# nRF91 AT Commands

## Command Reference Guide

v1.3

4418_963 v1.3 / 2020-07-06

NORDIC SEMICONDUCTOR

# Contents

Revision history . . . v

1 Introduction . . . 8

2 AT command syntax . . . 9

2.1 Set command **<CMD>[=...]** . . . 9
2.2 Read command **<CMD>?** . . . 9
2.3 Test command **<CMD>=?** . . . 10
2.4 Response . . . 10

3 Modem firmware versions . . . 11

4 General . . . 12

4.1 Request manufacturer identification +CGMI . . . 12
4.2 Request model identification +CGMM . . . 12
4.3 Request revision identification +CGMR . . . 13
4.4 Request product serial number identification +CGSN . . . 14
4.5 Request IMSI +CIMI . . . 15
4.6 Short software identification %SHORTSWVER . . . 16
4.7 Hardware identification %HWVERSION . . . 16
4.8 Request modem build UUID %XMODEMUUID . . . 17
4.9 Request ICCID %XICCID . . . 18
4.10 Set and read ODIS fields +ODIS . . . 18
4.11 Subscribe unsolicited ODIS notifications +ODISNTF . . . 19

5 Mobile termination control and status commands . . . 21

5.1 Functional mode +CFUN . . . 21
5.2 PIN code +CPIN . . . 23
5.3 Remaining PIN retries +CPINR . . . 24
5.4 List all available AT commands +CLAC . . . 25
5.5 Extended signal quality +CESQ . . . 26
5.6 Signal quality notification %CESQ . . . 28
5.7 Signal quality +CSQ . . . 29
5.8 SNR signal quality notification %XSNRSQ . . . 30
5.9 Restricted SIM access +CRSM . . . 32
5.10 Generic SIM access +CSIM . . . 34
5.11 Device activity status +CPAS . . . 35
5.12 Indicator control +CIND . . . 36
5.13 IP address format +CGPIAF . . . 38
5.14 Current band %XCBAND . . . 39
5.15 Read neighbor cells %NBRGRSRP . . . 40
5.16 Mode of operation (CS/PS) +CEMODE . . . 41
5.17 UICC state %XSIM . . . 42
5.18 Authenticated access %XSUDO . . . 43
5.19 Public key storage management %XPMNG . . . 44
5.20 RF test execution %XRFTEST . . . 45
5.21 Band lock %XBANDLOCK . . . 55
5.22 Data profile %XDATAPRFL . . . 57
5.23 Connectivity statistics %XCONNSTAT . . . 58

5.24 Battery voltage %XVBAT . . . 59
5.25 Battery voltage low level notification %XVBATLVL . . . 60
5.26 Set battery voltage low level %XVBATLOWLVL . . . 61
5.27 External Power off warnings %XPOFWARN . . . 62
5.28 Customer production done %XPRODDONE . . . 63
5.29 Credential storage management %CMNG . . . 64
5.30 Internal temperature %XTEMP . . . 67
5.31 High level for internal temperature %XTEMPHIGHLVL . . . 68
5.32 Clock +CCLK . . . 69
5.33 Proprietary clock %CCLK . . . 70
5.34 Modem trace activation %XMODEMTRACE . . . 71
5.35 Personalization of modem %XUSIMLCK . . . 72
5.36 Fallback to SMS only %XSMSFALLBACK . . . 75
5.37 System mode %XSYSTEMMODE . . . 75
5.38 PTW setting %XPTW . . . 77
5.39 Extra maximum TX power reduction %XEMPR . . . 80
5.40 Write content to file %XFILEWRITE . . . 81
5.41 Coverage enhancement mode information +CEINFO . . . 82

6 SiP pin configuration . . . 85

6.1 COEX0 pin control configuration %XCOEX0 . . . 85
6.2 MAGPIO configuration %XMAGPIO . . . 87
6.3 Antenna detection test %XANTDETMAGPIO . . . 89
6.4 SiP-external MIPI RFFE device introduction %XMIPIRFFEDEV . . . 90
6.5 SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL . . . 92
6.6 Alternative configuration of SiP antenna switch %XANTCFG . . . 95

7 Packet domain commands . . . 97

7.1 Define PDP context +CGDCONT . . . 97
7.2 Packet domain event reporting +CGEREP . . . 99
7.3 Packet domain event unsolicited result codes +CGEV . . . 100
7.4 PDP context activate +CGACT . . . 103
7.5 Allocate new CID %XNEWCID . . . 104
7.6 Map CID to PDN ID %XGETPDNID . . . 105
7.7 QoS dynamic params +CGEQOSRDP . . . 106
7.8 Show PDP address(es) +CGPADDR . . . 107
7.9 PDN connection dynamic parameters +CGCONTRDP . . . 108
7.10 PS attach or detach +CGATT . . . 109
7.11 Power preference indication for EPS +CEPPI . . . 110
7.12 Protocol configuration options notification %XPCO . . . 111
7.13 Usage of ePCO/PCO in PDN connection establishment %XEPCO . . . 112
7.14 APN class access %XAPNCLASS . . . 113
7.15 External IP stack IPv6 address resolution/refresh failure %XIPV6FAIL . . . 114
7.16 Define PDN connection authentication parameters +CGAUTH . . . 114
7.17 Signaling connection status +CSCON . . . 115
7.18 Use of APN %XAPNSTATUS . . . 117

8 Network service related commands . . . 119

8.1 PLMN selection +COPS . . . 119
8.2 Forced PLMN search %COPS . . . 121
8.3 Power saving mode setting +CPSMS . . . 122
8.4 eDRX setting +CEDRXS . . . 126
8.5 Read EDRX dynamic parameters +CEDRXRDP . . . 131

8.6 Subscriber number +CNUM . . . 134
8.7 Read operator name +COPN . . . 135
8.8 Facility lock +CLCK . . . 135
8.9 Change password +CPWD . . . 136
8.10 Network registration status +CEREG . . . 138
8.11 Subscribe unsolicited operator name indications %XOPNAME . . . 140
8.12 Subscribe unsolicited network time notifications %XTIME . . . 142
8.13 Set release assistance information %XRAI . . . 143
8.14 Operator ID %XOPERID . . . 144
8.15 Read modem parameters %XMONITOR . . . 145
8.16 Network time support %XNETTIME . . . 150
8.17 Support for averaging cell search mode to detect weak cells %XDEEPSEARCH . . . 151
8.18 Mobile network operator %XOPCONF . . . 153

9 Mobile termination errors . . . 155

9.1 Report mobile termination errors +CMEE . . . 155
9.2 Report network error codes +CNEC . . . 156
9.3 Extended error report +CEER . . . 158

10 SMS commands . . . 159

10.1 Message format +CMGF . . . 159
10.2 New message indications +CNMI . . . 160
10.3 Send message, PDU mode + CMGS . . . 161
10.4 Received SMS notification in PDU mode +CMT . . . 162
10.5 Delivery status notification in PDU mode +CDS . . . 163
10.6 New message ACK, PDU mode +CNMA . . . 163
10.7 New message ACK, text mode +CNMA . . . 165
10.8 Preferred message storage +CPMS . . . 166
10.9 Message service failure result code +CMS ERROR . . . 168
10.10 Select SMS service +CGSMS . . . 168
10.11 Short message memory available %XSMMA . . . 169

11 Authenticating AT command usage . . . 171

Glossary . . . 172

Acronyms and abbreviations . . . 178

Legal notices . . . 183

# Revision history

| Date | Version | Description |
|---|---|---|
| July 2020 | 1.3 | Updated to match modem firmware v1.1.3<br>• Added the following commands:<br>  • External Power off warnings %XPOFWARN on page 62<br>  • Mobile network operator %XOPCONF on page 153<br>• Updated the following commands:<br>  • Short software identification %SHORTSWVER on page 16<br>  • Hardware identification %HWVERSION on page 16<br>  • Request ICCID %XICCID on page 18<br>  • Functional mode +CFUN on page 21<br>  • UICC state %XSIM on page 42<br>  • RF test execution %XRFTEST on page 45<br>  • Credential storage management %CMNG on page 64<br>  • Internal temperature %XTEMP on page 67<br>  • Write content to file %XFILEWRITE on page 81<br>  • Alternative configuration of SiP antenna switch %XANTCFG on page 95<br>  • Usage of ePCO/PCO in PDN connection establishment %XEPCO on page 112<br>  • Define PDN connection authentication parameters +CGAUTH on page 114<br>  • Power saving mode setting +CPSMS on page 122<br>  • Operator ID %XOPERID on page 144<br>  • Read modem parameters %XMONITOR on page 145<br>• Removed the following commands:<br>  • Sending originating data through control plane +CSODCP<br>  • Stop transmitting exception data %XDATASTOP |
| April 2020 | 1.2 | Updated to match modem firmware v1.2.0<br>• Added the following commands:<br>  • Request modem build UUID %XMODEMUUID on page 17<br>  • Request ICCID %XICCID on page 18<br>  • Set and read ODIS fields +ODIS on page 18<br>  • Subscribe unsolicited ODIS notifications +ODISNTF on page 19<br>  • Battery voltage low level notification %XVBATLVL on page 60<br>  • Set battery voltage low level %XVBATLOWLVL on page 61<br>  • Extra maximum TX power reduction %XEMPR on page 80<br>  • Write content to file %XFILEWRITE on page 81<br>  • Coverage enhancement mode information +CEINFO on page 82 |

| Date | Version | Description |
|---|---|---|
| | | • Alternative configuration of SiP antenna switch %XANTCFG on page 95<br>• Sending originating data through control plane +CSODCP<br>• Stop transmitting exception data %XDATASTOP<br>• Use of APN %XAPNSTATUS on page 117<br>• Updated the following commands:<br>• Request manufacturer identification +CGMI on page 12<br>• Request model identification +CGMM on page 12<br>• Request product serial number identification +CGSN on page 14<br>• Functional mode +CFUN on page 21<br>• PIN code +CPIN on page 23<br>• SNR signal quality notification %XSNRSQ on page 30<br>• UICC state %XSIM on page 42<br>• RF test execution %XRFTEST on page 45<br>• Band lock %XBANDLOCK on page 55<br>• Customer production done %XPRODDONE on page 63<br>• PTW setting %XPTW on page 77<br>• SiP-external MIPI RFFE device introduction %XMIPIRFFEDEV on page 90<br>• SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL on page 92<br>• Packet domain event unsolicited result codes +CGEV on page 100<br>• Protocol configuration options notification %XPCO on page 111<br>• Usage of ePCO/PCO in PDN connection establishment %XEPCO on page 112<br>• Power saving mode setting +CPSMS on page 122<br>• eDRX setting +CEDRXS on page 126<br>• Facility lock +CLCK on page 135<br>• Set release assistance information %XRAI on page 143<br>• Operator ID %XOPERID on page 144<br>• Read modem parameters %XMONITOR on page 145 |
| March 2020 | 1.1.1 | Updated Customer production done %XPRODDONE on page 63 |
| November 2019 | 1.1 | Updated to match modem firmware v1.1.0<br>• Added the following commands:<br>• Short software identification %SHORTSWVER on page 16<br>• Hardware identification %HWVERSION on page 16<br>• Customer production done %XPRODDONE on page 63<br>• Proprietary clock %CCLK on page 70<br>• Personalization of modem %XUSIMLCK on page 72<br>• Fallback to SMS only %XSMSFALLBACK on page 75<br>• Antenna detection test %XANTDETMAGPIO on page 89<br>• Signaling connection status +CSCON on page 115<br>• Forced PLMN search %COPS on page 121 |

| Date | Version | Description |
|---|---|---|
| | | • %XRAI Read command on page 144<br>• Network time support %XNETTIME on page 150<br>• Support for averaging cell search mode to detect weak cells %XDEEPSEARCH on page 151<br>• Updated the following commands:<br>• Functional mode +CFUN on page 21<br>• Extended signal quality +CESQ on page 26<br>• Signal quality notification %CESQ on page 28<br>• Generic SIM access +CSIM on page 34<br>• Authenticated access %XSUDO on page 43<br>• Public key storage management %XPMNG on page 44<br>• Data profile %XDATAPRFL on page 57<br>• Battery voltage %XVBAT on page 59<br>• Clock +CCLK on page 69<br>• Modem trace activation %XMODEMTRACE on page 71<br>• PTW setting %XPTW on page 77<br>• Packet domain event unsolicited result codes +CGEV on page 100<br>• APN class access %XAPNCLASS on page 113<br>• PLMN selection +COPS on page 119<br>• eDRX setting +CEDRXS on page 126<br>• Read modem parameters %XMONITOR on page 145 |
| June 2019 | 1.0 | First release |

# 1 Introduction

This document describes the AT commands used to control the modem in nRF91 Series devices. The nRF91 series AT command API enables modem control for firmware running in the application core on nRF91 series devices.

The AT command API can also be exposed on one of the nRF91 serial interfaces by programming appropriate firmware in the application core. The nRF Connect SDK contains examples of such proxy firmware that can be run stand-alone or as part of other firmware functionality in the nRF91 application core. The stand-alone example is called at_client. This way, an external MCU or computer can get access to the modem API either exclusively or in addition to application firmware running on the nRF91 itself.

The AT commands described in this document apply to all versions of the nRF9160 module hardware. If a command applies only to a specific version of the module hardware, it is mentioned in the command description. The module hardware version is printed on the module label. For more information on nRF9160 module hardware versions, see nRF9160 Compatibility Matrix.

# 2 AT command syntax

The AT Commands have standardized syntax rules.

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

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

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

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

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

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

Set commands set values or perform actions.

Example:

```
AT+CMD=1
```

where

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

## 2.2 Read command `<CMD>?`

Read commands check the current values of subparameters.

Example:

```
AT+CMD?
```

where

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

## 2.3 Test command <CMD>=?

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

Example:

```
AT+CMD=?
```

where

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

## 2.4 Response

AT responds to all commands with a final response.

The response is one of the following:

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

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

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

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

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

```
AT+CGSN
490154203237518
OK
```

# 3 Modem firmware versions

This document describes AT commands used in all versions of the nRF91 Series modem firmware.

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

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

The version tags are read as follows:

- If a command or parameter is marked v1.0.x , it is supported by modem firmware versions where the first two digits are 1 and 0.
- If a command is marked v1.0.x v1.1.x v1.2.x , it is supported by modem firmware versions where the first two digits are 1 and 0, 1 and 1, or 1 and 2.
- If a command or parameter is marked v1.1.x≥3 , it is supported by modem firmware versions where the first two digits are 1 and 1 and the third digit is greater than or equal to 3.
- If a command or parameter is marked v1.1.3 , it is supported only by modem firmware version 1.1.3.

For nRF9160 modem firmware releases, see nRF9160 product page.

# 4 General

The general commands are for the identification of the device.

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

## 4.1 Request manufacturer identification +CGMI

The **+CGMI** command requests manufacturer identification. v1.0.x v1.1.x v1.2.x

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

### 4.1.1 Set command

The set command requests manufacturer identification.

Syntax:

```
+CGMI
```

Response syntax:

```
<manufacturer>
```

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

The following command example reads the manufacturer ID:

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

### 4.1.2 Read command

The read command is not supported.

### 4.1.3 Test command

The test command is not supported.

## 4.2 Request model identification +CGMM

For reference, see *3GPP 27.007 Ch. 5.2*. v1.0.x v1.1.x v1.2.x

### 4.2.1 Set command

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

Syntax:

```
+CGMM
```

Response syntax:

```
<model>
```

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

The following command example reads the model ID:

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

### 4.2.2 Read command

The read command is not supported.

### 4.2.3 Test command

The test command is not supported.

## 4.3 Request revision identification +CGMR

The **+CGMR** command requests modem firmware revision identification. v1.0.x v1.1.x v1.2.x

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

### 4.3.1 Set command

The set command requests revision identification.

Syntax:

```
+CGMR
```

Response syntax:

```
<revision>
```

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

The following command example reads the revision ID:

```
AT+CGMR
mfw_nrf9160_1.1.1
OK
```

### 4.3.2 Read command

The read command is not supported.

### 4.3.3 Test command

The test command is not supported.

# 4.4 Request product serial number identification +CGSN

The **+CGSN** command requests product serial number identification. v1.0.x v1.1.x v1.2.x

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

## 4.4.1 Set command

The set command requests product serial number identification.

Syntax:

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

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

**<snt>**

0 – Respond with <sn> (default)

1 – Respond with `+CGSN: <imei>`

2 – Respond with `+CGSN: <imeisv>`

3 – Respond with `+CGSN: <svn>`

**<sn>**

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

**<imei>**

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>.

The following command example reads the serial number:

```
AT+CGSN
352656100367872
OK
```

The following command example reads the IMEI:

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

### 4.4.2 Read command

The read command is not supported.

### 4.4.3 Test command

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

Response syntax:

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

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

**<snt>**

0 – Respond with <sn> (default)

1 – Respond with `+CGSN: <imei>`

2 – Respond with `+CGSN: <imeisv>`

3 – Respond with `+CGSN: <svn>`

Example:

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

## 4.5 Request IMSI +CIMI

The **`+CIMI`** command reads the *International Mobile Subscriber Identity (IMSI)* from the *Universal Subscriber Identity Module (USIM)* card. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 5.6*.

### 4.5.1 Set command

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

Syntax:

```
+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.

The following command example reads the IMSI string:

```
AT+CIMI
284011234567890
OK
```

### 4.5.2 Read command

The read command is not supported.

### 4.5.3 Test command

The test command is not supported.

## 4.6 Short software identification %SHORTSWVER

The Nordic-proprietary **`%SHORTSWVER`** command requests short software identification. v1.1.x v1.2.x

### 4.6.1 Set command

The set command requests short software identification.

Syntax:

```
%SHORTSWVER
```

Response syntax:

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

The response parameter and its defined value is the following:

**<version_string>**

A string without double quotes

The following command example requests short software identification:

```
AT%SHORTSWVER
%SHORTSWVER: nrf9160_1.1.2
OK
```

### 4.6.2 Read command

The read command is not supported.

### 4.6.3 Test command

The test command is not supported.

## 4.7 Hardware identification %HWVERSION

The Nordic-proprietary **`%HWVERSION`** command requests hardware identification. v1.1.x v1.2.x

### 4.7.1 Set command

The set command requests hardware identification.

Syntax:

```
%HWVERSION
```

Response syntax:

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

The response parameter and its defined value is the following:

**<version_string>**

A string without double quotes

The following command example requests hardware identification:

```
AT%HWVERSION
%HWVERSION: nRF9160 SICA B0A
OK
```

### 4.7.2 Read command

The read command is not supported.

### 4.7.3 Test command

The test command is not supported.

## 4.8 Request modem build UUID %XMODEMUUID

The Nordic-proprietary **%XMODEMUUID** command requests the Universally Unique Identifier (UUID) of a modem build. v1.2.x

### 4.8.1 Set command

The set command requests the UUID of a modem build.

Syntax:

```
%XMODEMUUID
```

Response syntax:

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

The response parameter and its defined value is the following:

**<UUID>**

UUID of the modem build. A string without double quotes.

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

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

### 4.8.2 Read command

The read command is not supported.

### 4.8.3 Test command

The test command is not supported.

## 4.9 Request ICCID %XICCID

The Nordic-proprietary **%XICCID** command reads the *Integrated Circuit Card Identifier (ICCID)* from the *USIM* card. v1.1.x≥3 v1.2.x

### 4.9.1 Set command

The set command reads the *ICCID* from the *USIM* card.

Syntax:

```
%XICCID
```

Response syntax:

```
%XICCID: <ICCID>
```

The response parameter and its defined value is the following:

**<ICCID>**

*ICCID* from the *USIM* card. A string without double quotes.

The following command example requests the *ICCID* of the *USIM* card:

```
AT%XICCID
%XICCID: 8901234567012345678F
OK
```

### 4.9.2 Read command

The read command is not supported.

### 4.9.3 Test command

The test command is not supported.

## 4.10 Set and read ODIS fields +ODIS

The **+ODIS** command sets and reads ODIS fields. v1.2.x

### 4.10.1 Set command

The set command sets ODIS fields.

Syntax:

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

> **Note:** ODIS fields are written to flash when *User Equipment (UE)* is powered down by the `AT +CFUN=0` command.

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

**<Host Device ID>**

String of host device ID in alphanumeric format.

**<Host Device Manufacturer>**

String of host device manufacturer in alphanumeric format.

**<Host Device Model>**

String of host device model in alphanumeric format.

**<Host Device Software Version>**

String of host device software version in alphanumeric format.

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

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

### 4.10.2 Read command

The command reads ODIS fields.

The response includes all values except the host device ID.

Response syntax:

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

The following command example reads the current values:

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

### 4.10.3 Test command

The test command is not supported.

## 4.11 Subscribe unsolicited ODIS notifications +ODISNTF

The **`+ODISNTF`** command subscribes and unsubscribes ODIS notifications. v1.2.x

### 4.11.1 Set command

The set command subscribes and unsubscribes ODIS notifications.

The notifications are triggered by modifications to ODIS fields.

Syntax:

```
+ODISNTF=<Reporting>
```

Notification syntax:

```
+CEINFO: <Host Device ID>, <Host Device Manufacturer>, <Host Device Model>, <Host Device
 Software Version>
```

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

**<reporting>**

0 – Disable unsolicited notifications

1 – Enable unsolicited notifications

The following command example subscribes ODIS notificatons:

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

The following is an example of an unsolicited ODIS notification:

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

### 4.11.2 Read command

The read command is not supported.

### 4.11.3 Test command

The test command is not supported.

# 5 Mobile termination control and status commands

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

## 5.1 Functional mode +CFUN

The **+CFUN** command sets and reads the modem functional mode. v1.0.x v1.1.x v1.2.x

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

### 5.1.1 Set command

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

Syntax:

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

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

**<fun>**

0 – Sets the device to minimum functionality. Disables both transmit and receive RF circuits and deactivates LTE and *Global Navigation Satellite System (GNSS)* services.

1 – Sets the device to full functionality.

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

20 – Deactivates LTE without shutting down *GNSS* services.

21 – Activates LTE without changing *GNSS*.

30 – Deactivates *GNSS* without shutting down LTE services.

31 – Activates *GNSS* without changing LTE.

40 – Deactivates *UICC*. v1.1.x≥3 v1.2.x

41 – Activates *UICC*. v1.1.x≥3 v1.2.x

44 – Sets the device to flight 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.
- `CFUN=41` is allowed only when LTE Cat-M1 or LTE Cat-NB1 is enabled by **%XSYSTEMMODE**.
- *UICC* initialization is started in modes CFUN=1, CFUN=21, and CFUN=41. **%XSIM** indications shall be followed for the *UICC* state.
- The configuration made with an AT command can be stored to *NVM* memory using `AT +CFUN=0` if the option is mentioned in the description of the command. This is useful if the command relates to a permanent hardware configuration, because it avoids the need to give the command every time in the beginning of the application.

The following command example activates the modem Normal mode:

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

## 5.1.2 Read command

The command reads the current functional mode.

Response syntax:

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

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

**<fun>**

0 – Power off and store. RF circuits are disabled by deactivating LTE and *GNSS* services.

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

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

The following command example reads the current functional mode:

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

## 5.1.3 Test command

The test command lists supported functional modes.

Response syntax:

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

The response parameters and their defined values are the following:

**<fun>**

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

1 – Sets the device to full functionality.

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

20 – Deactivates LTE without shutting down *GNSS* services.

21 – Activates LTE without changing *GNSS*.

30 – Deactivates *GNSS* without shutting down LTE services.

31 – Activates *GNSS* without changing LTE.

40 – Deactivates *UICC*. v1.1.x≥3 v1.2.x

41 – Activates *UICC*. v1.1.x≥3 v1.2.x

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

The following command example returns the supported functional modes.

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

# 5.2 PIN code +CPIN

The **+CPIN** command enters and checks the required *Personal Identification Number (PIN)*. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 8.3*.

## 5.2.1 Set command

The set command enters the *PIN*.

Syntax:

```
+CPIN=<pin>[,<newpin>]
```

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

**<pin>**

String of digits.

**<newpin>**

String of digits. Mandatory if the required code is *SIM Personal Unblocking Key (PUK)* or SIM PUK2.

**Note:** If no PIN is required, the response code is `ERROR`.

The following command example will enter PIN 1234.

```
AT+CPIN="1234"
OK
```

### 5.2.2 Read command

The read command checks if a *PIN* is needed or if a personalization lock is blocking the device start-up.

Response syntax:

```
+CPIN: <code>
```

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

**<code>**

READY – No PIN required

SIM PIN – PIN code required

SIM PUK – PUK code required

SIM PIN2 – PIN2 code required

SIM PUK2 – PUK2 code required

PH-SIM PIN – USIM depersonalization required v1.2.x

PH-NET PIN – Network depersonalization required v1.2.x

PH-NETSUB PIN – Network subset depersonalization required v1.2.x

PH-SP PIN – Service provider depersonalization required v1.2.x

PH-CORP PIN – Corporate depersonalization required v1.2.x

The following command example shows how to check if a PIN code is needed with the response that a PIN code is required:

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

**Note:** Use **`AT%XUSIMLCK`** when facility lock depersonalization is required.

### 5.2.3 Test command

The test command is not supported.

## 5.3 Remaining PIN retries +CPINR

The **`+CPINR`** command returns the number of remaining *PIN* retries for the *UE* passwords. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 8.65*.

### 5.3.1 Set command

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

Command syntax:

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

Response syntax for standard PINs:

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

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.

The following command example checks the remaining entries for PIN:

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

### 5.3.2 Read command

The read command is not supported.

### 5.3.3 Test command

The test command is not supported.

## 5.4 List all available AT commands +CLAC

The **+CLAC** command returns a list of all available AT commands. v1.0.x v1.1.x v1.2.x

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

### 5.4.1 Set command

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

Syntax:

```
+CLAC
```

Response syntax:

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

The following command example lists the supported AT commands:

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

### 5.4.2 Read command

The read command is not supported.

### 5.4.3 Test command

The test command is not supported.

## 5.5 Extended signal quality +CESQ

The **+CESQ** command returns received signal quality parameters. This command issues a valid response only when the modem is activated. v1.0.x v1.1.x v1.2.x

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

### 5.5.1 Set command

The set command returns received signal quality parameters.

Syntax:

```
+CESQ
```

Response syntax:

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

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

**<rxlev>**

99 – Not known or not detectable

**<ber>**

99 – Not known or not detectable

**<rscp>**

255 – Not known or not detectable

**<ecno>**

255 – Not known or not detectable

**<rsrq>**

0 rsrq < −19.5 dB

1 – When −19.5 dB ≤ RSRQ < −19 dB

2 – When −19 dB ≤ RSRQ < −18.5 dB

...

32 – When −4 dB ≤ RSRQ < −3.5 dB

33 – When −3.5 dB ≤ RSRQ < −3 dB

34 – When −3 dB ≤ RSRQ

255 – Not known or not detectable

**<rsrp>**

0 – RSRP < −140 dBm

1 – When −140 dBm ≤ RSRP < −139 dBm

2 – When −139 dBm ≤ RSRP < −138 dBm

...

95 – When −46 dBm ≤ RSRP < −45 dBm

96 – When −45 dBm ≤ RSRP < −44 dBm

97 – When −44 dBm ≤ RSRP

255 – Not known or not detectable

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

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

## 5.5.2 Read command

The read command is not supported.

## 5.5.3 Test command

The test command returns supported values as compound values.

Response syntax:

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

The following command example returns supported values as compound values.

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

# 5.6 Signal quality notification %CESQ

The Nordic-proprietary **%CESQ** command subscribes or unsubscribes notifications of changes in signal quality. v1.0.x v1.1.x v1.2.x

## 5.6.1 Set command

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

Syntax:

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

Notification syntax:

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

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

**<n>**

0 – Unsubscribe signal quality notifications

1 – Subscribe signal quality notifications

**<rsrp>**

0 – RSRP < −140 dBm

1 – When −140 dBm ≤ RSRP < −139 dBm

2 – When −139 dBm ≤ RSRP < −138 dBm

...

95 – When −46 dBm ≤ RSRP < −45 dBm

96 – When −45 dBm ≤ RSRP < −44 dBm

97 – When −44 dBm ≤ RSRP

255 – Not known or not detectable

**<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 – RSPR 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 RPSQ 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.

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

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

The example 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
```

### 5.6.2 Read command

The read command is not supported.

### 5.6.3 Test command

The test command is not supported.

## 5.7 Signal quality +CSQ

The **+CSQ** command reads 2G and 3G signal quality. v1.0.x v1.1.x v1.2.x

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

### 5.7.1 Set command

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

Syntax:

```
+CSQ
```

Response syntax:

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

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

The set command parameters and their value are the following:

**<rssi>, <ber>**

99 – Not detectable

The following command example reads signal quality:

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

### 5.7.2 Read command

The read command is not supported.

### 5.7.3 Test command

The test command lists supported signal quality values

Response syntax:

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

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

**<rssi>, <ber>**

99 – Not detectable

The following command example lists the supported signal quality values:

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

## 5.8 SNR signal quality notification %XSNRSQ

The Nordic-proprietary **%XSNRSQ** command subscribes notifications of changes in *Signal-to-Noise Ratio (SNR)* signal quality. v1.0.x v1.1.x v1.2.x

### 5.8.1 Set command

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

Syntax:

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

Notification syntax:

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

The parameters and their defined values are the following:

**<n>**

0 – Unsubscribe SNR signal quality notifications

1 – Subscribe SNR signal quality notifications

**<snr>**

0 – SNR < −24 dB

1 – When −24 dB ≤ SNR < −23 dB

2 – When −23 dB ≤ SNR < −22 dB

...

47 – When 22 dB ≤ SNR < 23 dB

48 – When 23 dB ≤ SNR < 24 dB

49 – When 24 dB ≤ SNR

**<threshold_index>**

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

0 – SNR is below the first threshold.

1 – SNR is between the first and second threshold.

2 – SNR is between the second and third threshold.

3 – SNR is between the third and fourth threshold.

4 – SNR is above the fourth threshold.

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

**<srxlev>**

0 – SRXLEV -127 or below -127

1...254 – SRXLEV -126...126

255 – SRXLEV 127 or above

32767 – Invalid or not know

**<ce_level>**

0 – CE Level 0

1 – CE Level 1

255 – Invalid or not known

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

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

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

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

### 5.8.2 Read command

The read command reads *SNR* signal quality.

Response syntax:

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

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

**<snr>**

0 – SNR < −24 dB

1 – When −24 dB ≤ SNR < −23 dB

2 – When −23 ≤ SNR < −22 dB

...

47 – When 22 ≤ SNR < 23 dB

48 – When 23 ≤ SNR < 24 dB

49– When 24 ≤ SNR

**<srxlev>**

0 – SRXLEV -127 or below -127

1...254 – SRXLEV -126...126

255 – SRXLEV 127 or above

32767 – Invalid or not know

**<ce_level>**

0 – CE Level 0

1 – CE Level 1

255 – Invalid or not known

The following command example reads SNR signal quality:

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

### 5.8.3 Test command

The test command is not supported.

## 5.9 Restricted SIM access +CRSM

The **+CRSM** command transmits restricted commands to *SIM*. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 8.18*.

## 5.9.1 Set command

The set command transmits restricted commands to the *SIM*.

Syntax:

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

Response syntax:

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

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.

The following command example reads the forbidden *Public Land Mobile Network (PLMN)* list:

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

### 5.9.2 Read command

The read command is not supported.

### 5.9.3 Test command

The test command is not supported.

## 5.10 Generic SIM access +CSIM

The **+CSIM** command transmits a command to the *SIM*. v1.0.x v1.1.x v1.2.x

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

To avoid conflicts with modem firmware, **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.

### 5.10.1 Set command

The set command transmits a command to the *SIM*.

Syntax:

```
+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.*

The following command example performs a **MANAGE CHANNEL** command to open a logical channel. The SIM card returns channel number '01' and success status '9000':

```
AT+CSIM=10,"0070000001"
+CSIM: 6,"019000"
OK
```

### 5.10.2 Read command

The read command is not supported.

### 5.10.3 Test command

The test command is not supported.

## 5.11 Device activity status +CPAS

The **+CPAS** command returns the device activity status. v1.0.x v1.1.x v1.2.x

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

### 5.11.1 Set command

The set command returns the device activity status.

Syntax:

```
+CPAS
```

Response syntax:

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

The command has the following parameter:

**<pas>**

Activity status.

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

The following command example checks the activity status:

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

### 5.11.2 Read command

The read command is not supported.

### 5.11.3 Test command

The test command is not supported.

## 5.12 Indicator control +CIND

The **+CIND** command sets indicator states. v1.0.x v1.1.x v1.2.x

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

### 5.12.1 Set command

The command sets indicator states.

Syntax:

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

Response syntax:

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

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

**<ind>**

Integer. 0 – Off.

Other values are <descr>-specific.

"service": 1 – On

"roam": 1 - On

"message": 1 - On

**<descr>**

"service" – Service availability

"roam" – Roaming indicator

"message" – Message received

**<value>**

Integer. Values are <descr>-specific.

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

"roam": 0 - Not roaming, 1 - Roaming

"message": 1 - Message received

The example enables service and message indicators:

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

The example notification indicates that the device is in service:

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

## 5.12.2 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
```

## 5.12.3 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
```

# 5.13 IP address format +CGPIAF

The **+CGPIAF** command returns information about IPv6 address format. v1.0.x v1.1.x v1.2.x

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

## 5.13.1 Set command

The set command is not supported.

## 5.13.2 Read command

The read command returns the IPv6 address format.

Response syntax:

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

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

**<IPv6_AddressFormat>**

1 – Use IPv6-like colon notation

**<IPv6_SubnetNotation>**

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

**<IPv6_LeadingZeros>**

1 – Leading zeros are included

**<IPv6_CompressZeros>**

0 – No zero compression

The following command example reads the current IPv6 address format:

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

## 5.13.3 Test command

The test command returns the supported IPv6 address formats.

Response syntax:

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

The 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

The following command example reads the current IPv6 address format:

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

# 5.14 Current band %XCBAND

The Nordic-proprietary **%XCBAND** command returns the current E-UTRA band. v1.0.x v1.1.x v1.2.x

## 5.14.1 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>
```

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

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

The following command example reads the current band:

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

### 5.14.2 Read command

The read command is not supported.

### 5.14.3 Test command

The test command returns a list of supported bands.

Response syntax:

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

The following command example returns a list of supported bands:

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

## 5.15 Read neighbor cells %NBRGRSRP

The Nordic-proprietary **%NBRGRSRP** command reads measured *RSRP* values of neighboring cells. The command issues a valid response only when the modem is activated. v1.0.x v1.1.x v1.2.x

### 5.15.1 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.

Syntax:

```
%NBRGRSRP
```

Response syntax:

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

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

**<phys_cellID>**

Integer. Physical cell ID.

**<EARFCN>**

Integer. 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

### 5.15.2 Read command

The read command is not supported.

### 5.15.3 Test command

The test command is not supported.

## 5.16 Mode of operation (CS/PS) +CEMODE

The **+CEMODE** command sets the device mode of operation. v1.0.x v1.1.x v1.2.x

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

### 5.16.1 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.

Syntax:

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

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

**<mode>**

0 – PS mode 2 of operation

2 – CS/PS mode 2 of operation

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

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

### 5.16.2 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

The following command example reads the current operating mode:

```
+CEMODE: 0
OK
```

### 5.16.3 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
```

## 5.17 UICC state %XSIM

The Nordic-proprietary **%XSIM** command subscribes *UICC* state notifications. v1.0.x v1.1.x v1.2.x

### 5.17.1 Set command

The set command subscribes *UICC* state notifications.

Syntax:

```
%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

The following command example subscribes *UICC* state notifications:

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

The following example notification indicates that *UICC* is not initialized:

```
%XSIM: 0
```

The following example notification indicates that *UICC* initialization is completed:

```
%XSIM: 1
```

### 5.17.2 Read command

The command reads the *UICC* state.

Response syntax:

```
%XSIM: <state>[,<cause>]
```

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

**<state>**

0 – UICC not initialized

1 – UICC initialization OK

The following command example reads the *UICC* state. The response indicates that *UICC* initialization is completed:

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

The following command example reads the *UICC* state. The response indicates that a *PIN* code is required:

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

### 5.17.3 Test command

The test command is not supported.

## 5.18 Authenticated access %XSUDO

The Nordic-proprietary **%XSUDO** command provides authenticated access for a restricted AT command.

v1.0.x v1.1.x v1.2.x

**Note:** This command is for future releases. In the current software release, the use of this command is not required.

For information on the usage of the command, see Authenticating AT command usage on page 171.

### 5.18.1 Set command

The set command provides authenticated access for a restricted AT command.

The restricted command is separated with a semicolon (;). The leading AT prefix is not included in the concatenated command.

Syntax:

```
%XSUDO=<data_len>,<signature>[,<sec_tag>]
```

**+CME ERROR codes**

513 – Not found, public key not found

520 – Authentication failed

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

**<data_len>**

Length of a signed command string.

Only the number of characters in <data_len> from an authenticated command is processed, the rest are ignored. <data_len> shall not be greater than the given command.

**<signature>**

Command signature in Base64 format

**<sec_tag>**

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

The following command example provides authenticated access for the restricted **+CMD** command:

```
AT%XSUDO=28,"c2lnbmF0dXJl";+CMD=...
OK
```

### 5.18.2 Read command

The read command is not supported.

### 5.18.3 Test command

The test command is not supported.

## 5.19 Public key storage management %XPMNG

The Nordic-proprietary **%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. v1.0.x v1.1.x v1.2.x

### 5.19.1 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.

The following command example writes the public key:

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

The following command example reads the public key:

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

### 5.19.2 Read command

The read command is not supported.

### 5.19.3 Test command

The test command is not supported.

## 5.20 RF test execution %XRFTEST

The Nordic-proprietary **%XRFTEST** command performs RF testing. v1.0.x v1.1.x v1.2.x

> **Note:** The use of this command can be permanently prevented with the **AT%XPRODDONE** command.

## 5.20.1 Set command

The set command performs RF testing.

Syntax:

```
%XRFTEST=<test>,<operation>,<param0>,<param1>,…,<param9>
```

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

**<test>**

0 – RX

1 – TX

2 – GPS SNR

3 – RX SNR

4 – AFC v1.2.x

10 – RF sensors v1.2.x

**<operation>**

0 – OFF

1 – ON

3 – SET v1.2.x

**<paramX>**

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

### 5.20.1.1 RX testing

The command enables RF receiver with the given parameters. It also measures signal power at the *SiP* antenna port 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. Use 24 for *Global Positioning System (GPS)* reception.

**<param1>**

Frequency 100 kHz.

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

**<param2>**

RX signal power at the *SiP* antenna port 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 the *SiP* antenna port in q8 dBm. q8 means that dividing the result by $2^8$ = 256 gives dBm.

The following command example enables the RF receiver for Band 1, 2140.0 MHz, −65 dBm, 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
OK
```

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

## 5.20.1.2 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 the following 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 the *SiP* antenna port [dBm].

Valid range from +23 to −50.

**<param3>**

System mode.

Valid range 0–1. NB1 (0) or M1 (1).

> **Note:** Some LTE bands may be supported in only one of the system modes. To ensure that the band can be used, set the correct mode with **`%XSYSTEMMODE`**.

**<param4>**

Modulation.

0 – QPSK

1 – 16QAM

2 – Reserved

3 – BPSK

4 – CW

M1: QPSK, 16QAM, and CW

NB1: QPSK, BPSK, and CW

> **Note:**
>
> - If `<param4> = CW`, in system modes NB1 and M1 the CW tone is offset by 255 kHz from the frequency given in <param1>. v1.0.x v1.1.x v1.2.x

**<param5>, <param6>, <param7>**

<param5> *Resource Block (RB)*/Tone count

<param6> *RB*/Tone start position

<param7> Subcarrier spacing 0 – 15 k, 1 – 3.75 k.

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

**<param8>** v1.2.x

System bandwidth.

0 – NB1

1 – 1.4 M

2 – reserved

3 – 5 MHz

4 - 6 – reserved

**<param9>** v1.2.x

Narrowband index.

0 – 1.4 M

0 - 3 – 5 M

Defined in *3GPP TS 36.211 Ch. 6.2.7*.

> **Note:** <param9> has effect only when <param8> is 3 (5 MHz).

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

| 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 |

*Table 1: Allowed parameter combinations for <param5>, <param6>, and <param7> in system modes NB1 and M1*

> **Note:** In system mode 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 the *SiP* antenna port 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, system bandwidth NB1, NB index 0:

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

**Note:** 271/16 = 16.9 dBm

The following command example enables the RF transmitter for Band 5, 830 MHz, +17 dBm, M1, 16-QAM, 6 *RB*, *RB* start position 0, subcarrier spacing 15 kHz, system bandwidth 5 MHz, NB index 3:

```
%XRFTEST=1,1,5,8300,17,1,1,6,0,0,3,3
%XRFTEST: 271
OK
```

### TX OFF

**<operation>**

0 – OFF

The following command example disables the RF transmitter:

```
%XRFTEST=1,0
OK
```

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

## 5.20.1.3 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 parameters and their defined values are the following:

**<test>**

2 – GPS SNR

**<operation>**

1 – ON

**Note:** "OFF" is not needed due to automatic stop.

GPS SNR ON has three parameters:

**<param0>**

RX signal power at the *SiP GPS* port in dBm.

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

**<param1>** v1.2.x

AFC correction.

1 – Enable AFC correction in measurement

0 – Disable AFC correction in measurement

**<param2>** v1.2.x

Frequency. Manually give RF frequency for nRF9160 in the 100 kHz format. Accepted value is +/- 10 MHz from the default *GPS* frequency of 1575.42 MHz. Set the signal generator to f = <param2> + 330 kHz.

> **Note:** If <param2> is not given, nRF9160 uses the default *GPS* L1 frequency 1575.42 MHz.

> **Note:** It is recommended to set the signal level defined in the GPS SNR ON parameters to the expected signal level at the *GPS* antenna port of the nRF9160 *SiP*.

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 signal power at the *SiP GPS* port in q8 dBm. q8 means that dividing the result by $2^8$ = 256 gives dBm.

The following command example executes a *GPS SNR* test, AFC disabled:

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

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

The following command example executes an *GPS SNR* test for -80 dBm. AFC correction is enabled and uses the result of the previous AFC measurement or AFC set. Instead of the default, the receiver frequency is set to 1575.0 MHz:

```
%XRFTEST=2,1,-80,1,15750
%XRFTEST: 312,-20480
OK
```

> **Note:** 312/16 = 19.5 dB and -20480/256 = -80 dBm.

### 5.20.1.4 RX SNR testing

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:** "OFF" is not needed due to automatic stop.

**<param0>**

3GPP band number. Use 24 for *GPS* reception.

**<param1>**

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

**<param2>**

RX signal power at the *SiP* antenna port in dBm.

Valid range from −127 to −25.

**<param3>**

System mode.

Valid range 0–1. NB1 (0) or M1 (1).

**<param4>** v1.2.x

Enabling AFC correction during *SNR* test.

1 – Enable

0 – Disable

If <param4> is not given, AFC correction is disabled.

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 signal power at the *SiP* antenna port 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, M1 mode, AFC correction enabled:

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

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

### 5.20.1.5 AFC measurement and set

The command measures or sets the AFC error value for a given frequency. The default error value is 0.
v1.2.x

Some frequency error may occur because the signal generator's and nRF9160's reference clocks are different. This test allows to measure or set frequency error and use the result value as compensation in subsequent tests. This improves the correctness of *SNR* tests and transmission frequency, but has no significant effect on the power measurement results in RX ON or TX ON tests. The relative Q30 formatted response value is stored in device memory and can be used in RX and *GPS SNR* tests if the AFC correction enabling parameter, for example <param4> in the RX *SNR* test, is 1.

The measurement expects the *CW* in the signal generator to be 120 kHz lower than the given frequency. For example, if DUT frequency is 830 MHz, the signal generator must be set to 829.880 MHz.

> **Note:** The same result value is used automatically in TX ON and RX ON tests.

The command parameters and their defined values are the following:

**<test>**

4 – AFC

#### AFC measurement

**<operation>**

1 – ON (measure)

> **Note:** "OFF" is not needed due to automatic stop. Mode is always M1. The stored value is reset when the device is booted.

AFC measurement has three parameters:

**<param0>**

3GPP band number

**<param1>**

Frequency (100 kHz)

**<param2>**

RX signal power at the *SiP* antenna port in dBm

Response syntax when <operation> is ON:

```
%XRFTEST: <AFC_error_value_Hz>
```

The response parameters and their values are the following:

**<AFC_error_value_Hz>**

Measured AFC error value in Hz format

The following command example measures the AFC error value for band 1, 2140.0 MHz, -40 dBm:

```
%XRFTEST=4,1,1,21400,-40
%XRFTEST:349
 OK
```

### AFC set

The command allows to manually set the AFC error value in Hz, which can be used in RX and *GPS SNR* tests if the AFC correction enabling parameter, for example <param4> in the RX SNR test, is 1. The value set with this command overwrites the value obtained with AFC measurement.

This value is used automatically in the RX ON and TX ON tests. To not use the AFC error value, for example in TX ON or RX ON tests, set it to 0.

The command parameter and its value are the following:

**<operation>**

3 – SET

AFC SET has the following parameter:

**<param0>**

AFC error value (Hz), if <param1> = 0 or not given.

Response syntax when <operation> is SET:

```
%XRFTEST: OK
```

The following command example sets AFC error value to 349 Hz:

```
%XRFTEST=4,3,349
OK
```

**Note:** In the AFC correction feature, if the set error value is given in Hz format, the relative signed Q30 value is automatically calculated when used in the next test for a given frequency. The relative error is used automatically in subsequent tests. The suitable error value can be calculated for any frequency with the following formula: error_in_hz_f2 = error_in_hz_f1 * f2 / f1. For example, if error is measured as 120 Hz at 2140 MHz, the error at 1950 MHz is 120 Hz * 1950 MHz/2140 MHz = 109 Hz.

## 5.20.1.6 RF sensor testing

The command allows to read the temperature sensor next to the *Power Amplifier (PA)* inside the *SiP* and another temperature sensor inside the nRF9120 *System on Chip (SoC)*. v1.2.x

When the modem is active (either LTE communication or *GPS* receiver), the command with `<param0>=0` returns the latest *PA* temperature sensor value measured automatically during modem wakeup or transmission. The temperature is not measured during LTE or *GPS* reception, because *PA* is not in use. During modem inactivity, the modem measures the *PA* temperature value when the command is received.

The command parameters and their defined values are the following:

**<test>**

10 – Sensor test

**<operation>**

1 – ON (measure)

**<param0>**

0 – *PA* temperature

1 – Reserved

2 – Reserved

3 – VBAT (same as **%XVBAT**)

4 – Reserved

Response syntax when <param0> = 0:

```
%XRFTEST: <temperature>
OK
```

When <param0> = 0, the response is given in millidegree Celsius, with a resolution of approximately 50 millidegree Celsius.

Response syntax when <param0> = 3:

```
%XRFTEST: <vbat>
OK
```

When <param0> = 3, the response is given in millivolts.

The following command example measures the *PA* temperature which is approximately 20.4 Celsius:

```
AT%XRFTEST=10,1,0
%XRFTEST: 20355
OK
```

### 5.20.2 Read command

The read command is not supported.

### 5.20.3 Test command

The test command is not supported.

## 5.21 Band lock %XBANDLOCK

The Nordic-proprietary **%XBANDLOCK** command sets locked bands. The band lock should be set before activating modem with **+CFUN**. v1.0.x v1.1.x v1.2.x

### 5.21.1 Set command

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

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

**Note:**

- Set band lock before activating modem with **+CFUN**.
- Permanent mask is written to flash when UE is powered down.

Syntax:

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

**+CME error code**

518 – Not allowed in active state

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

**<operation>**

0 – Remove band locks

1 – Set permanent band mask

2 – Set runtime band mask

**<band_mask><permanent_lock><run-time_lock>**

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

The following command example sets permanent band 4 lock:

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

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

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

## 5.21.2 Read command

The command reads locked bands.

Response syntax:

```
%XBANDLOCK: <permanent_lock>,<run-time_lock>
```

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

**<permanent_lock>,<run-time_lock>**

Bit string, 88 bits.

An empty string is returned if bandlock is not set.

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

```
AT%XBANDLOCK?
%XBANDLOCK:
 "","0000000000000000000000000000000000000000000000000000000000000000000000000000001000000001001"
OK
```

### 5.21.3 Test command

The test command is not supported.

## 5.22 Data profile %XDATAPRFL

The Nordic-proprietary **%XDATAPRFL** command can be used to provide information on the application use case to modem so that it can optimize power consumption. v1.0.x v1.1.x v1.2.x

> **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.

### 5.22.1 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.

Syntax:

```
%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

The following command example sets a low power level:

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

### 5.22.2 Read command

The read command reads the application data profile.

Syntax:

```
%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

The following command example reads the power level:

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

### 5.22.3 Test command

The test command is not supported.

## 5.23 Connectivity statistics %XCONNSTAT

The Nordic-proprietary **%XCONNSTAT** command sets the connectivity statistics command. v1.0.x v1.1.x v1.2.x

### 5.23.1 Set command

The set command sets the connectivity statistics command.

Syntax:

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

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

**<command>**

0 – Stop

1 – Start

The following command example makes the LwM2M application start and stop connectivity statistics:

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

### 5.23.2 Read command

The read command reads the connectivity statistics.

Syntax:

```
%XCONNSTAT: <SMS Tx>,<SMS Rx>,<Data Tx>,<Data Rx>,<Packet max>,<Packet average>
```

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

**<SMS Tx>**

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

**<SMS Rx>**

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

**<Data Tx>**

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

**<Data Rx>**

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

**<Packet max>**

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

**<Packet average>**

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

The following command example makes the LwM2M application read the connectivity statistics:

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

### 5.23.3 Test command

The test command is not supported.

## 5.24 Battery voltage %XVBAT

The Nordic-proprietary **%XVBAT** command reads battery voltage. v1.0.x v1.1.x v1.2.x

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.

### 5.24.1 Set command

The set command reads the battery voltage in mV.

Syntax:

```
%XVBAT
```

Response syntax:

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

The response parameter is the following:

**<vbat>**

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

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

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

### 5.24.2 Read command

The read command is not supported.

### 5.24.3 Test command

The test command is not supported.

## 5.25 Battery voltage low level notification %XVBATLVL

The **%XVBATLVL** command subscribes unsolicited battery voltage low level notifications. v1.1.x v1.2.x

The notification is sent when the battery voltage level is under the currently set low level.

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

### 5.25.1 Set command

The command subscribes unsolicited notifications of battery voltage low level.

Syntax:

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

Notification syntax:

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

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

**<n>**

0 – Unsubscribe unsolicited notifications of battery voltage low level

1 – Subscribe unsolicited notifications of battery voltage low level

The notification parameter and its defined values are the following:

**<battery_voltage>**

Integer. Millivolts between 3000 and 5000.

0 – No valid battery voltage available

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

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

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

```
%XVBATLOWLVL: 3700
```

### 5.25.2 Read command

The read command is not supported.

### 5.25.3 Test command

The test command is not supported.

## 5.26 Set battery voltage low level %XVBATLOWLVL

The **%XVBATLOWLVL** command sets the battery voltage low level for a modem. If notifications of battery voltage low level have been subscribed, the modem sends clients a notification when the measured battery voltage is below the defined level. v1.1.x v1.2.x

The notifications are subscribed using the %XVBATLVL command.

### 5.26.1 Set command

The command sets the battery voltage low level for a modem.

Syntax:

```
%XVBATLOWLVL=<battery level>
```

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

**<battery level>**

Integer. Millivolts between 3100 and 5000. Factory default 3300.

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

```
Set current battery voltage low level
AT%XVBATLOWLVL=3500
OK
```

### 5.26.2 Read command

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

Response syntax:

```
%XVBATLOWLVL?
```

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

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

### 5.26.3 Test command

The test command is not supported.

## 5.27 External Power off warnings %XPOFWARN

The Nordic-proprietary **%XPOFWARN** command controls the Power off warnings from the modem. The command is supported by nRF9160-SIxA-B1. v1.1.x≥3

### 5.27.1 Set command

The set command configures the Power off warnings from the modem. The warning is sent as an interrupt from the modem to the application. When a Power off warning is sent, the modem sets itself to Offline mode and sends `+CGEV: ME BATTERY LOW`.

Syntax:

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

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

**<state>**

0 – Disable power off warnings

1 – Enable power off warnings

**<voltage>**

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

28 – 2800 mV

29 – 2900 mV

30 – 3000 mV

31 – 3100 mV

32 – 3200 mV

33 – 3300 mV

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

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

### 5.27.2 Read command

The read command is not supported.

### 5.27.3 Test command

The test command is not supported.

# 5.28 Customer production done %XPRODDONE

The Nordic-proprietary **%XPRODDONE** command shall be sent after customer production is done. v1.0.x v1.1.x v1.2.x

## 5.28.1 Set command

The set command disables R&D features by closing the modem *Universal Asynchronous Receiver/ Transmitter (UART)* connection and optionally permanently disables the use of the **%XRFTEST** and **%XEMPR** commands. The command also permanently enables the downgrade prevention feature in modem firmware. Downgrade prevention cannot be disabled once it has been enabled.

Downgrade prevention means that it is not possible to flash an older firmware version to modem. Downgrade prevention applies to both FOTA and cable flash.

If the <value> parameter is not set or is 0, the **%XRFTEST** and **%XEMPR** commands are permanently disabled after the **%XPRODDONE** command has been performed. To keep the commands usable after **%XPRODDONE**, use `AT%XPRODDONE=1`. If commands have been disabled with **%XPRODDONE**, they cannot be re-enabled. If **%XRFTEST** is not disabled when issuing **%XPRODDONE**, it remains permanently enabled and cannot be disabled later.

Syntax:

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

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

**<value>**

0 – Permanently disable **%XRFTEST** and **%XEMPR**

1 – Leave **%XRFTEST** and **%XEMPR** usable

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

```
AT%XPRODDONE
OK
```

or

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

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

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

**CAUTION:** In modem firmware versions 1.0.x (x ≥ 2) and 1.1.0, downgrade prevention is enabled after the first modem flashing after issuing **%XPRODDONE**. In modem firmware versions 1.1.x (x ≥ 1) and 1.2.x, downgrade prevention becomes active immediately after the command is issued.

## 5.28.2 Read command

The read command is not supported.

### 5.28.3 Test command

The test command is not supported.

## 5.29 Credential storage management %CMNG

The Nordic-proprietary **%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. v1.0.x v1.1.x v1.2.x

### 5.29.1 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.

Syntax:

```
%CMNG=<opcode>[,<sec_tag>[,<type>[,<content>[,<passwd>]]]]
```

Response syntax for read operation:

```
%CMNG: <sec_tag>,<type>[,<sha>[,<content>]]
```

Response syntax for list operation:

```
%CMNG: <sec_tag>,<type>[,<sha>]
```

<sec_tag> <type> shall be a unique pair, no multiple items with the same <sec_tag> and <type> values are allowed.

**+CME ERROR 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 set command parameters and their defined values are the following:

**<opcode>**

0 – Write

1 – List

2 – Read

3 – Delete

**<sec_tag>**

Integer, 0 – 2147483647.

Mandatory for write, read, and delete operations. Optional for list operation.

**<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). Hexadecimal data in ASCII string format containing two *International Reference Alphabet (IRA)* characters per octet (*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.

The following command example writes the root certificate:

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

The following command example writes the client certificate:

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

The following command example writes the private key:

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

The following command example lists a single item by specifying tag and type:

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

The following command example lists a single tag:

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

The following command example lists all stored credentials:

```
AT%CMNG=1
%CMNG: 12345678, 0, "978C...02C4"
%CMNG: 567890, 1, "C485...CF09"
%CMNG: 123, 2, "92E1...8AC8"
%CMNG: 654321, 3, "E0C9...511D"
OK
```

The following command example reads the root certificate with tag 12345678:

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

The following command example deletes a client certificate with tag 123:

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

The following command example reads a non-existing root certificate with tag 4567. Error code 513 is returned:

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

## 5.29.2 Read command

The read command is not supported.

### 5.29.3 Test command

The test command is not supported.

## 5.30 Internal temperature %XTEMP

The Nordic proprietary **%XTEMP** command subscribes unsolicited internal temperature notifications.
v1.0.x v1.1.x v1.2.x

### 5.30.1 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.

Syntax:

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

Notification syntax:

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

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

**<n>**

0 – Unsubscribe unsolicited temperature indications

1 – Subscribe unsolicited temperature indications

The notification parameters and their defined values are the following:

**<temperature_level>**

1 – Normal temperature

2 – High temperature. Factory default 55. This can be changed with High level for internal temperature %XTEMPHIGHLVL on page 68.

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

**<temperature>**

Integer. Celsius degrees between −40 and 125.

The following command example subscribes notifications:

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

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

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

### 5.30.2 Read command

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

Syntax:

```
%XTEMP?
```

Response syntax:

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

The following command example reads the current modem temperature:

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

### 5.30.3 Test command

The test command is not supported.

## 5.31 High level for internal temperature %XTEMPHIGHLVL

The Nordic proprietary **%XTEMPHIGHLVL** command sets the high level to internal temperature in the modem. v1.0.x v1.1.x v1.2.x

### 5.31.1 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.

Syntax:

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

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

**<temperature>**

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

The following command example sets the high temperature level:

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

### 5.31.2 Read command

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

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

Syntax:

```
%XTEMPHIGHLVL?
```

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

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

### 5.31.3 Test command

The test command is not supported.

## 5.32 Clock +CCLK

The **+CCLK** command sets the clock of the device. v1.0.x v1.1.x v1.2.x

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

### 5.32.1 Set command

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

Syntax:

```
+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).

The following command example sets the real-time clock:

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

### 5.32.2 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).

The following command example reads the real-time clock:

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

### 5.32.3 Test command

The test command is not supported.

## 5.33 Proprietary clock %CCLK

The **%CCLK** command sets the real-time clock of the device. v1.1.x v1.2.x

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

### 5.33.1 Set command

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

Syntax:

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

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

**<time>**

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

The following command example sets the real-time clock:

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

### 5.33.2 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

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

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

### 5.33.3 Test command

The test command is not supported.

## 5.34 Modem trace activation %XMODEMTRACE

The Nordic-proprietary **%XMODEMTRACE** command activates modem traces. The trace data is in binary format and can help the Nordic customer support to analyze and resolve issues. v1.0.x v1.1.x v1.2.x

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

### 5.34.1 Set command

The set command activates and deactivates modem trace.

Syntax:

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

Response syntax for Read trace bitmap:

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

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

**<oper>**

Operation

0 – Deactivate traces

1 – Activate predefined trace set

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

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

**<set_id>**

Integer, predefined trace set identifier

1 – Coredump only

2 – 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.

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

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

The following command example deactivates trace:

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

### 5.34.2 Read command

The read command is not supported.

### 5.34.3 Test command

The test command is not supported.

## 5.35 Personalization of modem %XUSIMLCK

The Nordic-proprietary **%XUSIMLCK** command allows personalizing the modem to work with predefined USIM cards. v1.1.x v1.2.x

### 5.35.1 Set command

The set command allows locking the modem to work with predefined USIM cards. Using the command, the modem can be personalized, depersonalized, or the lock of a category can be disabled if the category is not depersonalized.

It is also possible to configure USIM personalization so that the device is locked to the first USIM that is inserted to it.

According to *3GPP TS 22.022*, the following personalization options are available:

- Network
- Network subset
- Service provider

- Corporate
- USIM

Syntax:

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

The modem supports a maximum of 24 personalization codes.

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

**<command>**

1 – Personalize

2 – Depersonalize

3 – Disable

4 – Lock device to the first inserted USIM. The value of <facility> must be PS.

**<facility>**

String:

PN – Network personalization

PU – Network subset personalization

PP – Service provider personalization

PC – Corporate personalization

PS – USIM personalization

**<pwd>**

String. A password for enabling or disabling personalization. Used for <command> values 1, 2, or 4. The length of the password is 6–16 digits.

If PN Network Control Key, (NCK)

If PU Network Subset Control Key, (NSCK)

If PP Service Provider Control Key, (SPCK)

If PC Corporate Control Key, (CCK)

If PS Personalization Control Key, (PCK)

**<permanent>**

Programmable selection of the Control Key. Used only when the value of <command> is 1. The permanent Control Key can be programmed once, and it is therefore immutable once programmed.

0 – Nonpermanent Control Key

1 – Permanent Control Key

**<pers_data>**

String. Used only when the value of <command> is 1.

When <facility> is PN, <pers_data> can contain a maximum of 24 pairs of MCC and MNC in the following format: `MCC1.MNC1:MCC2.MNC2:…:MCCn.MNCn`.

When <facility> is PU, <pers_data> can contain a maximum of 24 pairs of MCC +MNC+Network Subset Code (digits 6 and 7 of IMSI) in the following format: `MCC1.MNC1.D61.D71:MCC2.MNC2.D62.D72:...:MCCn.MNCn.D6n.D7n`, where D6x and D7x represent the sixth and seventh digits of IMSI.

When <facility> is PP, <pers_data> can contain a maximum of 24 USIM group identifiers for service provider personalization in the following format: `MCC1.MNC1.GID11:MCC2.MNC2.GID12:...:MCCn.MNCn.GID1n`. GID1x represents the first byte of EF_GID1 in USIM, see *3GPP TS 31.102* chapter 4.2.10 $EF_{GID1}$.

When <facility> is PC, <pers_data> can contain a maximum of 24 pairs of USIM group identifiers from $EF_{GID1}$ and 4.2.11 $EF_{GID2}$ for corporate personalization in the following format: `MCC1.MNC1.GID11.GID21:MCC2.MNC2.GID12.GID22:...:MCCn.MNCn.GID1n.GID2n`.

GID1x and GID2x represent the first bytes of EF_GID1 and EF_GID2, see *3GPP TS 31.102* chapters 4.2.10 $EF_{GID1}$ and 4.2.11 $EF_{GID2}$.

When <facility> is PS, <pers_data> can contain a maximum of 24 IMSIs as specified in *3GPP TS 31.102* chapter 4.2.2 $EF_{IMSI}$. Fifteen *IMSI* digits can be given. The format is the following: `IMSI1:IMSI2:...:IMSIn`.

.

The following command example creates a nonpermanent network personalization:

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

The following command depersonalizes the network personalization:

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

This command disables network personalization:

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

This command locks device to the first inserted USIM in a nonpermanent manner:

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

This command personalizes USIM to IMSI 100200777777777 (MCC=100, MNC=200, other digits are 7). The facility PS is permanently locked to password "12345678". After depersonalization, no other keys can be used for this facility:

```
AT%XUSIMLCK=1,"PS","12345678",1,"100200777777777"
OK
```

### 5.35.2 Read command

The read command is not supported.

### 5.35.3 Test command

The test command is not supported.

## 5.36 Fallback to SMS only %XSMSFALLBACK

The Nordic-proprietary **%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. v1.0.x v1.1.x v1.2.x

### 5.36.1 Set command

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

Syntax:

```
%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

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

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

### 5.36.2 Read command

The read command is not supported.

### 5.36.3 Test command

The test command is not supported.

## 5.37 System mode %XSYSTEMMODE

The Nordic-proprietary **%XSYSTEMMODE** command sets the modem system mode. v1.0.x v1.1.x v1.2.x

### 5.37.1 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.

Syntax:

```
%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 – *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

The following command example sets LTE Cat-M1 and GNSS as the system modes. No preferred LTE mode set:

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

## 5.37.2 Read command

The read command reads the supported modem system modes.

Response syntax:

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

The read response 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 – *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

The following command example reads the supported system mode:

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

### 5.37.3 Test command

The test command is not supported.

## 5.38 PTW setting %XPTW

The Nordic-proprietary **%XPTW** command sets the *Paging Time Window (PTW)*. v1.0.x v1.1.x v1.2.x

### 5.38.1 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 on page 126.

Syntax:

```
%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.

LTE Cat M1 mode

Bit

4 3 2 1 – Paging Time Window length

0 0 0 0 – 1.28 seconds

0 0 0 1 – 2.56 seconds

0 0 1 0 – 3.84 seconds

0 0 1 1 – 5.12 seconds

0 1 0 0 – 6.4 seconds

0 1 0 1 – 7.68 seconds

0 1 1 0 – 8.96 seconds

0 1 1 1 – 10.24 seconds

1 0 0 0 – 11.52 seconds

1 0 0 1 – 12.8 seconds

1 0 1 0 – 14.08 seconds

1 0 1 1 – 15.36 seconds

1 1 0 0 – 16.64 seconds

1 1 0 1 – 17.92 seconds

1 1 1 0 – 19.20 seconds

1 1 1 1 – 20.48 seconds

LTE Cat NB1 mode

Bit

4 3 2 1 – Paging Time Window length

0 0 0 0 – 2.56 seconds

0 0 0 1 – 5.12 seconds

0 0 1 0 – 7.68 seconds

0 0 1 1 – 10.24 seconds

0 1 0 0 – 12.8 seconds

0 1 0 1 – 15.36 seconds

0 1 1 0 – 17.92 seconds

0 1 1 1 – 20.48 seconds

1 0 0 0 – 23.04 seconds

1 0 0 1 – 25.6 seconds

1 0 1 0 – 28.16 seconds

1 0 1 1 – 30.72 seconds

1 1 0 0 – 33.28 seconds

1 1 0 1 – 35.84 seconds

1 1 1 0 – 38.4 seconds

1 1 1 1 – 40.96 seconds

The following command example sets the requested PTW value:

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

## 5.38.2 Read command

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

Response syntax:

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

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

**<AcT-type>**

4 – E-UTRAN (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

**<Requested_ptw_value>**

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

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

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

**Note:**

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

## 5.38.3 Test command

The test command is not supported.

# 5.39 Extra maximum TX power reduction %XEMPR

The Nordic-proprietary **%XEMPR** command allows to configure an extra reduction of 0.5 or 1 dB to the maximum transmission power on all or selected supported 3GPP bands separately in the NB1 and M1 modes. v1.2.x

> **Note:** The use of this command can be permanently prevented with the **AT%XPRODDONE** command.

## 5.39.1 Set command

The set command sets the extra maximum TX power reduction.

The command can be given separately to the NB1 and M1 modes. If a band is not mentioned in the command, the EMPR is zero for that band. The command cannot be used to increase transmission power. **%XEMPR** should be given before the activation of the modem to be effective. It can also be stored to nRF9160's memory using `AT+CFUN=0`. If a valid EMPR configuration exists, the reduction is automatically applied to the applicable transmissions.

Syntax:

```
AT%XEMPR=<nb1_or_m1_mode>,<k>,<band0>,<pr0>,<band1>,<pr1>,…,<bandk-1>,<prk-1>
```

or

```
AT%XEMPR=<nb1_or_m1_mode>, 0, <pr_for_all_bands>
```

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

**<nb1_or_m1_mode>**

0 – NB1

1 – M1

**<k>**

The number of bands to which EMPR is set. If <k> is 0, the next parameter <pr_for_all_bands> is applied to all supported 3GPP bands. The **%XEMPR** command supports listing the power reduction for all the supported bands of nRF9160 in one command for both NB1 and M1.

**<bandn>**

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

**<prn>**

EMPR for <bandn>

0 – 0 dB

1 – Maximum power reduced 0.5 dB

2 – Maximum power reduced 1.0 dB

> 2 is not allowed

The following command example reduces the maximum TX power on all bands in the NB1 mode by 1 dB:

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

The following command example reduces the maximum TX power on three (`<k>=3`) bands in the M1 mode (`<nb1_or_m1_mode>=1`). The maximum TX power is reduced by 1 dB on bands 5 and 8 and by 0.5 dB on band 13:

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

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

```
AT%XEMPR
OK
```

### 5.39.2 Read command

The read command reads the currently active configuration.

Syntax:

```
AT%XEMPR?
```

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

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

### 5.39.3 Test command

The test command is not supported.

## 5.40 Write content to file %XFILEWRITE

The Nordic-proprietary **`%XFILEWRITE`** command writes given content to a file. v1.2.x

### 5.40.1 Set command

The set command requests to write content to a file.

Syntax:

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

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

**<file>**

1 – *GNSS* almanac

**<content>**

Hexadecimal numbers containing two *IRA* characters per octet.

**<checksum>**

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

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

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

### 5.40.2 Read command

The read command is not supported.

### 5.40.3 Test command

The test command is not supported.

## 5.41 Coverage enhancement mode information +CEINFO

The **+CEINFO** command subcribes Coverage Enhancement (CE) notifications. v1.2.x

### 5.41.1 Set command

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

Syntax:

```
+CEINFO=<Reporting>
```

Notification syntax:

```
+CEINFO: <Reporting>,<CE Enabled>,<UE State>,<Downlink Repetition Factor>,<Uplink
 Repetition Factor>,<RSRP>,<CINR>
```

<UE State> is not between double quotation marks.

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

**<reporting>**

0 – Disable unsolicited notifications

1 – Enable unsolicited notifications

**<CE Enabled>**

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

1 – Serving cell supports CE mode A/B

**<UE State>**

*UE* state at the time of the report.

I – Idle

R – RACH

C – Connected

**<Downlink Repetition Factor>**

Downlink repetition factor.

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

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

**<Uplink Repetition Factor>**

Uplink repetition factor.

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

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

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

**<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

**<CINR>**

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

The following command example subscribes unsolicited CE notifications:

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

The following is an example of an unsolicited CE notification:

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

## 5.41.2 Read command

The command requests Coverage Enhancement (CE) mode information.

Response syntax:

```
+CEINFO: <Reporting>,<CE Enabled>,<UE State>,<Downlink Repetition Factor>,<Uplink
 Repetition Factor>,<RSRP>,<CINR>
```

The following command example reads CE mode information:

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

## 5.41.3 Test command

The test command is not supported.

# 6 SiP pin configuration

SiP pin configuration commands can be used to configure the behavior of selected pins of the nRF91 *SiP*. The pins that can currently be configured are **COEX0**, **MAGPIO[0:2]**, and **MIPI RFFE**.

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 COEX0 pin control configuration %XCOEX0

The Nordic-proprietary **%XCOEX0** command writes the COEX0 pin configuration to device's RAM memory.
v1.0.x v1.1.x v1.2.x

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.

Syntax:

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

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

**<count>**

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

**<state_x>**

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

**<freqlo_x>**

Low limit for the frequency range in MHz.

**<freqhi_x>**

High limit for the frequency range in MHz.

The following command example sets COEX0 to '1' when 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
```

This command example sets 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>
```

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

**<count>**

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

**<state_x>**

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

**<freqlo_x>**

Low limit for the frequency range in MHz.

**<freqhi_x>**

High limit for the frequency range in MHz.

The following command example returns the stored configuration:

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

## 6.1.3 Test command

The test command is not supported.

# 6.2 MAGPIO configuration %XMAGPIO

The Nordic-proprietary **`%XMAGPIO`** command writes the MAGPIO configuration to device's RAM memory.
v1.0.x v1.1.x v1.2.x

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.

This command has been updated in v0.7.1 of this document. The earlier format described in v0.7 of this document is still valid, but the new format is recommended.

Syntax:

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

A command without any parameters deletes the previously written values.

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

**<gpio_x>**

0 – MAGPIO_x is not used

1 – MAGPIO_x used

**<num_ranges>**

The number of frequency ranges, maximum value 12

**<state_y>**

Settings of the MAGPIO pins for the range x that follows

**<flo_y>**

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

**<fhi_y>**

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

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

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

*Table 2: Example configuration for an antenna tuner*

The following command example writes 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
```

This command example writes three ranges to device's RAM:

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

The following command example deletes the previously written values:

```
AT%XMAGPIO
OK
```

## 6.2.2 Read command

The command returns the stored MAGPIO configuration.

Response syntax:

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

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

**<gpio_x>**

0 – MAGPIO_x is not used

1 – MAGPIO_x used

**<num_ranges>**

The number of frequency ranges, maximum value 12

**<state_y>**

Settings of the MAGPIO pins for the range x that follows

**<flo_y>**

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

**<fhi_y>**

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

The following command example returns the stored configuration:

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

### 6.2.3 Test command

The test command is not supported.

## 6.3 Antenna detection test %XANTDETMAGPIO

The Nordic-proprietary **%XANTDETMAGPIO** command reads the MAGPIO pin status to detect if the antenna is connected. The antenna is detected when the pin is DC-grounded. v1.1.x v1.2.x

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 Set command

The set command is not supported.

### 6.3.2 Read command

The read command is not supported.

### 6.3.3 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.

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).

The following command example sets 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
```

# 6.4 SiP-external MIPI RFFE device introduction %XMIPIRFFEDEV

nRF91 can be configured to control a SiP-external, *MIPI RF Front-End Control Interface (RFFE)*-controlled[3] device to a limited extent. Antenna tuner is the primary use case. v1.0.x v1.1.x v1.2.x

The **`XMIPIRFFEDEV`** command introduces the device and its static parameters to nRF91. After introducing the *MIPI RFFE* device, the configuration for the various control phases can be given using the **`XMIPIRFFECTRL`** command (see SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL on page 92).

The **`XMIPIRFFEDEV`** command needs to be sent before any modem activity occurs. The configuration is stored to *NVM* when the device is powered off using `+CFUN=0`.

The stored configuration is applied when any modem/GPS RF activity occurs.

This section and SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL on page 92 provide the generic syntax of the *MIPI RFFE* set commands and examples of the read and delete commands. For examples using the **`%XMIPIRFFEDEV`** and **`%XMIPIRFFECTRL`** set commands, see nWP034 - nRF9160 Hardware Verification Guidelines.

## 6.4.1 Set command

The set command writes the XMIPIRFFEDEV configuration to nRF91 RAM memory.

Syntax:

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

The set command parameters and their descriptions are the following:

**<dev_id>**

Selectable identification number for the device. Non-zero. Valid range 1–255. The given dev_id is used with the **`%XMIPIRFFECTRL`** AT command. (See SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL on page 92.)

**<def_usid>**

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

**<prod_id>**

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

[3] MIPI RFFE[SM], MIPI RF Front-End Control Interface (RFFE)

**<man_id>**

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

**<pm_trig>**

An 8-bit content for PM_TRIG (address 0x1C = 28 dec) register. This is for setting the default power and triggering mode. Note that the setting of PM_TRIG can be also changed in the ON phase. See SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL on page 92.

All numbers should be given as decimals, i.e. not as hexadecimals. Currently, nRF91 supports only one *SiP*-external *MIPI RFFE*-controlled device.

## 6.4.2 Read command

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

Response syntax:

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

The read response parameters and their descriptions for the "%XMIPIRFFEDEV" row are as defined in Set command on page 90 if a valid **XMIPIRFFEDEV** command has been given earlier. Otherwise, the row is empty. The phase-specific rows that follow (INIT, ON, OFF, PWROFF) contain the parameters given for that phase or they are empty.

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

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

In the following command example, the read command returns:

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

## 6.4.3 Delete configuration

A *MIPI RFFE* device configuration and control phase information can be deleted from the nRF91 memory using this command.

Syntax:

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

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

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

**CAUTION:** The combined load of *Printed Circuit Board (PCB)* routing, the input load of the *MIPI RFFE*-controlled device, and any parasitic load from application shall not exceed 15 pF at SCLK or at SDATA pins. This load translates roughly to narrow transmission line length of less than 10 cm at the application board but it is dependent on the actual *PCB* design. A load higher than 15 pF at SCLK or SDATA pin will increase the risk of unwanted behavior of the nRF91 *SiP* itself and of *MIPI RFFE* control.

## 6.5 SiP-external MIPI RFFE device control configuration %XMIPIRFFECTRL

After the *MIPI RFFE*-controlled device has been introduced using **%XMIPIRFFEDEV**, its configuration in each control phase needs to be given using **XMIPIRFFECTRL**. v1.0.x v1.1.x v1.2.x

For information on **%XMIPIRFFEDEV**, see SiP-external MIPI RFFE device introduction %XMIPIRFFEDEV on page 90.

*MIPI RFFE* devices contain an internal register map described in the datasheet of the device. To control the device, the registers in the device must be written with appropriate values. This AT command allows to configure the nRF91 SiP to write the device's registers. The register addresses, the values, and timing (phase) can be configured as described below.

The external *MIPI RFFE* control in nRF9160 supports configuring the RFFE device for four different phases of RF operation. The phases are initializing (INIT), start receiving or transmitting (ON), stop receiving or transmitting (OFF), and going to sleep (PWROFF).

The **XMIPIRFFECTRL** command is to be sent separately for each phase. It is not mandatory to configure all phases.

The **%XMIPIRFFECTRL** command needs to be sent before any modem activity occurs. The configuration is stored to *NVM* using `+CFUN=0` when the device is powered off.

The phases are defined as follows:

**INIT**

Applied when RF is waking up. INIT is frequency-agnostic. Controls up to four *MIPI RFFE* device registers. The main purpose is to allow preparation or activation of the *MIPI RFFE* device if activation requires long settling.

**ON**

Applied when RF is starting for a specific frequency or when LTE M1 frequency hopping is performed by the modem RF. Controls a maximum of two frequency-agnostic registers that can be used for device activation, for instance. This phase also controls a maximum of two registers whose value can be defined to depend on the RF frequency of the modem. The table for the frequency-dependent control can have a maximum of 64 frequencies.

**OFF**

Applied when RF is stopping. The configuration is frequency-agnostic. Controls up to four *MIPI RFFE* device registers.

**PWROFF**

Applied when RF is going to sleep. The configuration is frequency-agnostic. Controls up to four *MIPI RFFE* device registers. The main purpose is to deactivate the *MIPI RFFE* device.

For detailed examples of using the commands in different phases, see nWP034 - nRF9160 Hardware Verification Guidelines.

## 6.5.1 Set command

The set command writes the XMIPIRFFECTRL configuration to the nRF91 RAM memory.

The command is given separately for each phase. It is not necessary to send the command for each phase, which means it is possible to configure only one phase.

Generic syntax:

```
AT%XMIPIRFFECTRL=<dev_id>,<phase#>,<variable_number_of_phase_specific_parameters>
```

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

**<dev_id>**

The identification number of the *MIPI RFFE* device given when it was introduced using **`MIPIRFFEDEV`** (see SiP-external MIPI RFFE device introduction %XMIPIRFFEDEV on page 90).

**<phase#>**

Number of the phase INIT = 0, ON = 1, OFF = 2, PWR_OFF = 3. All numbers must be given as decimals (hexadecimals are not allowed).

The following figure illustrates the RFFE device control in different phases:

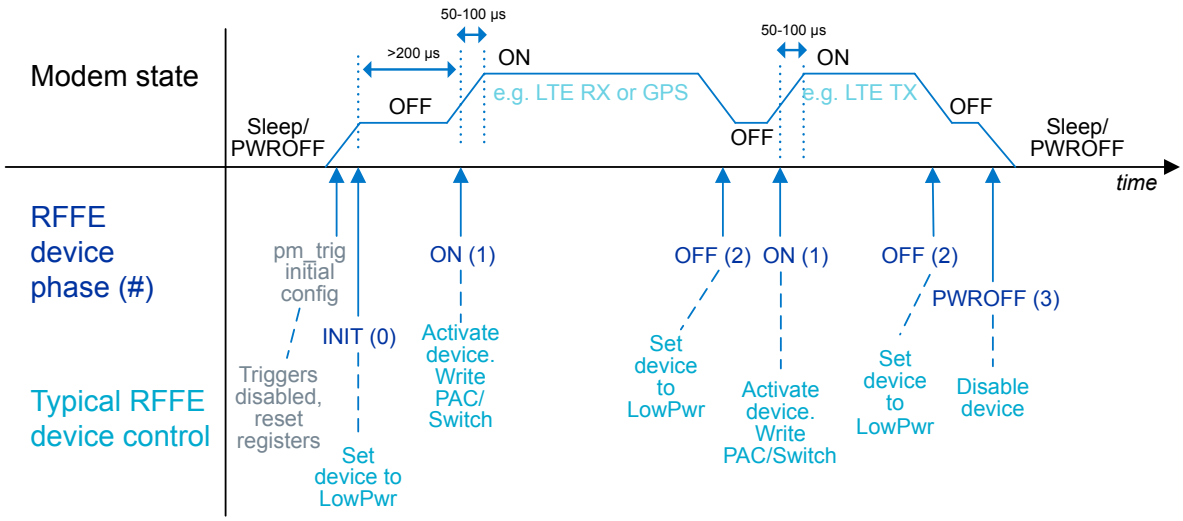


*Figure 1: RFFE device control and timing in different phases*

In the figure, PAC/Switch refers to a register in an example antenna tuner that controls the tunable capacitors and/or switches.

## 6.5.2 Phases INIT(0), OFF(2), and PWROFF(3)

Phases INIT(0), OFF(2), and PWROFF(3) are introduced here.

The syntax of each phase is the same except for the phase# parameter:

**INIT (0)**

```
AT%XMIPIRFFECTRL=<dev_id>,0,<n>,<address_0>,<data_0>,…,<address_n-1>,<data_n-1>
```

**OFF (2)**

```
AT%XMIPIRFFECTRL=<dev_id>,2,<n>,<address_0>,<data_0>,…,<address_n-1>,<data_n-1>
```

**PWROFF (3)**

```
AT%XMIPIRFFECTRL=<dev_id>,3,<n>,<address_0>,<data_0>,…,<address_n-1>,<data_n-1>
```

The parameters and their defined values are the following:

**<n>**

The number of address/data pairs. Valid values are 0, 1, 2, 3, 4. If the value is 0, all the following fields must be omitted.

**<address_x>**

The 8-bit address of the internal register in *MIPI RFFE* device. x = 0, ..., n−1.

**<data_x>**

The 8-bit data to be written to <address_x>. x = 0, ..., n−1.

For command examples, see nWP034 - nRF9160 Hardware Verification Guidelines.

## 6.5.3 Phase ON(1)

Phase ON(1) is introduced here. It contains the most options for configuring the device.

Syntax:

```
AT
%XMIPIRFFECTRL=<dev_id>,1,<n>,<act_addr_0>,<act_data_0>,<act_addr_n-1>,<act_data_n-1>,<k>,
<addr_0>,<addr_1>,<data_0_0>,<data_1_0>,<freq_0>,...,<data_0_k-1>,<data_1_k-1>,<freq_k-1>
```

The parameters and their descriptions are the following:

**<n>**

The number of activation register address-data pairs. Valid values are 0, 1, 2. If n = 0, act_addr_0/1 and act_data_0/1 must be omitted.

**<act_addr_x>**

8-bit address of the first register whose value is set to e.g. activate device. This is written each time RF starts. Must be given if <n> is greater than 0.

**<act_data_x>**

Optional 8-bit data for the register in <act_addr_x>.

Must be given if <n> is greater than 0.

**<k>**

The number of frequencies in the configuration. Valid values are 0–64. If k = 0, all the following fields must be omitted.

**<addr_0>**

The 8-bit address of the first register, whose value is changed on the basis of RF frequency.

**<addr_1>**

The 8-bit address of the other register, whose value is changed on the basis of RF frequency. If addr_1 == addr_0, then only <data_0_y> is written.

**<data_0_y>**

The 8-bit data for the register in <addr_0>, if frequency is smaller than or equal to <freq_y>.

**<data_1_y>**

The 8-bit data for the register in <addr_1>, if frequency is smaller than or equal to <freq_y>. Note that data_1_y must be given (e.g. as 0) even if addr_1 == addr_0.

**<freq_y>**

The frequency in MHz (integer), to which the current RF frequency is compared. If current RF frequency is smaller than or equal to <freq_y>, then <data_0_y> is written to <addr_0> and <data_1_y> is written to <addr_1>. Note that if the RF frequency is greater than <freq_k–1> (the last given frequency), then neither <addr_0> nor <addr_1> is written.

For command examples, see nWP034 - nRF9160 Hardware Verification Guidelines.

### 6.5.4 Delete configuration

The `AT%XMIPIRFFEDEV= <dev_id >` command deletes all configurations for the *MIPI RFFE* device, including phase configurations. To delete the configuration of each phase individually, set <n> = 0 or/and <k> = 0 in the phase-specific command.

For example, to delete only the ON phase configuration, send:

```
AT%XMIPIRFFECTRL=<dev_id>,1,0,0
```

To delete the PWROFF phase configuration, send:

```
AT%XMIPIRFFECTRL==<dev_id>,3,0
```

## 6.6 Alternative configuration of SiP antenna switch %XANTCFG

The Nordic-proprietary **`%XANTCFG`** command configures the *SiP*-internal antenna switch to an alternative predefined position. v1.1.x≥3 v1.2.x

### 6.6.1 Set command

The set command configures the *SiP*-internal antenna switch to an alternative predefined position.

The **%XANTCFG** command supports one predefined setting. Therefore, in the GPS mode the input signal to nRF9160's ANT input is routed back out from the AUX output. The configuration can be stored to device memory using `AT+CFUN=0` or given each time at boot before the modem is activated. After a valid configuration exists, nRF9160 automatically controls the switch during *GPS* reception. Adding further predefined settings requires a modification to MFW.

Syntax:

```
%XANTCFG=<cfg>
```

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

**<cfg>**

0 – Reserved, no action

1 – ANT input directed to AUX output in the GPS mode

2 – Reserved, no action

...

7 – Reserved, no action

The following command example directs ANT input to AUX output:

```
%XANTCFG=1
OK
```

The following command example deletes the previous configuration:

```
%XANTCFG
OK
```

## 6.6.2 Read command

The command reads the currently active configuration.

Syntax:

```
%XANTFCG?
```

The following command example returns the currently active configuration:

```
%XANTCFG?
%XANTCFG: 1
OK
```

## 6.6.3 Test command

The test command is not supported.

# 7 Packet domain commands

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

## 7.1 Define PDP context +CGDCONT

The **+CGDCONT** command defines *Packet Data Protocol (PDP) Context*. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.1*

### 7.1.1 Set command

The set command configures connection parameters.

Syntax:

```
+CGDCONT=<cid> [,<PDP_type> [,<APN> [,<PDP_addr> [,<d_comp> [,<h_comp> [,<IPv4AddrAlloc>
 [,<request_type> [,<P-CSCF_discovery> [,<IM_CN_Signalling_Flag_Ind> [,<NSLPI>
 [,<securePCO>]]]]]]]]]]]
```

**Note: +CGDCONT=<cid>** causes the values for context number <cid> to become undefined.

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

The following command example configures CID 1 to use IPv4 and access point "IOT_apn"

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

## 7.1.2 Read command

The 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

The following command example reads configured default bearers:

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

### 7.1.3 Test command

The test command is not supported.

## 7.2 Packet domain event reporting +CGEREP

The **+CGEREP** command enables or disables the sending of packet domain events. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.19*.

### 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 on page 100.

Syntax:

```
+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.

The following command example subscribes 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

The following command example reads the current mode:

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

### 7.2.3 Test command

The test command reads supported modes and buffering settings.

Response syntax:

```
+CGEREP: (list of supported <mode>s),(list of supported <bfr>s)
```

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

**<mode>**

0 – 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 Packet domain event unsolicited result codes +CGEV

Unsolicited packet domain notifications are sent when the device is detached from the network or when a packet data connection is activated, deactivated, or modified. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.19*.

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
```

Battery voltage is low:

```
+CGEV: ME BATTERY LOW
```

The *ME* has activated a default bearer:

```
+CGEV: ME PDN ACT <cid>[,<reason>]
```

The network has activated a dedicated bearer:

```
+CGEV: NW ACT <p_cid>, <cid>, <event_type>
```

The network has deactivated a default bearer:

```
+CGEV: NW PDN DEACT <cid>
```

The UE has deactivated a default bearer:

```
+CGEV: ME PDN DEACT <cid>
```

The network has deactivated a dedicated bearer:

```
+CGEV: NW DEACT <p_cid>, <cid>, <event_type>
```

The UE has deactivated a dedicated bearer:

```
+CGEV: ME DEACT <p_cid>, <cid>, <event_type>
```

The network has modified a bearer:

```
+CGEV: NW MODIFY <cid>, <change_reason>, <event_type>
```

The UE has modified a bearer:

```
+CGEV: ME MODIFY <cid>, <change_reason>, <event_type>
```

IPv6 link is up for the default bearer:

```
+CGEV: IPV6 <cid>
```

IPv6 address resolution or refresh failure:

```
+CGEV: IPV6 FAIL <cid>
```

Requested procedure restricted: v1.1.x v1.2.x

```
+CGEV: RESTR <cause>, <validity>
```

**<cid>**

0–11

**<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 – Permanent 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.

The example notification shows that an initial *Packet Data Network (PDN)* connection is activated:

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

The example notification shows that the device is detached from network:

```
+CGEV: ME DETACH
```

The example notification shows a restriction caused by throttling with temporary validity.

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

# 7.4 PDP context activate +CGACT

The **+CGACT** command activates or deactivates a *PDN* connection. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.10*.

## 7.4.1 Set command

The set command activates or deactivates a *PDN* connection.

> **Note:** Initial PDN connection (cid 0) could not be activated or deactivated.

First, the *Packet Data Protocol (PDP) Context* needs to be defined with the +CGDCONT command, see Define PDP context +CGDCONT on page 97.

Syntax:

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

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

**<state>**

0 – Deactivate

1 – Activate

**<cid>**

1–11

The following command example activates a bearer configured with CID 1:

```
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

The following command example returns a list of connections with states:

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

### 7.4.3 Test command

The test command returns a list of supported states.

Response syntax:

```
+CGACT: (list of supported <state>s)
```

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

**<state>**

0 – Deactivate

1 – Activate

Example:

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

## 7.5 Allocate new CID %XNEWCID

The Nordic-proprietary **%XNEWCID** command allocates a new context identifier. v1.0.x v1.1.x v1.2.x

### 7.5.1 Set command

The set command is not supported.

### 7.5.2 Read command

The read command allocates a new context identifier.

The command allocates a unique context identifier, which can be referenced with other commands like **+CGDCONT**. The allocated identifier can be deallocated with the **CGDCONT** command by giving only the <cid> parameter.

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

The following command example requests the allocation of a new context identifier:

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

### 7.5.3 Test command

The test command is not supported.

## 7.6 Map CID to PDN ID %XGETPDNID

The Nordic-proprietary **%XGETPDNID** command maps the context identifier to *PDN* ID. This command can be used only when the modem is activated. v1.0.x v1.1.x v1.2.x

### 7.6.1 Set command

The set command maps the context identifier to *PDN* ID.

PDN ID is used on a data path to select one of the existing connections for data transfer.

Syntax:

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

Response syntax:

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

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

**<cid>**

0–11

**<pdn_id>**

0–20

Example:

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

### 7.6.2 Read command

The read command is not supported.

### 7.6.3 Test command

The test command is not supported.

# 7.7 QoS dynamic params +CGEQOSRDP

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. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.27*.

## 7.7.1 Set command

The set command reads dynamic *EPS QoS* parameters.

Syntax:

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

Response syntax:

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

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

**<cid>**

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

The following command example returns a list of contexts with QoS parameters:

```
Get list of contexts with QOS parameters
AT+CGEQOSRDP
+CGEQOSRDP: 0,0,,
+CGEQOSRDP: 1,2,,
+CGEQOSRDP: 2,4,,,1,65280000
OK
```

## 7.7.2 Read command

The read command is not supported.

## 7.7.3 Test command

The test command is not supported.

# 7.8 Show PDP address(es) +CGPADDR

The +CGPADDR command returns a list of *Packet Data Protocol (PDP)* addresses for the specified context identifiers. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.14*.

## 7.8.1 Set command

The set command returns a list of *PDP* addresses for the specified context identifiers. This command issues a valid response only when the modem is activated.

Syntax:

```
+CGPADDR[=<cid>]
```

If <cid> is not present, all activated contexts are listed.

Response syntax:

```
[+CGPADDR: <cid>[,<PDP_addr_1>[,<PDP_addr_2>]]]
```

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

**<cid>**

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.

The following command example returns the IP address for context 1:

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

## 7.8.2 Read command

The read command is not supported.

## 7.8.3 Test command

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

Response syntax:

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

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

**<cid>**

0–11

Example:

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

# 7.9 PDN connection dynamic parameters +CGCONTRDP

The **+CGCONTRDP** command returns information for an active *PDN* connection. This command issues a valid response only when the modem is activated. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.23*.

## 7.9.1 Set command

The set command returns information for an active *PDN* connection.

Syntax:

```
+CGCONTRDP=<cid>
```

Response syntax:

```
+CGCONTRDP: <cid>,<bearer_id>,<apn>[,<local_addr and
 subnet_mask>[,<gw_addr>[,<DNS_prim_addr>[,<DNS_sec_addr>[,,,,,<IPv4_MTU]]]]]
```

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

**<cid>**

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.

The following command example reads dynamic parameters for an initial PDN connection:

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

### 7.9.2 Read command

The read command is not supported.

### 7.9.3 Test command

The test command is not supported.

## 7.10 PS attach or detach +CGATT

The **+CGATT** command attaches the *MT* to or detaches the MT from the Packet Domain services. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.9*.

### 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.

Syntax:

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

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

**<state>**

0 – Detached

1 – Attached

The following command example performs an EPS attach:

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

### 7.10.2 Read command

The read command reads the state.

Response syntax:

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

The response parameters and their defined values are the following:

**<state>**

0 – Detached

1 – Attached

The following command example reads the state in EPS attach state:

```
AT+CGATT?
+CGATT: 1
OK
```

### 7.10.3 Test command

The test command returns a list of supported states.

Response syntax:

```
+CGATT: (list of supported <state>s)
```

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

**<state>**

0 – Detached

1 – Attached

Example:

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

## 7.11 Power preference indication for EPS +CEPPI

The **+CEPPI** command selects the power saving preference. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.38*.

### 7.11.1 Set command

The set command selects if the *UE* indicates to the network during radio connection that it prefers low power configuration.

Syntax:

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

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

**<power preference>**

0 – Normal

1 – Low power consumption

The following command example selects the power saving preference:

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

### 7.11.2 Read command

The read command is not supported.

### 7.11.3 Test command

The test command lists the supported power preferences.

Syntax:

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

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

**<power preference>**

0 – Normal

1 – Low power consumption

The following command example lists the supported power saving preferences:

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

## 7.12 Protocol configuration options notification %XPCO

The Nordic-proprietary **%XPCO** command subscribes *PCO* notifications. v1.0.x v1.1.x v1.2.x

### 7.12.1 Set command

The set command subscribes *PCO* notifications.

Syntax:

```
%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 in decimal format

**<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
```

The following is an example of a PCO notification for a FF00h container:

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

### 7.12.2 Read command

The read command is not supported.

### 7.12.3 Test command

The test command is not supported.

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

The Nordic-proprietary **%XEPCO** command selects the usage of ePCO/PCO in PDN connection establishment. v1.0.x v1.1.x v1.2.x

### 7.13.1 Set command

The set command selects ePCO/PCO usage.

Syntax:

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

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

**<epco>**

0 – Use PCO

1 – Use ePCO

The following command example disables ePCO and selects PCO:

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

### 7.13.2 Read command

The command reads the state of ePCO/PCO usage. v1.1.x≥3

Response syntax:

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

The following command example reads the state of ePCO/PCO usage:

```
AT%XEPCO?
%XEPCO: 1
OK
```

### 7.13.3 Test command

The test command is not supported.

## 7.14 APN class access %XAPNCLASS

The Nordic-proprietary **%XAPNCLASS** command reads *APN* class data. v1.0.x v1.1.x v1.2.x

### 7.14.1 Set command

The set command reads *APN* class data.

Syntax:

```
%XAPNCLASS=<oper>,<class>[,<apn>]
```

Read response syntax:

```
%XAPNCLASS: <class>,<apn>,<addr_type>
```

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

**<oper>**

0 – Read

**<class>**

APN class

**<apn>**

APN name string

**<addr_type>**

String

IP – Internet Protocol

IPV6 – Internet Protocol version 6

IPV4V6 – Virtual type of dual IP stack

The following command example reads APN class 3:

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

### 7.14.2 Read command

The read command is not supported.

### 7.14.3 Test command

The test command is not supported.

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

The Nordic-proprietary **%XIPV6FAIL** indicates an external IP stack IPv6 address resolution or refresh failure. v1.0.x v1.1.x v1.2.x

### 7.15.1 Set command

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

Syntax:

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

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

**<cid>**

Context identifier

**<failure_type>**

0 – IPv6 address refresh failure

1 – IPv6 address resolution failure

The following command example indicates the modem an IPv6 address resolution failure in the default context identifier 0:

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

### 7.15.2 Read command

The read command is not supported.

### 7.15.3 Test command

The test command is not supported.

## 7.16 Define PDN connection authentication parameters +CGAUTH

The **+CGAUTH** command specifies authentication parameters. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.31*.

### 7.16.1 Set command

The set command specifies authentication parameters for a *PDN* connection specified by 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

The following command example sets authentication parameters for CID=1 context:

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

### 7.16.2 Read command

The read command is not supported.

### 7.16.3 Test command

The test command is not supported.

## 7.17 Signaling connection status +CSCON

The **+CSCON** command controls the presentation of an unsolicited result code. v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.30*.

### 7.17.1 Set command

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

Syntax:

```
+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

The following command example enables level 3 indications.

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

The following is an example of a level-3-related unsolicited indication:

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

## 7.17.2 Read command

The command returns the current status of unsolicited result code presentation <n>.

The parameter <mode> is returned always when <n> = 0 or when <n> = 1. The optional parameter <state> is returned when <n> = 2 and <access> when <n> = 3.

Response syntax:

```
+CSCON: <n>,<mode>[,<state>[,<access]]
```

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

**<n>**

0 – Unsolicited indications disabled

1 – Enabled: <mode>

2 – Enabled: <mode>[,<state>]

3 – Enabled: <mode>[,<state>[,<access>]]

**<mode>**

0 – Idle

1 – Connected

**<state>**

7 – E-UTRAN connected

**<access>**

4 – Radio access of type E-UTRAN FDD

When reading the current signaling connection status, the following response indicates that unsolicited indications are disabled, and the modem is an idle state:

```
AT+CSCON?
+CSCON: 0,0
OK
```

The following response indicates that unsolicited indications are enabled, the modem mode is 1, E-UTRAN is connected and the radio access type is E-UTRAN FDD:

```
AT+CSCON?
+CSCON: 3,1,7,4
OK
```

### 7.17.3 Test command

The test command returns a list of supported values of <n> as a compound value.

Response syntax:

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

The test command 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>]]

The following command example returns the supported values:

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

## 7.18 Use of APN %XAPNSTATUS

The Nordic-proprietary **%XAPNSTATUS** command enables and disables the use of an *APN*. v1.2.x

### 7.18.1 Set command

The set command enables or disables the use of an *APN*.

Syntax:

```
%XAPNSTATUS=<status>,<APN>
```

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

**<status>**

0 – Disable *APN*

1 – Enable *APN*

**<APN>**

String – *APN*

The following command example disables the use of an Internet *APN*:

```
AT%XAPNSTATUS=0,"internet"
```

The following command example enables the use of an Internet *APN*:

```
AT%XAPNSTATUS=1,"internet"
```

## 7.18.2 Read command

The command reads a list of disabled *APN*s. v1.2.x

Response syntax:

```
%XAPNSTATUS: [<apn_1>[,…<apn_n>]]
```

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

**<apn_x>**

APN string

The following command example reads a list of disabled *APN*s:

```
AT%XAPNSTATUS?
%XAPNSTATUS: ”APN_1”,”APN_2”
OK
```

## 7.18.3 Test command

The test command is not supported.

# 8 Network service related commands

For reference, see *3GPP 27.007 Ch. 7*.

## 8.1 PLMN selection +COPS

The **+COPS** command selects a *PLMN* automatically or manually, and reads and searches the current mobile network. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.3*

### 8.1.1 Set command

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

Syntax:

```
+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.

The following command example selects the automatic network selection:

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

The following command manually selects network 24407:

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

### 8.1.2 Read command

The command reads the current mobile network.

Response syntax:

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

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

**<mode>**

0 – Automatic network selection

1 – Manual network selection

2 – Deregistered. Only for the Read command.

**<format>**

0 – Long alphanumeric <oper> format

1 – Short alphanumeric <oper> format

2 – Numeric <oper> format

**<oper>**

A string consisting of the operator name in the alphanumeric format or a string of *MCC* and *MNC* values.

**<AcT>**

7 – E-UTRAN

9 – E-UTRAN (NB-S1 mode)

The following command example reads the current selection mode and network:

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

The following command example reads the current selection mode and network with the operator name in 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

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)

**Note:**

- The command fails if the device has an active radio connection. It returns `ERROR` or `+CME ERROR: 516`
- The time needed to perform a network search depends on device configuration and network conditions.

The following command example is used for a manual network search:

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

# 8.2 Forced PLMN search %COPS

The Nordic-proprietary **%COPS** command performs a forced *PLMN* search. v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.3*

## 8.2.1 Set command

The set command is not supported.

## 8.2.2 Read command

The read command is not supported.

## 8.2.3 Test command

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

The command is similar to **+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)

The following command example is used for a manual network search:

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

# 8.3 Power saving mode setting +CPSMS

The **+CPSMS** command controls *PSM* settings. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.38*.

## 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 manufacturer-specific default values.

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

**<mode>**

0 – Disable power saving mode

1 – Enable power saving mode

**<Requested_Periodic-RAU>**

Ignored

**<Requested_GPRS-READY-timer>**

Ignored

**<Requested_Periodic-TAU>**

String. One byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Periodic-TAU is set to the manufacturer-specific default. 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.

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the *General Packet Radio Services (GPRS)* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 10 minutes

0 0 1 – value is incremented in multiples of 1 hour

0 1 0 – value is incremented in multiples of 10 hours

0 1 1 – value is incremented in multiples of 2 seconds

1 0 0 – value is incremented in multiples of 30 seconds

1 0 1 – value is incremented in multiples of 1 minute

1 1 0 – value is incremented in multiples of 320 hours

> **Note:** Interpreted as 1 hour if the request sent to network is not integrity protected. After registration, check the value given by the network (see Network registration status +CEREG on page 138). If 1 hour unit is given, disable and enable *PSM* using commands `+CPSMS=0` and `+CPSMS=1`.

1 1 1 – value indicates that the timer is deactivated

> **Note:** If the USIM profile in use is a Verizon one, the minimum value for <Requested_Periodic-TAU> is 190 minutes.

**<Requested_Active-Time>**

String. One byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Active-Time is set to the manufacturer-specific default. For the coding and value range, see the GPRS Timer 2 IE in *3GPP TS 24.008* Table 10.5.163/3GPP TS 24.008.

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 2 seconds

0 0 1 – value is incremented in multiples of 1 minute

0 1 0 – value is incremented in multiples of 6 minutes

1 1 1 – value indicates that the timer is deactivated

The following command example enables power saving mode and set timer values. Set Periodic-TAU timer to 10 minutes and Active-Time to 1 minute.

```
AT+CPSMS=1,"","","10101010","00100001"
OK
```

The following command example disables power saving mode:

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

The following command example disables power saving mode and sets timer to default values:

```
AT+CPSMS=
OK
```

## 8.3.2 Read command

The command reads the current *PSM* settings.

Response syntax:

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

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

**<mode>**

0 – Disable power saving mode

1 – Enable power saving mode

**<Requested_Periodic-RAU>**

Ignored

**<Requested_GPRS-READY-timer>**

Ignored

**<Requested_Periodic-TAU>**

String. One byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Periodic-TAU is set to the manufacturer-specific default. 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.

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 10 minutes

0 0 1 – value is incremented in multiples of 1 hour

0 1 0 – value is incremented in multiples of 10 hours

0 1 1 – value is incremented in multiples of 2 seconds

1 0 0 – value is incremented in multiples of 30 seconds

1 0 1 – value is incremented in multiples of 1 minute

1 1 0 – value is incremented in multiples of 320 hours

> **Note:** Interpreted as 1 hour if the request sent to network is not integrity protected. After registration, check the value given by the network (see Network registration status +CEREG on page 138). If 1 hour unit is given, disable and enable *PSM* using commands `+CPSMS=0` and `+CPSMS=1`.

1 1 1 – value indicates that the timer is deactivated

> **Note:** If the USIM profile in use is a Verizon one, the minimum value for <Requested_Periodic-TAU> is 190 minutes.

**<Requested_Active-Time>**

String. One byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Active-Time is set to the manufacturer-specific default. For the coding and value range, see the GPRS Timer 2 IE in *3GPP TS 24.008* Table 10.5.163/3GPP TS 24.008.

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 2 seconds

0 0 1 – value is incremented in multiples of 1 minute

0 1 0 – value is incremented in multiples of 6 minutes

1 1 1 – value indicates that the timer is deactivated

The following command example reads the current power saving mode settings:

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

### 8.3.3 Test command

The test command is not supported.

## 8.4 eDRX setting +CEDRXS

The **+CEDRXS** command controls the setting of *eDRX* parameters. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.40*.

### 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 on page 77.

Syntax:

```
+CEDRXS=[<mode>,[,<AcT-type>[,<Requested_eDRX_value>]]]
```

Unsolicited result code syntax:

```
+CEDRXP: <AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>
 [,<Paging_time_window>]]]
```

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

**<mode>**

0 – Disable the use of eDRX

1 – Enable the use of eDRX

2 – Enable the use of eDRX and enable the unsolicited result code

3 – Disable the use of eDRX and discard all parameters for eDRX or, if available, reset to the manufacturer-specific default values

**<ActT-type>**

4 – E-UTRAN (WB-S1 mode)

5 – E-UTRAN (NB-S1 mode)

[6] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as if the Extended DRX parameters IE were not included in the message by this version of the protocol.

[7] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as 0010 by this version of the protocol.

[8] The value is applicable only in NB-S1 mode. If received in WB-S1 mode it is interpreted as 1101 by this version of the protocol.

**<Requested_eDRX_value>**

String. Half a byte in 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*). Optional. If not present, the value of the requested eDRX period is set to the manufacturer-specific default.

Bit

4 3 2 1 – E-UTRAN eDRX cycle length duration

0 0 0 0 – 5.12 seconds[6]

0 0 0 1 – 10.24 seconds[6]

0 0 1 0 – 20.48 seconds

0 0 1 1 – 40.96 seconds

0 1 0 0 – 61.44 seconds[7]

0 1 0 1 – 81.92 seconds

0 1 1 0 – 102.4 seconds[7]

0 1 1 1 – 122.88 seconds[7]

1 0 0 0 – 143.36 seconds[7]

1 0 0 1 – 163.84 seconds

1 0 1 0 – 327.68 seconds

1 0 1 1 – 655,36 seconds

1 1 0 0 – 1310.72 seconds

1 1 0 1 – 2621.44 seconds

1 1 1 0 – 5242.88 seconds[8]

1 1 1 1 – 10485.76 seconds[8]

**<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*).

Bit

4 3 2 1 – E-UTRAN eDRX cycle length duration

0 0 0 0 – 5.12 seconds[6]

0 0 0 1 – 10.24 seconds[6]

0 0 1 0 – 20.48 seconds

0 0 1 1 – 40.96 seconds

0 1 0 0 – 61.44 seconds[7]

0 1 0 1 – 81.92 seconds

0 1 1 0 – 102.4 seconds[7]

0 1 1 1 – 122.88 seconds[7]

1 0 0 0 – 143.36 seconds[7]

1 0 0 1 – 163.84 seconds

1 0 1 0 – 327.68 seconds

1 0 1 1 – 655,36 seconds

1 1 0 0 – 1310.72 seconds

1 1 0 1 – 2621.44 seconds

1 1 1 0 – 5242.88 seconds[8]

1 1 1 1 – 10485.76 seconds[8]

**<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*).

LTE Cat M1 mode

Bit

4 3 2 1 – Paging Time Window length

0 0 0 0 – 1.28 seconds

0 0 0 1 – 2.56 seconds

0 0 1 0 – 3.84 seconds

0 0 1 1 – 5.12 seconds

0 1 0 0 – 6.4 seconds

0 1 0 1 – 7.68 seconds

0 1 1 0 – 8.96 seconds

0 1 1 1 – 10.24 seconds

1 0 0 0 – 11.52 seconds

1 0 0 1 – 12.8 seconds

1 0 1 0 – 14.08 seconds

1 0 1 1 – 15.36 seconds

1 1 0 0 – 16.64 seconds

1 1 0 1 – 17.92 seconds

1 1 1 0 – 19.20 seconds

1 1 1 1 – 20.48 seconds

LTE Cat NB1 mode

Bit

4 3 2 1 – Paging Time Window length

0 0 0 0 – 2.56 seconds

0 0 0 1 – 5.12 seconds

0 0 1 0 – 7.68 seconds

0 0 1 1 – 10.24 seconds

0 1 0 0 – 12.8 seconds

0 1 0 1 – 15.36 seconds

0 1 1 0 – 17.92 seconds

0 1 1 1 – 20.48 seconds

1 0 0 0 – 23.04 seconds

1 0 0 1 – 25.6 seconds

1 0 1 0 – 28.16 seconds

1 0 1 1 – 30.72 seconds

1 1 0 0 – 33.28 seconds

1 1 0 1 – 35.84 seconds

1 1 1 0 – 38.4 seconds

1 1 1 1 – 40.96 seconds

The following command example enables eDRX and sets the requested eDRX value:

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

The unsolicited notification when <mode> 2 is used:

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

## 8.4.2 Read command

The command is used to read the requested *eDRX* parameters.

Response syntax:

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

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

**<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)

**<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*).

The following command example reads the requested eDRX value:

```
AT+CEDRXS?
+CEDRXS: 4,"0001"
OK
```

## 8.4.3 Test command

The test command is used to list the supported *eDRX* parameters.

Response syntax:

```
+CEDRXS: (list of supported <mode>s),(list of supported <AcT-type>s),(list of supported
 <Requested_eDRX_value>s)
```

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

**<mode>**

0 – Disable the use of eDRX

1 – Enable the use of eDRX

2 – Enable the use of eDRX and enable the unsolicited result code

3 – Disable the use of eDRX and discard all parameters for eDRX or, if available, reset to the manufacturer-specific default values

**<ActT-type>**

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*).

The following command example reads the supported parameter values:

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

# 8.5 Read EDRX dynamic parameters +CEDRXRDP

The **+CEDRXRDP** command reads dynamic eDRX parameters. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.41*.

## 8.5.1 Set command

The set command reads dynamic *eDRX* parameters.

Syntax:

```
+CEDRXRDP
```

Response syntax:

```
+CEDRXRDP: <AcT-type>[,<Requested_eDRX_value>[,<NW-
provided_eDRX_value>[,<Paging_time_window>]]]
```

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

**<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*).

Bit

4 3 2 1 – E-UTRAN eDRX cycle length duration

0 0 0 0 – 5.12 seconds

0 0 0 1 – 10.24 seconds

0 0 1 0 – 20.48 seconds

0 0 1 1 – 40.96 seconds

0 1 0 0 – 61.44 seconds

0 1 0 1 – 81.92 seconds

0 1 1 0 – 102.4 seconds

0 1 1 1 – 122.88 seconds

1 0 0 0 – 143.36 seconds

1 0 0 1 – 163.84 seconds

1 0 1 0 – 327.68 seconds

1 0 1 1 – 655,36 seconds

1 1 0 0 – 1310.72 seconds

1 1 0 1 – 2621.44 seconds

1 1 1 0 – 5242.88 seconds

1 1 1 1 – 10485.76 seconds

**<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*).

LTE Cat M1 mode

Bit

4 3 2 1 – Paging Time Window length

0 0 0 0 – 1.28 seconds

0 0 0 1 – 2.56 seconds

0 0 1 0 – 3.84 seconds

0 0 1 1 – 5.12 seconds

0 1 0 0 – 6.4 seconds

0 1 0 1 – 7.68 seconds

0 1 1 0 – 8.96 seconds

0 1 1 1 – 10.24 seconds

1 0 0 0 – 11.52 seconds

1 0 0 1 – 12.8 seconds

1 0 1 0 – 14.08 seconds

1 0 1 1 – 15.36 seconds

1 1 0 0 – 16.64 seconds

1 1 0 1 – 17.92 seconds

1 1 1 0 – 19.20 seconds

1 1 1 1 – 20.48 seconds

LTE Cat NB1 mode

Bit

4 3 2 1 – Paging Time Window length

0 0 0 0 – 2.56 seconds

0 0 0 1 – 5.12 seconds

0 0 1 0 – 7.68 seconds

0 0 1 1 – 10.24 seconds

0 1 0 0 – 12.8 seconds

0 1 0 1 – 15.36 seconds

0 1 1 0 – 17.92 seconds

0 1 1 1 – 20.48 seconds

1 0 0 0 – 23.04 seconds

1 0 0 1 – 25.6 seconds

1 0 1 0 – 28.16 seconds

1 0 1 1 – 30.72 seconds

1 1 0 0 – 33.28 seconds

1 1 0 1 – 35.84 seconds

1 1 1 0 – 38.4 seconds

1 1 1 1 – 40.96 seconds

The following command example reads eDRX parameters:

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

### 8.5.2 Read command

The read command is not supported.

### 8.5.3 Test command

The test command is not supported.

## 8.6 Subscriber number +CNUM

The **+CNUM** command returns the subscriber *Mobile Station International Subscriber Directory Number (MSISDN)*. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.1*.

### 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 set 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*)

The following command example reads the subscriber number stored in the SIM:

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

### 8.6.2 Read command

The read command is not supported.

### 8.6.3 Test command

The test command is not supported.

## 8.7 Read operator name +COPN

The **+COPN** command reads operator names. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.21*.

### 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
```

### 8.7.2 Read command

The read command is not supported.

### 8.7.3 Test command

The test command is not supported.

## 8.8 Facility lock +CLCK

The **+CLCK** command locks, unlocks, or interrogates a facility. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.4*.

### 8.8.1 Set command

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

Syntax:

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

**<fac>**

SC – SIM

PN – Network personalization v1.2.x

PU – Network subset personalization v1.2.x

PP – Service provider personalization v1.2.x

PC – Corporate personalization v1.2.x

PS – USIM personalization v1.2.x

**<mode>**

0 – Unlock

1 – Lock

2 – Query status v1.2.x

**<passwd>**

String. Password for the facility.

**Note:** SC is supported in modes 0 and 1. PN, PU, PP, PC, and PS are supported in modes 0 and 2.

The following command example disables *PIN* query:

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

The following command example checks network personalization status when it is not active:

```
AT+CLCK=”PN”,2
+CLCK: 0
OK
```

### 8.8.2 Read command

The read command is not supported.

### 8.8.3 Test command

The test command lists supported facilities.

Response syntax:

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

Example:

```
AT+CLCK=?
+CLCK: ("SC","PS","PN","PU","PP","PC")
OK
```

## 8.9 Change password +CPWD

The **+CPWD** command changes the password for the facility lock. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 7.5*.

## 8.9.1 Set command

The set command changes the password for the facility lock.

Syntax:

```
+CPWD=<fac>,<oldpwd>,<newpwd>
```

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

**<fac>**

"SC" – SIM PIN

"P2" – SIM PIN2

**<oldpwd>,<newpwd>**

String. Password.

> **Note:** Currently only "SC" supported.

The following command example changes the SIM *PIN*:

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

## 8.9.2 Read command

The read command is not supported.

## 8.9.3 Test command

The test command returns the supported facilities and password length.

Response syntax:

```
+CPWD: list of supported (<fac>,<pwdlength>)s
```

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

**<fac>**

"SC" – SIM PIN

"P2" – SIM PIN2

**<pwdlength>**

Integer. Maximum length of the password

Example:

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

# 8.10 Network registration status +CEREG

The **+CEREG** command subscribes unsolicited network status notifications. v1.0.x v1.1.x v1.2.x

## 8.10.1 Set command

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 `+CEREG:<stat>`

2 – Enable unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 – Enable unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 – Enable unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]`

5 – Enable unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]]`

For the notification syntax parameters, see Read command on page 138.

The following command example subscribes notifications with level 2:

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

Unsolicited notification level 1, trying to attach:

```
+CEREG: 2
```

Unsolicited notification level 2, registered:

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

## 8.10.2 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 `+CEREG:<stat>`

2 – Enable unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 – Enable unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 – Enable unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]`

5 – Enable unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<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 E-UTRAN. For the coding and value range, see the GPRS Timer 3 IE in *3GPP TS 24.008* Table 10.5.163a/3GPP TS 24.008.

The following command example reads the current registration status:

```
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)
```

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

**<n>**

0 – Disable unsolicited result codes

1 – Enable unsolicited result codes `+CEREG:<stat>`

2 – Enable unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>]`

3 – Enable unsolicited result codes `+CEREG:<stat>[,<tac>,<ci>,<AcT>[,<cause_type>,<reject_cause>]]`

4 – Enable unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]`

5 – Enable unsolicited result codes `+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]]`

The example shows supported unsolicited results codes:

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

# 8.11 Subscribe unsolicited operator name indications %XOPNAME

The Nordic proprietary **%XOPNAME** command subscribes unsolicited operator name notifications. v1.0.x v1.1.x v1.2.x

## 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.

Syntax:

```
%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.

The following command example subscribes notifications:

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

An example of an unsolicited notification for a full and a short operator name:

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

An example of an unsolicited notification for a short operator name:

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

## 8.11.2 Read command

The read command is not supported.

## 8.11.3 Test command

The test command is not supported.

# 8.12 Subscribe unsolicited network time notifications %XTIME

The Nordic proprietary **%XTIME** command subscribes unsolicited network time notifications. v1.0.x v1.1.x v1.2.x

## 8.12.1 Set command

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

Syntax:

```
%XTIME=<n>
```

Notification syntax:

```
%XTIME: [<local_time_zone>],[<universal_time>],[<daylight_saving_time>]
```

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

**<n>**

0 – Unsubscribe unsolicited network time

1 – Subscribe unsolicited network time

**<local_time_zone>**

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.

The following command example subscribes notifications:

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

An example of an unsolicited notification for network time with all parameters:

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

An example of an unsolicited notification for network time without local time zone:

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

### 8.12.2 Read command

The read command is not supported.

### 8.12.3 Test command

The test command is not supported.

## 8.13 Set release assistance information %XRAI

The Nordic-proprietary **%XRAI** command sets release assistance information. v1.0.x v1.1.x v1.2.x

It is designed for cases where an application sends one packet uplink and expects one response back from the network or server. This can occur also at the end of an application session negotiation, where the application knows that it is going to send the last packet, for example an ack to a server.

### 8.13.1 Set command

The set command sets release assistance information.

Syntax:

```
%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.
- This method does not function properly when an application or device sends multiple packets to uplink without receiving anything in between. The **XRAI** setting does not mark a specific packet as the last one, and the application does not know when a packet has actually been sent.

The following command example sets release assistance information when the application has one packet to be sent and no response from the network is expected:

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

This setting should be disabled only after it is clear that the data has been sent or the final receive has been done. This information could be received, for example, from **+CSCON** when the device enters LTE idle state.

### 8.13.2 Read command

The command reads release assistance information. v1.1.0

Response syntax:

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

The response 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*.

The following command example reads release assistance information, the response being "Control plane no response":

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

### 8.13.3 Test command

The test command is not supported.

## 8.14 Operator ID %XOPERID

The Nordic-proprietary **%XOPERID** command identifies the operator *USIM*. v1.0.x v1.1.x v1.2.x

### 8.14.1 Set command

The set command returns the operator ID.

Syntax:

```
%XOPERID
```

Response syntax:

```
%XOPERID: <oper_id>
```

The response parameter and its defined values are the following:

**<oper_id>**

0 – Operator not identified as any of those listed below.

1 – Verizon

2 – AT&T

3 – AT&T FirstNet

4 – AT&T Cricket

5 – AT&T Jasper

6 – China Telecom

7 – Softbank

8 – Telstra v1.2.x

The following command example returns the operator ID:

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

### 8.14.2 Read command

The read command is not supported.

### 8.14.3 Test command

The test command is not supported.

## 8.15 Read modem parameters %XMONITOR

The Nordic-proprietary **%XMONITOR** command reads a set of modem parameters. v1.0.x v1.1.x v1.2.x

### 8.15.1 Set command

The set command reads modem parameters.

Response syntax: v1.0.x v1.1.x

```
%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>]
```

Response syntax: v1.2.x

```
%XMONITOR: <reg_status>,[<full_name>,<short_name>,<plmn>,<tac>,<AcT>,<band>,<cell_id>,
<phys_cell_id>,<EARFCN>,<rsrp>,<snr>,<NW-provided_eDRX_value>,<Active-Time>,<Periodic-TAU-
ext>,<Periodic-TAU>]
```

The response parameters and their defined values are the following:

[9] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as if the Extended DRX parameters IE were not included in the message by this version of the protocol.

[10] The value is applicable only in WB-S1 mode. If received in NB-S1 mode it is interpreted as 0010 by this version of the protocol.

**<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*.

[11] The value is applicable only in NB-S1 mode. If received in WB-S1 mode it is interpreted as 1101 by this version of the protocol.

**<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 4-bit format. The *eDRX* value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element (see *3GPP TS 24.008, subclause 10.5.5.32*).

Bit

4 3 2 1 – E-UTRAN e-I-DRX cycle length duration

0 0 0 0 – 5.12 seconds[9]

0 0 0 1 – 10.24 seconds[9]

0 0 1 0 – 20.48 seconds

0 0 1 1 – 40.96 seconds

0 1 0 0 – 61.44 seconds[10]

0 1 0 1 – 81.92 seconds

0 1 1 0 – 102.4 seconds[10]

0 1 1 1 – 122.88 seconds[10]

1 0 0 0 – 143.36 seconds[10]

1 0 0 1 – 163.84 seconds

1 0 1 0 – 327.68 seconds

1 0 1 1 – 655,36 seconds

1 1 0 0 – 1310.72 seconds

1 1 0 1 – 2621.44 seconds

1 1 1 0 – 5242.88 seconds[11]

1 1 1 1 – 10485.76 seconds[11]

**<Active-Time>**

String. One byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Active-Time is set to the manufacturer-specific default. For the coding and value range, see the GPRS Timer 2 IE in *3GPP TS 24.008* Table 10.5.163/3GPP TS 24.008.

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 define the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 2 seconds

0 0 1 – value is incremented in multiples of 1 minute

0 1 0 – value is incremented in multiples of 6 minutes

1 1 1 – value indicates that the timer is deactivated

**<Periodic-TAU>** v1.0.x v1.1.x

String. One byte in 8-bit format.

Indicates the extended periodic *TAU* value (T3412_EXT extended value) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 3 IE in *3GPP TS 24.008* Table 10.5.163a/3GPP TS 24.008.

GPRS Timer 3 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 defines the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 10 minutes

0 0 1 – value is incremented in multiples of 1 hour

0 1 0 – value is incremented in multiples of 10 hours

0 1 1 – value is incremented in multiples of 2 seconds

1 0 0 – value is incremented in multiples of 30 seconds

1 0 1 – value is incremented in multiples of 1 minute

1 1 0 – value is incremented in multiples of 320 hours

1 1 1 – value indicates that the timer is deactivated

**<Periodic-TAU-ext>** v1.2.x

String. One byte in 8-bit format.

Indicates the extended periodic *TAU* value (T3412_EXT extended value) allocated to the device in E-UTRAN. For the coding and value range, see the GPRS Timer 3 IE in *3GPP TS 24.008* Table 10.5.163a/3GPP TS 24.008.

GPRS Timer 3 value (octet 3).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 defines the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 10 minutes

0 0 1 – value is incremented in multiples of 1 hour

0 1 0 – value is incremented in multiples of 10 hours

0 1 1 – value is incremented in multiples of 2 seconds

1 0 0 – value is incremented in multiples of 30 seconds

1 0 1 – value is incremented in multiples of 1 minute

1 1 0 – value is incremented in multiples of 320 hours

1 1 1 – value indicates that the timer is deactivated

**<Periodic-TAU>** v1.2.x

String. One byte in 8-bit format.

Optional. Timer value updated if present. If not present, the value of the requested Periodic-TAU is set to the manufacturer-specific default. For the coding and value range, see the GPRS Timer IE in *3GPP TS 24.008* Table 10.5.172/3GPP TS 24.008.

Timer value (octet 2).

Bits 5 to 1 represent the binary coded timer value.

Bits 8 to 6 defines the timer value unit for the *GPRS* timer as follows:

Bits

8 7 6

0 0 0 – value is incremented in multiples of 2 seconds

0 0 1 – value is incremented in multiples of 1 minute

0 1 0 – value is incremented in multiples of 6 minutes

1 1 1 – value indicates that the timer is deactivated

The following command example reads modem parameters: v1.0.x v1.1.x

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

The following command example reads modem parameters: v1.2.x

```
AT%XMONITOR
%XMONITOR: 1,"EDAV","EDAV","26295","00B7",7,4,"00011B07",7,2300,63,39,"",
"11100000","11100000","00000000"
OK
```

### 8.15.2 Read command

The read command is not supported.

### 8.15.3 Test command

The test command is not supported.

## 8.16 Network time support %XNETTIME

The Nordic proprietary **%XNETTIME** command controls if the time received from the network is used. v1.1.x v1.2.x

### 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*.

Syntax:

```
%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

The following command example disables network time support:

```
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

The following command example reads network time support :

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

### 8.16.3 Test command

The test command is not supported.

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

The Nordic proprietary **%XDEEPSEARCH** command supports averaging cell search mode to detect weak cells. v1.1.x v1.2.x

### 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.

Syntax:

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

The command parameter and its defined values are the following:

**<deep_search>**

0 – Disable deep search

1 – Enable deep search

The following command example enables deep search support:

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

## 8.17.2 Read command

The command reads the status of deep search.

Response syntax:

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

The response parameter and its defined values are the following:

**<deep_search>**

0 – Disable deep search

1 – Enable deep search

The following command example reads deep search availability:

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

## 8.17.3 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
```

The command example triggers deep search:

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

# 8.18 Mobile network operator %XOPCONF

The Nordic-proprietary **%XOPCONF** command configures modem for the selected mobile network operator. v1.1.x≥3

## 8.18.1 Set command

The set command configures the modem to comply with the requirements of various mobile network operators. The configuration is stored in *NVM* when the device is powered off with `+CFUN=0`. The stored configuration is in use when the device is powered on.

Syntax:

```
%XOPCONF=<op_conf>
```

The command parameter and its defined values are the following:

**<op_conf>**

1 – Automatically detected from IMSI

2 – Verizon

3 – AT&T

4 – China Telecom

5 – Softbank

6 – Telstra

7 – Bell

8 – LGU+

The example command sets Verizon operator configuration:

```
AT%XOPCONF=2
OK
```

## 8.18.2 Read command

The command reads the current mobile network operator configuration.

Response syntax:

```
%XOPCONF: <op_conf>
```

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

**<op_conf>**

1 – Automatically detected from IMSI

2 – Verizon

3 – AT&T

4 – China Telecom

5 – Softbank

6 – Telstra

7 – Bell

8 – LGU+

The example command reads the configured operator:

```
AT%XOPCONF?
AT%XOPCONF: 2
OK
```

## 8.18.3 Test command

The test command is not supported.

# 9 Mobile termination errors

For reference, see *3GPP 27.007 Ch. 9*.

## 9.1 Report mobile termination errors +CMEE

The **+CMEE** command disables or enables the use of the final result code `+CME ERROR`. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 9.1*.

### 9.1.1 Set command

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

Syntax:

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

The set command 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. <err> values are specified in *3GPP TS 27.007 Ch. 9.2*. Vendor-specific values listed in the command chapters, the value range starts from 512.

The following command example enables error codes in responses:

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

### 9.1.2 Read command

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

Response syntax:

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

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

**<n>**

0 – Disabled. `ERROR` used as the final response in case of failure.

1 – Enabled. `+CME ERROR: <err>` result code and numeric <err> values used.

The following command example reads the current error code setting:

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

### 9.1.3 Test command

The test command returns supported values as a compound value.

Response syntax:

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

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

**<n>**

0 – Disabled. `ERROR` used as the final response in case of failure.

1 – Enabled. `+CME ERROR: <err>` result code and numeric <err> values used.

The following command example returns the supported values:

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

## 9.2 Report network error codes +CNEC

The **+CNEC** command activates or deactivates unsolicited reporting of error codes sent by the network.
v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 9.1B*.

### 9.2.1 Set command

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

Syntax:

```
+CNEC=[<n>]
```

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

**<n>**

0 – Disable unsolicited error reporting

8 – Enable unsolicited result code `+CNEC_EMM: <error_code>[,<cid>]` to report *EPS* mobility management errors

16 – Enable unsolicited result code `+CNEC_ESM: <error_code>[,<cid>]` to report EPS session management errors

24 – Enable unsolicited result codes for `+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>.

The following command example enables CNEC_ESM error codes.

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

The notification example shows *EMM* Cause 22 (Congestion) received from the network:

```
+CNEC_EMM: 22
```

## 9.2.2 Read command

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

Response syntax:

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

**<n>**

0 – Disable

8 – +CNEC_EMM enabled

16 – +CNEC_ESM enabled

24 – +CNEC_EMM and +CNEC_ESM

The following command example reads **CNEC** error code setting, both CNEC_EMM and CNEC_ESM enabled.

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

## 9.2.3 Test command

The test command returns the supported values as compound values.

Response syntax:

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

**<n>**

0 – Disable

8 – **+CNEC_EMM** enabled

16 – **+CNEC_ESM** enabled

24 – **+CNEM_EMM** and **+CNEC_ESM**

The following command example returns **CNEC** error code setting values.

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

# 9.3 Extended error report +CEER

The **+CEER** command returns an extended error report. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 6.10*

## 9.3.1 Set command

The set command returns an extended error report.

Syntax:

```
+CEER
```

Response syntax:

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

The 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.

The following command example reads the latest failure stored by the modem:

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

## 9.3.2 Read command

The read command is not supported.

## 9.3.3 Test command

The test command is not supported.

# 10 SMS commands

For reference, see *3GPP 27.005 Ch. 3*.

## 10.1 Message format +CMGF

The **+CMGF** command sets message format. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.005 Ch. 3.2.3*.

### 10.1.1 Set command

The set command selects between *PDU* and text format

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

Syntax:

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

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

**<mode>**

0 – PDU mode, default value

The following command example sets the message format to PDU mode:

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

### 10.1.2 Read command

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

Response syntax:

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

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

**<mode>**

0 – *PDU* mode

The following command example reads the current message format:

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

### 10.1.3 Test command

The test command lists the supported message formats.

Response syntax:

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

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

**<mode>**

0 – *PDU* mode

Example:

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

# 10.2 New message indications +CNMI

The **+CNMI** command selects how new messages are indicated. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.005 Ch. 3.4.1*.

## 10.2.1 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.

Syntax:

```
+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.

The following command example registers as a client for mobile-terminated SMS and status reports:

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

### 10.2.2 Read command

The 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:

**<mode>**

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

3 – Forward unsolicited result codes directly to the TE.

**<mt>**

0 – No received message notifications, the modem acts as an SMS client.

2 – SMS-DELIVERs (except class 2 and message waiting indication group) are routed directly to the TE.

**<bm>**

No CBM notifications are routed to the TE.

**<ds>**

0 – No SMS-STATUS-REPORTs are routed to the TE.

1 – SMS-STATUS-REPORTs are routed to the TE using unsolicited result code: `+CDS: <length><CR><LF><pdu>`.

**<bfr>**

1 – The buffer of unsolicited result codes is cleared when <mode> 1...3 is entered

Example:

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

### 10.2.3 Test command

The test command is not supported.

## 10.3 Send message, PDU mode + CMGS

The command sends a message in *PDU* mode. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.005 Ch. 3.5.1* and *3GPP 27.005 Ch. 4.3*.

### 10.3.1 Set command

The command sends a message in *PDU* mode.

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

Syntax:

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

Response syntax:

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

The set 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 *IRA* characters per octet.

**<mr>**

Message reference value.

**<ackpdu>**

RP-User-Data element of RP-ACK PDU.

<pdu> is expected to be received in the same command after <CR>. Interactive mode is not supported. PDU consists of hexadecimal numbers containing two *IRA* characters per octet.

The following command example sends the message "Testing a SMS messaging over LTE" to +358401234567, Service Center Address +448888888:

```
AT+CMGS=42<CR>069144888888F811000C915348103254760000 0B20D4F29C9E769F4161
         D0BC3D07B5CBF379F89C769F416F7B590E62D3CB<ctrl-z>
+CMGS: 2
OK
```

## 10.3.2 Read command

The read command is not supported.

## 10.3.3 Test command

The test command is not supported.

# 10.4 Received SMS notification in PDU mode +CMT

**+CMT** notifies of an unsolicited received message in *PDU* mode. *TE* is expected to ack received message with **AT+CNMA**. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.005 Ch. 3.4.1*

The notification is subscribed using the **+CNMI** command.

Syntax:

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

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.

The example returns a notification of a received message "Testing a sms messaging over lte" from +358401234567, Service Center Address +44 888 8888:

```
+CMT: "+358401234567",28<CR><LF>069144888888F8D4F29C9E769F4161D0BC3D07B5CBF379F89C
       769F416F7B590E62D3CB
```

## 10.5 Delivery status notification in PDU mode +CDS

**+CDS** notifies of an unsolicited delivery status in *PDU* mode. *TE* is expected to ack received delivery report with **AT+CNMA**. v1.0.x v1.1.x v1.2.x

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.

The example returns a delivery status notification with the recipient address, service center timestamp, and message delivery time:

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

## 10.6 New message ACK, PDU mode +CNMA

The **+CNMA** command sends an ACK in *PDU* mode. v1.0.x v1.1.x v1.2.x

> **Note:** Text mode is not supported.

For reference, see *3GPP 27.005 Ch. 4.7*.

## 10.6.1 Set command

The set command sends a new message or delivery status ACK. A client receiving unsolicited notifications for new messages and delivery status is mandated to acknowledge those. This command can be used only when the modem is activated.

> **Note:**
> - This command can 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.

Syntax:

```
+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 on page 165

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.

The following command example confirms the reception of a message, timestamp 06/11/2071 13:26:31:

```
AT+CNMA=1,9<CR>010017116031621300<ctrl-z>
OK
```

## 10.6.2 Read command

The read command is not supported.

## 10.6.3 Test command

The test command lists supported <n>s.

Response syntax:

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

**<n>**

0 – The command operates in the same way as defined for the text mode.

1 – Send RP-ACK.

2 – Send RP-ERROR.

Example:

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

# 10.7 New message ACK, text mode +CNMA

The **+CNMA** command sends a new message ACK in text mode. v1.0.x v1.1.x v1.2.x

> **Note:** Text mode is not supported.

For reference, see *3GPP 27.005 Ch. 3.4.4*.

## 10.7.1 Set command

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

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 timeout), it should respond as specified in *3GPP TS 24.011* and the UE/*TA* shall automatically disable routing to *TE* by setting both <mt> and <ds> values of **+CNMI** to zero, i.e. the SMS client gets unregistered.

Syntax:

```
+CNMA
```

Example:

```
AT+CNMA
OK
```

## 10.7.2 Read command

The read command is not supported.

## 10.7.3 Test command

The test command lists supported <n>s.

Response syntax:

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

**<n>**

0 – The command operates in the same way as defined for the text mode.

1 – Send RP-ACK

2 – Send RP-ERROR

Example:

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

# 10.8 Preferred message storage +CPMS

The **+CPMS** command selects the memory storage. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.005 Ch. 3.2.2*.

## 10.8.1 Set command

The command sets the used memory.

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

Syntax:

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

Response syntax:

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

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
```

## 10.8.2 Read command

The command is used to query memory status.

Response syntax:

```
+CPMS: <mem1>,<used1>,<total1>,<mem2>,<used2>,<total2>,<mem3>,<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?
+CPMS: "MT",0,0,"MT",0,0,"MT",0,0
OK
```

## 10.8.3 Test command

The test command lists supported memories.

Response syntax:

```
+CPMS: (list of supported <mem1>s),(list of supported <mem2>s),(list of supported <mem3>s)
```

The 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

Example:

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

# 10.9 Message service failure result code +CMS ERROR

Message service failure result code **+CMS** is sent as error response to SMS-related commands. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.005 Ch. 3.2.5*.

Response syntax:

```
+CMS ERROR: <err>
```

The parameter and the values used by common messaging commands are the following:

**<err>**

0...127 – 3GPP TS 24.011 clause E.2 values

128...255 – 3GPP TS 23.040 clause 9.2.3.22 values

300...511 – 3GPP TS 27.005 Ch. 3.2.5

512... – Manufacturer specific

513 – Manufacturer-specific cause: Not found

514 – Manufacturer-specific cause: Not allowed

515 – Manufacturer-specific cause: Memory full

# 10.10 Select SMS service +CGSMS

The **+CGSMS** command selects the SMS service. v1.0.x v1.1.x v1.2.x

For reference, see *3GPP 27.007 Ch. 10.1.21*.

## 10.10.1 Set command

The set command selects the SMS service.

Syntax:

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

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

**<service>**

1 – Circuit-switched

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

The following command example selects the circuit-switched SMS service:

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

## 10.10.2 Read command

The command reads the current SMS service.

Response syntax:

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

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

**<service>**

1 – Circuit-switched

The following command example reads the current SMS service:

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

### 10.10.3 Test command

The command lists the supported SMS services.

Response syntax:

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

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

**<service>**

1 – Circuit-switched

The following command example lists the supported SMS services:

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

## 10.11 Short message memory available %XSMMA

The Nordic-proprietary **%XSMMA** command sends an *RP-SMMA* message. v1.0.x v1.1.x v1.2.x

### 10.11.1 Set command

The set command sends an *RP-SMMA* message.

The command is a trigger for the RP-SMMA message on the SMS stack to indicate to the Service Center that the *UE* has memory available and can receive mobile-terminated short messages. The client can set a memory full situation preventing incoming SMS messages by acking a mobile-terminated short message with `AT+CNMA=2` (the *PDU* parameter has to contain cause code 22 "Memory capacity exceeded"). Cause 300 is returned for all failures.

Command syntax

```
%XSMMA
```

The following command example triggers 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
```

## 10.11.2 Read command

The read command is not supported.

## 10.11.3 Test command

The test command is not supported.

# 11 Authenticating AT command usage

The **%XSUDO** is used to authenticate AT commands.

Before you start the authentication, perform the following two steps (only once):

1. Generate private and public keys with OpenSSL:

```
openssl ecparam -name prime256v1 -genkey -noout -out [private key PEM file]
openssl ec -in [private key PEM file] -out [public key PEM file] -pubout
```

2. Write the public key with the AT command:

```
AT%XPMNG=0,"<public key>"
OK
```

To authenticate an AT command, perform the following steps:

1. Calculate an AT command signature.
   a) Create an AT command text file for an authenticated AT command:
      Example:

```
%CMNG=0,1,0,"TEST ROOT CERTIFICATE"
```

   b) Create a digest file with OpenSSL from AT command that needs authentication:

```
openssl sha256 -binary [AT command text file] > [digest file]
```

   c) Create a signature file with OpenSSL from the digest file:

```
openssl pkeyutl -sign -in [digest file] -out [signature file] -inkey [private key PEM
 file]
```

   d) Convert the signature to Base64 format:

```
base64 < [signature file] > [signature base64 file]
```

2. Write the authenticated AT command.
   The **%XSUDO** command is used to authenticate the **%CMNG** command:

   Example:

```
AT%XSUDO=35,"<signature base64>";%CMNG=0,1,0,"TEST ROOT CERTIFICATE"
OK
```

   For more information on the command, see Authenticated access %XSUDO on page 43.

# Glossary

**16-state Quadrature Amplitude Modulation (16-QAM)**

A digital modulation technique used for signals in which four bits are modulated at once by selecting one of 16 possible combinations of carrier phase shift and amplitude.

**Access Point Name (APN)**

The name of a gateway between a mobile network and another computer network, usually the Internet.

**Application Protocol Data Unit (APDU)**

The communication unit between a terminal and smart card (*UICC*).

**Binary Phase-Shift Keying (BPSK)**

A digital modulation technique used for signals in which one bit is modulated by selecting one of two possible carrier phase shifts with a 180-phase difference.

**Carrier Wave (CW)**

A single-frequency electromagnetic wave that can be modulated in amplitude, frequency, or phase to convey information.

**Cat-M1**

LTE-M User Equipment (UE) category with a single RX antenna, specified in 3GPP Release 13.

**Cat-NB1**

Narrowband Internet of Things (NB-IoT) User Equipment (UE) category with 200 kHz UE bandwidth and a single RX antenna, specified in 3GPP Release 13.

**Check Digit (CD)**

The last one-digit number of the *IMEI* code used for error detection.

**Classless Inter-domain Routing (CIDR)**

A method for allocating IP (Internet Protocol) addresses.

**CS/PS Mode of Operation**

A *UE* mode of operation. The UE may either register to packet-switched services, circuit-switched services, or both based on the mode of operation. If both are registered, the mode of operation also contains a preference for either of them.

**Discontinuous Reception (DRX)**

A method in mobile communication to conserve the battery of a mobile device by turning the RF modem in a sleep state.

**Dynamic Host Configuration Protocol (DHCP)**

A network management protocol used for automatic and centralized management of IP addresses within a network.

**Electronic Serial Number (ESN)**

A unique number embedded on a microchip for identifying mobile devices.

**EPS Mobility Management (EMM)**

The *EPS* Mobility Management (EMM) sublayer in the *NAS* protocol provides mobility service to the *UE*.

**E-UTRA Absolute Radio Frequency Channel Number (EARFCN)**

LTE carrier channel number for unique identification of LTE band and carrier frequency.

**Evolved Packet System (EPS)**

A connection-oriented transmission network in LTE (Long-term Evolution) consisting of an EPC (Evolved Packet Core) and an E-UTRAN (Evolved Terrestrial Radio Access Network).

**Extended Discontinuous Reception (eDRX)**

A method to conserve the battery of an IoT (Internet of Things) device by allowing it to remain inactive for extended periods.

**Global Navigation Satellite System (GNSS)**

A satellite navigation system with global coverage. The system provides signals from space transmitting positioning and timing data to GNSS receivers, which use this data to determine location.

**General Packet Radio Services (GPRS)**

A packet-based mobile data service for 2G and 3G mobile networks with data rates of 56-114 kbps/ second and continuous connection to the Internet.

**Global Positioning System (GPS)**

A satellite-based radio navigation system that provides its users with accurate location and time information over the globe.

**Integrated Circuit Card Identifier (ICCID)**

A unique serial number of a SIM card.

**International Mobile (Station) Equipment Identity (IMEI)**

A unique code consisting of 14 digits and a check digit for identifying 3GPP-based mobile devices.

**International Mobile (Station) Equipment Identity, Software Version (IMEISV)**

A unique code consisting of 16 decimal digits and two software version digits for identifying 3GPP-based mobile devices.

**International Mobile Subscriber Identity (IMSI)**

A unique code, usually 15 digits, used for the identification of a mobile subscriber and consisting of an *MCC*, *MNC*, and MSIN (Mobile Subscription Identification Number).

**International Reference Alphabet (IRA)**

A seven-bit coded character set for information exchange.

**Low-Noise Amplifier (LNA)**

In a radio receiving system, an electronic amplifier that amplifies a very low-power signal without significantly degrading its signal-to-noise ratio.

**Maximum Transmission Unit (MTU)**

The largest packet or frame that can be sent in a single network layer transaction.

**MIPI RF Front-End Control Interface (RFFE)**

A dedicated control interface for the RF front-end subsystem. *MIPI Alliance*

**Mobile Country Code (MCC)**

A unique three-digit part of an IMSI code identifying the country of domicile of the mobile subscriber. MCC is used together with the Mobile Network Code (MNC).

**Mobile Equipment (ME)**

The physical *UE* consisting of one of more *MT* and one or more *TE*.

**Mobile Network Code (MNC)**

A code identifying the telecommunications network. The code is defined by ITU-T Recommendation E.212, consists of two or three decimal digits, and is used together with the Mobile Country Code (MCC).

**Mobile Station International Subscriber Directory Number (MSISDN)**

A number consisting of a maximum of 15 digits identifying a mobile subscriber by mapping the telephone number to the *SIM* card in a phone.

**Mobile Termination (MT)**

A component of the Mobile Equipment (ME) performing functions specific to management of the radio interface. The R interface between *TE* and MT uses the AT command set. The *IMEI* code is attached to the MT.

**Non-access Stratum (NAS)**

In telecom protocol stacks, the highest stratum of the control plane between the core network and *UE*. The layer is used to manage the establishment of communication sessions and for maintaining communications with the UE as it moves.

**Non-access Stratum (NAS) Signalling Low Priority Indication (NSLPI)**

Used by the network for NAS-level mobility management congestion control.

**Non-volatile Memory (NVM)**

Memory that can retrieve stored information even after having been power-cycled.

**Packet Data Network (PDN)**

A network that provides data services.

**Packet Data Protocol (PDP)**

A packet transfer protocol in wireless GPRS (General Packet Radio Services) and HSDPA (High-speed Downlink Packet Access) networks.

**Packet Data Protocol (PDP) Context**

In UMTS (Universal Mobile Telecommunications System) and GPRS (General Packet Radio Service), the record that specifies *UE* access to an external packet-switched network.

**Paging Time Window (PTW)**

The period of time during which the User Equipment (UE) attempts to receive a paging message.

**Personal Identification Number (PIN)**

An optional security feature in mobile devices used for identifying a user. PIN is a numeric code which must be entered each time a mobile device is started.

**Personal Unblocking Key (PUK)**

A digit sequence required in 3GPP mobile phones to unlock a *SIM* that has disabled itself after an in correct personal identification number has been entered multiple times.

**Power Amplifier (PA)**

A device used to increase the transmit power level of a radio signal.

**Power Saving Mode (PSM)**

A feature introduced in 3GPP Release 12 to improve battery life of IoT (Internet of Things) devices by minimizing energy consumption. The device stays dormant during the PSM window.

**Pre-shared Key (PSK)**

A password authentication method, a string of text, expected before a username and password to establish a secured connection. Also known as a "shared secret".

**Printed Circuit Board (PCB)**

A board that connects electronic components.

**Privacy Enhanced Mail (PEM)**

A public key certificate defined in the X.509 cryptography standard and used to privately transmit email.

**Protocol Configuration Options (PCO)**

An element of *NAS* message used for transferring parameters between the *UE* and the P-GW (Packet Data Network Gateway).

**Protocol Data Unit (PDU)**

Information transferred as a single unit between peer entities of a computer network and containing control and address information or data. PDU mode is one of the two ways of sending and receiving SMS messages.

**PS Mode of Operation**

A *UE* mode of operation. The UE registers only to *EPS* services.

**Public Land Mobile Network (PLMN)**

A network that provides land mobile telecommunications services to the public. A PLMN is identified by the *MCC* and *MNC*.

**Quadrature Phase-Shift Keying (QPSK)**

A digital modulation technique used for signals in which two bits are modulated at once, selecting one of four possible carrier phase shifts.

**Quality of Service (QoS)**

The measured overall performance of a service, such as a telephony or computer network, or a cloud computing service.

**Radio Policy Manager (RPM)**

A radio baseband chipset feature that protects the mobile network from signaling overload.

**Reference Signal Received Power (RSRP)**

The average power level received from a single reference signal in an LTE (Long-Term Evolution) network.

**Reference Signal Received Quality (RSRQ)**

The quality of a single reference signal received in an LTE (Long-Term Evolution) network and calculated from *RSRP*.

**Resource Block (RB)**

The smallest unit of resources that can be allocated to a user.

**RP-SMMA**

A message sent by the User Equipment to relay a notification to the network that the mobile has memory available to receive one or more short messages.

**Serial Number (SNR)**

A unique six-digit number part of the *IMEI* code identifying each equipment within each *TAC*.

**Signal-to-Noise Ratio (SNR)**

The level of signal power compared to the level of noise power, often expressed in decibels (dB).

**Software Version Number (SVN)**

Part of the *IMEI* code identifying the revision of the software installed on a mobile device.

**Subscriber Identity Module (SIM)**

A card used in *UE* containing data for subscriber identification.

**System in Package (SiP)**

A number of integrated circuits, often from different technologies, enclosed in a single module that performs as a system or subsystem.

**System on Chip (SoC)**

A microchip that integrates all the necessary electronic circuits and components of a computer or other electronic systems on a single integrated circuit.

**Terminal Adapter (TA)**

A device that connects a *UE* to a communications network. In mobile networks, the terminal adapter is used by the terminal equipment to access the mobile termination using AT commands.

**Terminal Equipment (TE)**

Communications equipment at either end of a communications link, used to permit the stations involved to accomplish the mission for which the link was established.

**Tracking Area Code (TAC)**

A unique code used to identify a tracking area within a particular network.

**Tracking Area Update (TAU)**

A procedure initiated by the *UE* when moving to a new tracking area in the LTE (Long-term Evolution) system.

**Type Allocation Code (TAC)**

The initial eight-digit part of an *IMEI* code used for identifying the model of a mobile phone.

**Universal Asynchronous Receiver/Transmitter (UART)**

A hardware device for asynchronous serial communication between devices.

**User Equipment (UE)**

Any device used by an end-user to communicate. The UE consists of the Mobile Equipment (ME) and the Universal Integrated Circuit Card (UICC).

**Universal Integrated Circuit Card (UICC)**

A new generation *SIM* used in *UE* for ensuring the integrity and security of personal data.

**Unique Slave Identifier (USID)**

A unique address for identifying each slave device in an RFFE (RF Front-End) system.

**Universal Subscriber Identity Module (USIM)**

A card used in *UE* containing data for subscriber identification.

# Acronyms and abbreviations

These acronyms and abbreviations are used in this document.

**16-QAM**
16-state Quadrature Amplitude Modulation

**APN**
Access Point Name

**APDU**
Application Protocol Data Unit

**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

**GNSS**
Global Navigation Satellite System

**GPS**
Global Positioning System

**GPRS**
General Packet Radio Services

**ICCID**
Integrated Circuit Card Identifier

**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

**MIPI RFFE**
MIPI RF Front-End Control Interface

**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

**PA**
Power Amplifier

**PCB**
Printed Circuit Board

**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

**RPM**
Radio Policy Manager

**RSRP**
Reference Signal Received Power

**RSRQ**
Reference Signal Received Quality

**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

**USIM**
Unique Slave Identifier

**USIM**
Universal Subscriber Identity Module

**UUID**
Universally Unique Identifier

# Legal notices

By using this documentation you agree to our terms and conditions of use. Nordic Semiconductor may change these terms and conditions at any time without notice.

## Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function, or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Nordic Semiconductor ASA does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. If there are any discrepancies, ambiguities or conflicts in Nordic Semiconductor's documentation, the Product Specification prevails.

Nordic Semiconductor ASA reserves the right to make corrections, enhancements, and other changes to this document without notice.

## Life support applications

Nordic Semiconductor products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury.

Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

## RoHS and REACH statement

Complete hazardous substance reports, material composition reports and latest version of Nordic's REACH statement can be found on our website www.nordicsemi.com.

## Trademarks

All trademarks, service marks, trade names, product names, and logos appearing in this documentation are the property of their respective owners.

## Copyright notice

© 2020 Nordic Semiconductor ASA. All rights are reserved. Reproduction in whole or in part is prohibited without the prior written permission of the copyright holder.

COMPANY WITH
QUALITY SYSTEM
CERTIFIED BY DNV GL
= ISO 9001 =