NL-SWN-LTE-NRF9160
AT Command Manual

Version 2
2020-08-04
NimbeLink Corp

Revision History

Revision	Date	Author(s)	Description
1	2020-06-30	KR	Initial release
2	2020-08-04	KR	Removed MIPI commands. Updated title page picture.

Contents

1 Introduction

1.1 Scope

This document serves as the AT command manual for the NL-SWN-LTE-NRF9160 Skywire Nano modem.

1.2 Contact Information

NimbeLink's goal is to make integrating Skywire modems into end-user applications as easy as possible. Please send any feedback, documentation requests, or technical support questions to NimbeLink's product support team at:

product.support@nimbelink.com

For purchasing information, please visit the "Part Ordering Information" section on the modem's product page:

https://nimbelink.com/products/4g-lte-m-global-nano/

Any additional sales questions or requests for quotation can be directed to NimbeLink's sales team at:

sales@nimbelink.com

1.3 Orderable Part Numbers

Orderable Devices	Description
NL-SWNDK	NimbeLink Skywire Nano Development Kit
NL-SWN-LTE-NRF9160	NimbeLink Skywire Nano

1.4 Additional Resources

- Skywire Nano NL-SWN-LTE-NRF9160 Embedded Modem Product Page
- Skywire Nano NL-SWN-LTE-NRF9160 Datasheet
- Skywire Nano NL-SWN-LTE-NRF9160 Mechanical Model (STEP)
- Skywire Nano Development Kit User Manual

2 AT Command Overview

Words enclosed in angled brackets (<>) are arguments for a given command. Words in square brackets ([]) are optional items and are not required for the command.

AT commands are issued by sending the command to the modem followed by a carriage return (hex value 0x0D).

Standard AT commands begin with a plus (+) sign. Commands written by Nordic Semiconductor begin with a percent (%) sign. Commands written by NimbeLink begin with a number (#) sign.

If a carriage return is printed by the modem, it is designated with the symbol <CR>. If a line feed character is printed, it is designated with <LF>.

2.1 Set Command

Set commands set parameters and perform actions.

Examples:

```
AT+CGMR
AT+CMGF=0
```

2.2 Read Command

Read commands check the parameters and the status of a command. The command always ends with a question mark (?).

Examples:

```
AT+CGDCONT?
AT+CGACT?
```

2.3 Test Command

Test commands display basic information about the parameters for a command. The command always ends with an equals sign, followed by a question mark (=?).

Examples:

```
AT+CEREG=?
AT+CFUN=?
```

2.4 Notes

If one or more of the Set, Read, or Test commands are not listed under an AT command, that function is not implemented.

3 General Commands

3.1 E - Command echo

The E command turns AT command echo on and off. When echo is enabled, commands that are sent to the modem will be echoed back to the user. When echo is turned off, the user will only see responses from the modem.

3.1.1 Set command

The set command turns echo on and off.

Command syntax:

```
ATE<n>
```

The set parameter and their defined values are the following:

<n>

0 - Echo off

1 - Echo on

Example - Turn echo off:

```
ATE0
OK
```

3.1.2 Read command

The read command reads the current echo state.

Response syntax:

```
<n>
```

Example:

```
ATE?
1
OK
```

3.2 +IPR - Set baud rate

The +IPR command sets the baud rate of the UART interface.

3.2.1 Set command

The set command changes the baud rate to the specified rate.

Command syntax:

```
AT+IPR=<baud>
```

The set parameter and their defined values are the following:

<baud>

300	600	1200	2400	4800
9600	14400	19200	28800	31250
38400	56000	57600	76800	115200
230400	250000	460800	921600	1000000

Note: The default baud rate is 115200.

Note: the response of 0K will appear at the new baud rate.

Example:

```
AT+IPR=921600
0K
```

3.2.2 Read command

The read command will display the current baudrate.

Response syntax:

```
+IPR: <baud>
```

Example:

```
AT+IPR?
+IPR: 115200
0K
```

3.2.3 Test command

The test command displays all baud rates supported by the modem.

Response syntax:

```
+IPR=(300,600,...)
```

Example:

```
AT+IPR=?
+IPR=(300,600,1200,2400,4800,9600,14400,19200,28800,
    31250,38400,56000,57600,76800,115200,230400,
    250000,460800,921600,1000000)
OK
```

3.3 +IFC - Toggle hardware flow control

The +IFC command enables or disables hardware flow control on the modem's UART port.

3.3.1 Set command

Command syntax:

```
AT+IFC=<rts>,<cts>
```

Note: the values provided for <rts> and <cts> must be the same.

The set parameters and their defined values are the following:

<rts>

0 - flow control disable

2 - C105 (RTS) enable

<cts>

0 - flow control disable

2 - C106 (CTS) enable

Example:

```
AT+IFC=2,2
OK
```

3.3.2 Read command

The read command will display the current flow control settings.

Response syntax:

```
+IFC: <rts>,<cts>
```

Example:

```
AT+IFC?
+IFC: 2,2
OK
```

3.3.3 Test command

The test command displays parameters for setting the baud rate.

Response syntax:

```
+IFC: (0,2),(0,2)
```

Example:

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

3.4 #URC - Enable or disable unsolicited response code messages

The #URC command enables or disables unsolocited response code (URC) messages.

3.4.1 Set command

The set command enables or disables URC messages.

Command syntax:

```
AT#URC="<urc>",<setting>
```

The set parameter and their defined values are the following:

<urc>

LWM2M - enable or disable LWM2M-related commands

GPS - enable or disable GPS-related commands

SOCK - enable or disable SOCK-related commands

Note: the command itself can be written, and the character between "AT" and the command is not required. See the example for more information.

<setting>

0 - disable URC messages

1 - enable URC messages

Example - Enable URC messages for GPS commands:

```
AT#URC="GPS",1
OK
```

Example - Disable URC messages for LWM2M commands:

```
AT#URC="LWM2M",0
OK
```

3.4.2 Read command

The read command reads current URC message settings.

Response syntax:

```
#URC: #LWM2M,<setting>
#URC: #GPS,<setting>
#URC: SOCK,<setting>
```

Example:

```
AT#URC?
#URC: #LWM2M,0
#URC: #GPS,0
#URC: SOCK,1
OK
```

3.4.3 Test command

The test command displays argument information.

Response syntax:

```
#URC="<urc>",[0,1]
```

3.5 +CGMI - Request manufacturer identification

The +CGMI command requests manufacturer identification.

3.5.1 Set command

Command syntax:

```
AT+CGMI
```

Response Syntax:

```
<manufacturer>
```

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

Example:

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

3.6 +CGMM - Request model identification

3.6.1 Set command

The set command requests model identification.

Command syntax:

```
AT+CGMM
```

Response syntax:

```
<model>
```

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

Example:

```
AT+CGMM
nRF9160-SICA
OK
```

3.7 +CGMR - Request cellular module revision identification

The +CGMR command request revision identification of the cellular module.

3.7.1 Set command

The set command requests revision information of the cellular module.

Command syntax:

```
AT+CGMR
```

Response syntax:

```
<revision>
```

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

Example:

```
AT+CGMR
mfw_nrf9160_1.1.0
OK
```

3.8 #APPVER - Request processor revision idenfication

The #APPVER command request revision identification of the on-board processor.

3.8.1 Set command

The set command requests revision information of the on-board processor.

Command syntax:

```
AT#APPVER
```

Response syntax:

```
#APPVER: <revision>
```

The <revision> parameter returns a string of up to 32 characters followed by <CR><LF>OK.

Example:

```
AT#APPVER
#APPVER: v0.1.2-3-abcdef45
OK
```

3.8.2 Read command

The read command functions the same as the set command.

3.9 +CGSN - Request product serial number identification

The +CGSN command requests product serial number identification.

3.9.1 Set command

The set command requests product serial number identification.

Command syntax:

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

The set parameter 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>

A 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>

A 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>

A 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>.

Example 1 - read the serial number:

```
AT+CGSN
490154203237518
OK
```

Example 2 - read the IMEI:

```
AT+CGSN=1
+CGSN: "490154203237518"
OK
```

3.9.2 Test command

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

Response syntax:

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

Example:

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

3.10 #SERNUM - Request serial number

The #SERNUM command returns the serial number of the modem.

3.10.1 Set command

The set command returns the serial number.

Command syntax:

```
AT#SERNUM
```

Response syntax:

```
#SERNUM: <serial_number>
```

<serial_number> is a string that is fifteen characters (15) long comprised of lowercase alphanumeric characters.

Example:

```
AT#SERNUM
#SERNUM: a01234abcdef567
OK
```

3.10.2 Read command

The read command functions the same as the set command.

3.10.3 Test command

The test command functions the same as the set command.

3.11 #ICCID - Request ICCID

The #ICCID command returns the currently-selected Integrated Circuit Card Identifier (ICCID), or SIM card.

3.11.1 Set command

The set command reads the currently-selected ICCID.

Command syntax:

```
AT#ICCID
```

Response syntax:

```
#ICCID: <id>
```

The <id> parameter returns a string of up to 20 characters followed by <CR><LF>.

Example:

```
AT#ICCID
#ICCID: 12345678901234567890
OK
```

3.11.2 Read command

The read command functions the same as the set command.

3.12 #SIMSELECT - Select SIM card interface

The #SIMSELECT command control the active SIM card interface.

3.12.1 Set command

The set command selects which SIM interface to use.

Command syntax:

```
AT#SIMSELECT=<interface>
```

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

<interface>

0 - Solder-down SIM card

1 - External SIM interface

Example - Select the external SIM interface:

```
AT#SIMSELECT=1
OK
```

3.12.2 Read command

The read command displays the currently-used SIM interface.

Response syntax:

```
#SIMSELECT: <interface>
```

Example:

```
AT#SIMSELECT?
#SIMSELECT: 0
OK
```

3.12.3 Test command

The test command displays what SIM interfaces are available.

Response syntax:

```
#SIMSELECT: (0,1)
```

Example:

```
AT#SIMSELECT=?
#SIMSELECT: (0,1)
OK
```

3.13 +CIMI - Request IMSI

The +CIMI command reads the International Mobile Subscriber Identity (IMSI) from the Universal Subscriber Identity Module (USIM) card.

3.13.1 Set command

The set command reads the IMSI from the Subscriber Identity Module (SIM) card.

Command syntax:

```
AT+CIMI
```

Response syntax:

```
<IMSI>
```

The response parameter and its defined value is the following:

<IMSI>

IMSI, a string without double quotes

Note: ERROR is returned if IMSI is not available.

Example:

```
AT+CIMI
284011234567890
OK
```

3.14 %SHORTSWVER - Short software identification

The %SHORTSWVER command requests short software identification.

3.14.1 Set command

The set command requests short software identification.

Command syntax:

```
AT%SHORTSWVER
```

Response syntax:

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

The response parameter and its defined value is the following:

<version_string>

A string without double quotes followed by <CR><LF>OK

Example:

```
AT%SHORTSWVER
%SHORTSWVER:
nrf9160_1_1_0
OK
```

3.15 %HWVERSION - Hardware identification

The %HWVERSION command requests hardware identification.

3.15.1 Set command

The set command requests hardware identification.

Command syntax:

```
AT%HWVERSION
```

Response syntax:

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

The response parameter and its defined value is the following:

<version_string>

A string without double quotes followed by <CR><LF>OK

Example:

```
AT%HWVERSION
%HWVERSION: B0A
OK
```

3.16 #REBOOT - Reboot the modem

The #REBOOT command gracefully reboots the modem.

3.16.1 Set command

The set command will reboot the modem. After you issue the command you will receive the +RESET URC, indicating that the modem has reset. This will be followed by the READY URC indicating that the modem is ready for use.

Example:

```
AT#REBOOT
OK
+RESET
```

3.17 #SHUTDOWN - Shutdown the modem

The #SHUTDOWN command gracefully shuts down the modem.

3.17.1 Set command

The set command will shutdown the modem. AFter you issue the command you will receive the #SHUTDOWN URC, indicating the modem has shutdown. At this time, it is safe to remove power.

Example:

```
AT#SHUTDOWN
OK
+SHUTDOWN
```

4 Mobile termination control and status commands

4.1 +CFUN - Functional mode

The +CFUN command sets and reads the modem functional mode.

4.1.1 Set command

The command sets the functional mode to Minimum (Power off), Normal, or Offline mode (Airplane mode). There is a specific mode for Airplane mode with SIM/Universal Integrated Circuit Card (UICC) on. It is also possible to activate or deactivate LTE or GNSS separately.

Command syntax:

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

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

<fun>

0 - Power off

1 - Normal mode

4 - Offline mode

20 - Deactivate LTE

21 - Activate LTE

30 - Deactivate GNSS

31 - Activate GNSS

44 - Offline mode without shutting down UICC

Note:

- %XSYSTEMMODE should be used for enabling system modes. It is possible to activate enabled modes.
- The response to changing to Normal mode could be ERROR if the SIM card has failed.
- Commanding the device to Power off or to Offline mode might take some time if signaling with the network is needed.

- When commanding the device to power off, wait for OK to make sure that Non-volatile Memory (NVM) has been updated.

Example - Activate Normal Mode:

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

4.1.2 Read command

The command reads the current functional mode.

Response syntax:

```
+CFUN: <funmode>
```

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

<funmode>

0 - Power off

1 - Normal mode. The active mode is either LTE or GNSS, or both.

4 - Flight mode

Example:

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

4.1.3 Test command

The test command lists supported functional modes.

Response syntax:

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

Example:

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

4.2 +CPIN - PIN code

The +CPIN command enters and checks the required Personal Identification Number (PIN).

4.2.1 Set command

The set command enters the PIN.

Command syntax:

```
AT+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 ERR0R.

Example - Enter PIN 1234

```
AT+CPIN="1234"
0K
```

4.2.2 Read command

The read command checks if a PIN is needed.

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

Example - Check if PIN is needed, with a response that it is required

```
AT+CPIN?
+CPIN: "SIM PIN"
OK
```

4.3 +CPINR - Remaining PIN retries

The +CPINR command returns the number of remaining PIN retries for the User Equipment (UE) passwords.

4.3.1 Set command

The set command returns the number of remaining PIN retries for the UE passwords.

Command syntax:

```
AT+CPINR=<sel_code>
```

Response syntax for standard PINs:

```
+CPINR: <code>,<retries>
```

Note: Manufacturer-specific PINs are not supported.

The 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.

Example - Check remaining retries for PIN:

```
AT+CPINR="SIM PIN"
+CPINR: "SIM PIN",3
OK
```

4.4 +CLAC - List all available AT commands

4.4.1 Set command

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

Command syntax:

```
AT+CLAC
```

Response syntax:

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

Example:

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

4.5 +CESQ Extended signal quality

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

4.5.1 Set command

The set command returns received signal quality parameters.

Command syntax:

```
AT+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

<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

Example:

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

4.5.2 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)
```

Example:

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

4.6 %CESQ - Signal quality notification

The %CESQ command subscribes or unsubscribes notifications of changes in signal quality.

4.6.1 Set command

The set command subscribes or unsubscribes notifications of changes in signal quality.

Command syntax:

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

Notification syntax:

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

The 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

<rsrp_threshold_index>

Index of RSRP threshold which is below measured RSRP value.

0 - RSRP is below the first threshold

1 - RSRP is between the first and second threshold

2 - RSRP is between the second and third threshold

3 - RSRP is between the third and fourth threshold

4 - RSRP is above the fourth threshold

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

<rsrq_threshold_index>

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

0 - RSRQ is below the first threshold

1 - RSRQ is between the first and second threshold

2 - RSRQ is between the second and third threshold

3 - RSRQ is between the third and fourth threshold

4 - RSRQ is above the fourth threshold

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

Example - Subscribe E-UTRA signal quality notifications:

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

Example - The notification 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
```

4.7 %XSNRSQ - SNR signal quality notification

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

4.7.1 Set command

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

Command syntax:

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

Notification syntax:

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

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

<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.

Example: Subscribes E-UTRA signal quality notifications:

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

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

```
AT%XSNRSQ: 39,3
OK
```

4.7.2 Read command

The read command reads SNR signal quality.

Response syntax:

```
%XSNRSQ: <snr>
```

Example:

```
AT%XSNRSQ?
%XSNRSQ: 39
OK
```

4.8 +CRSM - Restricted SIM access

The +CRSM command transmits restricted commands to SIM.

4.8.1 Set Command

The set command transmits restricted commands to the SIM.

Command Syntax:

```
AT+CRSM=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>[,<pathid>]]]]
```

Response syntax:

```
+CRSM: <sw1>,<sw2>[,<response>]
```

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 type. 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 type. Parameters passed on by the Mobile Termination (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 shall be written to the SIM.

<pathid>

String type. Contains the path of an elementary file on the SIM/UICC in hexadecimal format (e.g. "7F205F70" in SIM and UICC case). The <pathid> shall only be used in the mode "select by path from MF" as defined in ETSI TS 102 221.

<sw1>, <sw2>

Integer type. Information from the SIM about command execution. These parameters are delivered to the Terminal Equipment (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 will be returned. <response> is not returned after a successful UPDATE BINARY, UPDATE RECORD, or SET DATA command.

Example - Read the forbidden Public Land Mobile Network (PLMN) list:

```
AT+CRSM=176,28539,0,0,12
+CRSM: 144,0,"64F01064F040FFFFFFFFFFFF"
OK
```

4.9 +CSIM - Generic SIM access

The +CSIM command transmits a command to the SIM.

For reference, see 3GPP 27.007 Ch. 8.17 and ETSI TS 102 221 Ch. 10 and 11.

To avoid conflicts with modem firmware, AT+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)

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.

4.9.1 Set command

The set command transmits a command to the SIM.

Command syntax:

```
AT+CSIM=<length>,<command>
```

Response syntax:

```
+CSIM: <length>,<response>
```

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.

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
```

4.10 +CPAS - Device activity status

The +CPAS command returns the device activity status.

4.10.1 Set command

The set command returns the device activity status.

Command syntax:

```
AT+CPAS
```

Response syntax:

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

The command has the following parameter:

<pas>

Activity status.

0 - Ready (MT allows commands from Terminal Adapter (TA)/TE)

Example:

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

4.11 +CIND - Indicator control

The +CIND command sets indicator states.

4.11.1 Set command

The command sets indicator states.

Command syntax:

```
AT+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

Example: Enables service and message indicators:

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

Example: Notification indicates that the device is in service:

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

4.11.2 Read command

The command returns indicator states.

Response syntax:

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

The command has the following parameter:

<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

Example:

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

4.11.3 Test command

The 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

Example:

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

4.12 +CGPIAF - IP address format

The +CGPIAF command returns information about IPv6 address format.

4.12.1 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
```

4.12.2 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 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 CIDR notation

<IPv6_LeadingZeros>

1 - Leading zeros are included

<IPv6_CompressZeros>

0 - No zero compression

Example:

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

4.13 %XCBAND - Current band

The %XCBAND command returns the current E-UTRA band.

4.13.1 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, range 1-71. See 3GPP 36.101.

0 when current band information not available

Example:

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

4.13.2 Test command

The test command returns a list of supported bands.

Response syntax:

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

Example:

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

4.14 %NBRGRSRP - Read neighbor cells

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

4.14.1 Set command

The set command reads measured RSRP values of neighboring cells.

Note: Neighboring cell measurements are only valid and available when neighbors are monitored, i.e. current cell quality is low enough.

Command syntax:

```
AT%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. 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

Example:

```
AT%NBRGRSRP
%NBRGRSRP: 85,5230,14
OK
```

4.15 +CEMODE - Mode of operation (CS/PS)

The +CEMODE command sets the device mode of operation.

4.15.1 Set command

The command sets the CS/PS Mode of Operation. The mode is stored in the non-volatile memory when the device is powered off with +CFUN=0. The command should only be used when the modem is not activated.

Command syntax:

```
AT+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

Example:

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

4.15.2 Read command

The 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

Example:

```
+CEMODE: 0
OK
```

4.15.3 Test command

The test command lists the supported modes of operation.

Response syntax:

```
+CEMODE: (list of supported <mode>s)
```

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

Example:

```
+CEMODE: (0,2)
OK
```

4.16 %XSIM - UICC state

The %XSIM command subscribes UICC state notifications.

4.16.1 Set command

The set command subscribes UICC state notifications.

Command syntax:

```
AT%XSIM=<n>
```

Notification syntax:

```
%XSIM: <state>
```

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

<n>

0 - Unsubscribe XSIM notifications

1 - Subscribe XSIM notifications

<state>

0 - UICC not initialized

1 - UICC initialization OK

Example: Subscribe to UICC state notifications:

```
AT%XSIM=1
OK
```

Example: Notification indicates that UICC initialization is completed:

```
%XSIM: 1
```

4.16.2 Read command

The command reads the UICC state.

Response syntax:

```
%XSIM: <state>
```

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

<state>

0 - UICC not initialized

1 - UICC initialization OK

Example:

```
AT%XSIM?
%XSIM: 1
OK
```

4.17 %XPMNG - Public key storage management

The %XPMNG command writes and reads the public key. The public key can be written only if it does not exist. An existing key can be deleted with the %CMNG command.

4.17.1 Set command

The set command writes and reads the public key.

Syntax:

```
%XPMNG=<opcode>[,<content>[,<sec_tag>]]
```

Response syntax for read command:

```
%XPMNG: <content>
```

+CME ERROR codes

513 - For read: Not found

520 - For write: Already exists

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>

A secure tag for multiple public keys. Integer, 0-9. Optional

Example - Write the public key:

```
AT%XPMNG=0,"-----BEGIN PUBLIC KEY-----
    ...
    -----END PUBLIC KEY-----"
OK
```

Example - Read the public key:

```
AT%XPMNG=2
%XPMNG: "-----BEGIN PUBLIC KEY-----
    ...
    -----END PUBLIC KEY-----"
OK
```

4.18 %XRFTEST - RF test execution

The %XRFTEST command performs RF testing.

4.18.1 Set command

The set command performs RF testing.

Command syntax:

```
%XRFTEST=<test>,<operation>,<param0>,<param1>,...,<param7>
```

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

<test>

0 – RX

1 – TX

2 – GPS SNR

3 – RX SNR

<operation>

0 – OFF

1 – ON

<paramX>

One or more int16 values. The usage and number of parameters depends on <test> and <operation>. See the following sections.

4.18.1.1 RX testing The command enables RF receiver with the given parameters. It also measures antenna power with a time domain power meter and returns the measurement result.

The command parameter and its value are the following:

<test>

0 – RX

RX ON

<operation>

1 – ON

RX ON has a total of four parameters:

<param0>

3GPP band number.

<param1>

Frequency 100 kHz.

Valid range 6000–22000 (corresponds to 600.0 MHz–2200.0 MHz). Note that if CW is used, an offset of about 45 kHz for NB1 and 300 kHz for M1 is recommended.

<param2>

RX signal power at antenna in dBm.

Valid range from −127 to −25.

<param3>

System mode.

Valid range 0–1. NB1 (0) or M1 (1).

Response syntax when <operation> is ON:

```
%XRFTEST: <antenna_power>
```

The response value is the following:

<antenna_power>

Measured power at antenna, 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, NB1 mode:

```
%XRFTEST=0,1,1,21400,-65,0
%XRFTEST: -17002
OK
```

Note: −17002/256 = −66.4 dBm

RX OFF

<operation>

```
0 - OFF
```

The following command example disables the RF receiver:

```
%XRFTEST=0,0
0K
```

Note: Always send the OFF command before sending another ON command.

4.18.1.2 TX testing The command enables RF transmitter with the given parameters. It also measures TX power with an internal measurement receiver in time domain, and returns the measurement result.

CAUTION: This command transmits power to the selected RF band and may violate the radio directives of the region or country. Make sure that the equipment is in an RF-shielded room or connected to an RF cable so that RF power will not leak.

The command parameter and its value are the following:

<test>

1 – TX

TX ON

<operation>

1 – ON

TX ON has a total of seven parameters:

<param0>

3GPP band number.

<param1>

Frequency [100kHz].

Valid range 6000–22000 (corresponds to 600.0 MHz–2200.0 MHz).

<param2>

TX signal power at antenna [dBm].

Valid range from +23 to −50.

<param3>

System mode.

Valid range 0–1. NB1 (0) or M1 (1).

<param4>

Modulation.

0 – QPSK

1 – 16QAM

2 – Reserved

3 – BPSK

4 – CW

M1: QPSK, 16QAM, and CW

NB1: QPSK, BPSK, and CW

<param5>, <param6>, <param7>

<param5> RB/Tone count

<param6> RB/Tone start position

<param7> Subcarrier spacing

If <param4> = CW, then <param5>, <param6>, and <param7> = 0 (do not care)

The allowed combinations for <param5>, <param6>, and <param7> for both system mode (<param3>) values are listed in the table below:

System mode <param3>	RB/Tone count <param5>	RB/Tone start position <param6>	Subcarrier spacing <param7>
NB1 (0)	1	0-11	0
	3	0, 3, 6, 9	0
	6	0, 6	0
	12	0	0
	1	0-47	1
M1 (1)	1	0-5	0
	2	0-4	0
	3	0-3	0
	4	0-2	0
	5	0-1	0
	6	0	0

Note: In system modem M1, subcarrier spacing 3.75 kHz is not allowed.

Response syntax when <operation> is ON:

```
%XRFTEST: <antenna_power>
```

The response value is the following:

<antenna_power>

Internally measured TX power at antenna, in q4 dBm. q4 means that dividing the result by 2^4 = 16 gives dBm.

The following command example enables the RF transmitter for Band 5, 830.0 MHz, +17 dBm, NB1, BPSK, 12 tones, tone start position 0, subcarrier spacing 15 kHz:

```
%XRFTEST=1,1,5,8300,17,0,3,12,0,0
%XRFTEST: 271
0K
```

Note: 271/16 = 16.9 dBm

TX OFF

<operation>

0 – OFF

The following command example disables the RF transmitter:

```
%XRFTEST=1,0
0K
```

Note: Always send the OFF command before sending another ON command.

4.18.1.3 GPS SNR testing

The command executes a GPS SNR test.

GPS L1 frequency is 1575.42 MHz and this test expects the CW in signal generator to be 1575.750 MHz, i.e. the offset is 330 kHz. The measurement duration is 1 ms.

The command parameter and its value are the following:

<test>

2 – GPS SNR

GPS SNR

<operation>

1 – ON

Note: Automatic stop, i.e. no "OFF" needed.

GPS SNR ON has one parameter:

<param0>

RX signal power at antenna in dBm

Valid range from −127 to −25 or 0 = default gain −105 dBm.

Response syntax when <operation> is ON:

```
%XRFTEST: <snr>,<antenna_power>
```

The response value is the following:

<snr>

The result of the SNR measurement in q4 dB. q4 means that dividing the result by 2^4 = 16 gives dB.

<antenna_power>

Measured power at antenna, in q8 dBm. q8 means that dividing the result by 2^8 = 256 gives dBm.

The following command example executes an GPS SNR test:

```
%XRFTEST=2,1,0
%XRFTEST: 514,-19968
0K
```

Note: 514/16 = 32.125 dB and −19968/256 = −78 dBm.

4.18.1.4 RX SNR testing

In order to measure SNR correctly, the CW offset must be +330 kHz for the M1 mode and +45 kHz for NB1.

The parameters and their values are the following:

<test>

3 – RX SNR

<operation>

1 – ON

Note: Automatic stop, i.e. no "OFF" needed.

<param0>

3GPP band number.

<param1>

Frequency 100 kHz (,i.e. 2140 MHz is expressed as 21400).

<param2>

RX signal power at antenna in dBm.

Valid range from −127 to −25.

<param3>

System mode.

Valid range 0–1. NB1 (0) or M1 (1).

Response syntax when <operation> is ON:

```
%XRFTEST: <snr>,<antenna_power>
```

The response parameters and their values are the following:

<snr>

Result of the SNR measurement in q4 dB. q4 means that dividing the result by 2^4 = 16 dB.

<antenna_power>

Measured power at antenna, in q8 dBm. q8 means divided dividing the result by 2^8 = 256 gives dBm.

The following command example enables the RX SNR measurement and RF receiver for Band 1, 2140.0 MHz, −65 dBm, NB1 mode:

```
%RFTEST=3,1,1,21400,-65,0
%XRFTEST: 496,-17002
OK
```

Note: 496/16 = 31 dB and −17002/256 = −66.4 dBm.

4.19 %XDATAPRFL - Data profile

The %XDATAPRFL command can be used to provide information on the application use case to modem so that it can optimize power consumption.

Note: This command is for future releases and will be extended with new parameters later. In the current software release, the use of this command has limited impact on power consumption.

4.19.1 Set command

The set command provides information on the application use case to modem. The purpose of this command is to control the power-saving parameters of the modem.

Levels 4 and 3 are meant for devices that can prioritize the time spent on finding service over power consumption. Battery-operated devices should use levels 2, 1, or 0. In the current software release, the power-saving level has an effect on UICC deactivation and network search frequencies.

Command syntax:

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

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

<power level>

0 - Ultra-low power

1 - Low power

2 - Normal

3 - Performance

4 - High performance

Example:

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

4.19.2 Read command

The read command reads the application data profile.

Syntax:

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

Example:

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

4.20 %XCONNSTAT - Connectivity statistics

The %XCONNSTAT command sets the connectivity statistics command.

4.20.1 Set command

The set command sets the connectivity statistics command.

Command syntax:

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

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

<command>

0 - Stop

1 - Start

Example:

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

4.20.2 Read command

The read command reads the connectivity statistics.

Response 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.

Example:

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

4.21 %XVBAT - Battery voltage

The %XVBAT command reads battery voltage.

When the modem is active (either LTE communication or GPS 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 sleeps, such as eDRX and PSM, are modem active time. Therefore, in those cases the %XVBAT value returned is from the time just before entering the sleep or from previous GPS reception during the eDRX/PSM gap.

4.21.1 Set command

The set command reads the battery voltage in mV.

Command syntax:

```
AT%XVBAT
```

Response syntax:

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

The response parameter is the following:

<vbat>

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

Example:

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

4.22 %CMNG - Credential storage management

The %CMNG command is used for credential storage management. The command writes, reads, deletes, and checks the existence of keys and certificates. The credentials are stored in the non-volatile memory.

4.22.1 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 write and delete operations are allowed only when the modem is not activated.

Command syntax:

```
AT%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 codes

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

The 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.

<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)

Mandatory if <opcode> is write, read, or delete. Parameter <type> with the value Public Key can only be used when parameter <opcode> is delete.

<content>

String. 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). Base64- encoded string in double quotes (PSK).

<passwd>

String. PKCS#8 password. Mandatory for writing a type 2 encrypted private key, ignored for other types. Maximum length 32 characters.

<sha>

String. SHA-256 digest of the entity (DER, PEM) as stored in the filesystem, 64 hexadecimal 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.

Example - Write the root certificate:

```
AT%CMNG=0, 12345678, 0,"
-----BEGIN CERTIFICATE-----
MIIDSjCCA...
...bKbYK7p2CNTUQ
-----END CERTIFICATE-----”
OK
```

Example - Write client certificate:

```
AT%CMNG=0,567890,1,"
-----BEGIN CERTIFICATE-----
MIIBc464...
...bW9aAa4
-----END CERTIFICATE-----”
OK
```

Example - Writes the private key:

```
AT%CMNG=0,123,2,"
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIICz...
...ukBu
-----END ENCRYPTED PRIVATE KEY-----", "abcdefg"
OK
```

Example - List a single item by specifying tag and type:

```
AT%CMNG=1,12345678, 0
%CMNG: 12345678, 0, "978C...02C4"
OK
```

Example - Lists a single tag:

```
AT%CMNG=1,12345678
%CMNG: 12345678, 0, "978C...02C4"
%CMNG: 12345678, 1, "1A8C...02BB"
OK
```

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
```

Example - Read the root certificate with tag 12345678:

```
AT%CMNG=2, 12345678, 0
%CMNG: 12345678, 0, "978C...02C4",
"-----BEGIN CERTIFICATE-----
MIIBc464...
...bW9aAa4
-----END CERTIFICATE-----"
OK
```

Example - Delete a client certificate with tag 123:

```
AT%CMNG=3,123,1
OK
```

Example - Read a non-existing root certificate with tag 4567. Error code 513 is returned:

```
AT%CMNG=2,4567,0
+CME ERROR: 513
```

4.23 %XTEMP - Internal temperature

The %XTEMP command subscribes unsolicited internal temperature notifications.

4.23.1 Set command

The set command subscribes or unsubscribes 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.

Command syntax:

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

Notification syntax:

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

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

<n>

0 - Subscribe unsolicited temperature indications

1 - Unsubscribe 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 High level for internal temperature %XTEMPHIGHLVL.

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

<temperature>

Integer. Celsius degrees between −40 and 125.

Example - Subscribe to notifications:

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

Example - Unsolicited notification for an internal temperature level:

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

4.23.2 Read command

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

Command syntax:

```
AT%XTEMP?
```

Response syntax:

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

Example:

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

4.24 %XTEMPHIGHLVL - High level for internal temperature

The %XTEMPHIGHLVL command sets the high level to internal temperature in the modem.

4.24.1 Set command

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

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

Command syntax:

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

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

<temperature>

Integer. Celsius degrees between 1 and 85. Factory default 55.

Example:

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

4.24.2 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.

Command syntax:

```
AT%XTEMPHIGHLVL?
```

Example:

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

4.25 +CCLK Clock

The +CCLK command sets the clock of the device.

4.25.1 Set command

The set command sets the real-time clock of the UE.

Command syntax:

```
AT+CCLK=<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 −48...+48).

Example:

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

4.25.2 Read command

The read command reads the real-time clock.

Response syntax:

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

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 −48...+48).

Example:

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

4.26 %CCLK - Proprietary clock

The %CCLK command sets the real-time clock of the device.

4.26.1 Set command

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

Command syntax:

```
AT%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 −48...+48 and 99 for "not set" or "unknown").

<daylight_saving_time>

0 - No adjustment of daylight saving time

1 - +1 hour adjustment of daylight saving time

2 - +2 hours adjustment of daylight saving time

Example:

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

4.26.2 Read command

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

Response syntax:

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

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 −48...+48).

<daylight_saving_time>

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

0 - No adjustment of daylight saving time

1 - +1 hour adjustment of daylight saving time

2 - +2 hours adjustment of daylight saving time

Example:

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

4.27 %XMODEMTRACE - Modem trace activation

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

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

4.27.1 Set command

The set command activates and deactivates modem trace.

Command syntax:

```
AT%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 - Generic

3 - LWM2M

4 - IP only

5 - LWM2M_Generic

<bitmap_id>

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

<bitmap>

String, hexadecimal data represented with an IRA string. Used only with the assistance of Nordic customer support.

Example - Activates trace set 1 (Coredump only):

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

Example - Deactivate trace:

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

4.28 %XUSIMLCK - Personalization of modem

The %XUSIMLCK command allows personalizing the modem to work with predefined USIM cards.

4.28.1 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

Command syntax:

```
AT%XUSIMLCK=<command>,<facility>,[<pwd>,[<permanent>,
    [<pers_data>]]]
```

The modem supports a maximum of 24 personalization codes.

The 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 - Non-permanent 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 chapter 4.2.10 EF.

When <facility> is PC, <pers_data> can contain a maximum of 24 pairs of USIM group identifiers from EF and 4.2.11 EF 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 chapters 4.2.10 EF and 4.2.11 EF.

When <facility> is PS, <pers_data> can contain a maximum of 24 IMSIs as specified in 3GPP TS 31.102 chapter 4.2.2 EF. Fifteen IMSI digits can be given. The format is the following:

IMSI1:IMSI2:...:IMSIn.

Example - Create a non-permanent network personalization:

```
AT%XUSIMLCK=1,"PN","12345678",0,"100.200"
OK
```

Example - Depersonalize the network personalization:

```
AT%XUSIMLCK=2,"PN","12345678"
OK
```

Example - Disable network personalization:

```
AT%XUSIMLCK=3,"PN"
OK
```

Example - Lock device to first inserted USIM in a non permanent manner:

```
AT%XUSIMLCK=4,"PS","12345678",0
OK
```

Example - Personalized 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
```

4.29 %XSMSFALLBACK - Fallback to SMS only

The %XSMSFALLBACK command sets the SMS only fallback functionality. With SMS only fallback, UE triggers a Tracking Area Update (TAU) request for SMS only immediately 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.

4.29.1 Set command

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

Command syntax:

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

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

<fallback_status>

0 - Fallback is not performed

1 - Fallback is performed

Example:

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

4.30 %XSYSTEMMODE - System mode

The %XSYSTEMMODE command sets the modem system mode.

4.30.1 Set command

The set command sets the supported system modes of the modem.

Note: Only one supported LTE mode allowed at a time. This command is allowed only before activating the modem using the CFUN=1 command. If the mode needs to be changed,the modem must first be set to flight mode using the CFUN=4 command.

Note: NB1 is not supported at this time.

Command syntax:

```
AT%XSYSTEMMODE=<M1_support>,<NB1_support>,
    <GNSS_support>,<LTE_preference>
```

+CME error codes

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:

<M1_support>

0 - LTE Cat-M1 not supported

1 - LTE Cat-M1 supported

<NB1_support>

0 - LTE Cat-NB1 not supported

1 - LTE Cat-NB1 supported

<GNSS_support>

0 - Global Navigation Satellite System (GNSS) not supported

1 - GNSS supported

<LTE_preference>

<LTE preference> is for the coming releases. Not relevant in the current release.

0 - No preference

1 - LTE Cat-M1 preferred

2 - LTE Cat-NB1 preferred

Example - Set LTE Cat-M1 and GNSS as the system modes. No preferred LTE mode set:

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

4.30.2 Read command

The read command reads the supported modem system modes.

Response syntax:

```
%XSYSTEMMODE: <M1_support>,<NB1_support>,
    <GNSS_support>,<LTE_preference>
```

Example:

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

4.31 %XPTW - PTW setting

The %XPTW command sets the Paging Time Window (PTW).

4.31.1 Set command

The set command sets the requested Paging Time Window (PTW) parameters.

Note: Use the command with caution. The requested values must be compliant with the eDRX cycle values configured using the +CEDRXS command. The modem will use 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. If other than the default PTW has to be used, the %XPTW command shall be sent after the +CEDRXS command. See eDRX setting +CEDRXS.

Command syntax:

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

The set command 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 a 4-bit format. The PTW value refers to bits from 8 to 5 of octet 3 of the Extended Discontinuous Reception (eDRX) parameters information

element (see subclause 10.5.5.32 of 3GPP TS 24.008). Optional. If not present, the value of the requested AcT-type is reset to the manufacturer-specific default.

Example:

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

4.31.2 Read command

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

Response syntax:

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

Example:

```
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 IP socket dial commands

IP socket dial commands can be used to establish and use TCP and UDP connections. A maximum of four sockets can be created and used.

5.1 #XSOCKET - Manage sockets

The #XSOCKET command manages opening and closing IP sockets.

5.1.1 Set command

The set command manages opening and closing IP sockets.

Command syntax:

```
AT#XSOCKET=<sock>,<action>,<type>[,<protocol>]
```

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

<action>

0 - Free socket

1 - Create socket

For <sock> values, please see Section 5.1.1.1.

For <protocol> values, please see Section 5.1.1.2.

5.1.1.1 Free socket When <action> is 0, a socket is specified to be closed.

Command syntax:

```
AT#XSOCKET=<sock>,0
```

Response syntax:

```
#XSOCKET: <retval>
```

The free socket parameter and its defined values are the following:

<sock>

Socket index provided when a socket was opened with <action> set to 1.

<retval>

0 - Success

Example - Close socket 3:

```
AT#XSOCKET=0,3
#XSOCKET: 0
OK
```

5.1.1.2 Create socket When <action> is 1, a socket is requested to be created to be used.

Command syntax:

```
AT#XSOCKET=<sock>,1,<type>[,<protocol>]
```

Response syntax:

```
#XSOCKET: <sock>,<type>
```

The create socket command parameters and their defined values are the following:

<sock>

Index of created socket.

<type>

1 - Stream socket

2 - Datagram socket

3 - Raw socket

4 - Management socket

<protocol>

0 - IP

1 - ICMP

6 - TCP

17 - UDP

41 - IPv6

58 - ICMPv6

238 - TLS 1.2

255 - RAW IP

Example - Create a TCP socket, and socket is created at index 1:

```
AT#XSOCKET=1,1,1
#XSOCKET: 1,6
OK
```

5.1.2 Read command

The read command queries all created sockets.

Response syntax:

```
#XSOCKET: <sock1>,<type1>
[#XSOCKET: <sock2>,<type2>]
...
```

Example:

```
AT#XSOCKET?
#XSOCKET: 2,6
#XSOCKET: 3,17
OK
```

5.1.3 Test command

The test command lists possible commands for creating and freeing sockets.

Response syntax:

```
#XSOCKET=<sock>,0
#XSOCKET=<sock>,1,(1|2)[,(6|17)]
```

Example:

```
AT#XSOCKET=?
#XSOCKET=<sock>,0
#XSOCKET=<sock>,1,(1|2)[,(6|17)]
OK
```

5.2 #XBIND - Bind port to socket

The #XBIND command binds a given socket to a given port. Socket creating is handled with #XSOCKET.

5.2.1 Set command

The set command binds a given socket to a given port.

Command syntax:

```
AT#XBIND=<sock>,<port>
```

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

<sock>

Socket number provided by #XSOCKET.

<port>

Port number to use.

Example - Bind port 80 to socket 3:

```
AT#XBIND=3,80
OK
```

5.2.2 Test command

The test command displays arguments for the #XBIND command.

Response syntax:

```
#XBIND=<sock>,<port>
OK
```

Example:

```
AT#XBIND=<sock>,<port>
OK
```

5.3 #XUDPSENDTO - Send UDP data

The #XUDPSENDTO command sends data using UDP.

Note: you must first create a socket with #XSOCKET.

5.3.1 Set command

The set command sends data using UDP.

Command syntax:

```
AT#XUDPSENDTO=<sock>,"<url>",<port>[,<length>]
```

Once issued, a > prompt will appear. At this time, data can be sent to the modem. <length> is optional. If it is provided, once <length> bytes has been received by the modem, it will automatically send the data.

If <length> is not provided, enter the desired data and enter hex value 0x1A (ASCII SUB, CTRL+Z on the keyboard) to signal the modem to send data.

The set parameters and their defined values are the following:

<sock>

Socket index provided by #XSOCKET.

<url>

URL to connect to.

<port>

Port to use.

<length>

Length of the data to be send.

Example - Send 11 bytes of data (`hello world`) to example.com on port 123 using socket 6:

```
AT#XUDPSENDTO=6,”example.com”,123,11
> hello world
OK
```

5.3.2 Test command

The test command lists arguements for the set command.

Response syntax:

```
#XUDPSENDTO=<sock>,"<url>",<port>[,<length>]
><data>
OK
```

Example:

```
AT#XUDPSENDTO=?
#XUDPSENDTO=<sock>,"<url>",<port>[,<length>]
><data>
OK
```

5.4 #XUDPRECVFROM - Receive UDP data

The #XUDPRECVFROM command received data using UDP.

5.4.1 Set command

The set command receives data using UDP.

Command syntax:

```
AT#XUDPRECVFROM=<sock>,"<url>",<port>,<length>,<timeout>
```

Response syntax:

```
#XUDPRECVFROM: <len_received>,<data>
```

The set parameters and their defined values are the following:

<sock>

Socket index provided by #XSOCKET.

<url>

URL to connect to.

<port>

Port to use.

<length>

Length of the data to be received.

<timeout>

Timeout to receive data, in seconds

<len_received>

Actual amount of data received.

Example - Request 20 bytes from example.com, port 123, socket 6, with a timeout of 10 seconds. Actually received 11 bytes.

```
AT#XUDPRECVFROM=6,"example.com",123,20,10
#XUDPRECVFROM: 11,hello world
OK
```

5.4.2 Test command

The test command lists arguments for the set command.

Response syntax:

```
#XUDPRECVFROM=<sock>,"<url>",<port>,<length>,<timeout>
OK
```

Example:

```
AT#XUDPRECVFROM=?
#XUDPRECVFROM=<sock>,"<url>",<port>,<length>,<timeout>
OK
```

5.5 #XTCPCONN - Open a TCP connection

The #XTCPCONN command opens a TCP socket created by #XSOCKET.

5.5.1 Set command

The set command opens a TCP socket.

Command syntax:

```
AT#XTCPCONN=<sock>,”<url>”,<port>
```

Response syntax:

```
#XTCPCONN: <retval>
```

The set parameters and their defined values are the following:

<sock>

Socket index provided by #XSOCKET.

<url>

URL to connect to.

<port>

Port to use.

<retval>

1 - Success

Example - Open socket to example.com, port 80, using socket 3:

```
AT#XTCPCONN=3,”example.com”,80
#XTCPCONN: 1
OK
```

5.5.2 Test command

The test command lists arguments for the set command.

Response syntax:

```
#XTCPCONN: <sock>,”<url>”,<port>
```

Example:

```
AT#XTCPCONN=?
#XTCPCONN: <sock>,”<url>”,<port>
OK
```

5.6 #XTCPSEND - Send TCP data

The #XTCPSEND command sends data on TCP socket opened by #XTCPCONN.

5.6.1 Set command

The set command sends TCP data.

Command syntax:

```
AT#XTCPSEND=<sock>[,<length>]
```

Once issued, a > prompt will appear. At this time, data can be sent to the modem.

<length> is optional. If it is provided, once <length> bytes has been received by the modem, it will automatically send the data.

If <length> is not provided, enter the desired data and enter hex value 0x1A (ASCII SUB, CTRL+Z on the keyboard) to signal the modem to send data.

Note: Maximum size is 576 bytes regardless of the presence of <length>.

The set parameters and their defined values are the following:

<sock>

Socket index provided by #XSOCKET.

<length>

Optional. Number of bytes to buffer before sending.

Example - Send 11 bytes on socket 3:

```
AT#XTCPSEND=3,11
>hello world
OK
```

5.6.2 Test command

Response syntax:

```
#XTCPSEND=<sock>(,<length>)
><data>(^Z^Z)
OK
```

Example:

```
AT#XTCPSEND=?
#XTCPSEND=<sock>(,<length>)
><data>(^Z^Z)
OK
```

5.7 #XTCPRECV - Receive TCP data

The #XTCPRECV command received TCP data from a socket opened by #XTCPCONN.

5.7.1 Set command

The set command receives TCP data from a socket.

Command syntax:

```
AT#XTCPRECV=<sock>,<length>,<timeout>
```

Response syntax:

```
#XTCPRECV: <len_received>,<data>
```

The set parameters and their defined values are the following:

<sock>

Socket index provided by #XSOCKET.

<length>

Length to receive, in bytes.

<timeout>

Timeout to receive, in seconds.

<len_received>

Actual length of data received.

<data>

Data received.

Example - Recieve 11 bytes of data on socket 3 with a timeout of 10 seconds:

```
AT#XTCPRECV=3,11,10
#XTCPRECV: 11,hello world
OK
```

5.7.2 Test command

The test command displays parameters for the set command.

Response syntax:

```
#XTCPRECV=<sock>,<length>,<timeout>
```

Example:

```
AT#XTCPRECV=?
#XTCPRECV=<sock>,<length>,<timeout>
OK
```

5.8 #XTLSSOCKET - Manage TLS Sockets

The #XTLSSOCKET command manages opening and closing IP sockets.

5.8.1 Set command

The set command manages opening and closing IP sockets.

Command syntax:

```
AT#XTLSSOCKET=<sock>,<action>,"<hostname>",<stag>[,<stag>[...]]
```

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

<action>

0 - Close socket

1 - Open socket

For <sock> values, please see Section 5.8.1.1.

For <hostname> and <stag> values, please see Section 5.8.1.2.

5.8.1.1 Close and free socket

When <action> is 0, a socket is specified to be closed.

Command syntax:

```
AT#XTLSSOCKET=<sock>,0
```

Response syntax:

```
#XTLSSOCKET: <retval>
```

The free socket parameter and its defined values are the following:

<sock>

Socket index provided when a socket was opened with <action> set to 1.

<retval>

0 - Success

Example - Close socket 3:

```
AT#XTLSSOCKET=0,3
#XTLSSOCKET: 0
OK
```

5.8.1.2 Create and open socket

When <action> is 1, a socket is opened.

Command syntax:

```
AT#XTLSSOCKET=1,"<hostname>",<stag>[,<stag>[...]]
```

Response syntax:

```
#XTLSSOCKET: <sock>,<type>
```

The create socket command parameters and their defined values are the following:

<hostname>

Hostname to connect to

<stag>

Socket tag.

<sock>

Index of created socket.

<type>

6 - TCP

17 - UDP

238 - TLS 1.2

Example - Create a TCP socket, and socket is created at index 2:

```
AT#XTLSSOCKET=1,1
#XTLSSOCKET: 2,6
OK
```

5.8.2 Test command

The test command lists possible commands for creating and freeing TLS sockets.

Response syntax:

```
#XTLSSOCKET=<sock>,0
#XTLSSOCKET=<sock>,1,"<hostname>",<stag>[,<stag>[...]]
OK
```

Example:

```
AT#XTLSSOCKET=?
#XTLSSOCKET=<sock>,0
#XTLSSOCKET=<sock>,1,"<hostname>",<stag>[,<stag>[...]]
OK
```

6 SiP pin configuration

SiP pin configuration commands can be used to configure the behavior of selected pins of the nRF91 System in Package (SiP). The pins that can currently be configured are COEX0 and MAGPIO[0:2].

For more information on the nRF9160 SiP pins, see Pin assignments in nRF9160 Product Specification. The control of these pins is tied to the modem operations, i.e. the pins are only controllable when the modem is active. For example, if the modem goes to a long Power Saving Mode (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.

Note: The commands in this chapter are intended to be given only once at boot or, alternatively, e.g. in final device production where `AT+CFUN=0` must be given to store the command contents to flash memory. After giving the commands, the modem software will automatically toggle 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 %XCOEX0 - COEX0 pin control configuration

The `%XCOEX0` command writes the COEX0 pin configuration to device's RAM memory.

The COEX0 pin can be configured to switch its state based on the modem's RF frequency, for example, to enable external Low-Noise Amplifier (LNA) in GPS mode. The behavior is similar to the `%XMAGPIO` command with the difference that this command only controls one pin.

The AT 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. The configuration is stored to NVM using `+CFUN=0` when the device is powered off. The stored configuration is applied when the device is powered on. When RF is turned off, the given COEX0 state is inverted.

6.1.1 Set command

The set command writes the COEX0 pin configuration to device's RAM memory.

Command syntax:

```
AT%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.

Example - Set COEX0 to '1' when GPS is enabled (and '0' when GPS is turned off). COEX0 is not used with other frequencies (or LTE):

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

Example - Set COEX0 to '1' when GPS is enabled, or LTE frequency is 600-800 MHz or 2000-2180 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 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>
```

Example:

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

6.2 %XMAGPIO - MAGPIO configuration

The %XMAGPIO command writes the MAGPIO configuration to device's RAM memory.

The MAGPIO pins can be used, for example, to control an external antenna tuner, or any other GPIO-controlled device, whose state depends on modem's RF frequency. The AT 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. The configuration is stored to NVM when the device is powered off with +CFUN=0. The stored configuration is applied when the device is powered on.

6.2.1 Set command

The set command writes the MAGPIO configuration to device's RAM memory.

Command syntax:

```
AT%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

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
GPS	7	1	1	1	1574	1577

Example - Write seven ranges to device's 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
```

Example - Write three ranges to device's RAM:

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

Example - Delete the previously written values:

```
AT%XMAGPIO
OK
```

6.2.2 Read command

The 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>,...
```

Example:

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

6.3 %XANTDETMAGPIO - Antenna detection test

The `%XANTDETMAGPIO` command reads the MAGPIO pin status to detect if the antenna is connected. The antenna is detected when the pin is DC-grounded.

This command can be used, for example, in device production testing provided that the necessary circuitry between an MAGPIO pin and the antenna is in place. See Antenna presence test using MAGPIO in nWP033 - nRF9160 Antenna and RF Interface Guidelines.

6.3.1 Test command

The test command changes the pin specified in the command to input mode and sets internal pull-up for the corresponding pin.

After this, the pin state is read and reported in the command response.

Command syntax:

```
AT%XANTDETMAGPIO=<magpio_pin>
```

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

<magpio_pin>

0, 1, 2 - The MAGPIO pin whose state is read.

The response values are the following:

1 - Antenna connected

0 - Antenna not connected

After the command, the pin state is set back to normal (high-Z).

Example - Set MAGPIO pin 2 to input mode and sets internal pull-up for it. After this, the pin state is read and reported in the command response. After the command, the pin state is set back to normal (high impedance state):

```
AT%XANTDETMAGPIO=2
%XANTDETMAGPIO: 0
OK
```

7 Packet domain commands

Commands for the packet domain include commands that control packet-switched services.

7.1 +CGDCONT - Define PDP Context

The +CGDCONT command defines Packet Data Protocol (PDP) Context.

7.1.1 Set command

The set command configures connection parameters.

Command syntax:

```
AT+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.

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

<cid>

0-11 (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 type

IP - Internet Protocol

IPV6 - Internet Protocol version 6

IPV4V6 - Virtual type of dual IP stack

<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

Example - Configure CID 1 to use IPv4 and access point "IOT_apn"

```
AT+CGDCONT=1,"IP","IOT_apn"
OK
```

7.1.2 Read command

The 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>

0-11

<PDP_type>

String type

IP - Internet Protocol

IPV6 - Internet Protocol version 6

IPV4V6 - Virtual type of dual IP stack

<APN>

String - APN

<PDP_addr>

String - IP address

<d_comp>

0 - Compression not supported

<h_comp>

0 - Compression not supported

Example:

```
AT+CGDCONT?
+CGDCONT: 0,"IP","internet","10.0.1.1",0,0
+CGDCONT: 1,"IP","IOT_apn","10.0.1.2",0,0
OK
```

7.2 +CGEREP - Packet domain event reporting

The +CGEREP command enables or disables the sending of packet domain events.

7.2.1 Set command

The set command enables or disables the sending of packet domain events. The unsolicited result code is +CGEV: XXX.

For information on +CGEV, see Packet domain event unsolicited result codes +CGEV.

Command syntax:

```
AT+CGEREP=[<mode>]
```

The command parameter and its defined values are the following:

<mode>

0 - Do not forward unsolicited result codes to the TE (default).

1 - Discard unsolicited result codes when the MT TE link is reserved. Otherwise, forward them directly to the TE.

Example - Subscribe CGEV notifications:

```
AT+CGEREP=1
OK
```

7.2.2 Read command

The command reads the current mode and buffering settings.

Response syntax:

```
+CGEREP: <mode>,<bfr>
```

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

<mode>

0 - Do not forward unsolicited result codes to the TE (default).

1 - Discard unsolicited result codes when the MT TE link is reserved. Otherwise, forward them directly to the TE.

<bfr>

0 - MT buffer of unsolicited result codes is cleared when <mode> 1 is entered

Example:

```
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 - Do not forward unsolicited result codes to the TE (default).

1 - Discard unsolicited result codes when the MT TE link is reserved. Otherwise, forward them directly to the TE.

<bfr>

0 - MT buffer of unsolicited result codes is cleared when <mode> 1 is entered

Example:

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

7.3 +CGEV - Packet domain event unsolicited result codes

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.

These notifications are subscribed using the +CGEREP command.

Syntax descriptions are listed below:

Network detach:

```
+CGEV: NW DETACH
```

Mobile Equipment (ME) detach:

```
+CGEV: ME DETACH
```

ME overheated and flight mode enabled

```
+CGEV: ME OVERHEATED
```

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>
```

<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 summing all the applicable bits.

Bit 1 - TFT changed

Bit 2 - QoS changed

Bit 3 - WLAN offload changed

<cid_other>

1-11: Indicates the context identifier allocated for an MT-initiated context of a second address type. This parameter is included only if <reason> parameter indicates that only single address bearers are allowed.

<p_cid>

0-11: 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 - Permanen restriction. Enabling requires e.g. a power-off, UICC change, or a configuration change.

2 - Temporary restriction. Enabling requires e.g. back-off timer expiry.

Example - Notification shows that an initial Packet Data Network (PDN) connection is activated:

```
+CGEV: ME PDN ACT 0
```

Example - Notification shows that the device is detached from network:

```
+CGEV: ME DETACH
```

Example - Notification shows a restriction caused by throttling with temporary validity:

```
+CGEV: RESTR 2,2
```

7.4 +CGACT - PDP context activate

The +CGACT command activates or deactivates a PDN connection.

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.

Command syntax:

```
AT+CGACT=<state>,<cid>
```

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

<state>

0 - Deactivate

1 - Activate

<cid>

1-11

Example:

```
AT+CGACT=1,1
OK
```

7.4.2 Read command

The 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>

0-11

Example:

```
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)
```

Example:

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

7.5 %XNEWCID - Allocate new CID

The %XNEWCID command allocates a new context identifier.

7.5.1 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.

This command can be used instead of reading existing default and dedicated contexts with AT+CGDCONT? and finding an unused <cid> value before configuring new context.

Response syntax:

```
%XNEWCID: <cid>
```

The command parameter and its defined values are the following:

<cid>

1-11

Example:

```
AT%XNEWCID?
%XNEWCID: 2
OK
```

7.6 %XGETPDNID - Map CID to PDN ID

The %XGETPDNID command maps the context identifier to PDN ID. This command can be used only when the modem is activated.

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.

Command syntax:

```
AT%XGETPDNID=<cid>
```

Response syntax:

```
%XGETPDNID:P <pdn_id>
```

The command parameters and their defined values are the following:

<cid>

0-11

<pdn_id>

0-20

Example:

```
AT%XGETPDNID=0
%XGETPDNID: 1
OK
```

7.7 +CGEQOSRDP - QoS dynamic params

The +CGEQOSRDP command reads dynamic Evolved Packet System (EPS) Quality of Service (QoS) parameters. This command issues a valid response only when the modem is activated.

7.7.1 Set command

The set command reads dynamic EPS QoS parameters.

Command syntax:

```
AT+CGEQOSRDP[=<cid>]
```

Response syntax:

```
[+CGEQOSRDP: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],
    [<DL_MBR>,<UL_MBR>][,<DL_AMBR>,
    <UL_AMBR>]]
```

The command parameters and their defined values are the following:

<cid>

Context identifier, 0 - 11. 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_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.

<DL_GBR>, <UL_GBR>, <DL_MBR>, <UL_MBR>

Not supported

Example:

```
AT+CGEQOSRDP
+CGEQOSRDP: 0,0,,
+CGEQOSRDP: 1,2,,
+CGEQOSRDP: 2,4,,,1,65280000
OK
```

7.8 +CGPADDR - Show PDP address(es)

The +CGPADDR command returns a list of Packet Data Protocol (PDP) addresses for the specified context identifiers.

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.

Command syntax:

```
AT+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>

0-11

<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.

Example:

```
AT+CGPADDR=1
+CGPADDR: 1,"10.0.0.130",
    "1050:0000:0000:0000:0005:0600:300c:326b"
OK
```

7.8.2 Test command

The test command returns a list of defined <cid> values.

Response syntax:

```
+CGPADDR: (list of defined <cid>s)
```

Example:

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

7.9 +CGCONTRDP - PDN connection dynamic parameters

The +CGCONTRDP command returns information for an active PDN connection. This command issues a valid response only when the modem is activated.

7.9.1 Set command

The set command returns information for an active PDN connection.

Command syntax:

```
AT+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>

0-11 (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.

Example:

```
AT+CGCONTRDP=0
+CGCONTRDP: 0,,"internet","","","10.0.0.1","10.0.0.2",,,,,
    1028
OK
```

7.10 +CGATT - PS attach or detach

The +CGATT command attaches the MT to or detaches the MT from the Packet Domain services.

7.10.1 Set command

The set command attaches the UE to or detaches the UE from the Packet Domain services. The command is intended for testing purposes only.

Note: The UE performs an attach automatically when activated. In normal operation there is no need to issue the +CGATT command.

Command yntax:

```
+CGATT=<state>
```

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

<state>

0 - Detached

1 - Attached

Example:

```
AT+CGATT=1
OK
```

7.10.2 Read command

The read command reads the state.

Response syntax:

```
+CGATT: <state>
```

Example:

```
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)
```

Example:

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

7.11 +CEPPI - Power preference indication for EPS

The +CEPPI command selects the power saving preference.

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.

Command syntax:

```
AT+CEPPI=<power preference>
```

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

<power preference>

0 - Normal

1 - Low power consumption

Example:

```
AT+CEPPI=1
OK
```

7.11.2 Test command

The test command lists the supported power preferences.

Command syntax:

```
AT+CEPPI=(list of supported <power preference>s)
```

Example:

```
AT+CEPPI=?
+CEPPI: (0,1)
```

7.12 %XPCO - Protocol configuration options notification

The %XPCO command subscribes PCO notifications.

7.12.1 Set command

The set command subscribes PCO notifications.

Command syntax:

```
AT%XPCO=<n>
```

Notification syntax:

```
%XPCO: <id>,<container data>
```

The set command parameters and their 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

<container data>

Content of the container, hexadecimal data encoded with IRA characters. 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
```

Example:

```
%XPCO: 65280,"A1B1C1D1"
```

7.13 %XEPCO - Usage of ePCO/PCO in PDN connection establishment

The %XEPCO command selects the usage of ePCO/PCO in PDN connection establishment

7.13.1 Set command

The set command selects ePCO/PCO usage.

Command syntax:

```
AT%XEPCO=<epco>
```

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

<epco>

0 - Use PCO

1 - Use ePCO

Example:

```
AT%XEPCO=0
OK
```

7.14 %XAPNCLASS - APN class access

The %XAPNCLASS command reads APN class data.

7.14.1 Set command

The set command reads APN class data.

Command syntax:

```
AT%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

Example:

```
AT%XAPNCLASS=0,3
%XAPNCLASS: 3,"VZWAPN","IPV4V6"
OK
```

7.15 %XIPV6FAIL - External IP stack IPv6 address resolution/refresh failure

The %XIPV6FAIL indicates an external IP stack IPv6 address resolution or refresh failure.

7.15.1 Set command

The set command indicates the modem an external IP stack IPv6 address resolution or refresh failure.

Command syntax:

```
AT%XIPV6FAIL=<cid>,<failure_type>
```

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

<cid>

Context identifier

<failure_type>

0 - IPv6 address refresh failure

1 - IPv6 address resolution failure

Example:

```
AT%XIPV6FAIL=0,1
OK
```

7.16 +CGAUTH - Define PDN connection authentication parameters

The +CGAUTH command specifies authentication parameters.

7.16.1 Set command

The set command specifies authentication parameters for a PDN connection specified by parameter <cid>.

Syntax:

```
+CGAUTH=<cid>[,<auth_prot>[,<userid>[,<password>]]]
```

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

<cid>

0-11

<auth_prot>

0 - None. Username and password are removed if they have been specified.

1 - PAP

2 - CHAP

<userid>

String

<password>

String

Example:

```
AT+CGAUTH=1,"PAP","username","password"
OK
```

7.17 +CSCON - Signaling connection status

The +CSCON command controls the presentation of an unsolicited result code.

7.17.1 Set command

The set command controls the presentation of an unsolicited result code.

Command syntax:

```
AT+CSCON=[<n>]
```

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

<n>

0 - Unsolicited indications disabled
1 - Enabled: <mode>
2 - Enabled: <mode>[,<state>]
3 - Enabled: <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

Example - Enable level 3 indications:

```
AT+CSCON=3
OK
```

Example - Level-3-related unsolicited indication:

```
+CSCON: 1,7,4
```

7.17.2 Read command

The 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]]
```

Example - 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
```

Example - Indicate that unsoliticed 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)
```

Example:

```
AT+CSCON=?
+CSCON: (0,1,2,3)
OK
```

8 Network service related commands

8.1 +COPS - PLMN Selection

The +COPS command selects a PLMN automatically or manually, and reads and searches the current mobile network.

8.1.1 Set command

The set command selects a mobile network automatically or manually. The selection is stored in the nonvolatile memory during power-off.

Command syntax:

```
AT+COPS=[<mode>[,<format>[,<oper>]]]
```

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. Mobile Country Code (MCC) and Mobile Network Code (MNC) values. Only numeric string formats supported.

For manual selection, only the numeric string format is supported and <oper> is mandatory.

Example - Select the automatic network selection:

```
AT+COPS=0
OK
```

Example - Manually select network 24407:

```
AT+COPS=1,2,"24407"
OK
```

8.1.2 Read command

The command reads the current mobile network.

Response syntax:

```
+COPS: <mode>[,<format>,<oper>,[AcT>]]
```

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

<AcT>

7 - E-UTRAN

9 - E-UTRAN (NB-S1 mode)

Example - Read the current selection mode and network:

```
AT+COPS?
+COPS: 0,2,"26201",7
OK
```

Example - Read the current selection mode and network with the operator name in the 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 codes

516 - Radio connection is active

521 - PLMN search interrupted, partial results

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.

Example:

```
AT+COPS=?
+COPS: (2,"","","26201",7),(1,"","","26202",7)
OK
```

8.2 %COPS - Forced PLMN search

The %COPS command performs a forced PLMN search.

8.2.1 Test command

The test command searches the PLMN and returns a list of operators found.

The command is similar to +COPS with the exception that %COPS test command is considered a high priority search. This means that e.g. data transfer will be

suspended, pagings lost, and registration is not maintained. In other words, the search will not be 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. MCC and MNC values. Only numeric string formats supported.

<stat>

0 - Unknown

1 - Available

2 - Current

3 - Forbidden

<AcT>

7 - E-UTRAN

9 - E-UTRAN (NB-S1 mode)

Example:

```
AT%COPS=?
%COPS: (2,"","","26201",7),(1,"","","26202",7)
OK
```

8.3 +CPSMS - Power saving mode setting

The +CPSMS command controls PSM settings.

8.3.1 Set command

The command sets the power saving mode. Sets activity timer and PSM period after NAS signaling connection release. Configured values are stored in the non-volatile memory when the device is powered off with +CFUN=0.

Syntax:

```
+CPSMS=[<mode>[,<Requested_Periodic-RAU>,
    <Requested_GPRS-READY-timer>,
    <Requested_Periodic-TAU>[,
    <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 manufacturerspecific 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>

String. One byte in 8-bit format.

Optional. Timer value updated if present. 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.

Note: If the USIM profile in use is a Verizon profile, the minimum value for

<Requested_Periodic-TAU> is 190 minutes.

<Requested_Active-Time>

String. One byte in 8-bit format.

Optional. Timer value updated if present. 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.

Example - Enable 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
```

Example - Disable power saving mode:

```
AT+CPSMS=0
OK
```

Example - Disable power saving mode and sets timer to default values:

```
AT+CPSMS=
OK
```

8.3.2 Read command

The command reads the current PSM settings.

Response syntax:

```
+CPSMS: <mode>,[<Requested_Periodic-RAU>],
    [<Requested_GPRS-READY-timer>],
    [<Requested_Periodic-TAU>],
    [<Requested_Active-Time>]
```

Example:

```
AT+CPSMS?
+CPSMS: 1,,,"10101111","01101100"
OK
```

8.4 +CEDRXS - eDRX setting

The +CEDRXS command controls the setting of eDRX parameters.

8.4.1 Set command

The command sets the requested eDRX parameters.

When a eDRX parameter is changed, the default Paging Time Window (PTW) is set. If other than the default PTW has to be used, the %XPTW command shall be sent after the +CEDRX command. See PTW setting %XPTW.

Command syntax:

```
AT+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>

4 − E-UTRAN (WB-S1 mode)

5 − E-UTRAN (NB-S1 mode)

<Requested_eDRX_value>

String. Half a byte in a 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). Mandatory when enabling eDRX.

<NW-Provided_eDRX_value>

String. Half a byte in a 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).

<Paging_time_window>

String. Half a byte in a 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).

Example - Enable eDRX and sets the requested eDRX value:

```
AT+CEDRXS=1,4,"1000"
OK
```

Example - Unsolicited notification when <mode> 2 is used:

```
+CEDRXP: 4,"1000","0101","1011"
OK
```

8.4.2 Read command

The command is used to read the requested eDRX parameters.

Response syntax:

```
+CEDRXS: <AcT-type>,<Requested_eDRX_value>
```

Example:

```
AT+CEDRXS?
+CEDRXS: 4,"0110"
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)
```

Example:

```
AT+CEDRXS=?
+CEDRXS: (0-3),(4-5),("0000"-"1111")
OK
```

8.5 +CEDRXRDP - Read EDRX dynamic parameters

The +CEDRXRDP command reads dynamic eDRX parameters.

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 command parameters and their defined values are the following:

<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 a 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 a 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).

<Paging_time_window>

String. Half a byte in a 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).

Example:

```
AT+CEDRXRDP
+CEDRXRDP: 4,"0011","0010","1001"
OK
```

8.6 +CNUM - Subscriber number

The +CNUM command returns the subscriber Mobile Station International Subscriber Directory Number (MSISDN).

8.6.1 Set command

The +CNUM 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 command parameters and their defined values are the following:

<numberx>

String type phone number of format specified by <typex>

<typex>

Type of address octet in integer format (see 3GPP TS 24.008 subclause 10.5.4.7)

Example:

```
AT+CNUM
+CNUM: ,"+1234567891234",145
OK
```

8.7 +COPN - Read operator name

The +COPN command reads operator names.

8.7.1 Set command

The set command reads operator names.

Syntax:

```
+COPN
```

Note: The device does not have operator names stored in it.

Example:

```
AT+COPN
OK
```

8.8 +CLCK - Facility lock

The +CLCK command locks, unlocks, or interrogates a facility.

8.8.1 Set command

The set command locks, unlocks, or interrogates a facility.

Command syntax:

```
AT+CLCK=<fac>,<mode>[,<passwd>]
```

The command parameters and their defined values are the following:

<fac>

SC - SIM

<mode>

0 - Unlock

1 - Lock

<passwd>

String. Password for the facility.

Example - Disable PIN query:

```
AT+CLCK="SC",0,"<passwd>"
OK
```

8.8.2 Test command

The test command lists supported facilities.

Response syntax:

```
+CLCK: (list of supported <fac>s)
```

Example:

```
AT+CLCK=?
+CLCK: ("SC")
OK
```

8.9 +CPWD - Change password

The +CPWD command changes the password for the facility lock.

8.9.1 Set command

The set command changes the password for the facility lock.

Command syntax:

```
AT+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" supported.

Example:

```
AT+CPWD="SC","1234","5678"
OK
```

8.9.2 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:

<pwdlength>

Integer. Maximum length of the password

Example:

```
AT+CPWD=?
+CPWD: ("SC",8),("P2",8)
OK
```

8.10 +CEREG - Network registration status

The +CEREG command subscribes unsolicited network status notifications.

8.10.1 Set commands

The set command subscribes or unsubscribes unsolicited network status notifications.

Syntax:

```
+CEREG=<n>
```

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

<n>

0 - Disable unsolicited result codes

1 - Enable unsolicited result codes of the format:

`+CEREG:<stat>`

2 - Enable unsolicited result codes of the format:

`+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 - Enable unsolicited result codes of the format:

`+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 - Enable unsolicited result codes of the format:

```
+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],
   [<Periodic-TAU>]]]]
```

5 - Enable unsolicited result codes of the format:

```
+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],
   [<reject_cause>][,[<ActiveTime>],[<Periodic-TAU>]]]]
```

For the notification syntax parameters, see Read command.

Example - Subscribe to notifications with level 2:

```
AT+CEREG=2
OK
```

Example - Unsolicited notification level 1, trying to attach:

```
+CEREG: 2
```

Example - Unsolicited notification level 2, registered:

```
+CEREG: 1,"002F","0012BEEF",7
```

8.10.2 Read command

The command reads current network registration status.

Response syntax:

```
+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>][,<cause_type>],
    [<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]
```

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

<n>

0 - Disable unsolicited result codes

1 - Enable unsolicited result codes of the format:

`+CEREG:<stat>`

2 - Enable unsolicited result codes of the format:

`+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 - Enable unsolicited result codes of the format:

`+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 - Enable unsolicited result codes of the format:

`+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],`

[<Periodic-TAU>]]]]

5 - Enable unsolicited result codes of the format:

+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<ActiveTime>],[<Periodic-TAU>]]]]

<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 (e.g. out of E-UTRAN coverage).

5 - Registered, roaming.

8 - Attached for emergency bearer services only.

90 - Not registered due to UICC failure.

<tac>

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

<ci>

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

<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. One byte in an 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.

<Periodic-TAU>

String. One byte in an 8-bit format.

Indicates the extended periodic TAU value (T3412) allocated to the device in EUTRAN. 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.

Example:

```
AT+CEREG?
+CEREG: 2,1,"002F","0012BEEF",7
OK
```

8.10.3 Test command

The test command returns a list of supported modes as a compound value.

Response syntax:

```
+CEREG: (supported modes)
```

Example:

```
AT+CEREG=?
+CEREG: (0-5)
OK
```

8.11 %XOPNAME - Subscribe unsolicited operator name indications

The %XOPNAME command subscribes unsolicited operator name notifications.

8.11.1 Set command

The set command subscribes or unsubscribes unsolicited operator name notifications. The notification is sent when EMM information Protocol Data Unit (PDU) with the operator name is received.

Command syntax:

```
AT%XOPNAME=<n>
```

Notification syntax:

```
%XOPNAME: [<full_name>],[<short_name>],[<oper>]
```

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

<n>

0 - Unsubscribe unsolicited operator names

1 - Subscribe unsolicited operator names

<full_name>

A string in hexadecimal format. An 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>

A string in hexadecimal format. An 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>

A string of MCC and MNC values.

Example - Subscribe to notifications:

```
AT%XOPNAME=1
OK
```

Example - An unsolicited notification for a full and a short operator name:

```
%XOPNAME: "88D6B23CAD7FBB41D7B4BCCC2ECFE7","8B56FD15","556776"
```

Example - An unsolicited notification for a short operator name:

```
%XOPNAME: ,"8B56FD15","556776"
```

8.12 %XTIME - Subscribe unsolicited network time notifications

The %XTIME command subscribes unsolicited network time notifications.

8.12.1 Set command

The set command subscribes or unsubscribes unsolicited network time notifications.The notificaton is sent when EMM information PDU with time information is received.

Command syntax:

```
AT%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>

A string in hexadecimal format. A one-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>

A string in hexadecimal format. A seven-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>

A string in hexadecimal format. A one-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.

Example:

```
AT%XTIME=1
OK
```

Example - An unsolicited notification for network time with all parameters:

```
%XTIME: "08","81109251714208","01"
```

Example - An unsolicited notification for network time without local time zone:

```
%XTIME: ,"81109251714208","01"
```

8.13 %XRAI - Set release assistance information

The %XRAI command sets release assistance information.

8.13.1 Set command

The set command sets release assistance information.

Command syntax:

```
AT%XRAI[=<rai>]
```

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

<rai>

Release assistance information sent to the network.

0 - Undefined, default

3 - Control plane one response. For more information, see 3GPP TS 24.301, subclause 9.9.4.25 Release assistance indication.

4 - Control plane no response. For more information, see 3GPP TS 24.301, subclause 9.9.4.25 Release assistance indication.

Note:

- Release assistance information is used in control plane data. The current release supports control plane data only in NB-IoT
- When <rai> is set to 3 or 4, the UE includes release assistance information to the next control plane uplink data transmission until a new value is given. The network is not expecting more uplink data and will release the radio connection. Further uplink data transfer requires additional signaling for establishing a radio connection.

Release assistance information is used in control plane data. The current release supports control plane data only in NB-IoT.

Example - Set release assistance information when the application has one packet to be sent and no response from the network is expected:

```
AT%XRAI=4
OK
```

8.13.2 Read command

The command reads release assistance information.

Response syntax:

```
%XRAI: <rai>
```

Example:

```
AT%XRAI?
%XRAI: 4
OK
```

8.14 %XOPERID - Operator ID

The %XOPERID command identifies the operator USIM.

8.14.1 Set command

The set command returns the operator ID.

Command syntax:

```
AT%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

Example:

```
AT%XOPERID
%XOPERID: 1
OK
```

8.15 %XMONITOR - Read modem parameters

The %XMONITOR command reads a set of modem parameters.

8.15.1 Set command

The set command reads modem parameters.

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>]
```

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 (e.g. 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. Operator name in alphanumeric format.

<short_name>

String. Operator name in alphanumeric format.

<plmn>

String. MCC and MNC values.

<tac>

String. A 2-byte TAC in hexadecimal format.

<AcT>

7 - E-UTRAN

9 - E-UTRAN NB-S1

<band>

Integer. Range 1−68. See 3GPP 36.101. The value is 0 when current band information is not available.

<cell_id>

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

<phys_cell_id>

Integer. Physical cell ID.

<EARFCN>

Integer. E-UTRA Absolute Radio Frequency Channel Number (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

<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

<NW-provided_eDRX_value>

String. Half a byte in a 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see subclause 10.5.5.32 in 3GPP TS 24.008).

<Active-Time>

String. One byte in an 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.

Note: PSM is in use if other than a deactivated value for <Active-Time> is received.

<Periodic-TAU>

String. One byte in an 8-bit format.

Indicates the extended periodic TAU value (T3412_EXT) allocated to the device in EUTRAN. 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.

Example:

```
AT%XMONITOR
%XMONITOR: 1,"EDAV","EDAV","26295","00B7",7,4,"00011B07",
    7,2300,63,39,"","11100000","11100000"
OK
```

8.16 %XNETTIME - Network time support

The %XNETTIME command controls if the time received from the network is used.

8.16.1 Set command

The set command sets the requested network time support.

Network time support is enabled by default. The support setting is saved to the NVM.

Command syntax:

```
AT%XNETTIME=<network_time_support>
```

The command parameter and its defined values are the following:

<network_time_support>

0 - Disable network time support

1 - Enable network time support

Example:

```
AT%XNETTIME=0
OK
```

8.16.2 Read command

The command reads network time support.

Response syntax:

```
%XNETTIME: <network_time_support>
```

The response parameter and its defined values are the following:

<network_time_support>

0 - Disable network time support

1 - Enable network time support

Example:

```
AT%XNETTIME?
%XNETTIME: 0
OK
```

8.17 %XDEEPSEARCH - Support for averaging cell search mode to detect weak cells

The %XDEEPSEARCH command supports averaging cell search mode to detect weak cells.

8.17.1 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 will increase the probability to find weak cells. When the setting is disabled, it stops the possible ongoing deep searches immediately.

Note: Enabling this command reduces battery lifetime.

Command syntax:

```
AT%XDEEPSEARCH=<deep_search>
```

The command parameter and its defined values are the following:

<deep_search>

0 - Disable deep search

1 - Enable deep search

Example:

```
AT%XDEEPSEARCH=1
OK
```

8.17.2 Read command

The command reads the status of deep search.

Response syntax:

```
%XDEEPSEARCH: <deep_search>
```

Example:

```
AT%XDEEPSEARCH?
%XDEEPSEARCH:1
OK
```

8.17.3 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 succesfully performed.

Response syntax:

```
%XDEEPSEARCH
```

Example:

```
AT%XDEEPSEARCH=?
AT%XDEEPSEARCH
OK
```

9 Lightweight M2M (LWM2M) commands

LWM2M commands are used if your carrier requires them and supports them. Please contact your carrier for more information.

9.1 #LWM2M - LWM2M managament

The #LWM2M command sets a LWM2M URL.

9.1.1 Set command

The set command sets a LWM2M URL in the modem.

Command syntax:

```
AT#LWM2M=”<type>”,”<url>”
```

The command parameters and their defined values are the following:

<type>

VZWURI - Set VZWURI

<url>

LWM2M URL.

Example:

```
AT#LWM2M=”VZWURI”,”example.com”
OK
```

9.1.2 Read command

The read command reads the set URL.

Response syntax:

```
#VZWURI: <url>
```

9.1.3 Test command

The test command display parameters for the set command.

Response syntax:

```
#LWM2M="VZWURI","<url>"
```

Example:

```
AT#LWM2M=?
#LWM2M="VZWURI","<url>"
OK
```

10 GPS Commands

10.1 #GPS - Manage GPS

The #GPS command manages the GPS radio.

10.1.1 Set command

The set command starts and stops the GPS radio.

Command syntax:

```
AT#GPS=<action>[,<flags>]
```

Response syntax:

```
#GPS: <action>
```

The command parameters and their defined values are the following:

<action>

0 - Stop GPS radio

1 - Start GPS radio

<flags>

If <action> is 1:

TBD

Example:

```
AT#GPS=0
#GPS: 0
OK
```

10.1.2 Read command

The read command reads the status of the GPS radio.

Response syntax:

```
#GPS: <action>,<flags>
```

Example:

```
AT#GPS?
#GPS: 0
OK
```

10.1.3 Test command

The test command displays parameters for the set command.

Response syntax:

```
#GPS=0
#GPS=1,<flags>
```

Example:

```
AT#GPS=?
#GPS=0
#GPS=1,<flags>
OK
```

11 Mobile termination errors

11.1 +CMEE - Report mobile termination errors

The +CMEE command disables or enables the use of the final result code +CME ERROR.

11.1.1 Set command

The set command disables or enables the use of the final result code +CME ERROR.

Command syntax:

```
AT+CMEE=[<n>]
```

The set command parameters and their 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.

2 - Enable +CME ERROR: <err> result code and use verbose <err> values

<err> values are specified in Section 15. Vendor-specific values listed in the command chapters, the value range starts from 512.

Example:

```
AT+CMEE=1
OK
```

11.1.2 Read command

The read command returns the current setting of <n>.

Response syntax:

```
+CMEE: <n>
```

Example:

```
AT+CMEE?
+CMEE: 1
OK
```

11.1.3 Test command

The test command returns supported values as a compound value.

Response syntax:

```
+CMEE: (list of supported <n>s)
```

Example:

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

11.2 +CNEC - Report network error codes

The +CNEC command activates or deactivates unsolicited reporting of error codes sent by the network.

11.2.1 Set command

The set command activates or deactivates unsolicited reporting of error codes sent by the network.

Command syntax:

```
AT+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 +CNEM_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 - 11. <cid> is present if <error_code> is related to a specific <cid>.

Example:

```
AT+CNEC=16
OK
```

Notification example:

```
+CNEC_EMM: 22
```

11.2.2 Read command

The read command returns the current setting of <n>.

Response syntax:

```
+CNEC: <n>
```

Example:

```
AT+CNEC?
+CNEC: 24
OK
```

11.2.3 Test command

The test command returns the supported values as compound values.

Response syntax:

```
+CNEC: (list of supported <n>s)
```

Example:

```
AT+CNEC?
+CNEC: (0,8,16,24)
OK
```

11.3 +CEER - Extended error report

The +CEER command returns an extended error report.

11.3.1 Set command

The set command returns an extended error report.

Command syntax:

```
AT+CEER
```

Response syntax:

```
+CEER: <report>
```

The command has the following parameter:

<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.

Example - Read the last failure:

```
AT+CEER
+CEER: "SMS 301"
OK
```

12 SMS commands

12.1 +CMGF - Message format

The +CMGF command sets message format.

12.1.1 Set command

The set command selects between PDU and text format.

Note: This command can only be issued by a client registered with +CNMI.

Command syntax:

```
AT+CMGF=[<mode>]
```

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

<mode>

0 - PDU mode, default value

Example:

```
AT+CMFG=0
OK
```

12.1.2 Read command

The read command is used to query the current message format.

Response syntax:

```
+CMGF: <mode>
```

Example:

```
AT+CMGF?
+CMGF: 0
OK
```

12.1.3 Test command

The test command lists the supported message formats.

Response syntax:

```
+CMGF: (list of <mode>s)
```

Example:

```
AT+CMGF=?
+CMGF: (0)
OK
```

12.2 +CNMI - New message indications

The +CNMI command selects how new messages are indicated

12.2.1 Set command

The 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.

Command syntax:

```
AT+CNMI=[<mode>[,<mt>[,<bm>[,<ds>]]]]
```

The set 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. 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.

Example - Register as a client for mobile-terminated SMS and status reports:

```
AT+CNMI=3,2,0,1
OK
```

12.2.2 Read command

The command is used to query how new messages are indicated.

Response syntax:

```
+CNMI: <mode>,<mt>,<bm>,<ds>,<bfr>
```

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

<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...3 is entered

Example:

```
AT+CNMI?
+CNMI: 3,2,0,1,1
OK
```

12.3 +CMGS - Send message, PDU mode

The command sends a message in PDU mode.

12.3.1 Set command

The command sends a message in PDU mode.

Note: Only a client registered with +CNMI is allowed to send messages.

Command syntax:

```
AT+CMGS=<length><CR><pdu><ctrl-Z>
```

Response syntax:

```
+CMGS: <mr>[,<ackpdu>]
```

The command parameters and their defined values are the following:

<length>

Number of octets coded in the transport layer data unit to be given. 1-3 ASCII digits.

<pdu>

Hexadecimal numbers containing two International Reference Alphabet (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.

Example - Send the message "Testing a SMS messaging over LTE" to +358401234567, Service Center Address +448888888:

```
AT+CMGS=42<CR>069144888888F811000C915
    3481032547600000B20D4F29C9E7
    69F4161D0BC3D07B5CBF379F89C7
    69F416F7B590E62D3CB<ctrl-z>
+CMGS: 2
OK
```

12.4 +CMT - Received SMS notification in PDU mode

+CMT notifies of an unsolicited received message in PDU mode. TE is expected to ack received message with AT+CNMA.

The notification is subscribed using the +CNMI command.

Syntax:

```
+CMT: <alpha>,<length><CR><LF><pdu>
```

The notification parameters and their defined values are the following:

<alpha>

TP-Originating-Address in string format.

<length>

Number of hexadecimal octets in <pdu>. 1-3 ASCII digits.

<pdu>

Hexadecimal numbers containing two IRA characters per octet.

Example - Notification of a received message "Testing a sms messaging over lte" from +358401234567, Service Center Address +44 888 8888”:

```
+CMT: "+358401234567",28<CR><LF>069144888888F8D4F29C9E
    769F4161D0BC3D07B5CBF379F89C769F416F7B590E62D3CB
```

12.5 +CDS - Delivery status notification in PDU mode

+CDS notifies of an unsolicited delivery status in PDU mode. TE is expected to ack received delivery report with AT+CNMA.

The notification is subscribed using the +CNMI command.

Syntax:

```
+CDS: <length><CR><LF><pdu>
```

The notification parameters and their defined values are the following:

<length>

Number of hexadecimal octets in <pdu>. 1-3 ASCII digits.

<pdu>

Hexadecimal numbers containing two IRA characters per octet.

Example - A delivery status notification with the recipient address, service center timestamp, and message delivery time:

```
+CDS: 25<CR><LF>060C91534810325476171160316255001711603152120000
OK
```

12.6 +CNMA - New message ACK, PDU mode

The +CNMA command sends an ACK in PDU mode.

Note: Text mode is not supported.

12.6.1 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 only be issued by a client registered with +CNMI.
- 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), the 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.

Command syntax:

```
AT+CNMA[=<n>[,<length>[<CR>PDU is given<ctrl-Z/ESC>]]]
```

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

1 - Send RP-ACK

2 - Send RP-ERROR

<length>

Number of hexadecimal octets in <pdu>. 1-3 ASCII digits.

<pdu>

Hexadecimal numbers containing two IRA characters per octet.

Example - Confirms the reception of a message, timestamp 06/11/2071 13:26:31:

```
AT+CNMA=1,9<CR>01001711603162130 0<ctrl-z>
OK
```

12.6.2 Test command

The test command lists supported <n>s.

Response syntax:

```
+CNMA: (list of supported <n>s)
```

Example:

```
AT+CNMA=?
+CNMA: (0-2)
OK
```

12.7 +CNMA - New message ACK, text mode

The +CNMA command sends a new message ACK in text mode.

Note: Text mode is not supported.

12.7.1 Set command

The set command sends a new message ACK in text mode. This command can be used only when the modem is activated.

This command can only be issued by a client registered with +CNMI.

If the UE does not get an acknowledgement within the required time (network time-out), 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, i.e. the SMS client gets unregistered.

Command syntax:

```
AT+CNMA
```

Example:

```
AT+CNMA
OK
```

12.7.2 Test command

The test command lists supported <n>s.

Response syntax:

```
+CNMA: (list of supported <n>s)
```

Example:

```
AT+CNMA=?
+CNMA: (0-2)
OK
```

12.8 +CPMS - Preferred message storage

The +CPMS command selects the memory storage.

12.8.1 Set command

The command sets the used memory.

Note: The modem does not support SMS memory, only direct routing to TE.

Command syntax:

```
AT+CPMS=<mem1>[,<mem2>[,<mem3>]]
```

Response syntax:

```
+CPMS: <used1>,<total1>,<used2>,<total2>,<used3>,<total3>
```

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>

Example:

```
AT+CPMS="MT","MT","MT"
+CPMS: 0,0,0,0,0,0
OK
```

12.8.2 Read command

The command is used to query memory status.

Response syntax:

```
+CPMS: <mem1>,<used1>,<total1>,<mem2>,<used2>,
    <total2>,<mem3>,<used3>,<total3>
```

Example:

```
AT+CPMS?
+CPMS: "MT",0,0,"MT",0,0,"MT",0,0
OK
```

12.8.3 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)
```

Example:

```
AT+CPMS=?
+CPMS: ("MT"),("MT"),("MT")
OK
```

12.9 +CMS ERROR - Message service failure result code

Message service failure result code +CMS is sent as error response to SMS-related commands.

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

12.10 +CGSMS - Select SMS service

The +CGSMS command selects the SMS service.

12.10.1 Set command

The set command selects the SMS service.

Command syntax:

```
AT+CGSMS=[<service>]
```

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

<service>

1 - Circuit-switched

Note: In a failure case, the command response is ERROR or +CME ERROR.

Example:

```
AT+CGSMS=1
OK
```

12.10.2 Read command

The command reads the current SMS service.

Response syntax:

```
+CGSMS: <service>
```

Example:

```
AT+CGSMS?
+CGSMS: 1
OK
```

12.10.3 Test command

The command lists the supported SMS services.

Response syntax:

```
+CGSMS: (list of currently available <service>s)
```

Example:

```
AT+CGSMS=?
+CGSMS: (1)
OK
```

12.11 %XSMMA - Short message memory available

The %XSMMA command sends an RP-SMMA message.

12.11.1 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.

Command syntax

```
AT%XSMMA
```

Example: Trigger sending the RP-SMMA on the SMS layer to release a memory full situation and to receive a response:

A successful case:

```
AT%XSMMA
OK
```

13 Firmware-Over-The-Air Update Command

13.1 #XFOTA - Firmware Over The Air update

The #XFOTA command triggers a firmware-over-the-air (FOTA) update for the both the cellular module application image. This allows remote updates for devices deployed in the field.

13.1.1 Set command

The set command informs the modem of the location and URL, and starts the FOTA update.

Command syntax:

```
AT#XFOTA=<host>,<file>
```

Response syntax during download:

```
#XFOTA: <message_type>[,<percent>]
```

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

<host>

URL, maximum of 127 characters. This is the URL without the protocol, but the protocol used is HTTP.

For example, is the file is located at `http://example.com`, only use `example.com`.

<url>

File path, maximum of 255 characters.

<message_type>

0 - Update failure

2 - Download progress

3 - Successful FOTA update

<percent>

When <message_type> is 2:

File download percentage

Example:

```
AT#XFOTA="storage.example.com","filepath/v1-0.bin"
OK
#XFOTA: 2,1
#XFOTA: 2,3
#XFOTA: 2,4
...
#XFOTA: 2,99
#XFOTA: 3
+RESET
```

13.1.2 Test command

The test command displays the parameters for the command.

Response syntax:

```
#XFOTA=<host>,<file>
```

Example:

```
AT#XFOTA=?
#XFOTA=<host>,<file>
OK
```

14 GPIO Commands

14.1 #GPIO - GPIO interaction command

The #GPIO command handles reading from, writing to, and configuring GPIO pins.

14.1.1 Set command

The set command handles reading from, writing to, and configuring GPIO pins.

Command syntax:

```
AT#GPIO=(<pin>,<pin_mask>),<action>
    [,(<state>,<state_mask>)|(<dir>,<dir_mask>)]
```

Response syntax:

```
#GPIO: (<state>,<state_mask>)
```

Note: the response syntax is valid only for the read action.

There are a total of 32 GPIO pins available for use in the range [0,31]. Pins can be modified one-at-a-time, or by using a bitmask. When using the bitmask, you must specify all 32 pins at once. The MSB of the bitmask is pin 31, and the LSB is pin 0.

The parameter and the values used by the set command is the following:

<pin>

Which single GPIO pin that an <action> should be taken on.

<pin_mask>

32 character bit mask.

0 - Don't display the pin value at this index

1 - Display the pin value at this index

<action>

0 - Read GPIO pin state

1 - Write to GPIO pin

2 - Configure GPIO pin

<state>

When <action> is 1

0 - Logic level LOW

1 - Logic level HIGH

<state_mask>

32 character bit mask.

0 - Set pin at this index logic level LOW

1 - Set pin at this index logic level HIGH

<dir>

When <action> is 2

0 - Configure pin as output

1 - Configure pin as input

<dir_mask>

32 character bit mask.

0 - Configure pin at this index as output

1 - Configure pin at this index as input

Example - Read pin 2:

```
AT#GPIO=2,0
#GPIO: 0
OK
```

Example - Configure pin 3 as input:

```
AT#GPIO=3,2,1
OK
```

Example - Set pins 29, 30, and 31 as output:

```
AT#GPIO=11100000000000000000000000000000,2,0
OK
```

Example - Set pin 29, 30, and 31 as logic level HIGH

```
AT#GPIO=11100000000000000000000000000000,1,1
OK
```

Example - Set all pins as output. Set pin 29, 30, and 31 as logic level HIGH, and 0, 1, 2, and 3 as logic level LOW. Read pins 29-31, 0-3:

```
AT#GPIO=11111111111111111111111111111111,2,0
OK
AT#GPIO=11100000000000000000000000001111,1,
    11100000000000000000000000000000
OK
AT#GPIO=11100000000000000000000000001111,0
#GPIO: 11100000000000000000000000000000
OK
```

14.1.2 Test command

The test command list the supported states and parameters

Example:

```
AT#GPIO=?
#GPIO=(<pin>|<mask>),0
#GPIO=(<pin>|<mask>),1,(<state>|<mask>)
#GPIO=(<pin>|<mask>),2,(<dir>|<mask>)
OK
```

15 CME Errors

0 - phone failure

1 - no connection to phone

2 - phone-adaptor link reserved

3 - operation not allowed

4 - operation not supported

5 - PH-SIM PIN required

6 - PH-FSIM PIN required

7 - PH-FSIM PUK required

10 - SIM not inserted

11 - SIM PIN required

12 - SIM PUK required

13 - SIM failure

14 - SIM busy

15 - SIM wrong

16 - incorrect password

17 - SIM PIN2 required

18 - SIM PUK2 required

20 - memory full

21 - invalid index

22 - not found

23 - memory failure

24 - text string too long

25 - invalid characters in text string

26 - dial string too long

27 - invalid characters in dial string

30 - no network service

31 - network timeout

32 - network not allowed - emergency calls only

40 - network personalization PIN required

41 - network personalization PUK required

42 - network subset personalization PIN required

43 - network subset personalization PUK required

44 - service provider personalization PIN required

45 - service provider personalization PUK required

46 - corporate personalization PIN required

47 - corporate personalization PUK required

48 - hidden key required

49 - EAP method not supported

50 - Incorrect parameters

51 - command implemented but currently disabled

52 - command aborted by user

53 - not attached to network due to MT functionality restrictions

54 - modem not allowed, MT restricted to emergency calls only

55 - operation not allowed because of MT functionality restrictions

56 - fixed dial number only allowed, called number is not a fixed dial number

57 - temporarily out of service due to other MT usage

58 - language/alphabet not supported

59 - unexpected data value

60 - system failure

61 - data missing

62 - call barred

63 - message waiting indication subscription failure

100 - unknown

102 - IMSI unknown in HSS

103 - Illegal UE

105 - IMEI not accepted

106 - Illegal ME

107 - EPS services not allowed

108 - EPS services and non-EPS services not allowed

109 - UE identity cannot be derived by the network

110 - Implicitly detached

111 - PLMN not allowed

112 - Tracking area not allowed

113 - Roaming not allowed in this tracking area

114 - EPS services not allowed in this PLMN

115 - No suitable cells in tracking area

116 - MSC temporarily not reachable

117 - Network failure

118 - CS domain not available

119 - ESM failure

122 - Congestion

125 - Not authorized for this CSG

126 - Insufficient resources

127 - Missing or unknown APN

128 - Unknown PDN type

129 - User authentication or authorization failed

130 - Activation rejected by Serving GW or PDN GW

131 - Request rejected, unspecified

132 - Service option not supported

133 - Requested service option not subscribed

134 - Service option temporarily out of order

135 - PTI already in use

136 - Regular deactivation

137 - EPS QoS not accepted

139 - CS service temporarily not available

141 - Semantic error in the TFT operation

142 - Syntactical error in the TFT operation

143 - Invalid EPS bearer identity

144 - Semantic errors in packet filter(s)

145 - Syntactical errors in packet filter(s)

171 - Last PDN disconnection not allowed

172 - Semantically incorrect message

173 - Invalid mandatory information

174 - Message type non-existent or not implemented

175 - Conditional IE error

176 - Protocol error, unspecified

177 - Operator determined barring

178 - Maximum number of EPS bearers reached

179 - Requested APN not supported in current RAT and PLMN combination

181 - unsupported QCI value

184 - Invalid PTI value

185 - No EPS bearer context activated

186 - Message not compatible with protocol state

189 - Requested service option not authorized in this PLMN

190 - Network failure

191 - Reactivation requested

192 - PDN type IPv4 only allowed

193 - PDN type IPv6 only allowed

194 - Single address bearers only allowed

195 - Collision with network initiated request

196 - PDN type IPv4v6 only allowed

197 - PDN type non IP only allowed

198 - Bearer handling not supported

199 - APN restriction value incompatible with active PDP context

200 - Multiple accesses to a PDN connection not allowed

201 - ESM information not received

202 - PDN connection does not exist

203 - Multiple PDN connections for a given APN not allowed

204 - Severe network failure

208 - Message type not compatible with protocol state

209 - Information element non-existent or not implemented

16 Acronyms and abbreviations

These acronyms and abbreviations are used in this document.

16-QAM

16-state Quadrature Amplitude Modulation

APN

Access Point Name

BPSK

Binary Phase-Shift Keying

Cat-M1

Cat-NB1

CD

Check Digit

CIDR

Classless Inter-Domain Routing

CS

Circuit-Switched

DER

Distinguished Encoding Rules

DHCP

Dynamic Host Configuration Protocol

DRX

Discontinuous Reception

EARFCN

E-UTRA Absolute Radio Frequency Channel Number

eDRX

Extended Discontinuous Reception

EMM

EPS Mobility Management

EPS

Evolved Packet System

ESN

Electronic Serial Number

E-UTRA

Evolved Universal Terrestrial Radio Access

E-UTRAN

Evolved Terrestrial Radio Access Network

FOTA

Firmware Over The Air

GNSS

Global Navigation Satellite System

GPRS

General Packet Radio Services

IMEI

International Mobile (Station) Equipment Identity

IMEISV

International Mobile (Station) Equipment Identity, Software Version

IMSI

International Mobile Subscriber Identity

IRA

International Reference Alphabet

LNA

Low-Noise Amplifier

MCC

Mobile Country Code

ME

Mobile Equipment

MNC

Mobile Network Code

MSISDN

Mobile Station International Subscriber Directory Number

MT

Mobile Termination

MTU

Maximum Transmission Unit

NAS

Non-access Stratum

NSLPI

NAS Signalling Low Priority Indication

NVM

Non-volatile Memory

PCO

Protocol Configuration Options

PDP

Packet Data Protocol

PDN

Packet Data Network

PDU

Protocol Data Unit

PEM

Privacy Enhanced Mail

PIN

Personal Identification Number

PKCS

Public Key Cryptography Standards

PLMN

Public Land Mobile Network

PS

Packet-Switched

PSK

Pre-shared Key

PSM

Power Saving Mode

PSP

Paging Time Window

PUK

Personal Unblocking Key

QoS

Quality of Service

QPSK

Quadrature Phase-Shift Keying

RAU

Routing Area Update

RB

Resource Block

RP-ACK

Reply Path Acknowledgement

RP-ERROR

Reply Path Error

RSRP

Reference Signal Received Power

SIM

Subscriber Identity Module

SiP

System in Package

SNR

Serial Number

SNR

Signal-to-Noise Ratio

SVN

Software Version Number

TA

Terminal Adapter

TAC

Tracking Area Code
Type Allocation Code

TAU

Tracking Area Update

TE

Terminal Equipment

UART

Universal Asynchronous Receiver-Transmitter

UE

User Equipment

UICC

Universal Integrated Circuit Card

URC

Unsolicited response code

USIM

Universal Subscriber Identity Module

17 References and Acknowledgements

All standard and Nordic-proprietary AT commands' descriptions were taken from or are modified versions of the Nordic NRF AT command manual:

https://infocenter.nordicsemi.com/index.jsp?topic=%2Fref_at_commands%2FREF%2Fat_commands%2Fintro.html

For more information on standard AT commands, see 3GPP 27.005 and 3GPP 27.007.