requested *eDRX* parameters.

Response syntax:

```
+CEDRXS: <AcT-type>,<Requested_eDRX_value>
```

The read command parameters and their defined values are the following:

<ActT-type>

0 – Current cell is not using eDRX. Used only in the unsolicited result code.

4 – *E-UTRAN* (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

<Requestd_eDRX_value>

String. Half a byte in 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

The following command example reads the requested eDRX value:

```
AT+CEDRXS?
+CEDRXS: 4,"0001"
OK
```

8.4.3 Test command

The test command is used to list the supported *eDRX* parameters.

Response syntax:

```
+CEDRXS: (list of supported <mode>s),(list of supported <AcT-type>s),(list of supported
 <Requested_eDRX_value>s)
```

The test command parameters and their defined values are the following:

<mode>

0 – Disable the use of eDRX

1 – Enable the use of eDRX

2 – Enable the use of eDRX and enable the unsolicited result code

3 – Disable the use of eDRX and discard all parameters for eDRX or, if available, reset to the manufacturer-specific default values

<ActT-type>

0 – Current cell is not using eDRX. Used only in the unsolicited result code.

4 – *E-UTRAN* (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

<Requestd_eDRX_value>

String. Half a byte in 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

The following command example reads the supported parameter values:

```
AT+CEDRXS=?
+CEDRXS: (0-3),(4-5),("0000"-"1111")
OK
```

8.5 EDRX dynamic parameters +CEDRXRDP

The **+CEDRXRDP** command reads dynamic *eDRX* parameters. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.41*.

8.5.1 Set command

The set command reads dynamic *eDRX* parameters.

Syntax:

```
+CEDRXRDP
```

Response syntax:

```
+CEDRXRDP: <AcT-type>[,<Requested_eDRX_value>[,<NW-
provided_eDRX_value>[,<Paging_time_window>]]]
```

The set command parameters and their defined values are the following:

[5] The value is applicable only in WB-S1 mode. If received in NB-S1 mode, it is interpreted as if the Extended DRX parameters IE were not included in the message by this version of the protocol.

[6] The value is applicable only in WB-S1 mode. If received in NB-S1 mode, it is interpreted as 0010 by this version of the protocol.

[7] The value is applicable only in NB-S1 mode. If received in WB-S1 mode, it is interpreted as 1101 by this version of the protocol.

<ActT-type>

0 – Current cell not using eDRX

4 – *E-UTRAN* (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

<Requestd_eDRX_value>

String. Half a byte in 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

<NW-Provided_eDRX_value>

String. Half a byte in 4-bit format.

A configuration value that the network provides for the eDRX period. The value in this parameter does not necessarily indicate that eDRX is in use by the current cell to which the *UE* is registered. The use of eDRX can be dynamically monitored with the unsolicited notification **+CEDRXP** and its <ActT-type> parameter.

The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

Bit

4 3 2 1 – E-UTRAN eDRX cycle length duration

0 0 0 0 – 5.12 s[5]

0 0 0 1 – 10.24 s[5]

0 0 1 0 – 20.48 s

0 0 1 1 – 40.96 s

0 1 0 0 – 61.44 s[6]

0 1 0 1 – 81.92 s

0 1 1 0 – 102.4 s[6]

0 1 1 1 – 122.88 s[6]

1 0 0 0 – 143.36 s[6]

1 0 0 1 – 163.84 s

1 0 1 0 – 327.68 s

1 0 1 1 – 655,36 s

1 1 0 0 – 1310.72 s

1 1 0 1 – 2621.44 s

1 1 1 0 – 5242.88 s[7]

1 1 1 1 – 10485.76 s[7]

<Paging_time_window>

String. Half a byte in 4-bit format. The paging time window refers to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

LTE-M mode

Bit

4 3 2 1 – PTW length

0 0 0 0 – 1.28 s

0 0 0 1 – 2.56 s

0 0 1 0 – 3.84 s

0 0 1 1 – 5.12 s

0 1 0 0 – 6.4 s

0 1 0 1 – 7.68 s

0 1 1 0 – 8.96 s

0 1 1 1 – 10.24 s

1 0 0 0 – 11.52 s

1 0 0 1 – 12.8 s

1 0 1 0 – 14.08 s

1 0 1 1 – 15.36 s

1 1 0 0 – 16.64 s

1 1 0 1 – 17.92 s

1 1 1 0 – 19.20 s

1 1 1 1 – 20.48 s

NB-IoT mode

Bit

4 3 2 1 – PTW length

0 0 0 0 – 2.56 s

0 0 0 1 – 5.12 s

0 0 1 0 – 7.68 s

0 0 1 1 – 10.24 s

0 1 0 0 – 12.8 s

0 1 0 1 – 15.36 s

0 1 1 0 – 17.92 s

0 1 1 1 – 20.48 s

1 0 0 0 – 23.04 s

1 0 0 1 – 25.6 s

1 0 1 0 – 28.16 s

1 0 1 1 – 30.72 s

1 1 0 0 – 33.28 s

1 1 0 1 – 35.84 s

1 1 1 0 – 38.4 s

1 1 1 1 – 40.96 s

The following command example reads eDRX parameters:

```
AT+CEDRXRDP
+CEDRXRDP: 4,"0011","0010","1001"
OK
```

8.5.2 Read command

The read command is not supported.

8.5.3 Test command

The test command is not supported.

8.6 Operator name +COPN

The **+COPN** command reads operator names. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.21*.

8.6.1 Set command

The set command reads operator names.

Syntax:

```
+COPN
```

Note: The device does not have operator names stored in it.

The following command example reads operator names:

```
AT+COPN
OK
```

8.6.2 Read command

The read command is not supported.

8.6.3 Test command

The test command is not supported.

8.7 Network registration status notification +CEREG

The **+CEREG** command subscribes unsolicited network status notifications. v2.0.x

8.7.1 Set command

The set command subscribes unsolicited network status notifications.

Syntax:

```
+CEREG=<n>
```

The set command parameter and its defined values are the following:

<n>

0 – Unsubscribe unsolicited result codes

1 – Subscribe unsolicited result codes `+CEREG:<stat>`

2 – Subscribe unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 – Subscribe unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 – Subscribe unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU-ext>]]]]`

5 – Subscribe unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU-ext>]]]]`

For the notification syntax parameters, see Read command on page 147.

The following command example subscribes notifications with level 2:

```
AT+CEREG=2
OK
```

Unsolicited notification level 1, trying to attach:

```
+CEREG: 2
```

Unsolicited notification level 2, registered:

```
+CEREG: 1,"002F","0012BEEF",7
```

8.7.2 Read command

The read command reads current network registration status.

Response syntax:

```
+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>][,<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU-ext>]]]]
```

The read command parameters and their defined values are the following:

<n>

0 – Unsubscribe unsolicited result codes

1 – Subscribe unsolicited result codes `+CEREG:<stat>`

2 – Subscribe unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 – Subscribe unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 – Subscribe unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU-ext>]]]]`

5 – Subscribe unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU-ext>]]]]`

<stat>

0 – Not registered. *UE* is not currently searching for an operator to register to.

1 – Registered, home network

2 – Not registered, but UE is currently trying to attach or searching an operator to register to

3 – Registration denied

4 – Unknown (for example, out of *E-UTRAN* coverage)

5 – Registered, roaming

90 – Not registered due to *UICC* failure

<tac>

String in hexadecimal format. 2-byte *TAC*.

<ci>

String in hexadecimal format. 4-byte E-UTRAN cell ID.

<AcT>

7 – E-UTRAN

9 – E-UTRAN NB-S1

<cause_type>

0 – <reject_cause> contains an *EPS Mobility Management (EMM)* cause value. See *3GPP TS 24.301 Annex A*.

<reject_cause>

EMM cause value. See *3GPP TS 24.301 Annex A*

<Active-Time>

String. 1 byte in 8-bit format.

Indicates the Active Time value (T3324) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 2 IE in *3GPP TS 24.008* Table 10.5.163/3GPP TS 24.008.

The device can enter *PSM* state in LTE idle mode when the <Active-Time> parameter has a valid value where at least one of bits 6–8 is set to `0`. The length of PSM sleep is the value that remains from the value configured for periodic TAU after active time has elapsed, which is T3324 subtracted from T3412.

GPRS Timer 2 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the GPRS timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 2 s

0 0 1 – Value is incremented in multiples of 1 min

0 1 0 – Value is incremented in multiples of 6 min

1 1 1 – Value indicates that the timer is deactivated

<Periodic-TAU-ext>

String. 1 byte in 8-bit format.

Indicates the extended periodic *TAU* value (T3412ext) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 3 IE in *3GPP TS 24.008* Table 10.5.163a/3GPP TS 24.008.

A deactivated value where bits 6–8 are set to `1` after a successful LTE Attach or TAU means that the network configures the periodic TAU or PSM cycle length using a non-extended format with *GPRS* timer instead of GPRS timer 3. In this case, the configured value of T3412 can be read with the **`%XMONITOR`** command. The response includes the value in a format selected by the network.

GPRS Timer 3 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the GPRS timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 10 min

0 0 1 – Value is incremented in multiples of 1 h

0 1 0 – Value is incremented in multiples of 10 h

0 1 1 – Value is incremented in multiples of 2 s

1 0 0 – Value is incremented in multiples of 30 s

1 0 1 – Value is incremented in multiples of 1 min

1 1 0 – Value is incremented in multiples of 320 h

1 1 1 – Value indicates that the timer is deactivated

The following command example reads the current registration status:

```
AT+CEREG?
+CEREG: 2,1,"002F","0012BEEF",7
OK
```

8.7.3 Test command

The test command returns a list of supported modes as a compound value.

Response syntax:

```
+CEREG: (supported modes)
```

The test command parameter and its defined values are the following:

<n>

0 – Unsubscribe unsolicited result codes

1 – Subscribe unsolicited result codes `+CEREG:<stat>`

2 – Subscribe unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 – Subscribe unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 – Subscribe unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU-ext>]]]]`

5 – Subscribe unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU-ext>]]]]`

The following command example shows supported unsolicited results codes:

```
AT+CEREG=?
+CEREG: (0-5)
OK
```

8.8 Operator name notification %XOPNAME

The proprietary **%XOPNAME** command subscribes unsolicited operator name notifications. v2.0.x

8.8.1 Set command

The set command subscribes unsolicited operator name notifications. The notification is sent when *EMM* information *Protocol Data Unit (PDU)* with the operator name is received.

Syntax:

```
%XOPNAME=<n>
```

Notification syntax:

```
%XOPNAME: [<full_name>],[<short_name>],[<oper>]
```

The set command and notification parameters and their defined values are the following:

<n>

0 – Unsubscribe unsolicited operator names

1 – Subscribe unsolicited operator names

<full_name>

String in hexadecimal format. Optional field for the full operator name as specified in *3GPP TS 24.008 Ch. 10.5.3.5a Network Name* and received from network. The first octet describes the number of spare bits in the last octet, usage of country initials, and the coding scheme of the network name. Octets 2−n specify the network name.

<short_name>

String in hexadecimal format. Optional field for a short operator name as specified in *3GPP TS 24.008 Ch. 10.5.3.5a Network Name* and received from network. The first octet describes the number of spare bits in the last octet, usage of country initials, and the coding scheme of the network name. Octets 2−n specify the network name.

<oper>

String of digits. *MCC* and *MNC* values.

The following command example subscribes notifications:

```
AT%XOPNAME=1
OK
```

An example of an unsolicited notification for a full and a short operator name:

```
%XOPNAME: "88D6B23CAD7FBB41D7B4BCCC2ECFE7","8B56FD15","556776"
```

An example of an unsolicited notification for a short operator name:

```
%XOPNAME: ,"8B56FD15","556776"
```

8.8.2 Read command

The read command is not supported.

8.8.3 Test command

The test command is not supported.

8.9 Network time notification %XTIME

The proprietary **%XTIME** command subscribes unsolicited network time notifications. v2.0.x

8.9.1 Set command

The set command subscribes unsolicited network time notifications. The notification is sent when *EMM* information *PDU* with time information is received.

Syntax:

```
%XTIME=<n>
```

Notification syntax:

```
%XTIME: [<local_time_zone>],[<universal_time>],[<daylight_saving_time>]
```

The set command and notification parameters and their defined values are the following:

<n>

0 – Unsubscribe unsolicited network time

1 – Subscribe unsolicited network time

<local_time_zone>

String in hexadecimal format. 1-byte optional field for the local time zone as specified in *3GPP TS 24.008 Ch. 10.5.3.8 Time Zone* and received from network.

<universal_time>

String in hexadecimal format. 7-byte optional field for universal time as specified in *3GPP TS 24.008 Ch. 10.5.3.9 Time Zone and Time* and received from network.

<daylight_saving_time>

String in hexadecimal format. 1-byte optional field for daylight saving time as specified in *3GPP TS 24.008 Ch. 10.5.3.12 Daylight Saving Time* and received from network.

The following command example subscribes notifications:

```
AT%XTIME=1
OK
```

An example of an unsolicited notification for network time with all parameters:

```
%XTIME: "08","81109251714208","01"
```

An example of an unsolicited notification for network time without local time zone:

```
%XTIME: ,"81109251714208","01"
```

8.9.2 Read command

The read command is not supported.

8.9.3 Test command

The test command is not supported.

8.10 Operator ID %XOPERID

The proprietary **%XOPERID** command identifies the operator *USIM*. v2.0.x

8.10.1 Set command

The set command returns the operator ID.

Syntax:

```
%XOPERID
```

Response syntax:

```
%XOPERID: <oper_id>
```

The response parameter and its defined values are the following:

<oper_id>

0 – Operator not identified as any of those listed below

1 – Verizon

2 – AT&T

3 – AT&T FirstNet

4 – AT&T Cricket

5 – AT&T Jasper

6 – China Telecom

7 – Softbank

8 – Telstra

9 – Bell

10 – LG U+

11 – KDDI

12 – Truphone

13 – Telus Jasper V2

14 – Vodafone

15 – T-Mobile US

16 – Deutsche Telekom

17 – Rogers

18 – Orange

19 – KT

20 – Telefonica

The following command example returns the operator ID:

```
AT%XOPERID
%XOPERID: 1
OK
```

8.10.2 Read command

The read command is not supported.

8.10.3 Test command

The test command is not supported.

8.11 Modem parameters %XMONITOR

The proprietary **%XMONITOR** command reads a set of modem parameters. v2.0.x

Note: When *NB-IoT* system mode is used and the device is in RRC CONNECTED state, old signal quality parameter values are reported. The values are recorded and reported from the previous RRC IDLE state.

8.11.1 Set command

The set command reads modem parameters.

Syntax:

```
%XMONITOR
```

Response syntax:

```
%XMONITOR: <reg_status>,[<full_name>,<short_name>,<plmn>,<tac>,<AcT>,<band>,<cell_id>,
<phys_cell_id>,<EARFCN>,<rsrp>,<snr>,<NW-provided_eDRX_value>,<Active-Time>,<Periodic-TAU-
ext>,<Periodic-TAU>]
```

The response parameters and their defined values are the following:

<reg_status>

0 – Not registered. *UE* is not currently searching for an operator to register to.

1 – Registered, home network

2 – Not registered, but UE is currently trying to attach or searching an operator to register to

3 – Registration denied

4 – Unknown (for example, out of *E-UTRAN* coverage)

5 – Registered, roaming

90 – Not registered due to *UICC* failure

Note: The optional part is included in the response only when <reg_status> is 1 or 5. Some parameters may not be present in specific circumstances. For example, <phys_cell_id>, <EARFCN>, <rsrp>, and <snr> are not available when the device is not camped on a cell.

<full_name>

String in alphanumeric format. Operator name.

<short_name>

String in alphanumeric format. Operator name.

<plmn>

String of digits. *MCC* and *MNC* values.

[8] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as if the Extended DRX parameters IE were not included in the message by this version of the protocol.

[9] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as 0010 by this version of the protocol.

[10] The value is applicable only in NB-S1 mode. If received in WB-S1 mode it is interpreted as 1101 by this version of the protocol.

<tac>

String in hexadecimal format. 2-byte *TAC*.

<AcT>

7 – E-UTRAN

9 – E-UTRAN NB-S1

<band>

Integer, 1−68. See *3GPP 36.101*. The value is 0 when current band information is not available.

<cell_id>

String in hexadecimal format. 4-byte E-UTRAN cell ID.

<phys_cell_id>

Integer. Physical cell ID.

<EARFCN>

Integer. *EARFCN* for a given cell where EARFCN is as defined in *3GPP TS 36.101*.

<rsrp>

0 – RSRP < −140 dBm

1 – When −140 dBm ≤ RSRP < −139 dBm

2 – When −139 dBm ≤ RSRP < −138 dBm

...

95 – When −46 dBm ≤ RSRP < −45 dBm

96 – When −45 dBm ≤ RSRP < −44 dBm

97 – When −44 dBm ≤ RSRP

255 – Not known or not detectable

The index value of RSRP can be converted to decibel with the following formula: Index – 140 = dBm. For example, 95 – 140 = –45 dBm.

<snr>

0 – SNR < −24 dB

1 – When −24 dB ≤ SNR < −23 dB

2 – When −23 dB ≤ SNR < −22 dB

...

47 – When 22 dB ≤ SNR < 23 dB

48 – When 23 dB ≤ SNR < 24 dB

49 – When 24 dB ≤ SNR

127 – Not known or not detectable

The index value of SNR can be converted to decibel with the following formula: Index – 24 = dB. For example, 1 – 24 = –23 dB and 47 – 24 = 23 dB.

<NW-provided_eDRX_value>

String. Half a byte in 4-bit format.

A configuration value that the network provides for the eDRX period. The value in this parameter does not necessarily indicate that eDRX is in use by the current cell to which the UE is registered. The use of eDRX can be dynamically monitored with the unsolicited notification **+CEDRXP** and its <ActT-type> parameter.

The *eDRX* value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

Bit

4 3 2 1 – E-UTRAN e-I-DRX cycle length duration

0 0 0 0 – 5.12 s[8]

0 0 0 1 – 10.24 s[8]

0 0 1 0 – 20.48 s

0 0 1 1 – 40.96 s

0 1 0 0 – 61.44 s[9]

0 1 0 1 – 81.92 s

0 1 1 0 – 102.4 s[9]

0 1 1 1 – 122.88 s[9]

1 0 0 0 – 143.36 s[9]

1 0 0 1 – 163.84 s

1 0 1 0 – 327.68 s

1 0 1 1 – 655,36 s

1 1 0 0 – 1,310.72 s

1 1 0 1 – 2,621.44 s

1 1 1 0 – 5,242.88 s[10]

1 1 1 1 – 10,485.76 s[10]

<Active-Time>

String. 1 byte in 8-bit format.

Indicates the Active Time value (T3324) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 2 IE in *3GPP TS 24.008* Table 10.5.163/3GPP TS 24.008.

The device can enter *PSM* state in LTE idle mode when the <Active-Time> parameter has a valid value where at least one of bits 6–8 is set to `0`. The length of PSM sleep is the value that remains from the value configured for periodic TAU after active time has elapsed, which is T3324 subtracted from T3412.

GPRS Timer 2 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the GPRS timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 2 s

0 0 1 – Value is incremented in multiples of 1 min

0 1 0 – Value is incremented in multiples of 6 min

1 1 1 – Value indicates that the timer is deactivated

<Periodic-TAU-ext>

String. 1 byte in 8-bit format.

Indicates the extended periodic TAU value (T3412_EXT extended value) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 3 IE in *3GPP TS 24.008* Table 10.5.163a/3GPP TS 24.008.

GPRS Timer 3 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the GPRS timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 10 min

0 0 1 – Value is incremented in multiples of 1 h

0 1 0 – Value is incremented in multiples of 10 h

0 1 1 – Value is incremented in multiples of 2 s

1 0 0 – Value is incremented in multiples of 30 s

1 0 1 – Value is incremented in multiples of 1 min

1 1 0 – Value is incremented in multiples of 320 h

1 1 1 – Value indicates that the timer is deactivated

<Periodic-TAU>

String. 1 byte in 8-bit format.

Indicates the periodic TAU value (T3412) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer IE in *3GPP TS 24.008* Table 10.5.172/3GPP TS 24.008.

Timer value (octet 2).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the GPRS timer as follows:

Bits

8 7 6

0 0 0 – Value is incremented in multiples of 2 s

0 0 1 – Value is incremented in multiples of 1 min

0 1 0 – Value is incremented in multiples of 6 min

1 1 1 – Value indicates that the timer is deactivated

The following command example reads modem parameters:

```
AT%XMONITOR
        %XMONITOR: 1,"EDAV","EDAV","26295","00B7",7,4,"00011B07",7,2300,63,39,"",
        "11100000","11100000","11100000"
        OK
```

8.11.2 Read command

The read command is not supported.

8.11.3 Test command

The test command is not supported.

8.12 Network time support %XNETTIME

The proprietary **%XNETTIME** command controls if the time received from the network is used. v2.0.x

8.12.1 Set command

The set command sets the requested network time support.

Network time support is enabled by default. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

The network time may be received in NITZ IE as a part of EMM information message or from SIB16. If both are available, the SIB16 time is prioritized due to higher accuracy. SIB16 reception has its own control options in addition to the network time support. If optional parameter for SIB16 support is not set, the functionality is as specified in parameter value 2.

Syntax:

```
%XNETTIME=<network_time_support>[,<sib16_support>]
```

The set command parameter and its defined values are the following:

<network_time_support>

0 – Disable network time support.

1 – Enable network time support.

<sib16_support>

0 – Support for time information from SIB16 is disabled.

1 – The modem supports time information from SIB16. When a cell is changed before starting idle mode operations or entering a connection, the modem attempts to receive SIB16 until it is received successfully. The cell selections might take more time, but the time information from SIB16 is more likely available for applications after a cell has been selected, for example, during the first connection that is established after modem activation.

2 – The modem supports time information from SIB16. The SIB16 reception does not cause delays in cell selections, but the time information from SIB16 is available only after the modem has been in RRC IDLE state long enough to capture the broadcast message.

Note:

- Setting <sib16_support> to `1` might cause a delay in establishing a connection after modem activation. The delay can be up to 40 seconds in *LTE-M* and up to 5 minutes 30 seconds in *NB-IoT* in every cell where connection is attempted until time information from SIB16 is received.
- Setting <sib16_support> to `1` or `2` does not stop the modem from selecting a cell if SIB16 reception fails, for example, due to poor radio conditions. The modem makes more attempts later to receive time information from SIB16 during normal idle mode operation before establishing a connection.

The following command example disables network time support:

```
AT%XNETTIME=0
OK
```

8.12.2 Read command

The read command reads network time support.

Response syntax:

```
%XNETTIME: <network_time_support>[,<sib16_support>]
```

The response parameter and its defined values are the following:

<network_time_support>

0 – Disable network time support.

1 – Enable network time support.

<sib16_support>

0 – Support for time information from SIB16 is disabled.

1 – The modem supports time information from SIB16. When a cell is changed before starting idle mode operations or entering a connection, the modem attempts to receive SIB16 until it is received successfully. The cell selections might take more time, but the time information from SIB16 is more likely available for applications after a cell has been selected, for example, during the first connection that is established after modem activation.

2 – The modem supports time information from SIB16. The SIB16 reception does not cause delays in cell selections, but the time information from SIB16 is available only after the modem has been in RRC IDLE state long enough to capture the broadcast message.

The following command example reads network time support:

```
AT%XNETTIME?
%XNETTIME: 0
OK
```

8.12.3 Test command

The test command is not supported.

8.13 Weak cell detection %XDEEPSEARCH

The proprietary **%XDEEPSEARCH** command supports averaging cell search mode to detect weak cells. v2.0.x

8.13.1 Set command

The set command sets the support for averaging cell search mode to detect weak cells.

The feature is available in *NB-IoT* and it increases the probability to find weak cells. When the setting is disabled, it stops the possible ongoing deep searches immediately. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **%XPOFWARN** warning blocks the storing to NVM.

> **Note:** Enabling this command reduces battery lifetime.

Syntax:

```
%XDEEPSEARCH=<deep_search>
```

The set command parameter and its defined values are the following:

<deep_search>

0 – Disable deep search

1 – Enable deep search

The following command example enables deep search support:

```
AT%XDEEPSEARCH=1
OK
```

8.13.2 Read command

The read command reads the status of deep search.

Response syntax:

```
%XDEEPSEARCH: <deep_search>
```

The response parameter and its defined values are the following:

<deep_search>

0 – Disable deep search

1 – Enable deep search

The following command example reads deep search availability:

```
AT%XDEEPSEARCH?
%XDEEPSEARCH:1
OK
```

8.13.3 Test command

The test command triggers averaging cell search mode to detect weak cells. The search is initiated when the next search due to unavailable network services is started.

> **Note:** The feature must be enabled using the set command before the test command can be successfully performed.

Response syntax:

```
%XDEEPSEARCH
```

The following command example triggers deep search:

```
AT%XDEEPSEARCH=?
AT%XDEEPSEARCH
OK
```

8.14 Mobile network operator %XOPCONF

The proprietary **%XOPCONF** command configures the modem for the selected mobile network operator. v2.0.x

> **Note:**
>
> - The **%XOPCONF** command is for testing purposes only.
> - The use of this command in the nRF91x1 modem firmware image can be permanently prevented with the **%XPRODDONE** command.

8.14.1 Set command

The set command configures the modem to comply with the requirements of various mobile network operators.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. The stored configuration is in use when the device is powered on.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

Syntax:

```
%XOPCONF=<op_conf>
```

The set command parameter and its defined values are the following:

<op_conf>

Integer

1 – Automatically detected from *International Mobile Subscriber Identity (IMSI)*

2 – Verizon

3 – AT&T

4 – China Telecom

5 – Softbank

6 – Telstra

7 – Bell

8 – LG U+

9 – KDDI

10 – Truphone

11 – Telus Jasper V2

12 – Vodafone

13 – T-Mobile

14 – Deutsche Telekom

15 – Rogers

16 – Orange

17 – KT

18 – Telefonica

The following command example sets Verizon operator configuration:

```
AT%XOPCONF=2
OK
```

8.14.2 Read command

The read command reads the current mobile network operator configuration.

Response syntax:

```
%XOPCONF: <op_conf>
```

The read command response parameter and its defined values are the following:

<op_conf>

Integer

1 – Automatically detected from *IMSI*

2 – Verizon

3 – AT&T

4 – China Telecom

5 – Softbank

6 – Telstra

7 – Bell

8 – LG U+

9 – KDDI

10 – Truphone

11 – Telus Jasper V2

12 – Vodafone

13 – T-Mobile

14 – Deutsche Telekom

15 – Rogers

16 – Orange

17 – KT

18 – Telefonica

The following command example reads the configured operator:

```
AT%XOPCONF?
AT%XOPCONF: 2
OK
```

8.14.3 Test command

The test command is not supported.

8.15 Release assistance indication %RAI

The proprietary **%RAI** command activates and deactivates *Release Assistance Indication (RAI)* support for access stratum and control plane. v2.0.x

In LTE, a device is not allowed to leave a power consuming connection by its own decision. The **%RAI** command activates support for the RAI feature where the device can indicate when it does not need the data connection anymore. The **%RAI** command sets on support for protocols, which means that the modem indicates to the network that it supports the RAI feature when connections are set up. The RAI feature can be used if also the network supports it.

In addition to using the **%RAI** command, the application must tell when the data connection is not needed anymore through the application domain data socket option interface. With the interface, the information can be given with the last data packet that is sent or after the application has received

the data packet that ends the need for the connection. Each socket client tells their own need for the connection, and the information is combined in the modem domain.

8.15.1 Set command

The set command activates and deactivates the support of the *RAI* feature. The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

Syntax:

```
%RAI[=<rai>]
```

The set command parameter and its defined values are the following:

<rai>

0 – Deactivate

1 – Activate

The following command example activates RAI feature support:

```
AT%RAI=1
OK
```

8.15.2 Read command

The read command reads the *RAI* support value.

Syntax:

```
%RAI: <rai>
```

The following command example reads the RAI support value:

```
AT%RAI?
%RAI: 1
OK
```

8.15.3 Test command

The test command is not supported.

8.16 Periodic cell search configuration %PERIODICSEARCHCONF

The proprietary **`%PERIODICSEARCHCONF`** command is used to configure periodic cell searches in limited service state. v2.0.x

8.16.1 Set command

The set command can be used to write, read, and clear periodic search configuration and to start a search. The configuration affects the periodic searches that the *UE* performs in limited service state to obtain normal service.

Syntax:

```
%PERIODICSEARCHCONF=<mode>[,<loop>,<return_to_pattern>,<band_optimization>,<pattern_1>
[,<pattern_2>][,<pattern_3>][,<pattern_4>]]
```

Response syntax for mode 1:

```
%PERIODICSEARCHCONF:
 <loop>,<return_to_pattern>,<band_optimization>,<pattern_1>[,<pattern_2>][,<pattern_3>]
[,<pattern_4>]
```

+CME ERROR code

513 – Not found.

527 – Invalid content.

The set command parameters and their defined values are the following:

<mode>

Selects the performed operation. The additional optional parameters are valid only for mode 0.

0 – Write.

1 – Read.

2 – Clear.

3 – Search now.

> **Note:** When <mode> is 3, the response is always `OK`, which indicates that the request has been delivered to the modem. The search is performed only when the modem is in sleep state between periodic searches.

<loop>

Indicates if the last given search pattern is looped from the beginning when the pattern has ended. If several search patterns are configured, <loop> impacts only the last pattern.

0 – No loop.

1 – Loop last search pattern.

<return_to_pattern>

Indicates if the modem can return to a given search pattern with shorter sleep periods, for example, when radio conditions change and the given pattern index has already been exceeded.

0 – No return.

1–4 – Return to search pattern index 1–4.

<band_optimization>

0 – No optimization. Every periodic search is an all band search.

1 – Use default optimizations predefined by the modem. Predefinition depends on the active data profile, which can be configured with **%XDATAPRFL**.

2 – Every second periodic search is an all band search.

3 – Every third periodic search is an all band search.

4 – Every fourth periodic search is an all band search.

…

20 – Every 20th periodic search is an all band search.

<pattern_1…pattern_4>

The write command and response to the read command can contain 1–4 search pattern entries. Each pattern entry is range or table type and enclosed in quotation marks ("pattern_1", "pattern_2").

Range type search pattern: "<type>,<initial_sleep>, <final_sleep>, [<time_to_final_sleep>], <pattern_end_point>"

Table type search pattern: "<type>,<val1>[,<val2>][,<val3>][,<val4>][,<val5>]"

The following parameters describe the content of <pattern 1...pattern4>:

<type>

0 – Range.

1 – Table.

<initial_sleep>

0–65535 s.

Sleep time between searches in the beginning of the range.

<final_sleep>

0–65535 s.

Sleep time between searches in the end of the range.

<time_to_final_sleep>

0–1080 min. Optional.

Target time for achieving the <final_sleep> value. This can be used to determine the angle factor between the <initial_sleep> and <final_sleep> times. The timeline for the <time_to_final_sleep> starts from the beginning of the search pattern.

If given, the value cannot be greater than the value of <pattern_end_point> in the same search pattern. If not given, the angle factor is calculated using the <pattern_end_point> value so that the <final_sleep> value is reached at the point of <pattern_end_point>.

<pattern_end_point>

0–1080 min.

Time that must elapse before entering the next search pattern. The timeline for <pattern_end_point> starts from the beginning of the limited service starting point which is the moment when the first sleep period started.

<val1...val5>

0–65535 s.

1–5 sleep time values between searches. After going through all values, the last value of the last search pattern is repeated if not configured differently by the <loop> or <return_to_pattern> parameters.

The following diagram shows an example of two configured search patterns' behavior on a timeline. The first search pattern is table type, and the second search pattern is range type.

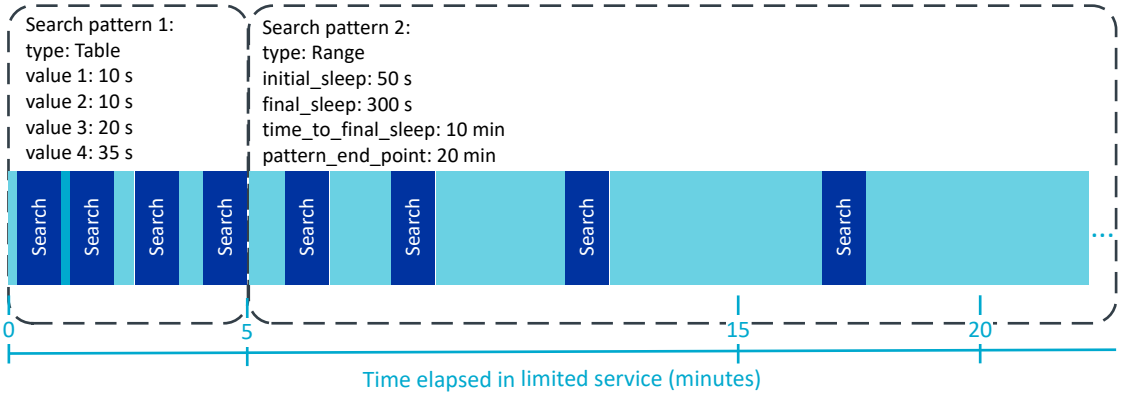

Figure 2: Example of two different types of search patterns

The following diagram shows the sleep lengths in search pattern 1 in Figure 2: Example of two different types of search patterns on page 167.

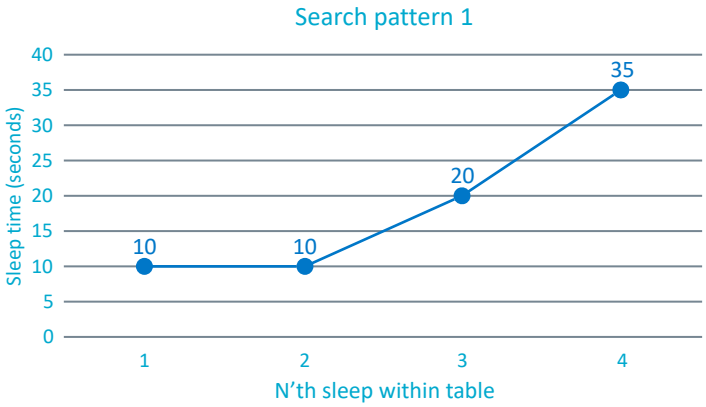

Figure 3: Sleep lengths in search pattern 1

The following diagram shows the sleep lengths in search pattern 2 in Figure 2: Example of two different types of search patterns on page 167. The sleep time reaches the <final_sleep> value in 10 minutes according to the <time_to_final_sleep> parameter.

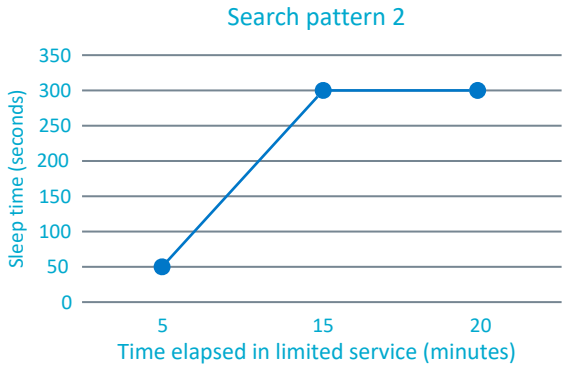

Figure 4: Sleep lengths in search pattern 2

The following command example configures one range type search pattern without the optional <time_to_final_sleep> parameter:

```
AT%PERIODICSEARCHCONF=0,0,0,10,"0,60,3600,,600"
OK
```

The following command example reads the configuration:

```
AT%PERIODICSEARCHCONF=1
%PERIODICSEARCHCONF: 0,0,10,"0,60,3600,,600"
OK
```

The following command example configures one range type search pattern with the optional <time_to_final_sleep> parameter and additional table type search pattern with three sleep time values that are looped when the pattern has ended:

```
AT%PERIODICSEARCHCONF=0,1,0,10,"0,60,3600,300,600","1,60,120,3600"
OK
```

The following command example requests an extra search now:

```
AT%PERIODICSEARCHCONF=3
OK
```

The following command example clears the configuration:

```
AT%PERIODICSEARCHCONF=2
OK
```

The following command example reads the configuration when no configuration is stored:

```
AT%PERIODICSEARCHCONF=1
%PERIODICSEARCHCONF:
OK
```

The modem uses specific periodic search patterns for each power level defined in the **%XDATAPRFL** command. The following commands correspond to the search pattern configurations used for each power level. They are default configurations, and any specific periodic cell search configuration overrides them.

Ultra low power:

```
AT%PERIODICSEARCHCONF=0,0,0,1,"0,10,40,,5","1,300,600,1800,1800,3600"
```

Low power:

```
AT%PERIODICSEARCHCONF=0,0,0,1,"0,10,40,,5","1,300,600"
```

Normal:

```
AT%PERIODICSEARCHCONF=0,0,0,1,"0,10,40,5,15","1,60,90,300"
```

Performance:

```
AT%PERIODICSEARCHCONF=0,0,0,1,"0,10,40,5,15","1,60,90"
```

High performance:

```
AT%PERIODICSEARCHCONF=0,0,0,1,"1,10","0,10,40,,15"
```

For more information on data profile power levels, see Data profile %XDATAPRFL on page 40.

8.16.2 Read command

The read command is not supported.

8.16.3 Test command

The test command is not supported.

8.17 Reduced mobility feature configuration %REDMOB

The proprietary **%REDMOB** command is used to reduce LTE idle mode mobility, such as swapping between cells in stationary use cases. The command can be used according to the relaxed monitoring standard defined in *3GPP TS 36.304 Ch. 5.2.4.12* or in a Nordic-proprietary way. v2.0.x

The 3GPP relaxed monitoring feature and the Nordic-proprietary reduced mobility feature are designed to reduce power consumption in stable radio conditions by limiting mobility and reducing measurements required for mobility. The actual power savings depend on the radio environment.

The main difference between the 3GPP relaxed monitoring feature and the Nordic-proprietary reduced mobility feature is that the 3GPP feature requires support from the network and the Nordic-proprietary feature does not.

> **Note:** The reduced mobility feature is designed to be used when the *UE* is stationary. Using it in mobile conditions can lead to suboptimal connectivity performance and increased power consumption.

8.17.1 Set command

The set command is used to control features that aim to reduce LTE idle mode mobility.

Syntax:

```
%REDMOB=<mode>
```

The set command parameter and its defined values are the following:

<mode>

0 – Default. Functionality according to the 3GPP relaxed monitoring feature.

1 – Enable Nordic-proprietary reduced mobility feature.

2 – Full measurements for best possible mobility. Disable the 3GPP relaxed monitoring and Nordic-proprietary reduced mobility features.

The following command example enables the Nordic-proprietary reduced mobility feature:

```
AT%REDMOB=1
OK
```

The following command example disables all reduced mobility functionality:

```
AT%REDMOB=2
OK
```

8.17.2 Read command

The read command returns the current configuration.

Response syntax:

```
%REDMOB: <mode>
```

The read command parameters and their defined values are the following:

<mode>

0 – Reduced mobility features disabled.

1 – Nordic-proprietary reduced mobility enabled.

2 – Reduced mobility features disabled.

The following command example reads the configuration when the Nordic-proprietary reduced mobility feature is enabled:

```
AT%REDMOB?
%REDMOB: 1
OK
```

The following command example reads the configuration when reduced mobility feature is disabled:

```
AT%REDMOB?
%REDMOB: 2
OK
```

8.17.3 Test command

The test command is not supported.

8.18 Feature configuration %FEACONF

The proprietary **%FEACONF** command is used to set support for the configurable features of the modem.
v2.0.x

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command. Active **`%XPOFWARN`** warning blocks the storing to NVM.

All configurable features are disabled by default.

> **Note:** Operator configurations can override values set by the user.

8.18.1 Set command

The set command sets support for the configurable features of the modem.

Syntax:

```
%FEACONF=<operation>[,<feature_id>[,<state>]]
```

Response syntax for <operation> read mode:

```
%FEACONF: <feature_id>,<state>
```

Response syntax for <operation> list mode:

```
%FEACONF: <feature_id_1>,<state_1>,<feature_id_2>,<state_2>,…<feature_id_n>,<state_n>
```

+CME ERROR code

518 – Not allowed in active state.

The set command parameters and their defined values are the following:

<operation>

0 – Write.

1 – Read.

2 – List.

<feature_id>

0 – Proprietary *PSM*.

Performs a PSM-like sleep when the network does not allow normal PSM usage. When taken into use, the modem goes to sleep the same way as it would if the network allowed the use of PSM. The sending of MO data or MO SMS automatically wakes up the modem the same way as from normal PSM sleep.

Proprietary PSM can be used only in application use cases where the device always initiates the data connection. This is because when the device is in proprietary PSM the network cannot reach it. If the network notices this, it can perform local deregistration for the device which would lead to a new registration and extra signalling when the device wakes up. The operator can take some unwanted actions if a large amount of devices are constantly unreachable when there is downlink data to be sent to the devices.

The length of the proprietary PSM sleep depends on the MO data or MO SMS cycles but also on the need to perform periodic *TAU* which is controlled by timer T3412. Typically, the T3412 timer is shorter when PSM is not used.

Can be changed when the modem is active.

1 – HPLMN skip in *PLMN* selection.

Used to optimize time and power consumption used for PLMN selection by ignoring *IMSI* HPLMN information during PLMN selection procedures.

The feature shall be used only when the home network (HPLMN) of the UICC does not physically exist. The application should consider this feature when using the UICC of a virtual operator.

Can be changed only when the modem is not activated.

2 – Recovery from unexpected release.

Forces the *UE* to perform IMSI Attach faster than normal when GUTI is used in Attach or TAU procedure and there are consecutive problems, such as unexpected RRC connection releases by the network or the network being unresponsive. The connection might be recovered by initiating IMSI Attach.

Can be changed only when the modem is not activated.

<state>

0 – Disabled.

1 – Enabled.

The following command example enables proprietary PSM:

```
AT%FEACONF=0,0,1
OK
```

The following command example reads proprietary PSM state:

```
AT%FEACONF=1,0
%FEACONF: 0,1
OK
```

The following command example lists the states of the supported features:

```
%FEACONF=2
%FEACONF: 0,1,1,1,2,0
OK
```

8.18.2 Read command

The read command is not supported.

8.18.3 Test command

The test command is not supported.

9 Mobile termination errors

For reference, see *3GPP 27.007 Ch. 9*.

9.1 Mobile termination error notification +CMEE

The **+CMEE** command disables or enables the use of the final result code `+CME ERROR`. v2.0.x

The following `+CME ERROR` codes can be received by all commands.

+CME ERROR code

0 – Phone failure. Invalid command, parameter, or other unexpected error.

23 – Memory failure. Unexpected error in memory handling.

50 – Incorrect parameters in a command.

60 – System error.

Other `+CME ERROR` codes are listed in command descriptions.

For reference, see *3GPP 27.007 Ch. 9.1*.

9.1.1 Set command

The set command disables or enables the use of the final result code `+CME ERROR`.

Syntax:

```
+CMEE=[<n>]
```

The set command parameter and its defined values are the following:

<n>

0 – Disable and use `ERROR` instead (default)

1 – Enable `+CME ERROR: <err>` result code and use numeric <err> values. <err> values are specified in *3GPP TS 27.007 Ch. 9.2*. Vendor-specific values listed in the command chapters, the value range starts from 512.

The following command example enables error codes in responses:

```
AT+CMEE=1
OK
```

9.1.2 Read command

The read command returns the current setting of <n>.

Response syntax:

```
+CMEE: <n>
```

The read command parameter and its defined values are the following:

<n>

0 – Disabled. `ERROR` used as the final response in case of failure.

1 – Enabled. `+CME ERROR: <err>` result code and numeric <err> values used.

The following command example reads the current error code setting:

```
AT+CMEE?
+CME: 1
OK
```

9.1.3 Test command

The test command returns supported values as a compound value.

Response syntax:

```
+CME: (list of supported <n>s)
```

The test command parameter and its defined values are the following:

<n>

0 – Disabled. `ERROR` used as the final response in case of failure.

1 – Enabled. `+CME ERROR: <err>` result code and numeric <err> values used.

The following command example returns the supported values:

```
AT+CMEE=?
+CME:(0,1)
OK
```

9.2 Network error code notification +CNEC

The **+CNEC** command activates or deactivates unsolicited reporting of error codes sent by the network.
v2.0.x

For reference, see *3GPP 27.007 Ch. 9.1B*.

9.2.1 Set command

The set command activates or deactivates unsolicited reporting of error codes sent by the network.

Syntax:

```
+CNEC=[<n>]
```

The set command parameters and their defined values are the following:

<n>

0 – Disable unsolicited error reporting

8 – Enable unsolicited result code `+CNEC_EMM: <error_code>[,<cid>]` to report *EPS* mobility management errors

16 – Enable unsolicited result code `+CNEC_ESM: <error_code>[,<cid>]` to report EPS session management errors

24 – Enable unsolicited result codes for `+CNEC_EMM: <error_code>[,<cid>]` and `+CNEC_ESM: <error_code>[,<cid>]`

<error_code>

3GPP TS 24.301 Table 9.9.3.9.1 for EPS mobility management errors codes

3GPP TS 24.301 Table 9.9.4.4.1 for EPS session management errors codes

<cid>

0–10. <cid> is present if <error_code> is related to a specific <cid>.

The following command example enables `CNEC_ESM` error codes:

```
AT+CNEC=16
OK
```

The following notification example shows *EMM* Cause 22 (Congestion) received from the network:

```
+CNEC_EMM: 22
```

9.2.2 Read command

The read command returns the current setting of <n>.

Response syntax:

```
+CNEC: <n>
```

The read command parameter and its defined values are the following:

<n>

0 – Disable

8 – `+CNEC_EMM` enabled

16 – `+CNEC_ESM` enabled

24 – `+CNEC_EMM` and `+CNEC_ESM`

The following command example reads the **+CNEC** error code setting, both `CNEC_EMM` and `CNEC_ESM` enabled:

```
AT+CNEC?
+CNEC: 24
OK
```

9.2.3 Test command

The test command returns the supported values as compound values.

Response syntax:

```
+CNEC: (list of supported <n>s)
```

The test command parameter and its defined values are the following:

<n>

0 – Disable

8 – **+CNEC_EMM** enabled

16 – **+CNEC_ESM** enabled

24 – **+CNEC_EMM** and **+CNEC_ESM**

The following command example returns the **+CNEC** error code setting values:

```
AT+CNEC?
+CNEC: (0,8,16,24)
OK
```

9.3 Extended error report +CEER

The **+CEER** command returns an extended error report. v2.0.x

For reference, see *3GPP 27.007 Ch. 6.10*.

9.3.1 Set command

The set command returns an extended error report.

Syntax:

```
+CEER
```

Response syntax:

```
+CEER: <report>
```

The set command parameter and its defined value are the following:

<report>

String. Information related to the last failure. Contains module information and the cause value. The module is one of the following values: OTHER, ESM, EMM, PDP, UICC, SMS.

The following command example reads the latest failure stored by the modem:

```
AT+CEER
+CEER: "SMS 301"
OK
```

9.3.2 Read command

The read command is not supported.

9.3.3 Test command

The test command is not supported.

10 SMS commands

For reference, see *3GPP 27.005 Ch. 3*.

10.1 Message format +CMGF

The **+CMGF** command sets message format. v2.0.x

For reference, see *3GPP 27.005 Ch. 3.2.3*.

10.1.1 Set command

The set command selects between *PDU* and text format.

This command can be issued only by a client registered with **+CNMI**.

Syntax:

```
+CMGF=[<mode>]
```

+CMS ERROR code

300 – ME failure.

302 – Not allowed.

The set command parameter and its defined value are the following:

<mode>

0 – PDU mode, default value

The following command example sets the message format to PDU mode:

```
AT+CMFG=0
OK
```

10.1.2 Read command

The read command is used to query the current message format.

Response syntax:

```
+CMGF: <mode>
```

The read command parameter and its defined value are the following:

<mode>

0 – *PDU* mode

The following command example reads the current message format:

```
AT+CMGF?
+CMGF: 0
OK
```

10.1.3 Test command

The test command lists the supported message formats.

Response syntax:

```
+CMGF: (list of <mode>s)
```

The test command parameter and its defined value are the following:

<mode>

0 – *PDU* mode

The following command example lists the supported message formats:

```
AT+CMGF=?
+CMGF: (0)
OK
```

10.2 New message indications +CNMI

The **+CNMI** command subscribes and configures new message indications. v2.0.x

For reference, see *3GPP 27.005 Ch. 3.4.1*.

10.2.1 Set command

The set command registers or unregisters an SMS client. Only one AT client can be registered as an SMS client. An existing registration must be released before registering a new client.

Syntax:

```
+CNMI=[<mode>[,<mt>[,<bm>[,<ds>]]]]
```

+CMS ERROR code

301 – SMS service of ME reserved.

302 – Not allowed.

The set command parameters and their defined values are the following:

<mode>

0 – Unsubscribe unsolicited result codes.

3 – Subscribe unsolicited result codes.

<mt>

0 – No received message notifications, the modem acts as an SMS client. Forces also <ds> to 0.

2 – SMS-DELIVERs (except class 2 and message waiting indication group) are routed directly to the TE using unsolicited result code `+CMT: [<alpha>],<length><CR><LF><pdu>`. TE needs to ACK with **+CNMA**.

<bm>

Ignored

<ds>

0 – No SMS-STATUS-REPORTs are routed to the TE. The only option if <mt> is set to 0.

1 – SMS-STATUS-REPORTs are routed to the TE using unsolicited result code: `+CDS: <length><CR><LF><pdu>`. TE needs to ACK with **+CNMA**.

The TE needs to handle both SMS-DELIVER and SMS-STATUS-REPORT or neither of them, <mt> and <ds> shall both be set to 0 at the same time, equals to <mode> 0.

The following command example registers as a client for mobile-terminated SMS and status reports:

```
AT+CNMI=3,2,0,1
OK
```

10.2.2 Read command

The read command is used to query how new messages are indicated.

Response syntax:

```
+CNMI: <mode>,<mt>,<bm>,<ds>,<bfr>
```

The read command parameters and their defined values are the following:

<mode>

0 – Do not forward unsolicited result codes to the *TE* (default)

3 – Forward unsolicited result codes directly to the TE

<mt>

0 – No received message notifications, the modem acts as an SMS client

2 – SMS-DELIVERs (except class 2 and message waiting indication group) are routed directly to the TE

<bm>

No CBM notifications are routed to the TE

<ds>

0 – No SMS-STATUS-REPORTs are routed to the TE

1 – SMS-STATUS-REPORTs are routed to the TE using unsolicited result code: `+CDS: <length><CR><LF><pdu>`

<bfr>

1 – The buffer of unsolicited result codes is cleared when <mode> 1 to 3 is entered

The following command example reads the configuration of how new messages are indicated:

```
AT+CNMI?
+CNMI: 3,2,0,1,1
OK
```

10.2.3 Test command

The test command is not supported.

10.3 Send message, PDU mode +CMGS

The **+CMGS** command sends a message in *PDU* mode. v2.0.x

For reference, see *3GPP 27.005 Ch. 3.5.1* and *3GPP 27.005 Ch. 4.3*.

10.3.1 Set command

The set command sends a message in *PDU* mode.

> **Note:** This command can be issued only by a client registered with the **+CNMI** command.

Syntax:

```
+CMGS=<length><CR><pdu><ctrl-Z>
```

Response syntax:

```
+CMGS: <mr>[,<ackpdu>]
```

+CMS ERROR code

300 – ME failure.

301 – SMS service of ME reserved.

304 – Invalid PDU mode parameter.

330 – SMSC address unknown.

331 – No network service.

332 – Network timeout.

517 – Modem not activated.

The set command parameters and their defined values are the following:

<length>

ASCII, 1–3 characters. Number of hexadecimal octets in <pdu>.

<pdu>

Hexadecimal numbers containing two *IRA* characters per octet.

<mr>

Message reference value.

<ackpdu>

RP-User-Data element of RP-ACK PDU.

<pdu> is expected to be received in the same command after <CR>. Interactive mode is not supported. PDU consists of hexadecimal numbers containing two IRA characters per octet.

The following command example sends the message "Testing a SMS messaging over LTE" to +358401234567, Service Center Address +448888888:

```
AT+CMGS=42<CR>069144888888F811000C915348103254760000 0B20D4F29C9E769F4161
         D0BC3D07B5CBF379F89C769F416F7B590E62D3CB<ctrl-z>
+CMGS: 2
OK
```

10.3.2 Read command

The read command is not supported.

10.3.3 Test command

The test command is not supported.

10.4 Received SMS notification in PDU mode +CMT

The **+CMT** command notifies of an unsolicited received message in *PDU* mode. v2.0.x

TE is expected to ACK received message with the **+CNMA** command.

For reference, see *3GPP 27.005 Ch. 3.4.1*.

The notification is subscribed using the **+CNMI** command.

Syntax:

```
+CMT: <alpha>,<length><CR><LF><pdu>
```

+CMS ERROR code

524 – Manufacturer-specific cause: SMS client has been unregistered.

The notification parameters and their defined values are the following:

<alpha>

String. TP-Originating-Address.

<length>

ASCII, 1–3 characters. Number of hexadecimal octets in <pdu>.

<pdu>

Hexadecimal numbers containing two *IRA* characters per octet.

The following example returns a notification of a received message "Testing a sms messaging over lte" from +358401234567, Service Center Address +44 888 8888:

```
+CMT: "+358401234567",28<CR><LF>069144888888F8D4F29C9E769F4161D0BC3D07B5CBF379F89C
       769F416F7B590E62D3CB
```

10.5 Delivery status notification in PDU mode +CDS

The **+CDS** command notifies of an unsolicited delivery status in *PDU* mode. v2.0.x

TE is expected to ACK received delivery report with the **+CNMA** command.

The notification is subscribed using the **+CNMI** command.

Syntax:

```
+CDS: <length><CR><LF><pdu>
```

+CMS ERROR code

524 – Manufacturer-specific cause: SMS client has been unregistered.

The notification parameters and their defined values are the following:

<length>

ASCII, 1–3 characters. Number of hexadecimal octets in <pdu>.

<pdu>

Hexadecimal numbers containing two *IRA* characters per octet.

The following example returns a delivery status notification with the recipient address, service center timestamp, and message delivery time:

```
+CDS: 25<CR><LF>060C91534810325476171160316255001711603152120000
OK
```

10.6 New message ACK, PDU mode +CNMA

The **+CNMA** command sends an ACK in *PDU* mode. v2.0.x

> **Note:** Text mode is not supported.

For reference, see *3GPP 27.005 Ch. 4.7*.

10.6.1 Set command

The set command sends a new message or delivery status ACK. A client receiving unsolicited notifications for new messages and delivery status is mandated to acknowledge those. This command can be used only when the modem is activated.

> **Note:**
>
> - This command can be issued only by a client registered with the **+CNMI** command.
> - After sending cause 22, the **%XSMMA** command needs to be used when memory is available.

If the *UE* does not get an acknowledgement within the required time (network timeout), it should respond as specified in *3GPP TS 24.011*, and UE/*TA* shall automatically disable routing to the *TE* by setting both <mt> and <ds> values of **+CNMI** to zero, that is, the SMS client gets unregistered. In that case, the *ME* informs the TE client with a `CMS ERROR: 524` notification.

Syntax:

```
+CNMA[=<n>[,<length>[<CR>PDU is given<ctrl-Z/ESC>]]]
```

+CMS ERROR code

300 – ME failure.

302 – Not allowed.

The set command parameters and their defined values are the following:

<n>

0 – The command operates in the same way as defined for the text mode, see New message ACK, text mode +CNMA on page 185

1 – Send RP-ACK

2 – Send RP-ERROR

<length>

ASCII, 1–3 characters. Number of hexadecimal octets in <pdu>.

<pdu>

Hexadecimal numbers containing two *IRA* characters per octet.

The following command example confirms the reception of a message, timestamp 06/11/2071 13:26:31:

```
AT+CNMA=1,9<CR>010017116031621300<ctrl-z>
OK
```

10.6.2 Read command

The read command is not supported.

10.6.3 Test command

The test command lists supported <n>s.

Response syntax:

```
+CNMA: (list of supported <n>s)
```

The test command parameter and its defined values are the following:

<n>

0 – The command operates in the same way as defined for the text mode

1 – Send RP-ACK

2 – Send RP-ERROR

The following command example lists the supported delivery statuses:

```
AT+CNMA=?
+CNMA: (0-2)
OK
```

10.7 New message ACK, text mode +CNMA

The **+CNMA** command sends a new message ACK in text mode. v2.0.x

> **Note:** Text mode is not supported.

For reference, see *3GPP 27.005 Ch. 3.4.4*.

10.7.1 Set command

The set command sends a new message ACK in text mode. This command can be used only when the modem is activated.

> **Note:** This command can be issued only by a client registered with the **+CNMI** command.

If the *UE* does not get an acknowledgement within the required time (network timeout), it should respond as specified in *3GPP TS 24.011* and the UE/*TA* shall automatically disable routing to *TE* by setting both <mt> and <ds> values of **+CNMI** to zero, that is, the SMS client gets unregistered.

Syntax:

```
+CNMA
```

The following command example sends a new message ACK in text mode:

```
AT+CNMA
OK
```

10.7.2 Read command

The read command is not supported.

10.7.3 Test command

The test command lists supported <n>s.

Response syntax:

```
+CNMA: (list of supported <n>s)
```

The test command parameter and its defined values are the following:

<n>

0 – The command operates in the same way as defined for the text mode

1 – Send RP-ACK

2 – Send RP-ERROR

The following command example lists the supported delivery statuses:

```
AT+CNMA=?
+CNMA: (0-2)
OK
```

10.8 Preferred message storage +CPMS

The **+CPMS** command selects the memory storage. v2.0.x

For reference, see *3GPP 27.005 Ch. 3.2.2*.

10.8.1 Set command

The set command sets the used memory.

Note: The modem does not support SMS memory, only direct routing to *TE*.

Syntax:

```
+CPMS=<mem1>[,<mem2>[,<mem3>]]
```

Response syntax:

```
+CPMS: <used1>,<total1>,<used2>,<total2>,<used3>,<total3>
```

+CMS ERROR code

303 – Not supported.

The set command parameters and their defined values are the following:

<mem1>

"MT" – Refers to all message storage areas associated with the modem

<mem2>

"MT" – Refers to all message storage areas associated with the modem

<mem3>

"MT" – Refers to all message storage areas associated with the modem

<usedx>

Integer. The number of messages currently in <memx>.

<totalx>

Integer. The number of messages currently in <memx>.

The following command example configures the used memory storages:

```
AT+CPMS="MT","MT","MT"
+CPMS: 0,0,0,0,0,0
OK
```

10.8.2 Read command

The read command is used to query memory status.

Response syntax:

```
+CPMS: <mem1>,<used1>,<total1>,<mem2>,<used2>,<total2>,<mem3>,<used3>,<total3>
```

The read command parameters and their defined values are the following:

<mem1>

"MT" – Refers to all message storage areas associated with the modem

<mem2>

"MT" – Refers to all message storage areas associated with the modem

<mem3>

"MT" – Refers to all message storage areas associated with the modem

<usedx>

Integer. The number of messages currently in <memx>.

<totalx>

Integer. The number of messages currently in <memx>.

The following command example reads the memory storage configurations:

```
AT+CPMS?
+CPMS: "MT",0,0,"MT",0,0,"MT",0,0
OK
```

10.8.3 Test command

The test command lists supported memories.

Response syntax:

```
+CPMS: (list of supported <mem1>s),(list of supported <mem2>s),(list of supported <mem3>s)
```

The test command parameters and their defined values are the following:

<mem1>

"MT" – Refers to all message storage areas associated with the modem

<mem2>

"MT" – Refers to all message storage areas associated with the modem

<mem3>

"MT" – Refers to all message storage areas associated with the modem

The following command example lists the supported memories:

```
AT+CPMS=?
+CPMS: ("MT"),("MT"),("MT")
OK
```

10.9 Message service failure result code +CMS ERROR

Message service failure result code **+CMS** is sent as error response to SMS-related commands. v2.0.x

For reference, see *3GPP 27.005 Ch. 3.2.5*.

Response syntax:

```
+CMS ERROR: <err>
```

The parameter and the values used by common messaging commands are the following:

<err>

0–127 – *3GPP TS 24.011 clause E.2* values.

128–255 – *3GPP TS 23.040 clause 9.2.3.22* values.

300–511 – *3GPP TS 27.005 Ch. 3.2.5*.

512≤ – Manufacturer specific.

513 – Manufacturer-specific cause: Not found.

514 – Manufacturer-specific cause: Not allowed.

515 – Manufacturer-specific cause: Memory full.

524 – Manufacturer-specific cause: SMS client has been unregistered.

10.10 SMS service +CGSMS

The **+CGSMS** command selects the SMS service. v2.0.x

For reference, see *3GPP 27.007 Ch. 10.1.21*.

10.10.1 Set command

The set command selects the SMS service.

Syntax:

```
+CGSMS=[<service>]
```

The set command parameter and its defined value are the following:

<service>

1 – Circuit-switched

Note: In a failure case, the command response is `ERROR` or `+CME ERROR`.

The following command example selects the circuit-switched SMS service:

```
AT+CGSMS=1
OK
```

10.10.2 Read command

The read command reads the current SMS service.

Response syntax:

```
+CGSMS: <service>
```

The read command parameter and its defined value are the following:

<service>

1 – Circuit-switched

The following command example reads the current SMS service:

```
AT+CGSMS?
+CGSMS: 1
OK
```

10.10.3 Test command

The test command lists the supported SMS services.

Response syntax:

```
+CGSMS: (list of currently available <service>s)
```

The test command parameter and its defined value are the following:

<service>

1 – Circuit-switched

The following command example lists the supported SMS services:

```
AT+CGSMS=?
+CGSMS: (1)
OK
```

10.11 Short message memory available %XSMMA

The proprietary **%XSMMA** command sends an *RP-SMMA* message. v2.0.x

10.11.1 Set command

The set command sends an *RP-SMMA* message.

The command is a trigger for the RP-SMMA message on the SMS stack to indicate to the Service Center that the *UE* has memory available and can receive mobile-terminated short messages. The client can set a memory full situation preventing incoming SMS messages by acking a mobile-terminated short message with `AT+CNMA=2` (the *PDU* parameter has to contain cause code 22 "Memory capacity exceeded"). Cause 300 is returned for all failures.

Syntax

```
%XSMMA
```

+CMS ERROR code

300 – ME failure.

517 – Modem not activated.

The following command example triggers sending the RP-SMMA on the SMS layer to release a memory full situation and to receive a response:

```
AT%XSMMA
OK
```

10.11.2 Read command

The read command is not supported.

10.11.3 Test command

The test command is not supported.

11 Production test commands

Production test AT commands can be used to test and verify products in device manufacturing. The *PTI* modem firmware supports a subset of AT commands for test purposes in non-signaling mode.

11.1 RX test %XRFTEST

The proprietary **%XRFTEST** command can be used to test the nRF91x1 receiver. The command enables the RF receiver and measures signal power at the *SiP* antenna port with a time domain power meter and returns the measurement result and headroom. v2.0.x

The **%XRFTEST** command allows starting RF to a desired frequency to test, debug, and verify configurations that depend on RF frequency, such as **%XMAGPIO** and **%XMIPIRFFECTRL**.

> **Note:** **%XRFTEST** cannot be used if the modem has already been activated with the **+CFUN** command.

11.1.1 Set command

The set command controls the RF receiver.

Syntax:

```
%XRFTEST=<test>,<operation>[,<param0>,<param1>,<param2>,<param3>]
```

The set command parameters and their defined values are the following:

<test>

0 – RX.

<operation>

1 – ON.

0 – OFF.

> **Note:** Always send OFF <operation> before sending another ON <operation>.

<param0>

3GPP band number. Use 24 for *GNSS* reception.

<param1>

Frequency 100 kHz.

Valid range 6000–22000 (corresponds to 600.0 MHz–2200.0 MHz).

> **Note:** If *Carrier Wave (CW)* is used, an offset of about 45 kHz for *NB-IoT* and 300 kHz for *LTE-M* is recommended.

<param2>

RX signal power at the SiP antenna port.

Valid range from −127 dBm to −25 dBm.

<param3>

System mode.

0 – NB-IoT.

1 – LTE-M.

Response syntax when <operation> is ON:

```
%XRFTEST: <antenna_power>, <headroom>
```

The response parameter and its defined value are the following:

<antenna_power>

Measured power at the SiP antenna port in q8 dBm. q8 means that dividing the result by 2^8 = 256 gives dBm.

The following command example enables the RF receiver for Band 1, 2140.0 MHz, −65 dBm, NB-IoT mode:

```
AT%XRFTEST=0,1,1,21400,-65,0
%XRFTEST: -17002, -16
OK
```

Note: −17002/256 = −66.4 dBm

The following command example disables the RF receiver:

```
AT%XRFTEST=0,0
OK
```

11.1.2 Read command

The read command is not supported.

11.1.3 Test command

The test command is not supported.

12 Security commands

Security commands can be used for device and data security.

12.1 PIN code +CPIN

The **+CPIN** command enters and checks the required *Personal Identification Number (PIN)*. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.3*.

12.1.1 Set command

The set command enters the *PIN*.

Syntax:

```
+CPIN=<pin>[,<newpin>]
```

The set command parameters and their defined values are the following:

<pin>

String of digits

<newpin>

String of digits. Mandatory if the required code is *SIM Personal Unblocking Key (PUK)* or SIM PUK2.

Note: If no PIN is required, the response code is `ERROR`.

The following command example enters PIN 1234:

```
AT+CPIN="1234"
OK
```

12.1.2 Read command

The read command checks if a *PIN* is needed or if a personalization lock is blocking the device start-up.

Response syntax:

```
+CPIN: <code>
```

The read command parameter and its defined values are the following:

<code>

READY – No PIN required

SIM PIN – PIN code required

SIM PUK – PUK code required

SIM PIN2 – PIN2 code required

SIM PUK2 – PUK2 code required

PH-SIM PIN – USIM depersonalization required

PH-NET PIN – Network depersonalization required

PH-NETSUB PIN – Network subset depersonalization required

PH-SP PIN – Service provider depersonalization required

PH-CORP PIN – Corporate depersonalization required

The following command example shows how to check if a PIN code is needed with the response that a PIN code is required:

```
AT+CPIN?
+CPIN: "SIM PIN"
OK
```

Note: Use **%XUSIMLCK** when facility lock depersonalization is required.

12.1.3 Test command

The test command is not supported.

12.2 Remaining PIN retries +CPINR

The **+CPINR** command returns the number of remaining *PIN* retries for the *UE* passwords. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.65*.

12.2.1 Set command

The set command returns the number of remaining *PIN* retries for the *UE* passwords.

Syntax:

```
+CPINR=<sel_code>
```

Response syntax for standard PINs:

```
+CPINR: <code>,<retries>
```

Manufacturer-specific PINs are not supported.

The set command parameters and their defined values are the following:

<sel_code>, <code>

SIM PIN

SIM PIN2

SIM PUK

SIM PUK2

Wildcard not supported.

<retries>

Integer. Number of remaining retries.

The following command example checks the remaining entries for PIN:

```
AT+CPINR="SIM PIN"
+CPINR: "SIM PIN",3
OK
```

12.2.2 Read command

The read command is not supported.

12.2.3 Test command

The test command is not supported.

12.3 Facility lock +CLCK

The **+CLCK** command locks, unlocks, or interrogates a facility. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.4*.

12.3.1 Set command

The set command locks, unlocks, or interrogates a facility.

Syntax:

```
+CLCK=<fac>,<mode>[,<passwd>]
```

+CME ERROR code

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<fac>

SC – SIM

PN – Network personalization

PU – Network subset personalization

PP – Service provider personalization

PC – Corporate personalization

PS – *USIM* personalization

<mode>

0 – Unlock

1 – Lock

2 – Query status

<passwd>

String. Password for the facility.

Note: SC is supported in modes 0 and 1. PN, PU, PP, PC, and PS are supported in modes 0 and 2.

The following command example disables *PIN* query:

```
AT+CLCK="SC",0,"<passwd>"
OK
```

The following command example checks network personalization status when it is not active:

```
AT+CLCK=”PN”,2
+CLCK: 0
OK
```

12.3.2 Read command

The read command is not supported.

12.3.3 Test command

The test command lists supported facilities.

Response syntax:

```
+CLCK: (list of supported <fac>s)
```

The following command example returns a list of supported facilities:

```
AT+CLCK=?
+CLCK: ("SC","PS","PN","PU","PP","PC")
OK
```

12.4 Change password +CPWD

The **+CPWD** command changes the password for the facility lock. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.5*.

12.4.1 Set command

The set command changes the password for the facility lock.

Syntax:

```
+CPWD=<fac>,<oldpwd>,<newpwd>
```

The set command parameters and their defined values are the following:

<fac>

SC – *SIM PIN*

P2 – SIM PIN2

<oldpwd>,<newpwd>

String. Password.

Note: Currently only SC is supported.

The following command example changes the SIM PIN:

```
AT+CPWD="SC","1234","5678"
OK
```

12.4.2 Read command

The read command is not supported.

12.4.3 Test command

The test command returns the supported facilities and password length.

Response syntax:

```
+CPWD: list of supported (<fac>,<pwdlength>)s
```

The test command parameters and their defined values are the following:

<fac>

SC – *SIM PIN*

P2 – SIM PIN2

<pwdlength>

Integer. Maximum length of the password.

The following command example returns a list of supported facilities and password lengths:

```
AT+CPWD=?
+CPWD: ("SC",8),("P2",8)
OK
```

12.5 Personalization of modem %XUSIMLCK

The proprietary **%XUSIMLCK** command allows personalizing the modem to work with predefined *USIM* cards. v2.0.x

12.5.1 Set command

The set command allows locking the modem to work with predefined *USIM* cards. Using the command, the modem can be personalized, depersonalized, or the lock of a category can be disabled if the category is not depersonalized.

It is also possible to configure USIM personalization so that the device is locked to the first USIM that is inserted to it.

According to *3GPP TS 22.022*, the following personalization options are available:

- Network
- Network subset
- Service provider
- Corporate
- USIM

Syntax:

```
%XUSIMLCK=<command>,<facility>,[<pwd>,[<permanent>,[<pers_data>]]]
```

The modem supports a maximum of 24 personalization codes.

+CME ERROR code

518 – Not allowed in active state.

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<command>

1 – Personalize

2 – Depersonalize

3 – Disable

4 – Lock device to the first inserted USIM. The value of <facility> must be PS.

<facility>

String

PN – Network personalization

PU – Network subset personalization

PP – Service provider personalization

PC – Corporate personalization

PS – USIM personalization

<pwd>

String. A password for enabling or disabling personalization. Used for <command> values 1, 2, or 4. The length of the password is 6–16 digits.

If PN Network Control Key, (NCK)

If PU Network Subset Control Key, (NSCK)

If PP Service Provider Control Key, (SPCK)

If PC Corporate Control Key, (CCK)

If PS Personalization Control Key, (PCK)

<permanent>

Programmable selection of the Control Key. Used only when the value of <command> is 1. The permanent Control Key can be programmed once, and it is therefore immutable once programmed.

0 – Nonpermanent Control Key

1 – Permanent Control Key

<pers_data>

String. Used only when the value of <command> is `1`.

When <facility> is PN, <pers_data> can contain a maximum of 24 pairs of *MCC* and *MNC* in the following format: `MCC1.MNC1:MCC2.MNC2:…:MCCn.MNCn`.

When <facility> is PU, <pers_data> can contain a maximum of 24 pairs of MCC +MNC+Network Subset Code (digits 6 and 7 of *IMSI*) in the following format: `MCC1.MNC1.D61.D71:MCC2.MNC2.D62.D72:...:MCCn.MNCn.D6n.D7n`, where D6x and D7x represent the sixth and seventh digits of IMSI.

When <facility> is PP, <pers_data> can contain a maximum of 24 USIM group identifiers for service provider personalization in the following format: `MCC1.MNC1.GID11:MCC2.MNC2.GID12:...:MCCn.MNCn.GID1n`. GID1x represents the first byte of EF_GID1 in USIM, see *3GPP TS 31.102 Ch. 4.2.10* EF_{GID1}.

When <facility> is PC, <pers_data> can contain a maximum of 24 pairs of USIM group identifiers from EF_{GID1} and 4.2.11 EF_{GID2} for corporate personalization in the following format: `MCC1.MNC1.GID11.GID21:MCC2.MNC2.GID12.GID22:...:MCCn.MNCn.GID1n.GID2n`.

GID1x and GID2x represent the first bytes of EF_GID1 and EF_GID2, see *3GPP TS 31.102 Ch. 4.2.10* EF_{GID1} and *4.2.11* EF_{GID2}.

When <facility> is PS, <pers_data> can contain a maximum of 24 IMSIs as specified in *3GPP TS 31.102 Ch. 4.2.2* EF_{IMSI}. Fifteen IMSI digits can be given. The format is the following: `IMSI1:IMSI2:...:IMSIn`.

The following command example creates a nonpermanent network personalization:

```
AT%XUSIMLCK=1,"PN","12345678",0,"100.200"
OK
```

The following command depersonalizes the network personalization:

```
AT%XUSIMLCK=2,"PN","12345678"
OK
```

This command disables network personalization:

```
AT%XUSIMLCK=3,"PN"
OK
```

This command locks device to the first inserted USIM in a nonpermanent manner:

```
AT%XUSIMLCK=4,"PS","12345678",0
OK
```

This command personalizes USIM to IMSI 100200777777777 (MCC=100, MNC=200, other digits are 7). The facility PS is permanently locked to password "12345678". After depersonalization, no other keys can be used for this facility:

```
AT%XUSIMLCK=1,"PS","12345678",1,"100200777777777"
OK
```

12.5.2 Read command

The read command is not supported.

12.5.3 Test command

The test command is not supported.

12.6 Authenticated access %XSUDO

The proprietary **%XSUDO** command provides authenticated access for a restricted AT command. v2.0.x

> **Note:** This command is for future releases. In the current software release, the use of this command is not required.

For information on the usage of the command, see Authenticating AT command usage on page 237.

12.6.1 Set command

The set command provides authenticated access for a restricted AT command.

The restricted command is separated with a semicolon (;). The leading AT prefix is not included in the concatenated command.

Syntax:

```
%XSUDO=<data_len>,<signature>[,<sec_tag>]
```

+CME ERROR code

513 – Not found, public key not found.

514 – Not allowed.

520 – Authentication failed.

The set command parameters and their defined values are the following:

<data_len>

Length of a signed command string.

Only the number of characters in <data_len> from an authenticated command is processed, the rest are ignored. <data_len> shall not be greater than the given command.

<signature>

Command signature in Base64 format.

<sec_tag>

Integer, 0–9 (optional). A secure tag for multiple public keys.

The following command example provides authenticated access for the restricted **+CMD** command:

```
AT%XSUDO=28,"c2lnbmF0dXJl";+CMD=...
OK
```

12.6.2 Read command

The read command is not supported.

12.6.3 Test command

The test command is not supported.

12.7 Public key storage management %XPMNG

The proprietary **%XPMNG** command writes and reads the public key. The public key can be written only if it does not exist. v2.0.x

An existing key can be deleted with the **%CMNG** command.

For information on the usage of the command, see Authenticating AT command usage on page 237.

12.7.1 Set command

The set command writes and reads the public key. The public key is updated directly to *NVM*. The write operation is allowed only when the modem is not activated.

Active **%XPOFWARN** warning blocks the update.

Syntax:

```
%XPMNG=<opcode>[,<content>[,<sec_tag>]]
```

Response syntax for read command:

```
%XPMNG: <content>
```

+CME ERROR code

513 – For read: Not found.

514 – Not allowed.

515 – Memory full.

518 – Not allowed in active state.

519 – Already exists, for write.

520 – For write: Already exists.

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<opcode>

0 – Write

2 – Read

<content>

String. Mandatory if parameter <opcode> is "Write". An empty string is not allowed. Parameter <content> is enclosed in double quotes. ASN.1 DER encoding in Base64 encoded with the header and footer of begin key and end key.

<sec_tag>

Integer, 0–9 (optional). A secure tag for multiple public keys.

The following command example writes the public key:

```
AT%XPMNG=0,"-----BEGIN PUBLIC KEY-----...-----END PUBLIC KEY-----"
OK
```

The following command example reads the public key:

```
AT%XPMNG=2
%XPMNG: "-----BEGIN PUBLIC KEY-----...-----END PUBLIC KEY-----"
OK
```

12.7.2 Read command

The read command is not supported.

12.7.3 Test command

The test command is not supported.

12.8 Credential storage management %CMNG

The proprietary **%CMNG** command is used for credential storage management. The command writes, reads, deletes, and checks the existence of keys and certificates. v2.0.x

12.8.1 Set command

The set command is used for credential storage management. The command writes, reads, deletes, and checks the existence of keys and certificates. The keys and certificates are updated directly to *NVM* when the set command is issued.

Active **%XPOFWARN** warning blocks the update.

The write and delete operations are allowed only when the modem is not activated.

Syntax:

```
%CMNG=<opcode>[,<sec_tag>[,<type>[,<content>[,<passwd>]]]]
```

Response syntax for read operation:

```
%CMNG: <sec_tag>,<type>[,<sha>[,<content>]]
```

Response syntax for list operation:

```
%CMNG: <sec_tag>,<type>[,<sha>]
```

<sec_tag> <type> shall be a unique pair, no multiple items with the same <sec_tag> and <type> values are allowed.

+CME ERROR code

513 – Not found. Applies to read, write, and delete.

514 – No access. Applies to read, write, and delete.

515 – Memory full. Applies to write.

518 – Not allowed in active state.

519 – Already exists.

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<opcode>

0 – Write

1 – List

2 – Read

3 – Delete

<sec_tag>

Integer, 0–2147483647.

Mandatory for write, read, and delete operations. Optional for list operation.

Values above 2147483647 are reserved for the modem. <sec_tag>s with values above 2147483647 might be visible in the **%CMNG** list command.

Transport Layer Security (TLS) traffic can be decrypted with Nordic tools if the TLS session is created using certificates stored to <sec_tag>s 2147483648– 2147483667. These <sec_tag>s shall be used only for test and development purposes.

<type>

0 – Root CA certificate (ASCII text).

1 – Client certificate (ASCII text).

2 – Client private key (ASCII text).

3 – *Pre-shared Key (PSK)* (ASCII text in hexadecimal string format).

4 – PSK identity (ASCII text).

5 – Public key (ASCII text). Used in authenticated AT commands.

6 – Device identity public key.

7 – Reserved.

8 – Endorsement private key, public key for <opcode> 2 – read. Used for content encryption. See **%KEYGEN** and **%KEYINJECT**.

9 – Ownership key. Used for signature verification. See **%KEYGEN** and **%KEYINJECT**.

10 – Nordic identity root CA. Root CA for Nordic identity server.

11 – Nordic base public key. Used for ownership key provisioning message signature verification.

13 – Asset encryption key

<content>

ASCII text in hexadecimal string format containing two *IRA* characters per octet (PSK). Mandatory if <opcode> is `write`. An empty string is not allowed. A *Privacy Enhanced Mail (PEM)* file enclosed in double quotes (X.509 PEM entities).

<passwd>

String. PKCS#8 password. Mandatory for writing a type 2 encrypted private key, ignored for other types. Maximum length 32 characters.

Not supported in current modem firmware versions.

<sha>

String in hexadecimal format. SHA-256 digest of the entity (*DER*, PEM) as stored in the filesystem. 64 characters representing a 256-bit vector.

Note:

- <content> in the read response is exactly what is written, including `<CR>`, `<LF>`, and other characters. The characters outside the double quotes are part of the AT response format.
- Reading types 1, 2, and 3 are not supported.
- Writing and deleting types 8, 10, and 11 are not supported.
- Overwriting and deleting type 9 is not supported.

The following command example writes the root certificate:

```
AT%CMNG=0, 12345678, 0,"
-----BEGIN CERTIFICATE-----
MIIDSjCCA...
...bKbYK7p2CNTUQ
-----END CERTIFICATE-----”
OK
```

The following command example writes the client certificate:

```
AT%CMNG=0,567890,1,"
-----BEGIN CERTIFICATE-----
MIIBc464...
...bW9aAa4
-----END CERTIFICATE-----”
OK
```

The following command example writes the private key:

```
AT%CMNG=0,123,2,"
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIICz...
...ukBu
-----END ENCRYPTED PRIVATE KEY-----”, ”abcdefg”
OK
```

The following command example lists a single item by specifying tag and type:

```
AT%CMNG=1,12345678, 0
%CMNG: 12345678, 0, "978C...02C4"
OK
```

The following command example lists a single tag:

```
AT%CMNG=1,12345678
%CMNG: 12345678, 0, "978C...02C4"
%CMNG: 12345678, 1, "1A8C...02BB"
OK
```

The following command example lists all stored credentials:

```
AT%CMNG=1
%CMNG: 12345678, 0, "978C...02C4"
%CMNG: 567890, 1, "C485...CF09"
%CMNG: 123, 2, "92E1...8AC8"
%CMNG: 654321, 3, "E0C9...511D"
OK
```

The following command example reads the root certificate with tag 12345678:

```
AT%CMNG=2, 12345678, 0
%CMNG: 12345678, 0, "978C...02C4",
"-----BEGIN CERTIFICATE-----
MIIBc464...
...bW9aAa4
-----END CERTIFICATE-----"
OK
```

The following command example deletes a client certificate with tag 123:

```
AT%CMNG=3,123,1
OK
```

The following command example reads a non-existing root certificate with tag 4567. Error code 513 is returned:

```
AT%CMNG=2,4567,0
+CME ERROR: 513
```

12.8.2 Read command

The read command is not supported.

12.8.3 Test command

The test command is not supported.

12.9 Attestation token generation %ATTESTTOKEN

The proprietary **%ATTESTTOKEN** command requests an attestation token. Attestation tokens are used to verify the authenticity of a Nordic device. v2.0.x

12.9.1 Set command

The set command requests an attestation token.

The response contains a device identity attestation message including the device type, device UUID, and COSE authentication metadata joined by a dot "." and coded to Base64url format.

Syntax:

```
%ATTESTTOKEN=[<challenge>]
```

Response syntax:

```
%ATTESTTOKEN: <response>
```

The set command parameters and their defined values are the following:

<challenge>

Optional.

32-character string in hexadecimal format.

If present, <challenge> is included in the device identity attestation message instead of device-generated nonce.

<response>

Device identity attestation message and COSE_Sign1 signature as Base64Url encoded strings separated by a dot '.':

`"Base64Url(Attestation message).Base64Url(Cose_Sign1)"`

The following command example requests an attestation token:

```
AT%ATTESTTOKEN
%ATTESTTOKEN: "9aiUFOq7SOwtdPfEgdkPxV_vTYLoGeE0-gzV2e5c.jUfK3OpdVBVYWnso-
aOjIZcSTAwdqQQJWrrY2Lnc"
OK
```

The following command example requests an attestation token with input challenge:

```
AT%ATTESTTOKEN="90115c20ac3f381b520b43c76bd5d99c"
%ATTESTTOKEN: "ksjdfEjAgfgdgfg.fghgehhgSyrlaskj"
OK
```

12.9.2 Read command

The read command is not supported.

12.9.3 Test command

The test command is not supported.

12.10 Key generation %KEYGEN

The proprietary **%KEYGEN** command allows creating keys for different purposes. v2.0.x

12.10.1 Set command

The set command creates keys for different purposes.

The purposes are the following:

Client private key and certificate signing request (CSR) (<key_type> 2, <response_content> 0)

The CSR can be passed to a Certificate Authority (CA) for requesting a client certificate for the device. The modem stores the generated private key to a <sec_tag>. The client certificate must be stored to the device in the same <sec_tag> as the client private key. The client device certificate and the corresponding private key can be used as TLS client credentials.

The **%KEYGEN** response contains a Certificate Signing Request (CSR) and COSE signature for the CSR separated by a dot "." with both sides Base64Url-encoded: `"Base64Url(CSR_DER).Base64Url(cose_sign)"`.

Client private key and public key (<key_type> 2, <response_content> 1)

The **%KEYGEN** response contains a public key in CBOR device pubkey message and COSE signature separated by dot "." with both sides Base64Url-encoded: `"Base64Url(public_key_DER).Base64Url(cose_sign)"`.

Device Endorsement key pair (<key_type> 8)

Endorsement keys can be used to pass encrypted data to a device. The endorsement private key is stored to the device in a <sec_tag>, and the public key is returned in a **%KEYGEN** response. If the private key already exists in <sec_tag>, the response is created based on the existing key instead of creating a new key.

The **%KEYGEN** response contains an Endorsement Public key in CBOR device pubkey message and COSE signature separated by dot "." with both sides Base64Url-encoded: `"Base64Url(public_key_DER).Base64Url(cose_sign)"`.

Note:

- The **%KEYGEN** command is allowed only when the modem is deactivated.
- The **%CMNG** list operation can be used to check the availability of generated keys.

Syntax:

```
%KEYGEN=<sec_tag>,<key_type>[,<response_content>][,<attributes>][,<key_usage>]
```

+CME ERROR code

514 – Not allowed.

515 – Memory full.

518 – Not allowed in active state.

519 – Already exists.

523 – Key generation failed.

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<sec_tag>

Integer, 1–2147483647.

<key_type>

Integer

Type of the decryption key. Values corresponding to <type> in **%CMNG**.

Supported values:

2 – Client private key

8 – Endorsement key

<response_content>

Integer

Mandatory for <key_type> 2. Optional for <key_type> 8.

Content of the response:

0 – Certificate signing request returned in response (<key_type> 2 only)

1 – Public key returned in response (<key_type> 2 or 8)

<attributes>

String

Optional. Valid only for CSR (<response_content> 0), otherwise ignored. The content of the string is a comma-separated list of attribute ID and value pairs, for example, "ID1=value1,ID2=value2", where the ID can be one of the supported attributes listed below, and the values provide the corresponding information (see *RFC 5280*).

Supported attributes:

commonName (CN)

locality (L)

stateOrProvinceName (ST)

organizationName (O)

organizationalUnitName (OU)

countryName (C)

domainComponent (DC)

surName (SN)

givenName (GN)

emailAddress (R)

serialNumber

postalAddress

postalCode

dnQualifier

title

initials

pseudonym

generationQualifier

If the <attributes> parameter is omitted, the default is "CN=<device-uuid>".

<key_usage>

String

Optional. Valid only for CSR (response_content 0), otherwise ignored. The string contains 9 binary characters (0/1) corresponding to the KeyUsage bit string defined in *RFC 5280*. The digits or bits in the string are numbered 0–8 from left to right, and they have the following meaning (see *RFC 5280* for more details).

0 – digitalSignature (the first digit)

1 – nonRepudiation

2 – keyEncipherment

3 – dataEncipherment

4 – keyAgreement

5 – keyCertSign

6 – cRLSign

7 – encipherOnly

8 – decipherOnly (the last digit)

> **Note:** Setting decipherOnly bit to `1` is not supported.

If the <key_usage> parameter is omitted, default is `111010000`, which means that the digitalSignature, nonRepudiation, keyEncipherment, and keyAgreement bits are set as `1`.

The following command example generates a key for TLS. The key is stored to <sec_tag> 11. The <key_type> is client private key, and it includes the optional <key_usage> parameter:

```
AT%KEYGEN=11,2,0,,"101010000"
%KEYGEN: "ckijfiok.jdiqhdn"
OK
```

The following command example generates a key for TLS. The key is stored to <sec_tag> 11. The <key_type> is client private key, and it includes the optional parameters <attributes> and <key_usage>:


```
AT%KEYGEN=11,2,0,"O=Nordic Semiconductor,L=Trondheim,C=no","101010000"
%KEYGEN: "MIIBHDCBwAIBADBAMR0wGwYDVQQKDBROb3JkaWMgU2VtaWNvbmR1Y3RvcjESMBAGA1UEB
wwJVHJvbmRoZWltMQswCQYDVQQGEwJubzBZMBMGByqGSM49AgEGCCqGSM49AwEHA0IABAP6p4ZjYNG0
FFdz5axZkbcQzfO8Fkzws4o2unhS5qqgON5BA6jj0TbHJ5imPbVXWNiDVCNc0us41NtL3TygDdqgHjA
cBgkqhkiG9w0BCQ4xDzANMAsGA1UdDwQEAwIDqDAMBggqhkjOPQQDAgUAA0kAMEYCIQDnFWQl-x25qj
EPBtLrvH5wwecrA5KmjUcK_b3eok5nmQIhALsSKRmuqDM3H09-q4R0MSpYprWZZp5iPjRMtZbjataH.
0oRDoQEmoQRBIVhL2dn3hQlQUFAwQTYzQ7CAwwEaNJbawEELWCD7er2zKHc8yAMi8x-1jAuCEqaLmJ3
PMZTHIYSrmIR0tlD7CDRaW--s0c7XA4pI3r9wWEAQOGVg5VejmHx9o_sF1RcGBMClw-MkN_wMrS2ryM
8CdA427j96z_4wjLgBvejVnYGAl7qqL77qlEbBEzXOkLx4"
OK
```


The following command example generates an endorsement key. The key is stored to <sec_tag> 33. The <key_type> is endorsement private key:

```
AT%KEYGEN=33,8
%KEYGEN: "vdofjihew.cnjeiwhfok"
OK
```

12.10.2 Read command

The read command is not supported.

12.10.3 Test command

The test command is not supported.

12.11 Key injection %KEYINJECT

The proprietary **%KEYINJECT** command injects encrypted provisioning messages to the nRF91x1 modem. The command can be used to inject encrypted certificates and keys to the device. v2.0.x

12.11.1 Set command

The set command injects encrypted provisioning messages to the nRF91x1 modem.

Syntax:

```
%KEYINJECT=<msg>
```

Response syntax:

```
%KEYINJECT: <response>
```

+CME ERROR code

50 – Incorrect parameters.

51 – Command is disabled in this software build.

513 – Signing or decrypting key not found on device.

515 – Not enough memory for processing.

518 – Not allowed in active state.

528 – Not allowed when power off warning is active.

The set command parameters and their defined values are the following:

<msg>

Provisioning message and signature separated by a dot ".". The actual message is encrypted and the ciphertext is delivered in a COSE_Encrypt object.

The format is "Base64url(COSE_Encrypt(provisioning_message)).Base64url(COSE_Sign1)".

<response>

The response format is base64url(cbor_resp).

The following command example shows the nRF91x1 modem receiving an encrypted provisioning message:

```
%KEYINJECT="2GCEQ6EBCqEFTS6b97seIkh_t5cLNhpYikK5miurjQrYHxWuPaA_LQFo5Qk2s9g3xO71g8ec-
AVvKRyk4Gdo2DCRL9q3gGOZxgrYEN9R44F1WOdaGhnaypjtdNwqgu-
Kv-6DlZRqyxKkSvrZLuOq3R6GdLPK2BZ5pQNYkeN
8dIPhJujbA5Bb_bt_iUrlYA2jIF3MR395PDtd4BlTb4ika-
tVeoGDRKEBOBiiIKQBAiABIVgg8zFHgdz6a3LjQ2Nq7V-jM
u9CM3KxI9XjPajLT8fOKGkiWCDi4YnIiCDQisIPWoCaucAnUUti4h-
eazw1UMX3LR3X_ARCGC1A.0oRYH6MBJjoAAQAAUL
UTc_owres5hDTJSUsd4r46AAEAAQShBEMZAtP2WEAIQ9Gw3O2-
ymC1_bMChHdYorwSPGSK7okxl9Xkj3V7JVrUW1UK7QeW
M1ajtaxNYs7wJARQdeSmMbNTF6R0_VuD"
%KEYINJECT: "2dn3gwVQ_______RK2Afv__z-ihGIEA"
OK
```

12.11.2 Read command

The read command is not supported.

12.11.3 Test command

The test command is not supported.

12.12 Creating JWT %JWT

The proprietary **%JWT** command creates a JSON Web Token (JWT). v2.0.x

12.12.1 Set command

The set command creates a *JSON Web Token (JWT)*.

Syntax:

```
%JWT=[<alg>],[<exp_delta>],[<subject>],[<audience>][,<sec_tag>,<key_type>]
```

Response syntax:

```
%JWT: <jwt>
```

+CME ERROR code

50 – Incorrect parameters.

51 – Command is disabled in this software build.

513 – Key not found (<sec_tag>, <key_type>).

514 – Could not read key (<sec_tag>, <key_type>).

525 – Error in JWT creation.

Note: If **%JWT** is given without the optional <sec_tag> and <key_type> parameters, device identity private key is used to sign the JWT.

The set command parameters and their defined values are the following:

<alg>

Integer

Defines the JWT signing algorithm. The default value is 0. Other values are currently not supported.

0 – ES256

<exp_delta>

Integer

The number of seconds before expiry. The modem adds the <exp_delta> value to the current time to determine the "exp" (expiration time) claim defined in *RFC 7519 4.1.4* identifying the time when or after which the JWT must not be accepted for processing.

If <exp_delta> is not given, or the value is 0, the "exp" and "iat" claims are not included in the JWT.

> **Note:** To get valid "exp" and "iat" claims, the device must have a valid date and time that can be received from the network or set manually (see the `+CCLK` and `%CCLK` commands).

<subject>

String

The "sub" (subject) claim for the JWT as defined in *RFC 7519 4.1.2*. Identifies the principal that is the subject of the JWT.

<audience>

String

The "aud" (audience) claim for the JWT as defined in *RFC 7519 4.1.3*. Identifies the recipients that the JWT is intended for.

> **Note:** Entering an array of multiple strings is not supported. Only a single audience string supported.

<sec_tag>

Integer

Identifies the key to be used for signing the JWT. The values correspond to the <sec_tag> parameter in `%CMNG`.

<key_type>

Integer

Type of the key to be used for signing the JWT. The values correspond to the <type> parameter in `%CMNG`.

2 – Client private key

<jwt>

String

Created JWT as Base64Url encoded string.

The following command example creates a JWT:

```
AT%JWT=0,3600,"ClientAuth","urn:server"
```

12.12.2 Read command

The read command is not supported.

12.12.3 Test command

The test command is not supported.

12.13 Certificate expiry information %CERTEXPIRY

The proprietary **%CERTEXPIRY** command reads the *TLS* certificate expiry information. v2.0.x

12.13.1 Set command

The set command reads the *TLS* certificate expiry information.

Syntax:

```
%CERTEXPIRY=<sec_tag>,<type>
```

Response syntax:

```
%CERTEXPIRY: <serial number>,<notBefore>,<notAfter>
```

The set command parameters and their defined values are the following:

<sec_tag>

Integer, 1–2147483647.

<type>

0 – Root CA certificate.

1 – Client certificate.

10 – Nordic identity root CA.

<serial number>

String in hexadecimal format. Certificate serial number.

<notBefore>

Certificate not valid before.

Format: yyyymmddhhmmssZ

yyyy – Four digits for the year.

mm – Two digits for the month.

dd – Two digits for the date.

hh – Two digits for hours.

mm – Two digits for minutes.

ss – Two digits for seconds.

Z – Time zone Zulu/UTC.

<NotAfter>

Certificate not valid after.

Format: yyyymmddhhmmssZ

+CME ERROR code

513 – Not found.

527 – Invalid content.

The following command example reads the expiry information of the root certificate stored in <sec_tag> 1:

```
%CERTEXPIRY=1,0
```

The following response example reads the expiry information of a certificate with serial number 00AEF87681FA968B61 that is valid from August 23, 2017 8:47:37 AM GMT to August 23, 2018 8:47:37 AM GMT:

```
%CERTEXPIRY: 00AEF87681FA968B61,20170823084737Z,20180823084737Z
```

12.13.2 Read command

The read command is not supported.

12.13.3 Test command

The test command is not supported.

12.14 Claim token generation %CLAIMTOKEN

The proprietary **%CLAIMTOKEN** command requests the device's claim token. v2.0.x

Claim token is proof that the ownership public key has been provisioned onto a particular device. The device signs the digest of the ownership public key with its identity key. The owner then signs the claim tokens with the ownership private key. Using the ownership public key, the nRF Identity service can determine that the ownership claim of the device is valid.

The response contains a device claim token message including the device type, device *UUID*, firmware UUID, ownership public key digest, and COSE authentication metadata COSE_Sign1 signature separated by a dot '.' and encoded to Base64Url format:

```
"Base64Url(Device claim token).Base64Url(Cose_Sign1)". <response>
Device claim token message and COSE_Sign1 signature as Base64Url
encoded strings separated by a dot ".": "Base64Url(Device claim
token).Base64Url(Cose_Sign1)"
```

12.14.1 Set command

The set command requests the device's claim token.

Syntax:

```
%CLAIMTOKEN=<sec_tag>[,<challenge>]
```

Response syntax:

```
%CLAIMTOKEN: <response>
```

+CME ERROR code

529 – Failed to create response. Ownership key is not found at the given <sec_tag> or an error has occurred in secure side processing.

The set command parameters and their defined values are the following:

<sec_tag>

Integer, 0–2147483647.

Identifies the <sec_tag> where the ownership public key is stored on the device. The values correspond to the <sec_tag> in **%CMNG**.

<challenge>

Optional.

32-character string in hexadecimal format.

Used to generate a response.

<response>

Device claim token message and COSE_Sign1 signature as Base64Url encoded strings separated by a dot ".": `"Base64Url(Device claim token).Base64Url(Cose_Sign1)"`

The following command example requests the claim token. The ownership key is stored at <sec_tag> 22:

```
AT%CLAIMTOKEN=22
%CLAIMTOKEN: "RGV2aWNlIGNsYWltIHRva2Vu.Q29zZV9TaWduMQ"
OK
```

The following command example requests the claim token including challenge. The ownership key is stored at <sec_tag> 34:

```
AT%CLAIMTOKEN=34,"90115c20ac3f381b520b43c76bd5d99c"
%CLAIMTOKEN: "RGV2aWNlIGNsYWltIHRva2Vu.Q29zZV9TaWduMQ"
OK
```

12.14.2 Read command

The read command is not supported.

12.14.3 Test command

The test command is not supported.

13 UICC access commands

UICC access commands can be used to access UICC and receive notifications of the UICC state.

13.1 UICC state notification %XSIM

The proprietary **%XSIM** command subscribes *UICC* state notifications. v2.0.x

13.1.1 Set command

The set command subscribes *UICC* state notifications.

Syntax:

```
%XSIM=<n>
```

Notification syntax:

```
%XSIM: <state>[,<cause>]
```

The set command parameters and their defined values are the following:

<n>

0 – Unsubscribe UICC notifications

1 – Subscribe UICC notifications

<state>

0 – UICC not initialized

1 – UICC initialization OK

<cause>

<cause> is included if it has a non-zero value.

0 – No specific cause (<cause> omitted)

1 – *PIN* required

2 – PIN2 required

3 – *PUK* required (PIN blocked)

4 – PUK2 required (PIN2 blocked)

5 – PUK blocked

6 – PUK2 blocked

7 – Device personalization blocked

8 – *IMEI* lock blocked

9 – *USIM* card failure

10 – USIM card changed

11 – USIM profile changed

The following command example subscribes UICC state notifications:

```
AT%XSIM=1
OK
```

The following notification example indicates that UICC is not initialized:

```
%XSIM: 0
```

The following notification example indicates that UICC initialization is completed:

```
%XSIM: 1
```

13.1.2 Read command

The read command reads the *UICC* state.

Response syntax:

```
%XSIM: <state>[,<cause>]
```

The read command parameter and its defined values are the following:

<state>

0 – UICC not initialized

1 – UICC initialization OK

<cause>

<cause> is included if it has a non-zero value.

0 – No specific cause (<cause> omitted)

1 – *PIN* required

2 – PIN2 required

3 – *PUK* required (PIN blocked)

4 – PUK2 required (PIN2 blocked)

5 – PUK blocked

6 – PUK2 blocked

7 – Device personalization blocked

8 – *IMEI* lock blocked

9 – *USIM* card failure

10 – USIM card changed

11 – USIM profile changed

12 – GNSS mode only (UICC not initialized)

The following command example reads the UICC state. The response indicates that UICC initialization is completed:

```
AT%XSIM?
%XSIM: 1
OK
```

The following command example reads the UICC state. The response indicates that a *PIN* code is required:

```
AT%XSIM?
%XSIM: 0,1
OK
```

13.1.3 Test command

The test command is not supported.

13.2 Request IMSI +CIMI

The **+CIMI** command reads the *IMSI* from the *USIM* card. v2.0.x

For reference, see *3GPP 27.007 Ch. 5.6*.

13.2.1 Set command

The set command reads the *IMSI* from the *SIM* card.

Syntax:

```
+CIMI
```

Response syntax:

```
<IMSI>
```

The response parameter and its defined value are the following:

<IMSI>

String without double quotes. IMSI.

> **Note:** ERROR is returned if IMSI is not available.

The following command example reads the IMSI string:

```
AT+CIMI
284011234567890
OK
```

13.2.2 Read command

The read command is not supported.

13.2.3 Test command

The test command is not supported.

13.3 Request ICCID %XICCID

The proprietary **%XICCID** command reads the *Integrated Circuit Card Identifier (ICCID)* from the *USIM* card. v2.0.x

13.3.1 Set command

The set command reads the *ICCID* from the *USIM* card.

Syntax:

```
%XICCID
```

Response syntax:

```
%XICCID: <ICCID>
```

The response parameter and its defined value are the following:

<ICCID>

String without double quotes. ICCID from the USIM card.

The following command example requests the ICCID of the USIM card:

```
AT%XICCID
%XICCID: 8901234567012345678F
OK
```

13.3.2 Read command

The read command is not supported.

13.3.3 Test command

The test command is not supported.

13.4 Subscriber number +CNUM

The **+CNUM** command returns the subscriber *Mobile Station International Subscriber Directory Number (MSISDN)*. v2.0.x

For reference, see *3GPP 27.007 Ch. 7.1*.

13.4.1 Set command

The set command returns the subscriber *MSISDN*.

Syntax:

```
+CNUM
```

Response syntax:

```
+CNUM: ,<number1>,<type1>
```

An `ERROR` response is returned if MSISDN is not available on *SIM* card or if SIM card is not initialized.

The set command parameters and their defined values are the following:

<numberx>

String. Phone number of format specified by <typex>.

<typex>

Integer. Type of address octet (see *3GPP TS 24.008, subclause 10.5.4.7*).

The following command example reads the subscriber number stored in the SIM:

```
AT+CNUM
+CNUM: ,"+1234567891234",145
OK
```

13.4.2 Read command

The read command is not supported.

13.4.3 Test command

The test command is not supported.

13.5 Restricted SIM access +CRSM

The **+CRSM** command transmits restricted commands to *SIM*. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.18*.

13.5.1 Set command

The set command transmits restricted commands to the *SIM*.

Syntax:

```
+CRSM=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>[,<pathid>]]]]
```

Response syntax:

```
+CRSM: <sw1>,<sw2>[,<response>]
```

+CME ERROR code

11 – SIM PIN required.

12 – SIM PUK required.

13 – SIM failure.

14 – SIM busy.

17 – SIM PIN2 required.

18 – SIM PUK2 required.

The set command parameters and their defined values are the following:

<command>

Integer

176 – READ BINARY

178 – READ RECORD

192 – GET RESPONSE

214 – UPDATE BINARY

220 – UPDATE RECORD

242 – STATUS

203 – RETRIEVE DATA

219 – SET DATA

<fileid>

Integer. Identifier of an elementary data file on SIM. Mandatory for every command except `STATUS`. The range of valid file identifiers depends on the actual SIM and is defined in *3GPP TS 51.011*. Optional files may not be present at all.

<P1>, <P2>, <P3>

Integer. Parameters passed on by the *MT* to the SIM. These parameters are mandatory for every command, except `GET RESPONSE` and `STATUS`. The values are described in *3GPP TS 51.011*.

<data>

String in hexadecimal format. Information that is written to the SIM.

<pathid>

String in hexadecimal format. Contains the path of an elementary file on the SIM/*UICC* (for example, "7F205F70" in SIM and UICC case). The <pathid> is used only in the mode "select by path from MF" as defined in *ETSI TS 102 221*.

<sw1>, <sw2>

Integer. Information from the SIM about command execution. These parameters are delivered to the *TE* in both cases, on successful or failed command execution.

<response>

String in hexadecimal format. Issued once a command is successfully completed. `STATUS` and `GET RESPONSE` return data which provides information about the current elementary data field. This information includes file type and size (see *3GPP TS 51.011*). After `READ BINARY`, `READ RECORD`, or `RETRIEVE DATA` command, the requested data is returned. <response> is not returned after a successful `UPDATE BINARY`, `UPDATE RECORD`, or `SET DATA` command.

The following command example reads the forbidden *PLMN* list:

```
AT+CRSM=176,28539,0,0,12
+CRSM: 144,0,"64F01064F040FFFFFFFFFFFF"
OK
```

13.5.2 Read command

The read command is not supported.

13.5.3 Test command

The test command is not supported.

13.6 Generic SIM access +CSIM

The **+CSIM** command transmits a command to the *SIM*. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.17* and *ETSI TS 102 221 Ch. 10 and 11*.

To avoid conflicts with modem firmware, **+CSIM** is limited so that only the following commands are allowed on a basic channel (channel 0 encoded in CLA):

- **STATUS**, with P1="No indication"
- **MANAGE CHANNEL**, open/close logical channels
- PIN-code-related commands (**VERIFY**, **UNBLOCK**, **ENABLE**, **DISABLE**, **CHANGE**)
- ENVELOPE, send application toolkit-specific information to *UICC*

To use other commands, use **MANAGE CHANNEL** to open a logical channel, encode the channel number in the CLA byte of the subsequent commands, and close the logical channel when SIM card access is finished.

13.6.1 Set command

The set command transmits a command to the *SIM*.

Syntax:

```
+CSIM=<length>,<command>
```

Response syntax:

```
+CSIM: <length>,<response>
```

+CME ERROR code

11 – SIM PIN required.

12 – SIM PUK required.

13 – SIM failure.

14 – SIM busy.

17 – SIM PIN2 required.

18 – SIM PUK2 required.

The set command parameters and their defined values are the following:

<length>

Integer. The number of hexadecimal characters.

<command>

The command passed to the SIM in hexadecimal format. Two characters per byte. Contains CLA, INS, P1, P2, and optionally Lc, Data, and Le bytes according to the command *Application Protocol Data Unit (APDU)* structure specification in *ETSI TS 102 221, Ch. 10.1*.

<response>

The response from the SIM in hexadecimal format. Two characters per byte. Contains optional data bytes and SW1, SW2 according to the response APDU structure specification in *ETSI TS 102 221, Ch. 10.2.*

The following command example performs a **MANAGE CHANNEL** command to open a logical channel. The SIM card returns channel number `01` and success status `9000`:

```
AT+CSIM=10,"0070000001"
+CSIM: 6,"019000"
OK
```

13.6.2 Read command

The read command is not supported.

13.6.3 Test command

The test command is not supported.

13.7 Open logical channel +CCHO

The **+CCHO** command opens a new logical channel in the currently selected *UICC*. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.45*.

> **Note:** The **+CCHO** command is for testing purposes only.

13.7.1 Set command

The set command causes the currently selected *UICC* to open a new logical channel. The UICC selects the application identified by the *Dedicated File (DF)* name that is received with this command and returns a session ID as a response.

Syntax:

```
+CCHO=<dfname>
```

Response syntax:

```
+CCHO: <sessionid>
```

+CME ERROR code

11 – SIM PIN required.

12 – SIM PUK required.

13 – SIM failure.

14 – SIM busy.

17 – SIM PIN2 required.

18 – SIM PUK2 required.

The set command parameters and their defined values are the following:

<dfname>

1–16 bytes. Application selected in the UICC referenced by a DF name.

<sessionid>

Integer.

Session ID used to target a specific application on a smart card, such as *USIM*, using the opened logical channel.

The following command example opens a logical channel and selects an *IP Multimedia Services Identity Module (ISIM)* application in the GSMA Generic eUICC Test Profile:

```
AT+CCHO="A0000000871004FF49FF0589"
+CCHO: 2
OK
```

13.7.2 Read command

The read command is not supported.

13.7.3 Test command

The test command is not supported.

13.8 Close logical channel +CCHC

The **+CCHC** command closes a previously opened logical channel in the selected *UICC*. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.46*.

> **Note:** The **+CCHC** command is for testing purposes only.

13.8.1 Set command

The set command closes a communication session with the active *UICC*. The UICC closes the logical channel when it receives the command.

Syntax:

```
+CCHC=<sessionid>
```

+CME ERROR code

11 – SIM PIN required.

12 – SIM PUK required.

13 – SIM failure.

14 – SIM busy.

17 – SIM PIN2 required.

18 – SIM PUK2 required.

The set command parameter and its defined values are the following:

<sessionid>

Integer.

Session ID used to target a specific application on a smart card, such as *USIM*, using the opened logical channel.

The following command example closes a previously opened logical channel:

```
AT+CCHC=2
OK
```

13.8.2 Read command

The read command is not supported.

13.8.3 Test command

The test command is not supported.

13.9 Generic UICC access through logical channel +CGLA

The **+CGLA** command transmits a command to the *UICC* through an opened logical channel. v2.0.x

For reference, see *3GPP 27.007 Ch. 8.43*.

> **Note:** The **+CGLA** command is for testing purposes only.

13.9.1 Set command

The set command transmits a command to a selected *UICC* using an opened logical channel. The UICC response is sent back to the *TA* as it is.

The command allows direct control of the currently selected UICC by a distant application on the *TE*. The TE processes the UICC information.

The command transmitted to the UICC is formatted as a command *APDU*. *ETSI TS 102 221, Ch. 10.1* defines four different cases of the command's APDU structure. The structure depends on whether the command sends data to UICC and whether the expected response has data bytes or only status bytes.

The cases are the following:

- Case 1: CLA INS P1 P2 – No data to send. No response data expected.
- Case 2: CLA INS P1 P2 Le – No data to send. Expecting Le bytes of response data.
- Case 3: CLA INS P1 P2 Lc Data – Sending Lc bytes of data. No response data expected. The data to send is entered into the command string after Lc byte.
- Case 4: CLA INS P1 P2 Lc Data Le – Sending Lc bytes of data. Expecting Le bytes of response data.

Le can be set to `00` if the exact amount of response data is not known. In this case, the maximum number of available data (up to 256 bytes) is returned in the <response> string.

If Le is omitted and the UICC has response data to send, or the amount of response data on the UICC is different from the Le, the UICC can indicate it by status byte values `61xx` or `6Cxx` where `xx` indicates the number of bytes available on the UICC. If the status byte value is `61xx`, the response data can be read by issuing the `GET RESPONSE` command with Le value `xx`. If the status byte value is `6Cxx`, the previous command must be repeated with the Le value `xx` to read the response data from the UICC.

This can result in failure due to a crossover if the modem firmware has sent another APDU between the original command and `GET RESPONSE` which clears the previous response data. The risk of crossover

can be reduced by activating the modem in UICC only mode with the `+CFUN=41` command. Depending on the UICC, there can still be some interfering *Universal Subscriber Identity Module Application Toolkit (USAT)* activity. If the `GET RESPONSE` command fails, the command must be issued again.

> **Note:** Executing a Run GSM Algorithm command or an Authenticate command from the TE using the **+CGLA** command is not supported.

Syntax:

```
+CGLA=<sessionid>,<length>,<command>
```

Response syntax:

```
+CGLA: <length>,<response>
```

+CME ERROR code

11 – SIM PIN required.

12 – SIM PUK required.

13 – SIM failure.

14 – SIM busy.

17 – SIM PIN2 required.

18 – SIM PUK2 required.

The set command parameters and their defined values are the following:

<sessionid>

Integer.

Session ID used to target a specific application on a smart card, such as *USIM*, using the opened logical channel.

The <sessionid> must be acquired with the **+CCHO** command before entering **+CGLA** commands. It must be released with the **+CCHC** command after the last **+CGLA** command.

<length>

Integer. Number of hexadecimal characters in the <command> or <response> string.

<command>

The command passed to the SIM in hexadecimal format. Two characters per byte. Contains CLA, INS, P1, P2, and optionally Lc, Data, and Le bytes according to the command *APDU* structure specification in *ETSI TS 102 221, Ch. 10.1*.

<response>

The response from the SIM in hexadecimal format. Two characters per byte. Contains optional data bytes and SW1, SW2 according to the response APDU structure specification in *ETSI TS 102 221, Ch. 10.2.*

The following command example shows the `STATUS` command returning the currently selected AID on the channel when CLA is `80`, INS is `F2`, P1 is `00`, P2 is `01`, and Le is `00`. The response contains the DF name TLV object (tag, length, and AID) and status bytes SW1 and SW2:

```
AT+CGLA=2,10,"80F2000100"
+CGLA: 32,"840CA0000000871004FF49FF05899000"
OK
```

The following command example shows the `SELECT` command sending four bytes of data to UICC. The Le byte is omitted, so the UICC indicates only the available amount of response data by status bytes `611E` with 30 bytes available. The CLA is `00`, INS is `A4`, P1 is `08`, P2 is `04`, Lc is `04`, and the sent data is `7FFF6F7B`.

The `GET RESPONSE` command reads the response data of the previously issued `SELECT` command. The CLA is `00`, INS is `C0`, P1 is `00`, P2 is `00`, and Le is `1E`:

```
AT+CGLA=1,18,"00A40804047FFF6F7B"
+CGLA: 4,"611E"
OK

AT+CGLA=1,10,"00C000001E"
+CGLA: 64,"621C8202412183026F7BA5038001618A01058B036F06028002001288016890 00"
OK
```

The following command example shows the `SELECT` command sending four bytes of data to UICC. Le byte is set to `00`. The CLA is `00`, INS is `A4`, P1 is `08`, P2 is `04`, Lc is `04`, the sent data is `7FFF6F7B`, and Le is `00`. When the Le byte is `00`, the modem automatically issues the `GET RESPONSE` command and returns full response data:

```
AT+CGLA=1,20,"00A40804047FFF6F7B00"
+CGLA: 64,"621C8202412183026F7BA5038001618A01058B036F06028002001288016890 00"
OK
```

13.9.2 Read command

The read command is not supported.

13.9.3 Test command

The test command is not supported.

13.10 USAT event notification %USATEV

The proprietary **`%USATEV`** command subscribes *USAT* event notifications. v2.0.x

For more information on USAT, see *3GPP TS 31.111* and *ETSI TS 102 223*.

13.10.1 Set command

The set command subscribes *USAT* event notifications. The unsolicited result code is `%USATEV: XXX`.

The application can use the notifications to get information on USAT events. The application can ensure *UICC* data transmission by not deactivating the modem when the *Bearer Independent Protocol (BIP)* channel is open.

Syntax:

```
%USATEV=<subscribe>[,<event_mask>]
```

The set command parameters and their defined values are the following:

<subscribe>

0 – Unsubscribe and disable USAT event notifications.

1 – Subscribe and enable USAT event notifications.

<event_mask>

String. Bit string where LSB is BIP events. Leading zeros can be omitted. Maximum length 3 characters. If the parameter is omitted, notifications of all events are subscribed.

Allows to select a subset of USAT events to be reported. Can be used only in a subscribe command (`<subscribe>=1`). One or more event categories can be selected by setting bits to `1`. If the <event_mask> parameter is omitted, all events are enabled (same as entering <event_mask> "111").

Bit 1 – BIP events.

Bit 2 – REFRESH events.

Bit 3 – SMS delivery to or from UICC.

<refresh_type>

REFRESH mode received from UICC.

Valid range 0–10.

Values as specified in *ETSI TS 102 223 Ch. 8.6*.

<refresh_result>

Result of REFRESH command.

0 – REFRESH performed successfully.

1 – REFRESH rejected, for example, device is busy on data transfer.

2 – REFRESH failed.

The notification syntax descriptions are as follows:

SMS has been received and delivered to UICC by using envelope command SMS-PP data download:

```
%USATEV: SMS Receive
```

Sending SMS requested by UICC by using proactive command SEND SHORT MESSAGE:

```
%USATEV: SMS Send
```

BIP channel connect process has been started as requested by UICC by using proactive command OPEN CHANNEL:

```
%USATEV: BIP Connecting
```

BIP channel connect process has been finished successfully, providing the UICC access to, for example, SIM *Over-the-Air (OTA)* server:

```
%USATEV: BIP Connected
```

BIP channel connection attempt failed or the channel has been closed by UICC by using proactive command CLOSE CHANNEL:

```
%USATEV: BIP Closed
```

Proactive command REFRESH has been received from UICC indicating that UICC content has been changed:

```
%USATEV: REFRESH Requested <refresh_type>
```

REFRESH command is completed with result indicated in <refresh_result>:

```
%USATEV: REFRESH End <refresh_result>
```

The following command example subscribes notifications of all available USAT events:

```
AT%USATEV=1
OK
```

The following command example subscribes notifications of BIP channel related USAT events only:

```
AT%USATEV=1,"001"
OK
```

The following command example subscribes notifications of BIP channel and SMS related USAT events only:

```
AT%USATEV=1,"101"
OK
```

The following command example unsubscribes all USAT event notifications:

```
AT%USATEV=0
OK
```

13.10.2 Read command

The read command is not supported.

13.10.3 Test command

The test command is not supported.

13.11 USAT REFRESH and BIP state information %USATINFO

The proprietary **%USATINFO** command requests the statuses of the REFRESH command and *BIP* channel.
v2.0.x

13.11.1 Set command

The set command provides information about the current state of the REFRESH command and *BIP* channel.

Syntax:

```
%USATINFO
```

Response syntax:

```
%USATINFO: <refresh_state>,<bip_state>
```

The set command parameters and their defined values are the following:

<refresh_state>

Integer.

0 – REFRESH command handling is not ongoing.

1 – REFRESH command handling is ongoing.

<bip_state>

Integer.

0 – BIP channel is closed. No ongoing activity.

1 – BIP channel is connecting to server.

2 – BIP channel is connected. *UICC* can perform data transfer.

The following command example gets USAT information when BIP or REFRESH activity is not ongoing:

```
AT%USATINFO
%USATINFO: 0,0
OK
```

The following command example gets USAT information when REFRESH is ongoing:

```
AT%USATINFO
%USATINFO: 1,0
OK
```

The following command example receives USAT information when BIP channel is connected to, for example, a SIM OTA server:

```
AT%USATINFO
%USATINFO: 0,2
OK
```

13.11.2 Read command

The read command is not supported.

13.11.3 Test command

The test command is not supported.

13.12 UICC suspend and deactivate control +SSRDA

The **+SSRDA** command controls the suspend and resume UICC and deactivate and activate UICC capability in the device. v2.0.x

13.12.1 Set command

The set command sets the parameters to enable or disable the suspend and resume *UICC* and deactivate and activate UICC capability in the device. When either or both features are enabled, the command allows for a delay in executing a suspend and deactivate of the UICC when the conditions to allow start of suspend and deactivate are met.

UICC suspension and deactivation are enabled with no delay by default. UICC can be suspended or deactivated when allowed by conditions specified in *ETSI TS 102 221* and *3GPP TS 31.102, 5.1.10 (PSM)* or *3GPP TS 31.102, 5.1.11* (*eDRX*).

Note: In an AT&T subscription, the minimum sleep time for suspend and deactivate is two minutes.

Syntax:

```
+SSRDA=<sus>,<deac>,<del>
```

The set command parameters and their defined values are the following:

<sus>

Controls suspend and resume UICC capability.

0 – Disable suspend and resume UICC.

1 – Enable suspend and resume UICC.

<deac>

Controls deactivate and activate UICC capability.

0 – Disable deactivate and activate UICC.

1 – Enable deactivate and activate UICC.

<del>

Delay in executing a suspend or deactivate of the UICC upon meeting the conditions to allow start of suspend or deactivate in decimal format.

0–10 s.

The following command example enables UICC suspension and deactivation functionality with no delay after meeting conditions to start using the default parameters:

```
AT+SSRDA=1,1,0
OK
```

The following command example enables UICC suspension with five seconds delay after meeting conditions to start suspension and disables UICC deactivation:

```
AT+SSRDA=1,0,5
OK
```

The following command example disables UICC suspension and enables UICC deactivation with ten seconds delay:

```
AT+SSRDA=0,1,10
OK
```

13.12.2 Read command

The read command returns the current parameters for the *UICC* suspend and deactivate capability. If the set command has not been executed, the default parameters are returned with both features enabled and no delay.

Response syntax:

```
+SSRDA: <sus>,<deac>,<del>
```

The read command parameter and its defined values are the following:

<sus>

0 – Suspend and resume UICC disabled.

1 – Suspend and resume UICC enabled.

<deac>

0 – Deactivate and activate UICC disabled.

1 – Deactivate and activate UICC enabled.

<del>

Delay in executing a suspend or deactivate of the UICC upon meeting the conditions to allow start of suspend or deactivate in decimal format.

0–10 s.

The following command example reads the parameters when UICC suspension and deactivation are enabled and there is no delay, which are the default parameters:

```
AT+SSRDA?
+SSRDA: 1,1,0
OK
```

The following command example reads the configuration when UICC suspension is disabled and deactivation is enabled with ten seconds delay:

```
AT+SSRDA?
+SSRDA: 0,1,10
OK
```

13.12.3 Test command

The test command is not supported.

13.13 Selecting UICC slot %CSUS

The proprietary **%CSUS** command is used to select the *UICC* slot to be used by the modem. v2.0.x

All *USIM* and UICC activity is directed to the selected slot the next time the modem or UICC is activated. The slot covers also Soft SIM solutions, such as virtual iUICC, in addition to physical UICC slot.

The command configuration is stored to *NVM* approximately every 48 hours and when the modem is set to minimum functionality mode with the `+CFUN=0` command.

Active **`%XPOFWARN`** warning blocks the storing to NVM.

The stored configuration is reset to 0 with the **`%XFACTORYRESET`** command's <reset_type>s `0` and `1`.

If the UICC slot has been set to `2` and modem firmware is updated to a version that does not support slot 2, the modem uses the default slot 0.

> **Note:** The command is allowed only when the modem is set to minimum functionality mode with the `+CFUN=0` command or deactivated with the `+CFUN=4` command.

13.13.1 Set command

The set command is used to select the *UICC* slot to be used by the modem.

Syntax:

```
%CSUS=<slot>
```

+CME ERROR code

518 – Not allowed in active state.

The set command parameter and its defined values are the following:

<slot>

0 – Physical UICC slot 1 (default)

1 – Reserved

2 – Virtual iUICC

3 – Reserved

The following command example selects virtual iUICC slot:

```
AT%CSUS=2
OK
```

The following command example selects physical UICC slot 1:

```
AT%CSUS=0
OK
```

13.13.2 Read command

The read command returns the currently selected *UICC* slot.

Response syntax:

```
%CSUS: <slot>
```

The read command parameter and its defined values are the following:

<slot>

0 – Physical UICC slot 1 (default)

1 – Reserved

2 – Virtual iUICC

3 – Reserved

The following command example reads the slot when virtual iUICC is selected:

```
AT%CSUS?
%CSUS: 2
OK
```

The following command example reads the slot when physical UICC slot 1 is selected:

```
AT%CSUS?
%CSUS: 0
OK
```

13.13.3 Test command

The test command returns a list of supported *UICC* slot numbers. The supported UICC slots depend on the available Soft SIM solutions in the modem firmware.

Response syntax:

```
%CSUS: <slot1>[,<slot2>...]
```

The test command parameter and its defined values are the following:

<slot>

0 – Physical UICC slot 1 (default)

1 – Reserved

2 – Virtual iUICC

3 – Reserved

The following command example returns a list of supported slot values when slots 0 and 2 are supported:

```
AT%CSUS=?
%CSUS: 0,2
OK
```

14 Authenticating AT command usage

The **%XSUDO** command is used to authenticate AT commands.

Before you start the authentication, perform the following two steps (only once):

1. Generate private and public keys with OpenSSL:

```
openssl ecparam -name prime256v1 -genkey -noout -out [private key PEM file]
openssl ec -in [private key PEM file] -out [public key PEM file] -pubout
```

2. Write the public key with the AT command:

```
AT%XPMNG=0,"<public key>"
OK
```

To authenticate an AT command, perform the following steps:

1. Calculate an AT command signature.
   a) Create an AT command text file for an authenticated AT command:
      Example:

```
%CMNG=0,1,0,"TEST ROOT CERTIFICATE"
```

   b) Create a digest file with OpenSSL from AT command that needs authentication:

```
openssl sha256 -binary [AT command text file] > [digest file]
```

   c) Create a signature file with OpenSSL from the digest file:

```
openssl pkeyutl -sign -in [digest file] -out [signature file] -inkey [private key PEM
 file]
```

   d) Convert the signature to Base64 format:

```
base64 < [signature file] > [signature base64 file]
```

2. Write the authenticated AT command.
   The **%XSUDO** command is used to authenticate the **%CMNG** command:

   Example:

```
AT%XSUDO=35,"<signature base64>";%CMNG=0,1,0,"TEST ROOT CERTIFICATE"
OK
```

   For more information on the command, see Authenticated access %XSUDO on page 201.

Glossary

16-state Quadrature Amplitude Modulation (16-QAM)

A digital modulation technique used for signals in which four bits are modulated at once by selecting one of 16 possible combinations of carrier phase shift and amplitude.

Access Point Name (APN)

The name of a gateway between a mobile network and another computer network, usually the Internet.

Application Protocol Data Unit (APDU)

The communication unit between a terminal and smart card (*UICC*).

AT command

A command used to control the modem.

Bearer Independent Protocol (BIP)

A set of commands and events that allows a smart card to establish a communication channel with a terminal and through the terminal with a remote server or a remote device.

Binary Phase-Shift Keying (BPSK)

A digital modulation technique used for signals in which one bit is modulated by selecting one of two possible carrier phase shifts with a 180-phase difference.

Carrier Wave (CW)

A single-frequency electromagnetic wave that can be modulated in amplitude, frequency, or phase to convey information.

Cat-M1

LTE-M UE category with a single RX antenna, specified in 3GPP Release 13.

Cat-NB1

NB-IoT *UE* category with 200 kHz UE bandwidth and a single RX antenna, specified in 3GPP Release 13.

Cat-NB2

An upgraded version of *Cat-NB1*, specified in 3GPP Release 14.

Check Digit (CD)

The last one-digit number of the *IMEI* code used for error detection.

Classless Inter-domain Routing (CIDR)

A method for allocating IP (Internet Protocol) addresses.

CS/PS Mode of Operation

A *UE* mode of operation. The UE may either register to packet-switched services, circuit-switched services, or both based on the mode of operation. If both are registered, the mode of operation also contains a preference for either of them.

DECT NR+

A non-cellular radio standard included as part of the 5G standards by the ITU.

Dedicated File (DF)

A functional grouping of files consisting of a dedicated file and all files which contain the dedicated file in their parental hierarchy.

Discontinuous Reception (DRX)

A method in mobile communication to conserve the battery of a mobile device by turning the RF modem in a sleep state.

DER

Distinguished Encoding Rules

Dynamic Host Configuration Protocol (DHCP)

A network management protocol used for automatic and centralized management of IP addresses within a network.

Electronic Serial Number (ESN)

A unique number embedded on a microchip for identifying mobile devices.

Enhanced Machine Type Communication (eMTC)

A low-power wide area network radio technology standard developed by 3GPP to enable a wide range of cellular devices and services.

EPS Mobility Management (EMM)

The *EPS* Mobility Management (EMM) sublayer in the *NAS* protocol provides mobility service to the *UE*.

E-UTRA Absolute Radio Frequency Channel Number (EARFCN)

LTE carrier channel number for unique identification of LTE band and carrier frequency.

Evolved Packet System (EPS)

A connection-oriented transmission network in LTE (Long-term Evolution) consisting of an EPC (Evolved Packet Core) and an E-UTRAN (Evolved Terrestrial Radio Access Network).

Evolved Terrestrial Radio Access Network (E-UTRAN)

The network architecture defined for the E-UTRA radio interface consisting of evolved Node Bs (eNB) and providing user plane and control plane protocol terminations towards the User Equipment (UE).

Evolved Universal Terrestrial Radio Access (E-UTRA)

The air interface of LTE.

Extended Discontinuous Reception (eDRX)

A method to conserve the battery of an IoT (Internet of Things) device by allowing it to remain inactive for extended periods.

Firmware-Over-The-Air (FOTA)

A firmware update performed remotely over the air (OTA).

General Packet Radio Services (GPRS)

A packet-based mobile data service for 2G and 3G mobile networks with data rates of 56-114 kbps/ second and continuous connection to the Internet.

Global Navigation Satellite System (GNSS)

A satellite navigation system with global coverage. The system provides signals from space transmitting positioning and timing data to GNSS receivers, which use this data to determine location.

Global Positioning System (GPS)

A satellite-based radio navigation system that provides its users with accurate location and time information over the globe.

General-Purpose Input/Output (GPIO)

A digital signal pin that can be used as input, output, or both. It is uncommitted and can be controlled by the user at runtime.

Hybrid Automatic Repeat Request (HARQ)

Any combined Automatic Repeat Request (ARQ) and Forward Error Coding (FEC) method that saves failed decoding attempts for future joint decoding.

Integrated Circuit Card Identifier (ICCID)

A unique serial number of a SIM card.

IP Multimedia Services Identity Module (ISIM)

An application which resides on a mobile device's *SIM* card (*UICC*).

International Mobile (Station) Equipment Identity (IMEI)

A unique code consisting of 14 digits and a check digit for identifying 3GPP-based mobile devices.

International Mobile (Station) Equipment Identity, Software Version (IMEISV)

A unique code consisting of 16 decimal digits and two software version digits for identifying 3GPP-based mobile devices.

International Mobile Subscriber Identity (IMSI)

A unique code, usually 15 digits, used for the identification of a mobile subscriber and consisting of an *MCC*, *MNC*, and MSIN (Mobile Subscription Identification Number).

Internet of Things (IoT)

Physical objects that are embedded with sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems of the Internet or other communications networks.

Internet Protocol (IP)

The network layer communications protocol in the Internet protocol suite for relaying datagrams across network boundaries. Its routing function enables internetworking, and essentially establishes the Internet.

International Reference Alphabet (IRA)

A seven-bit coded character set for information exchange.

JSON Web Token (JWT)

An Internet-proposed standard for creating data with optional signature and optional encryption whose payload holds JSON that asserts some number of claims.

Low-Noise Amplifier (LNA)

In a radio receiving system, an electronic amplifier that amplifies a very low-power signal without significantly degrading its signal-to-noise ratio.

Long-Term Evolution (LTE)

A wireless broadband communication standard for mobile devices and data terminals, based on the GSM/EDGE and UMTS/HSPA technologies.

LTE-M

An open standard that is most suitable for medium throughput applications requiring low power, low latency, and/or mobility, like asset tracking, wearables, medical, POS, and home security applications. Also known as Cat-M1.

Maximum Transmission Unit (MTU)

The largest packet or frame that can be sent in a single network layer transaction.

MIPI RF Front-End Control Interface (RFFE)

A dedicated control interface for the RF front-end subsystem. *MIPI Alliance*

Mobile Country Code (MCC)

A unique three-digit part of an IMSI code identifying the country of domicile of the mobile subscriber. MCC is used together with the Mobile Network Code (MNC).

Mobile Equipment (ME)

The physical *UE* consisting of one of more *MT* and one or more *TE*.

Mobile Network Code (MNC)

A code identifying the telecommunications network. The code is defined by ITU-T Recommendation E.212, consists of two or three decimal digits, and is used together with the Mobile Country Code (MCC).

Mobile Station International Subscriber Directory Number (MSISDN)

A number consisting of a maximum of 15 digits identifying a mobile subscriber by mapping the telephone number to the *SIM* card in a phone.

Mobile Termination (MT)

A component of the Mobile Equipment (ME) performing functions specific to management of the radio interface. The R interface between *TE* and MT uses the AT command set. The *IMEI* code is attached to the MT.

Narrowband Internet of Things (NB-IoT)

A narrowband technology standard with longer range, lower throughput, and better penetration in, for example, cellars and parking garages compared to LTE-M. NB-IoT is most suitable for static, low throughput applications like smart metering, smart agriculture, and smart city applications. Also known as *Cat-NB1* or *Cat-NB2*.

Non-access Stratum (NAS)

In telecom protocol stacks, the highest stratum of the control plane between the core network and User Equipment (UE). The layer is used to manage the establishment of communication sessions and for maintaining communications with the UE as it moves.

Non-access Stratum (NAS) Signalling Low Priority Indication (NSLPI)

Used by the network for NAS-level mobility management congestion control.

Non-volatile Memory (NVM)

Memory that can retrieve stored information even after having been power-cycled.

Over-the-Air (OTA)

Refers to any type of wireless transmission. For Matter, OTA is synonymous with the Over-the-Air software update mechanism.

Packet Data Network (PDN)

A network that provides data services.

Packet Data Protocol (PDP)

A packet transfer protocol in wireless GPRS (General Packet Radio Services) and HSDPA (High-speed Downlink Packet Access) networks.

Packet Data Protocol (PDP) Context

In UMTS (Universal Mobile Telecommunications System) and GPRS (General Packet Radio Service), the record that specifies *UE* access to an external packet-switched network.

Paging Time Window (PTW)

The period of time during which the User Equipment (UE) attempts to receive a paging message.

Personal Identification Number (PIN)

An optional security feature in mobile devices used for identifying a user. PIN is a numeric code which must be entered each time a mobile device is started.

Personal Unblocking Key (PUK)

A digit sequence required in 3GPP mobile phones to unlock a *SIM* that has disabled itself after an in correct personal identification number has been entered multiple times.

Power Amplifier (PA)

A device used to increase the transmit power level of a radio signal.

Power Saving Mode (PSM)

A feature introduced in 3GPP Release 12 to improve battery life of IoT (Internet of Things) devices by minimizing energy consumption. The device stays dormant during the PSM window.

Pre-shared Key (PSK)

A password authentication method, a string of text, expected before a username and password to establish a secured connection. Also known as a shared secret.

Printed Circuit Board (PCB)

A board that connects electronic components.

Privacy Enhanced Mail (PEM)

A public key certificate defined in the X.509 cryptography standard and used to privately transmit email.

Production Test Image (PTI)

A modem firmware version used in the device manufacturing phase.

Protocol Configuration Options (PCO)

An element of *NAS* message used for transferring parameters between the *UE* and the P-GW (Packet Data Network Gateway).

Protocol Data Unit (PDU)

Information transferred as a single unit between peer entities of a computer network and containing control and address information or data. PDU mode is one of the two ways of sending and receiving SMS messages.

PS Mode of Operation

A *UE* mode of operation. The UE registers only to *EPS* services.

Public Land Mobile Network (PLMN)

A network that provides land mobile telecommunications services to the public. A PLMN is identified by the *MCC* and *MNC*.

Quadrature Phase-Shift Keying (QPSK)

A digital modulation technique used for signals in which two bits are modulated at once, selecting one of four possible carrier phase shifts.

Quality of Service (QoS)

The measured overall performance of a service, such as a telephony or computer network, a connection, or a cloud computing service.

Radio Policy Manager (RPM)

A radio baseband chipset feature that protects the mobile network from signaling overload.

Release Assistance Indication (RAI)

A 3GPP feature that allows an *LTE-M* or *NB-IoT* device to indicate to the eNB that it has no more UL data and that the device does not anticipate receiving further DL data.

Reference Signal Received Power (RSRP)

The average power level received from a single reference signal in an LTE (Long-Term Evolution) network.

Reference Signal Received Quality (RSRQ)

The quality of a single reference signal received in an LTE (Long-Term Evolution) network and calculated from *RSRP*.

Resource Block (RB)

The smallest unit of resources that can be allocated to a user.

RP-SMMA

A message sent by the User Equipment to relay a notification to the network that the mobile has memory available to receive one or more short messages.

Serial Number (SNR)

A unique six-digit number part of the *IMEI* code identifying each equipment within each *TAC*.

Signal-to-Noise Ratio (SNR)

The level of signal power compared to the level of noise power, often expressed in decibels (dB).

Software Version Number (SVN)

Part of the *IMEI* code identifying the revision of the software installed on a mobile device.

Subscriber Identity Module (SIM)

A card used in *UE* containing data for subscriber identification.

System in Package (SiP)

Several integrated circuits, often from different technologies, enclosed in a single module that performs as a system or subsystem.

System on Chip (SoC)

A microchip that integrates all the necessary electronic circuits and components of a computer or other electronic systems on a single integrated circuit.

Terminal Adapter (TA)

A device that connects a *UE* to a communications network. In mobile networks, the terminal adapter is used by the terminal equipment to access the mobile termination using AT commands.

Terminal Equipment (TE)

Communications equipment at either end of a communications link, used to permit the stations involved to accomplish the mission for which the link was established.

Tracking Area Code (TAC)

A unique code used to identify a tracking area within a particular network.

Tracking Area Update (TAU)

A procedure initiated by the *UE* when moving to a new tracking area in the LTE (Long-term Evolution) system.

Transport Layer Security (TLS)

A cryptographic protocol that provides end-to-end security of data sent over a computer network.

Type Allocation Code (TAC)

The initial eight-digit part of an *IMEI* code used for identifying the model of a mobile phone.

Universal Asynchronous Receiver/Transmitter (UART)

A hardware device for asynchronous serial communication between devices.

User Equipment (UE)

Any device used by an end-user to communicate. The UE consists of the Mobile Equipment (ME) and the Universal Integrated Circuit Card (UICC).

Universal Integrated Circuit Card (UICC)

A new generation *SIM* used in *UE* for ensuring the integrity and security of personal data.

Unique Slave Identifier (USID)

A unique address for identifying each slave device in an RFFE (RF Front-End) system.

Universal Subscriber Identity Module (USIM)

A card used in *UE* containing data for subscriber identification.

Universal Subscriber Identity Module Application Toolkit (USAT)

A standard that provides mechanisms which allow applications in the *USIM* to interact and operate with *ME*.

UUID

Universally Unique Identifier

Legal notices

By using this documentation you agree to our terms and conditions of use. Nordic Semiconductor may change these terms and conditions at any time without notice.

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function, or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Nordic Semiconductor ASA does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. If there are any discrepancies, ambiguities or conflicts in Nordic Semiconductor's documentation, the Product Specification prevails.

Nordic Semiconductor ASA reserves the right to make corrections, enhancements, and other changes to this document without notice.

Life support applications

Nordic Semiconductor products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury.

Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

RoHS and REACH statement

Complete hazardous substance reports, material composition reports and latest version of Nordic's REACH statement can be found on our website www.nordicsemi.com.

Trademarks

All trademarks, service marks, trade names, product names, and logos appearing in this documentation are the property of their respective owners.

Copyright notice

© 2023 Nordic Semiconductor ASA. All rights are reserved. Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.