# BC680Z-EU
# AT Commands Manual

**NB-IoT Module Series**

Version: 1.0

Date: 2024-12-26

Status: Released

**At Quectel, our aim is to provide timely and comprehensive services to our customers. If you require any assistance, please contact our headquarters:**

**Quectel Wireless Solutions Co., Ltd.**
Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai 200233, China
Tel: +86 21 5108 6236
Email: info@quectel.com

**Or our local offices. For more information, please visit:**
http://www.quectel.com/support/sales.htm.

**For technical support, or to report documentation errors, please visit:**
http://www.quectel.com/support/technical.htm.
Or email us at: support@quectel.com.

# Legal Notices

We offer information as a service to you. The provided information is based on your requirements and we make every effort to ensure its quality. You agree that you are responsible for using independent analysis and evaluation in designing intended products, and we provide reference designs for illustrative purposes only. Before using any hardware, software or service guided by this document, please read this notice carefully. Even though we employ commercially reasonable efforts to provide the best possible experience, you hereby acknowledge and agree that this document and related services hereunder are provided to you on an “as available” basis. We may revise or restate this document from time to time at our sole discretion without any prior notice to you.

# Use and Disclosure Restrictions

## License Agreements

Documents and information provided by us shall be kept confidential, unless specific permission is granted. They shall not be accessed or used for any purpose except as expressly provided herein.

## Copyright

Our and third-party products hereunder may contain copyrighted material. Such copyrighted material shall not be copied, reproduced, distributed, merged, published, translated, or modified without prior written consent. We and the third party have exclusive rights over copyrighted material. No license shall be granted or conveyed under any patents, copyrights, trademarks, or service mark rights. To avoid ambiguities, purchasing in any form cannot be deemed as granting a license other than the normal non-exclusive, royalty-free license to use the material. We reserve the right to take legal action for noncompliance with abovementioned requirements, unauthorized use, or other illegal or malicious use of the material.

## Trademarks

Except as otherwise set forth herein, nothing in this document shall be construed as conferring any rights to use any trademark, trade name or name, abbreviation, or counterfeit product thereof owned by Quectel or any third party in advertising, publicity, or other aspects.

## Third-Party Rights

This document may refer to hardware, software and/or documentation owned by one or more third parties ("third-party materials"). Use of such third-party materials shall be governed by all restrictions and obligations applicable thereto.

We make no warranty or representation, either express or implied, regarding the third-party materials, including but not limited to any implied or statutory, warranties of merchantability or fitness for a particular purpose, quiet enjoyment, system integration, information accuracy, and non-infringement of any third-party intellectual property rights with regard to the licensed technology or use thereof. Nothing herein constitutes a representation or warranty by us to either develop, enhance, modify, distribute, market, sell, offer for sale, or otherwise maintain production of any our products or any other hardware, software, device, tool, information, or product. We moreover disclaim any and all warranties arising from the course of dealing or usage of trade.

# Privacy Policy

To implement module functionality, certain device data are uploaded to Quectel's or third-party's servers, including carriers, chipset suppliers or customer-designated servers. Quectel, strictly abiding by the relevant laws and regulations, shall retain, use, disclose or otherwise process relevant data for the purpose of performing the service only or as permitted by applicable laws. Before data interaction with third parties, please be informed of their privacy and data security policy.

# Disclaimer

a) We acknowledge no liability for any injury or damage arising from the reliance upon the information.
b) We shall bear no liability resulting from any inaccuracies or omissions, or from the use of the information contained herein.
c) While we have made every effort to ensure that the functions and features under development are free from errors, it is possible that they could contain errors, inaccuracies, and omissions. Unless otherwise provided by valid agreement, we make no warranties of any kind, either implied or express, and exclude all liability for any loss or damage suffered in connection with the use of features and functions under development, to the maximum extent permitted by law, regardless of whether such loss or damage may have been foreseeable.
d) We are not responsible for the accessibility, safety, accuracy, availability, legality, or completeness of information, advertising, commercial offers, products, services, and materials on third-party websites and third-party resources.

***Copyright © Quectel Wireless Solutions Co., Ltd. 2024. All rights reserved.***

# About the Document

## Revision History

| Version | Date | Author | Description |
|---|---|---|---|
| - | 2023-11-14 | Sven DING | Creation of the document |
| 1.0 | 2024-12-26 | Howie DONG | First official release |

# Contents

**About the Document** ..... **3**
**Contents** ..... **4**
**Table Index** ..... **7**

**1 Introduction** ..... **8**
1.1. Definitions ..... 9
1.2. AT Command Syntax ..... 9
1.3. AT Command Responses ..... 10
1.4. Declaration of AT Command Examples ..... 11
1.5. 3GPP Compliance ..... 11

**2 General Commands** ..... **12**
2.1. ATI Display Product Identification Information ..... 12
2.2. ATE Set Command Echo Mode ..... 13
2.3. AT+CGMI Request Manufacturer Identification ..... 14
2.4. AT+CGMM Request Manufacturer Model ..... 14
2.5. AT+CIMI Request IMSI ..... 15
2.6. AT+CGSN Request IMEI Information ..... 16
2.7. AT+CGMR Request Software Version Number ..... 18
2.8. AT+CFUN Set UE Functionality ..... 19
2.9. AT+CMEE Report UE Error ..... 20
2.10. AT+NRB Reboot UE ..... 21
2.11. AT+CEER Extended Error Report ..... 22
2.12. AT+CLAC List Available Commands ..... 23
2.13. AT+CCLK Return Current Date and Time ..... 24
2.14. AT+NITZ Set Time Update Mode ..... 25
2.15. AT+CPIN Enter PIN ..... 26
2.16. AT+CPINR Query Remaining PIN Retries ..... 27
2.17. AT+QLEDMODE Configure Network-Status-Indication Light ..... 29

**3 (U)SIM Card Related Commands** ..... **31**
3.1. AT+CRSM Restricted (U)SIM Access ..... 31
3.2. AT+CSIM Generic (U)SIM Access ..... 32
3.3. AT+CCHO Open Logical Channel ..... 34
3.4. AT+CCHC Close Logical Channel ..... 35
3.5. AT+CGLA Generic UICC Logical Channel Access ..... 36
3.6. AT+NUICC Set UICC Power Mode ..... 37

**4 SMS Function Commands** ..... **39**
4.1. AT+CMGS Send SMS Messages ..... 39
4.2. AT+CSCA Service Centre Address ..... 40
4.3. AT+CSMS Select Message Service ..... 41
4.4. AT+CMGF Message Format ..... 42

4.5. AT+CNMA New Message Acknowledgement ..... 43
4.6. AT+CMGC Send SMS Command ..... 45
4.7. AT+CMMS Send More Messages ..... 45
4.8. +CMT Report SMS-DELIVER ..... 47

**5 Network Status Related Commands ..... 48**
5.1. AT+COPS Operator Selection ..... 48
5.2. AT+CPSMS Power Saving Mode Setting ..... 50
5.3. AT+CEDRXS eDRX Setting ..... 52
5.4. AT+CEREG EPS Network Registration Status ..... 55
5.5. AT+CTZR Time Zone Reporting ..... 58
5.6. AT+CSQ Signal Quality Report ..... 60
5.7. AT+NPTWEDRXS PTW and eDRX Setting ..... 61
5.8. AT+CESQ Extended Signal Quality ..... 64

**6 PS Commands ..... 67**
6.1. AT+CGDCONT Define PDP Context ..... 67
6.2. AT+CGATT PS Attach or Detach ..... 69
6.3. AT+CGACT PDP Context Activate/Deactivate ..... 71
6.4. AT+NCPCDPR Configure PDP Context Dynamic Parameters to be Read ..... 72
6.5. AT+CSODCP Sending Originating Data via Control Plane ..... 74
6.6. AT+NQPODCP Query Pending Originating Data List via the Control Plane ..... 75
6.7. AT+CRTDCP Report CP Data ..... 76
6.8. AT+CSCON Signalling Connection Status ..... 78
6.9. AT+NIPINFO IP Address Information Report ..... 79
6.10. AT+CGPADDR Show PDP Addresses ..... 81
6.11. AT+CGAPNRC APN Rate Control ..... 82
6.12. AT+CEDRXRDP Read Dynamic eDRX Parameters ..... 84
6.13. AT+CCIOTOPT CIoT Optimization Configuration ..... 86
6.14. AT+CGCONTRDP Read PDP Context Dynamic Parameters ..... 88
6.15. AT+CIPCA Initial PDP Context Activation ..... 90
6.16. AT+CGAUTH Define PDP Context Authentication Parameters ..... 91
6.17. AT+CNMPSD No More PS Data ..... 93
6.18. AT+NCIDSTATUS Get <cid> Status ..... 94
6.19. AT+NGACTR Report PDP Context Activation/Deactivation Result ..... 95
6.20. AT+NSNPD Send Non-IP Data ..... 97
6.21. AT+NRNPDM Set Reporting of Non-IP Data Mode ..... 98
6.22. AT+NQPNPD Query Pending Non-IP Data List ..... 100

**7 Socket Commands ..... 102**
7.1. AT+NSOCR Create a Socket ..... 102
7.2. AT+NSOST Send Data (UDP Only) ..... 103
7.3. AT+NSOSTF Send Messages with Flags (UDP Only) ..... 105
7.4. AT+NQSOS Query the List of Pending Socket Message ..... 106
7.5. AT+NSONMI Set Reporting Mode of Downlink Data ..... 108
7.6. AT+NSORF Receive Data ..... 110

7.7. AT+NSOCO Connect Command (TCP Only) ..... 111
7.8. AT+NSOSD Send Message (TCP Only) ..... 112
7.9. AT+NSOCL Close a Socket ..... 114
7.10. AT+NSOSTATUS Get Socket Status ..... 115
7.11. AT+QIDNSCFG Configure DNS ..... 117
7.12. AT+QDNS Trigger DNS Domain Name Resolution ..... 118
7.13. AT+NSOSTEX Send Messages (UDP Only) ..... 119
7.14. AT+NSOSTFEX Send Messages with Flags (UDP Only) ..... 122
7.15. AT+NSOSDEX Send Messages (TCP Only) ..... 124
7.16. Matters Needing Attention ..... 127

**8 Protocol extension Command ..... 128**
8.1. AT+NEARFCN Specify EARFCN ..... 128
8.2. AT+NCSEARFCN Clear Stored EARFCN ..... 129
8.3. AT+NBAND Set Supported Bands ..... 130
8.4. AT+NCONFIG Configure UE Behaviour ..... 131
8.5. AT+NUESTATS Query UE Statistics ..... 135
8.6. AT+NCCID USIM Card Identification ..... 141
8.7. AT+NSET Extend the Configuration ..... 142
8.8. AT+NPOWERCLASS Set the Mapping for Band and Power Class ..... 144
8.9. AT+QPLMNS Search PLMN ..... 146

**9 Platform Extended Commands ..... 148**
9.1. AT+NPSMR Report Power Saving Mode Status ..... 148
9.2. AT+QCHIPINFO Read System Information ..... 149
9.3. AT+NPING Test IP Network Connectivity to a Remote Host ..... 150
9.4. AT+NPINGSTOP Stop Sending Ping Packets ..... 152
9.5. AT+NATSPEED Configure Baud Rate of UART Port ..... 152
9.6. AT+NPIN PIN Operations ..... 154
9.7. AT+QCFG Extended Configuration Settings ..... 156
9.7.1. AT+QCFG="slplocktimes" Configure Countdown to Entering Sleep Mode ..... 156

**10 Error Codes ..... 158**

**11 Reboot Reasons ..... 163**

**12 Examples ..... 164**
12.1. Network Attachment ..... 164
12.1.1. Automatic Network Attachment ..... 164
12.1.2. Manual Network Attachment ..... 165
12.2. Send/Receive/Read a UDP Message ..... 166
12.3. Release Assistance Indication（RAI） ..... 166

**13 Appendix A AT Command Delay Response Description ..... 169**
**14 Appendix B Terms and Abbreviations ..... 170**

# Table Index

Table 1: Unsolicited Result Code for Modules Update State Indication ..... 8
Table 2: Types of AT Commands ..... 9
Table 3: General Error Codes (27.007) ..... 158
Table 4: General Error Codes (27.005) ..... 159
Table 5: Platform-level Extended Error Codes ..... 160
Table 6: Error Codes in +CME ERROR ..... 160
Table 7: Reboot Reasons ..... 163
Table 8: Terms and Abbreviations ..... 170

# 1 Introduction

This document provides details of the AT commands supported by Quectel NB-IoT BC680Z-EU module.

After the module is booted, the string **<CR><LF>OK<CR><LF>OK<CR><LF>** will be output. After this string has been received, the AT command processor will be ready to receive AT commands.

In the case that an external MCU intervenes in the process of firmware update via DFOTA, the module will output URC to inform the current update state of the external MCU.

For "This command takes effect after the module is rebooted" mentioned in this document, the rebooting method is to execute **AT+NRB**.

**Table 1: Unsolicited Result Code for Modules Update State Indication**

| URC | Description |
|---|---|
| **<CR><LF>FIRMWARE DOWNLOADING<CR><LF>** | Indicates that the module is downloading update package. |
| **<CR><LF>FIRMWARE DOWNLOAD FAILED<CR><LF>** | Indicates that the download failed. |
| **<CR><LF>FIRMWARE DOWNLOADED<CR><LF>** | Indicates that the download finished. |
| **<CR><LF>FIRMWARE UPDATING<CR><LF>** | Indicates that the module is updating. |
| **<CR><LF>FIRMWARE UPDATE SUCCESS<CR><LF>** | Indicates that the update is successful, but update state has not been reported to firmware package server yet. |
| **<CR><LF>FIRMWARE UPDATE FAILED<CR><LF>** | Indicates that the update failed. |
| **<CR><LF>FIRMWARE UPDATE OVER<CR><LF>** | Indicates that the update has been completed, and the update state has been reported to firmware package server. Update is done after this URC. |

**NOTE**

During updating procedure, the module should not be operated until **FIRMWARE UPDATE OVER** is reported, otherwise an error will occur.

## 1.1. Definitions

- **<CR>** Carriage return character.
- **<LF>** Line feed character.
- **<...>** Parameter name. Angle brackets do not appear on the command line.
- **[...]** Optional parameter of a command or an optional part of TA information response. Square brackets do not appear on the command line. When an optional parameter is not given in a command, the new value equals to its previous value or the default settings, unless otherwise specified.
- **Underline** Default setting of a parameter.

## 1.2. AT Command Syntax

All command lines must start with **AT** or **at** and end with **<CR>**. Information responses and result codes always start and end with a carriage return character and a line feed character: **<CR><LF><response><CR><LF>**. In tables presenting commands and responses throughout this document, only the commands and responses are presented, and **<CR>** and **<LF>** are deliberately omitted.

AT commands implemented by BC680Z-EU fall into two categories syntactically: “Basic” and “Extended”, as listed below:

- **Basic Command**

Basic command format is **AT<x><n>**, or **AT&<x><n>**, where **<x>** is the command, and **<n>** is/are the argument(s) of the command. For example, **ATE<n>** tells the DCE (Data Circuit-terminating Equipment) whether received characters should be echoed back to the DTE (Data Terminal Equipment) according to the value of **<n>**. **<n>** is optional and a default will be used if it is omitted.

- **Extended Command**

There are several types of extended commands as shown in the following table.

**Table 2: Types of AT Commands**

| Command Type | Syntax | Description |
|---|---|---|
| Test Command | **AT+<cmd>=?** | Test the existence of the corresponding command and return information about the type, value, or range of its parameter. |
| Read Command | **AT+<cmd>?** | Check the current parameter value of the corresponding command. |

| | | |
|---|---|---|
| Write Command | **AT+<cmd>=<p1>[,<p2>[,<p3>[...]]]** | Set user-definable parameter value. |
| Execution Command | **AT+<cmd>** | Return a specific information parameter or perform a specific action. |

Multiple commands can be placed on a single line using a semi-colon (**;**) between commands. In such cases, only the first command should have **AT** prefix. Commands can be in upper or lower case.

Spaces should be ignored when you enter AT commands, except in the following cases:

- Within quoted strings, where spaces are preserved;
- Within an unquoted string or numeric parameter;
- Within an IP address;
- Within the AT command name up to and including a **=**, **?** or **=?**.

On input, at least a carriage return is required. A newline character is ignored so it is permissible to use carriage return/line feed pairs on the input.

If no command is entered after the **AT** token, **OK** will be returned. If an invalid command is entered, **ERROR** will be returned.

Optional parameters, unless explicitly stated, need to be provided up to the last parameter being entered.

## 1.3. AT Command Responses

When the AT command processor has finished processing a line, it will output **OK**, **ERROR** or **+CME ERROR: <err>** to indicate that it is ready to accept a new command. Solicited information responses are sent before the final **OK**, **ERROR** or **+CME ERROR: <err>**.

Responses will be in the format of:

**<CR><LF>+CMD1:<parameters><CR><LF>**
**<CR><LF>OK<CR><LF>**

Or

**<CR><LF><parameters><CR><LF>**
**<CR><LF>OK<CR><LF>**

## 1.4. Declaration of AT Command Examples

The AT command examples in this document are provided to help you learn about the use of the AT commands introduced herein. The examples, however, should not be taken as Quectel's recommendations or suggestions about how to design a program flow or what status to set the module into. Sometimes multiple examples may be provided for one AT command. However, this does not mean that there is a correlation among these examples, or that they should be executed in a given sequence. The URLs, domain names, IP addresses, usernames/accounts, and passwords (if any) in the AT command examples are provided for illustrative and explanatory purposes only, and they should be modified to reflect your actual usage and specific needs.

## 1.5. 3GPP Compliance

3GPP commands are compliant with the *3GPP TS 27.007 v14.3.0 (2017-03)*.

# 2 General Commands

## 2.1. ATI Display Product Identification Information

The Execution Command returns product identification information.

| ATI Display Product Identification Information | |
|---|---|
| Execution Command<br>**ATI** | Response<br>**Quectel**<br>**<objectID>**<br>**Revision:<revision>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<objectID>** | String type. Identifier of device type. |
| **<revision>** | String type. Software revision number. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
ATI
Quectel
BC680Z-EU
Revision:BC680ZEU04AAR01A01

OK
```

## 2.2. ATE Set Command Echo Mode

The Execution Command determines whether the UE echoes characters will be received from external MCU.

| ATE Set Command Echo Mode | |
|---|---|
| Execution Command<br>**ATE<value>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<value>** | Integer type.<br>0 Turn off Echo mode<br>1 Turn on Echo mode |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
ATE1
OK
ATI
Quectel
BC680Z-EU
Revision:BC680ZEU04AAR01A01

OK
```

## 2.3. AT+CGMI Request Manufacturer Identification

The Execution Command returns manufacturer information. By default, it will return **Quectel**.

| AT+CGMI Request Manufacturer Identification | |
|---|---|
| Execution Command<br>**AT+CGMI** | Response<br>**<manufacturer>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Test Command<br>**AT+CGMI=?** | Response<br>**OK** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<manufacturer>** | String type. Manufacturer information. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CGMI
Quectel

OK
```

## 2.4. AT+CGMM Request Manufacturer Model

The Execution Command returns manufacturer model information.

| AT+CGMM Request Manufacturer Model | |
|---|---|
| Test Command<br>**AT+CGMM=?** | Response<br>**OK** |

| | |
|---|---|
| Execution Command<br>**AT+CGMM** | Response<br>**<model>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<model>** | String type. Manufacturer model information. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CGMM
Quectel_BC680Z-EU

OK
```

## 2.5. AT+CIMI Request IMSI

This command returns IMSI (International Mobile Subscriber Identity) of (U)SIM card.

| **AT+CIMI Request IMSI** | |
|---|---|
| Execution Command<br>**AT+CIMI** | Response<br>**<IMSI>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Test Command<br>**AT+CIMI=?** | Response<br>**OK** |

| | |
|---|---|
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<IMSI>** | String without double quotes, in decimal. IMSI value stored in (U)SIM card. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CIMI
460111174746006

OK
```

## 2.6. AT+CGSN Request IMEI Information

This command returns the IMEI number and related information. The Test Command returns the range of supported **<snt>**.

| **AT+CGSN Request IMEI Information** | |
|---|---|
| Test Command<br>**AT+CGSN=?** | Response<br>When UE supports **<snt>** and the command is executed successfully:<br>**+CGSN:(**list of supported **<snt>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution/Write Command<br>**AT+CGSN[=<snt>]** | Response<br>When **<snt>**=0 (or omitted) and the command is executed successfully:<br>**<sn>**<br><br>**OK**<br><br>When **<snt>**=1 and the command is executed successfully: |

| | |
|---|---|
| | **+CGSN:<imei>**<br><br>**OK**<br><br>When **<snt>**=2 and the command is executed successfully:<br>**+CGSN:<imeisv>**<br><br>**OK**<br><br>When **<snt>**=3 and the command is executed successfully:<br>**+CGSN:<svn>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<snt>** | Integer type. The type that has been requested.<br>0 Returns SN, product serial number<br>1 Returns the IMEI number<br>2 Returns the IMEISV number<br>3 Returns the SVN (Software Version Number) |
| **<sn>** | String type. Product serial number. Also known as machine code, authentication code, registration application code, etc., it is used to identify and verify the "legal identity of the product". The maximum length of SN code is 64 digits and consists of numbers, uppercase and lowercase letters. |
| **<imei>** | String type. IMEI number. |
| **<imeisv>** | String type. IMEISV number. |
| **<svn>** | String type. SVN value, 2 bytes in length. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CGSN=1                          //Request the IMEI number.
+CGSN:869213050001188
```

```
OK
```

## 2.7. AT+CGMR Request Software Version Number

This command returns the software version number.

| AT+CGMR Request Software Version Number | |
|---|---|
| Test Command<br>**AT+CGMR=?** | Response<br>**OK** |
| Execution Command<br>**AT+CGMR** | Response<br>**Software Version:<software_version>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<mfr_revision>** | Software version number. See example for details. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CGMR
Software Version:V1200FB10002R00C0003

OK
```

## 2.8. AT+CFUN Set UE Functionality

This command sets UE mode or reboots UE.

The Read Command returns the current setting of **<fun>**.

The Test Command returns values supported by the UE.

| AT+CFUN Set UE Functionality | |
|---|---|
| Test Command<br>**AT+CFUN=?** | Response<br>**+CFUN:(**list of supported **<fun>**s**),(**list of supported **<rst>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CFUN?** | Response<br>**+CFUN:<fun>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CFUN=<fun>[,<rst>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 10 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<fun>** | Integer type. UE functionality level.<br>0 Minimum functionality |

| | | |
|---|---|---|
| | 1 | Full functionality |
| | 5 | Local quick shutdowns (no network interaction), enter minimum functionality mode |
| **<rst>** | Integer type. Whether to reboot before **<fun>** setting. | |
| | 0 | Not reboot |
| | 1 | Reboot |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

### Example

```
AT+CFUN=0
OK
AT+CFUN?
+CFUN:0

OK
```

**NOTE**

The maximum response time of **AT+CFUN=0** may exceed 5 seconds and not exceed 10 seconds.

## 2.9. AT+CMEE Report UE Error

The Write Command disables or enables the use of final result code **+CME ERROR:<err>** as an indication of an error relating to the functionality of the UE. When enabled, UE related errors cause **+CME ERROR:<err>** final result code instead of the regular **ERROR** final result code. When disabled, **ERROR** is returned normally when error is related to syntax, invalid parameters or UE functionality.

| **AT+CMEE Report UE Error** | |
|---|---|
| Test Command<br>**AT+CMEE=?** | Response<br>**+CMEE:(**list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CMEE?** | Response<br>**+CMEE:<n>** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CMEE=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configuration is not saved. |

### Parameter

| | |
|---|---|
| **<n>** | Integer type. Error modes.<br>0 Disable **+CME ERROR:<err>** result code and use **ERROR** instead<br>1 Enable **+CME ERROR:<err>** result code and use numeric **<err>** values |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

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

OK
```

## 2.10. AT+NRB Reboot UE

This command controls the module to enter software rebooting process.

| **AT+NRB Reboot the UE** | |
|---|---|
| Execution Command | Response |

| AT+NRB | REBOOTING |
|---|---|
| Maximum Response Time | 1 s |
| Characteristics | - |

## 2.11. AT+CEER Extended Error Report

The Execution Command queries the cause for the following errors:

- The latest failure in the PDP context deactivation/activation.
- The latest failure in the PDP context deregistration.

| AT+CEER Extended Error Report | |
|---|---|
| Test Command<br>**AT+CEER=?** | Response<br>**OK** |
| Execution Command<br>**AT+CEER** | Response<br>**[+CEER:<report>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<report>** | String type. Description of the cause of the error. The length is less than 100 bytes. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CEER
+CEER:No suitable cells in tracking area

OK
```

## 2.12. AT+CLAC List Available Commands

This command lists available commands supported by the protocol stack, including 3GPP standard AT commands and extended AT commands.

| AT+CLAC List Available Commands | |
|---|---|
| Test Command<br>**AT+CLAC=?** | Response<br>**OK** |
| Execution Command<br>**AT+CLAC** | Response<br>**<AT_command>**<br><br>**[<AT_command>]**<br><br>**[...]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<AT_command>** | Defines the AT command including the prefix AT. Text shall not contain the sequence **0<CR>** or **OK<CR>**. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CLAC
AT+CGSN

AT+CEREG

AT+CGATT
…
AT+COPS
```

```
AT+NCONFIG

AT+NCCID
...

OK
```

**NOTE**

This command only displays the list of AT commands supported by the protocol stack.

## 2.13. AT+CCLK Return Current Date and Time

This command returns the current date and time.

| AT+CCLK Return Current Date and Time | |
|---|---|
| Read Command<br>**AT+CCLK?** | Response<br>**+CCLK:<time>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CCLK=<time>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configuration is not saved. |

### Parameter

| | |
|---|---|
| **<time>** | String type. The format is “yy/MM/dd,hh:mm:ss±zz”, where characters indicate year (two last digits), month, day, hour, minute, second and time zone (indicates the difference, expressed in quarters of an hour, between the local time and GMT). The value of year does not exceed 99. If the value is 0–69, it corresponds to the year 2000–2069, and if the value is 70–99, it corresponds to the year 1970–1999, where ±*zz* is a sub-time zone, and each sub-time zone is 15 minutes. For instance, "19/05/06,22:10:00+08" represents May 6, 2019, 22:10:00 GMT+2 hours. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NITZ=0
OK
AT+CCLK="18/11/09,05:36:42+32"
OK
AT+CCLK?
+CCLK:18/11/09,05:36:45+32

OK
```

## 2.14. AT+NITZ Set Time Update Mode

This command sets the time update mode. When **AT+NITZ=1**, **AT+CCLK** cannot be used for setting, but can be used for query.

| **AT+NITZ Set Time Update Mode** | |
|---|---|
| Test Command<br>**AT+NITZ=?** | Response<br>**+NITZ:(**list of supported **<mode>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NITZ?** | Response<br>**+NITZ:<mode>**<br><br>**OK** |

| | |
|---|---|
| | If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NITZ=<mode>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configuration is not saved. |

### Parameter

| | |
|---|---|
| **<mode>** | Integer type. Time update mode.<br>0 Update time via local<br>1 Update time via NITZ |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NITZ=0
OK
```

## 2.15. AT+CPIN Enter PIN

The Write Command verifies and unlocks the PIN. If the current password input request is PIN, enter **AT+CPIN=<pin>** for verification. If the current password input request is PUK, enter **AT+CPIN=<pin>,<newpin>** to unlock. The first parameter **<pin>** is the (U)SIM PUK code, and the second parameter **<newpin>** is the new PIN code. The Read Command is used to indicate whether there is a password entry request.

| **AT+CPIN Enter PIN** | |
|---|---|
| Test Command<br>**AT+CPIN=?** | Response<br>**OK** |
| Read Command<br>**AT+CPIN?** | Response<br>**+CPIN: <code>** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CPIN=<pin>[,<newpin>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<pin>** | String type. PIN or PUK. |
| **<newpin>** | String type. New PIN. |
| **<code>** | String type without double quotes.<br>READY No password requests<br>SIM PIN UICC/SIM PIN requests<br>SIM PUK UICC/SIM PUK requests |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CPIN?
+CPIN: READY

OK
```

## 2.16. AT+CPINR Query Remaining PIN Retries

The Write Command returns the number of remaining PIN retries for the UE passwords with intermediate result code **+CPINR:<code>,<retries>[,<default_retries>]**. A **<code>** result specified by the **<sel_code>** will be returned. When a parameter **<sel_code>** is omitted, all supported **<code>** results are returned.

| AT+CPINR Query Remaining PIN Retries | |
|---|---|
| Test Command<br>**AT+CPINR=?** | Response<br>**+CPINR:(**list of supported **<code>**s**)**<br><br>**OK** |
| Write Command<br>**AT+CPINR[=<sel_code>]** | Response<br>**[+CPINR: <code>,<retries>[,<default_retries>]]**<br>**[…]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<code>** | String type. Type of PIN. Supports SIM PIN, SIM PUK, SIM PIN2 and SIM PUK2. |
| **<retries>** | Integer type. Number of remaining retries per PIN. |
| **<default_retries>** | Integer Type. Number of initial retries per PIN. |
| **<sel_code>** | String type. It is the same as **<code>** or be as a substring of **<code>**. The start/end of a substring can be fuzzily matched using a *.<br>Note: * can only be at the beginning of a substring. When there are no **<code>** matches, an empty will be returned. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CPINR
+CPINR: SIM PIN,3,3
+CPINR: SIM PUK,10,10
+CPINR: SIM PIN2,3,3
+CPINR: SIM PUK2,10,10

OK
```

## 2.17. AT+QLEDMODE Configure Network-Status-Indication Light

This command configures network-status-indication light.

| AT+QLEDMODE Configure Network-Status-Indication Light | |
|---|---|
| Test Command<br>**AT+QLEDMODE=?** | Response<br>**+QLEDMODE:(**list of supported **<LED_mode>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+QLEDMODE?** | Response<br>**+QLEDMODE:<LED_mode>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+QLEDMODE=<LED_mode>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | The command takes effect immediately.<br>The configurations are not saved to NVRAM. |

### Parameter

| | |
|---|---|
| **<LED_mode>** | Integer type. Network-status-indication light mode.<br>0 Disable network-status-indication light, and NETLIGHT pin outputs low level<br>1 Enable network-status-indication light, and NETLIGHT pin outputs PWM signal<br>Different durations of high and low levels indicate different network status:<br>Network search: high level 64 milliseconds (light on) / low level 800 milliseconds (light off) |

| | |
|---|---|
| | Connection status: high level 64 milliseconds (light on) / low level 2000 milliseconds (light off) |
| **<err>** | Integer type. Error code. See ***Chapter 10*** for details. |

### Example

```
AT+QLEDMODE=1
OK
```

# 3 (U)SIM Card Related Commands

## 3.1. AT+CRSM Restricted (U)SIM Access

This command allows the user to perform limited (U)SIM card operations via MT. Each time a command is executed, MT internally performs all card operations related to the command, including selecting the target file and executing the actual command. After the command is executed, the (U)SIM execution result and response data are returned.

| AT+CRSM Restricted (U)SIM Access | |
|---|---|
| Test Command<br>**AT+CRSM=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CRSM=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>[,<pathid>]]]]** | Response<br>**+CRSM: <sw1>,<sw2>[,<response>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | | |
|---|---|---|
| **<command>** | Integer type. Command passed on by the MT to (U)SIM; please refer to *3GPP TS 51.011 [28]*. | |
| | 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. This is the identifier of an elementary data file on (U)SIM. Mandatory for every command except **<command>** equal to STATUS. | |
| **<P1>, <P2>, <P3>** | Integer type. Parameters passed on by the MT to (U)SIM. These parameters are mandatory for every command, except **<command>** equal to GET RESPONSE and STATUS. The values are described in *3GPP TS 51.011 [28].* | |
| **<data>** | String type. Information which shall be written to (U)SIM (hexadecimal character format). | |
| **<pathid>** | String type. Contain the path of an elementary file on the (U)SIM/UICC in hexadecimal format. If the elementary file is in the current path, this parameter can be omitted. | |
| **<sw1>, <sw2>** | Integer type. Information from (U)SIM about the execution of the actual command. | |
| **<response>** | Response of a successful completion of the command previously issued (hexadecimal character format). STATUS and GET RESPONSE return data, which gives information about the current elementary data field. This information includes the type of file and its size. After READ BINARY, READ RECORD or RETRIEVE DATA command return the requested data. **<response>** is not returned after a successful UPDATE BINARY, UPDATE RECORD or SET DATA command. | |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

**Example**

```
AT+CRSM=?
OK
```

## 3.2. AT+CSIM Generic (U)SIM Access

This command allows the user to send directly to the (U)SIM via MT, and after the (U)SIM executes this command, the response is returned to the user in the form of **<response>**. Compared with **AT+CRSM**, this command releases almost unlimited (U)SIM operation permissions to users, and the **AT+CSIM** command operates on (U)SIM as the original operation, and MT just transmits **<command>** to (U)SIM, no other processing will be performed. For example, to read a specified file, you need to deliver two related commands: the Select File command and the Read File command. Since the execution of the associated command of the file operation is original, for example, you must ensure that the basic file selected by the file selection command will not be changed before the file reading command is executed, otherwise the 0x6A82 will be reported.

Currently, after the command is executed, ensure that the file selected by the associated command will not be invalid within 15 seconds due to (U)SIM card power-off. Therefore, if multiple associated commands need to be executed, you must execute the next command within 15 seconds after the previous associated command is executed.

| AT+CSIM Generic (U)SIM Access | |
|---|---|
| Test Command<br>**AT+CSIM=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CSIM=<length>,<command>** | Response<br>**+CSIM: <length>,<response>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<length>** | Integer type. Length of the characters of **<command>** or **<response>**. |
| **<command>** | String type. Hexadecimal format. APDU commands for users to perform (U)SIM operations. (See *TS 102 221* for details.) |
| **<response>** | String type. Hexadecimal format. The actual result returned by (U)SIM. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CSIM=?
OK
```

**NOTE**

1. Since the system operates on (U)SIM card on logical channel 0, it is restricted that **AT+CSIM** cannot switch applications on logical channel 0.
2. If the logic channel is opened through **AT+CSIM**, you need to use **AT+CSIM**/**+CCHC** to close the logic channel after the operation is completed.

## 3.3. AT+CCHO Open Logical Channel

This command opens a new logical channel on UICC and selects the application identified by the **<dfname>**. The new logical channel number is returned by **<sessionid>**. This **<sessionid>** is to be used when sending commands with Generic UICC Logical Channel access **AT+CGLA** commands.

Due to the need to save power on the UICC, the card will be powered off when it is idle, and it will be re-powered and initialized when the next operation is performed. If the card is powered off and then is powered on after the logical channel is opened, the previously opened logic channel will be invalid. Once the logical channels are opened, the UICC will no longer enter the power-saving state until all open logical channels are closed. Therefore, as a constraint rule, the user opens a logical channel via **AT+CCHO**, and after all operations on the logical channel are completed, the channel needs to be closed via **AT+CCHC** (see ***Chapter 3.4***).

| **AT+CCHO Open Logical Channel** | |
|---|---|
| Test Command<br>**AT+CCHO=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CCHO=<dfname>** | Response<br>**<sessionid>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<dfname>** | String type. All selectable applications in the UICC are referenced by a DF name coded on 1 to 16 bytes. |
| **<sessionid>** | Integer type. The number of the logical channel that is opened, returned by the UICC. Range: 1–19. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CCHO=?
OK
```

## 3.4. AT+CCHC Close Logical Channel

This command closes the specified logical channel and needs to be used in conjunction with **AT+CCHO**.

| **AT+CCHC Close Logical Channel** | |
|---|---|
| Test Command<br>**AT+CCHC=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CCHC=<sessionid>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<sessionid>** | Integer type. The logical channel number to be closed. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CCHC=?
OK
```

## 3.5. AT+CGLA Generic UICC Logical Channel Access

The Write Command executes the operation **<command>** on a specified logical channel. Send UICC operation **<command>** to MT, it is then forwarded by MT to UICC. After the UICC executes this command, it will return **<response>** to the upper-layer application through MT. This command allows a remote application on the TE to directly control the currently selected UICC.

| AT+CGLA Generic UICC Logical Channel Access | |
|---|---|
| Test Command<br>**AT+CGLA=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CGLA=<sessionid>,<length>,<command>** | Response<br>**+CGLA: <length>,<response>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<sessionid>** | Integer type. The specified logical channel number, which is enabled by **AT+CCHO** and cannot be set to 0. |
| **<length>** | Integer type. Length of the characters of **<command>** or **<response>** (two times the actual length of the command or response). |
| **<command>** | APDU command in hexadecimal format. (see *TS 102 221* for details). |
| **<response>** | The result of the operation returned by UICC in hexadecimal format. |

| | |
|---|---|
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

The logical channel number identified by the CLA in **<command>** must be the same as that of in **<sessionid>**.

### Example

```
AT+CGLA=?
OK
```

## 3.6. AT+NUICC Set UICC Power Mode

This command powers on/off UICC.

| AT+NUICC Set UICC Power Mode | |
|---|---|
| Test Command<br>**AT+NUICC=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NUICC=<mode>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<mode>** | Integer type. UICC power mode.<br>0 Power off |

| | |
|---|---|
| | 1 Power on |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+NUICC=1
OK
```

# 4 SMS Function Commands

## 4.1. AT+CMGS Send SMS Messages

This command sends SMS messages. Currently SMS message supports PDU mode only and does not support text mode.

| AT+CMGS Send SMS Message | |
|---|---|
| Test Command<br>**AT+CMGS=?** | Response<br>**OK** |
| Write Command<br>**AT+CMGS=<length><CR>PDU is given<CTRL + Z/ESC>** | Response<br>**+CMGS:<mr>[,<ackpdu>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 60 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<length>** | Integer type. Indicates the length of the actual TP data unit (i.e. the SMSC address octets are not counted in the length). |
| **<mr>** | Integer type. *3GPP TS 23.040* TP-Message-Reference. |
| **<ackpdu>** | *3GPP TS 23.040* RP-User-Data element of RP-ACK PDU. This parameter is optional when **<service>** of **AT+CSMS** is set to 1. Currently this parameter is not returned. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## 4.2. AT+CSCA Service Centre Address

This Write Command updates the SMSC address. In PDU mode, the configuration in this command is used to send SMS messages only when the length of the SMSC address (as the destination address) is 0. This Read Command returns the current setting information.

| AT+CSCA Service Centre Address | |
|---|---|
| Test Command<br>**AT+CSCA=?** | Response<br>**OK** |
| Read Command<br>**AT+CSCA?** | Response<br>**+CSCA:<sca>[,<tosca>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CSCA=<sca>[,<tosca>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<sca>** | String type. SC address. |
| **<tosca>** | Integer type. Type of **<sca>**. It can be omitted in the command. When omitted, it is set to 129 or 145. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CSCA="8610000000000"
OK
AT+CSCA?
+CSCA:"8610000000000",145
```

```
OK
AT+CSCA=?
OK
```

## 4.3. AT+CSMS Select Message Service

The Write Command selects messaging service. It returns the types of messages supported by the UE: **<mt>** for mobile terminated messages, **<mo>** for mobile originated messages and **<bm>** for broadcast type messages.

| AT+CSMS Select Message Service | |
|---|---|
| Test Command<br>**AT+CSMS=?** | Response<br>**+CSMS:(**list of supported **<service>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CSMS?** | Response<br>**+CSMS:<service>,<mt>,<mo>,<bm>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CSMS=<service>** | Response<br>**+CSMS:<mt>,<mo>,<bm>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | | |
|---|---|---|
| **<service>** | Integer type. Message service | |
| | 0 | *3GPP TS 23.040 [3]* and *3GPP TS 23.041 [4]* |
| | 1 | *3GPP TS 23.040 [3]* and *3GPP TS 23.041 [4]* (the requirement of **<service>** setting 1 is mentioned under corresponding command descriptions) |
| **<mt>** | Integer type. Mobile terminated messages | |
| | 0 | Type not supported |
| | 1 | Type supported |
| **<mo>** | Integer type. Mobile originated messages | |
| | 0 | Type not supported |
| | 1 | Type supported |
| **<bm>** | Integer type. Broadcast type messages | |
| | 0 | Type not supported |
| | 1 | Type supported |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

### Example

```
AT+CSMS=1
+CSMS:1,1,0

OK
AT+CSMS?
+CSMS:1,1,1,0

OK
AT+CSMS=?
+CSMS:(0,1)

OK
```

## 4.4. AT+CMGF Message Format

This command specifies the input and output format of the short messages. Currently it only supports PDU mode.

| AT+CMGF Message Format | |
|---|---|
| Test Command<br>**AT+CMGF=?** | Response<br>**+CMGF: (**list of supported **<n>**s**)** |

| | |
|---|---|
| | **OK** |
| Read Command<br>**AT+CMGF?** | Response<br>**+CMGF: <n>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CMGF=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<mode>** | Integer type. Input and output format of the short messages.<br>0 PDU mode |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

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

OK
```

## 4.5. AT+CNMA New Message Acknowledgement

This command confirms reception of a new message (SMS-DELIVER or SMS-STATUS-REPORT). This acknowledgement command shall be used when the parameter **<service>** of **AT+CSMS** equals 1. Currently only PDU mode is supported. In PDU mode, it is possible to send either positive (RPACK) or negative (RP-ERROR) acknowledgement to the network. When **<length>** is greater than 0, the message of RP-ACK or RP-ERROR carries TPDU. The entering of PDU is done similarly as specified in **AT+CMGS**,

except that the format of **<ackpdu>** is used instead of **<pdu>** (i.e. SMSC address field is not present).

This command must be executed when receiving the downlink SMS message and waiting for the UE to respond RP-ACK or RP-ERROR to the SMS center, otherwise an error is returned.

| **AT+CNMA New Message Acknowledgement** | |
|---|---|
| Test Command<br>**AT+CNMA=?** | Response<br>**+CNMA:(**list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CNMA[=<n>[,<length>[<CR>PDU is given<CTRL + Z/ESC>]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<n>** | Integer type.<br>1 Send RP-ACK<br>2 Send RP-ERROR |
| **<length>** | Integer type. Indicates the length of the actual TP data unit (i.e. the SMSC address octets are not counted in the length). |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CNMA=?
+CNMA:(1,2)

OK
```

## 4.6. AT+CMGC Send SMS Command

This command sends a command message (SMS-COMMAND) to the network. Currently only PDU mode is supported. In PDU mode, the entering of PDU is done similarly as specified in **AT+CMGS**. Message reference value **<mr>** and the optional value **<ackpdu>** are returned to the TE on successful message delivery.

| AT+CMGC Send SMS Command | |
|---|---|
| Test Command<br>**AT+CMGC=?** | Response<br>**OK** |
| Write Command<br>**AT+CMGC=<length><CR>PDU is given<CTRL + Z/ESC>** | Response<br>**+CMGC:<mr>[,<ackpdu>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<length>** | Integer type. Indicates the length of the actual TP data unit (i.e. the SMSC address octets are not counted in the length). |
| **<mr>** | Integer type. *3GPP TS 23.040* TP-Message-Reference. |
| **<ackpdu>** | *3GPP TS 23.040* RP-User-Data element of RP-ACK PDU. This parameter is optional when **<service>** of **AT+CSMS** is set to 1. Currently this parameter is not returned. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## 4.7. AT+CMMS Send More Messages

This command controls the continuity of SMS relay protocol link. When feature is enabled (and supported by network), multiple messages can be sent much faster as link is kept open.

| AT+CMMS Send More Messages | |
|---|---|
| Test Command<br>**AT+CMMS=?** | Response<br>**+CMMS:(**list of supported **<n>**s**)** |

| | |
|---|---|
| | **OK** |
| Read Command<br>**AT+CMMS?** | Response<br>**+CMMS:<n>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CMMS=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<n>** | Integer type.<br>0 Disable<br>1 Keep enabled until the time between the response of the latest message sending and the next sending exceeds 5 seconds, then TA switches **<n>** automatically back to 0<br>2 Feature enabled (if the time between the response of the latest message sending and the next sending exceeds 5 seconds. TA shall keep **<n>** unchanged |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CMMS=?
+CMMS:(0,1,2)

OK
```

## 4.8. +CMT Report SMS-DELIVER

This URC is reported when SMS-DELIVER message is received.

| +CMT Report SMS-DELIVER | |
|---|---|
| **+CMT:<alpha>,<length><CR><LF><pdu>** | Notify that the SMS-DELIVER message is received |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<alpha>** | String type. Identifies the PDU format. Supports "HEX" (hexadecimal format) only. |
| **<length>** | Integer type. PDU length. Unit: Bytes. |
| **<pdu>** | PDU data of the SMS-DELIVER message. |

# 5 Network Status Related Commands

## 5.1. AT+COPS Operator Selection

This Write Command forces an attempt to select and register the EPS network operator.

This Read Command can be executed in Idle State. It is not allowed to be execute during the business process.

This Read Command returns the current mode, the currently selected operator and the current access technology. If no operator is selected, **<format>**, **<oper>** and **<AcT>** are omitted.

This command is an asynchronous command. After the module receives the command, it responds immediately. The response result only represents the analysis feedback of the command on this command, and does not reflect the specific search results.

| AT+COPS Operator Selection | |
|---|---|
| Test Command<br>**AT+COPS=?** | Response<br>**+COPS:[(**list of supported **<stat>**s**,,,**list of supported **<oper>**s**[,<AcT>])][,,**(list of supported **<mode>**s**),(**list of supported **<format>**s**)]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+COPS?** | Response<br>**+COPS: <mode>[,<format>,<oper>][,<AcT>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |

| | |
|---|---|
| Write Command<br>**AT+COPS=<mode>[,<format>[,<oper>[,<AcT>]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 600 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<mode>** | Integer type. Network selection mode.<br>0 Automatic mode (**<oper>**, **<format>** and **<AcT>** are ignored)<br>1 Manual operator selection (**<oper>** and **<format>** shall be present and **<AcT>** is optional). The module enters automatic mode if manual operator selection fails<br>2 Manually deregister from network |
| **<format>** | Integer type. Format of **<oper>**.<br>2 Numeric **<oper>** |
| **<oper>** | String type. **<format>** indicates if the format is numeric. Numeric format is the NB-IoT network location area identification number (see *3GPP TS 24.008 subclause 10.5.1.3*) which consists of a three BCD digit ITU-T country code, plus a two or three BCD digit network code (refer to *ITU-T Recommendation E.212 Annex A* for coding). |
| **<stat>** | Integer type. Network status.<br>0 Unknown<br>1 Operator available<br>2 Operator currently selected<br>3 Operator forbidden to be selected |
| **<AcT>** | Integer type. Access technology selected.<br>9 E-UTRAN (NB-S1 mode) |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

1. Currently **AT+COPS=?** does not support HHPLMN search.
2. **AT+COPS?** returns the operator selection mode configured by the module. When the module defaults to be in automatic mode, it is an operation of manually deregistering from network if you execute **AT+COPS=2** because it is an operation without changing the network selection mode. Therefore, the return value of **AT+COPS?** is still 0.

**Example**

```
AT+COPS?
+COPS: 0,2,"46000"

OK
```

## 5.2. AT+CPSMS Power Saving Mode Setting

This Write Command controls the setting of the UE's power saving mode (PSM) parameters. It controls whether the UE applies PSM or not, as well as the requested extended periodic TAU value and the requested Active Time value.

| AT+CPSMS Power Saving Mode Setting | |
|---|---|
| Test Command<br>**AT+CPSMS=?** | Response<br>**+CPSMS:(**list of supported **<mode>**s)**,,,(**list of supported **<Requested_Periodic-TAU>**s)**,(**list of supported **<Requested_Active-Time>**s)<br><br>**OK** |
| Read Command<br>**AT+CPSMS?** | Response<br>**+CPSMS:<mode>,,,[<Requested_Periodic-TAU>],[<Requested_Active-Time>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CPSMS=<mode>[,<Requested_Periodic-RAU>[,<Requested_GPRS-READY-timer>[,<Requested_Periodic-TAU>[,<Requested_Active-Time>]]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configurations are saved automatically. |

## Parameter

| | |
|---|---|
| **<mode>** | Integer type. Disable or enable the use of PSM in the UE.<br>0 Disable the use of PSM<br>1 Enable the use of PSM<br>2 Disable the use of PSM and discard all parameters for PSM |
| **<Requested_Periodic-TAU>** | String type. One byte in an 8-bit format. Requested extended periodic TAU value (T3412). The coding method is:<br>Bits 5 to 1 represents the binary coded timer value.<br>Bits 8 to 6 defines the timer value unit as follows:<br>Bits<br>8 7 6<br>0 0 0 Value is incremented in multiples of 10 minutes<br>0 0 1 Value is incremented in multiples of 1 hour<br>0 1 0 Value is incremented in multiples of 10 hours<br>0 1 1 Value is incremented in multiples of 2 seconds<br>1 0 0 Value is incremented in multiples of 30 seconds<br>1 0 1 Value is incremented in multiples of 1 minute<br>1 1 0 Value is incremented in multiples of 320 hours (See details in the Note)<br>1 1 1 Value indicates that the timer is deactivated |
| **<Requested_Active-Time>** | String type. One byte in an 8-bit format. Requested Active Time value (T3324). The coding method is:<br>Bits 5 to 1 represent the binary coded timer value.<br>Bits 8 to 6 defines the valid timer value unit for the GPRS timer as follows:<br>Bits<br>8 7 6<br>0 0 0 Value is incremented in multiples of 2 seconds<br>0 0 1 Value is incremented in multiples of 1 minute<br>0 1 0 Value is incremented in multiples of 6 minutes<br>1 1 1 Value indicates that the timer is deactivated |
| **<Requested_Periodic-RAU>** | Reserved. |
| **<Requested_GPRS-READY-timer>** | Reserved. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

This timer value unit is only applicable to the T3412 extended value IE. If it is received in an integrity protected message, the value shall be interpreted as multiples of 320 hours. Otherwise, the value shall be interpreted as multiples of 1 hour.

**Example**

```
AT+CPSMS?
+CPSMS:1,,,01000001,00000101

OK
```

## 5.3. AT+CEDRXS eDRX Setting

This Write Command controls the setting of the UE's eDRX parameters. It controls whether the UE applies eDRX or not, as well as the requested eDRX value for each specified type of access technology.

| AT+CEDRXS eDRX Setting | |
|---|---|
| Test Command<br>**AT+CEDRXS=?** | Response<br>**+CEDRXS:(**list of supported **<mode>**s**),(**list of supported**<AcT-type>**s**),(**list of supported **<Requested_eDRX_value>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CEDRXS?** | Response<br>**[+CEDRXS:<AcT-type>,<Requested_eDRX_value>**<br>**[+CEDRXS:<AcT-type>,<Requested_eDRX_value>**<br>**[...]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CEDRXS=<mode>,<AcT-type>[,<Requested_eDRX_value>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |

| Characteristics | The command takes effect immediately.<br>The configurations are saved automatically. |
|---|---|

## Parameter

**<mode>**

Integer type. Disable or enable the use of eDRX. This parameter is applicable to all specified types of access technologies, i.e. the most recent setting of **<mode>** takes effect for all specified values of **<AcT-type>**.

0 Disable the use of eDRX

1 Enable the use of eDRX

2 Enable the use of eDRX and enable URC **+CEDRXP:<AcT-type>[,<Requested_eDRX_value>[,<NW-Provided_eDRX_value>[,<Paging_time_window>]]]**

3 Disable the use of eDRX and discard all other parameters in the Write Command and reset to default values

**<AcT-type>**

Integer type. The type of access technology. **AT+CEDRXS?** specifies the relationship between the type of access technology and the requested eDRX value.

0 The currently access network does not use eDRX (terminal or network does not support eDRX). This value is only used to actively report URC **+CEDRXP**. When the value is 0, other parameters are omitted

5 E-UTRAN (NB-S1 mode)

**<Requested_eDRX_value>**

String type. Requested eDRX cycle. Half a byte in a 4-bit format. NB-S1 mode.

| Bits | | | | |
|---|---|---|---|---|
| 4 | 3 | 2 | 1 | E-UTRAN eDRX cycle length duration |
| 0 | 0 | 1 | 0 | 20.48 seconds |
| 0 | 0 | 1 | 1 | 40.96 seconds |
| 0 | 1 | 0 | 1 | 81.92 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 |

**<NW-provided_eDRX_value>**

String type. The eDRX cycle provided by the network. Half a byte in a 4-bit format. NB-S1 mode.

| Bits | | | | |
|---|---|---|---|---|
| 4 | 3 | 2 | 1 | E-UTRAN eDRX cycle length duration |
| 0 | 0 | 1 | 0 | 20.48 seconds |

| | | | | | |
|---|---|---|---|---|---|
| | 0 | 0 | 1 | 1 | 40.96 seconds |
| | 0 | 1 | 0 | 1 | 81.92 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 type. Paging time window provided by the network. Half a byte in a 4-bit format. NB-S1 mode.

Bits

| 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 |

**<err>** Error code. See ***Chapter 10*** for details.

## Example

```
AT+CEDRXS=1,5,"0010"
OK
AT+CEDRXS?
+CEDRXS:5,"0010"

OK
AT+CEDRXS=?
+CEDRXS:(0,1,2,3),(5),("0010","0011","0101","1001","1010","1011","1100","1101","1110","1111")

OK
```

## 5.4. AT+CEREG EPS Network Registration Status

This command configures the presentation of unsolicited result codes for EPS network registration status.

- When **<n>**=1 and there is a change in the UE's EPS network registration status, an unsolicited result code (URC) **+CEREG:<stat>** is presented.
- When **<n>**=2 and there is a change of the network cell, URC **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>]]** is presented.
- When **<n>**=3 and the value of **<stat>** changes, URC **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]** is presented.

If the UE applies for entering PSM to reduce power consumption:

- When **<n>**=4, and the cell information changes, the following URC is present: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]**
- When **<n>**=5, and network status changes, the following URC is present: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<ActiveTime>],[<Periodic-TAU>]]]]**

This Read Command returns the status of result code presentation and an integer **<stat>** which shows the EPS registration status of the UE. Other parameters are present based on **<n>** and **<stat>**. Their presence is the same as that in the URC.

| **AT+CEREG EPS Network Registration Status** | |
|---|---|
| Test Command<br>**AT+CEREG=?** | Response<br>**+CEREG:(**list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CEREG?** | Response<br>When **<n>**=0, 1, 2 or 3 and the command is executed successfully:<br>**+CEREG:<n>,<stat>[,[<tac>],[<ci>],[<AcT>[,<cause_type>,<reject_cause>]]]**<br><br>When**<n>**=4 or 5 and the command is executed successfully:<br>**+CEREG:<n>,<stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]]**<br><br>**OK** |

| | |
|---|---|
| | If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CEREG=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configurations are not saved. |

## Parameter

| | |
|---|---|
| **<n>** | Integer type.<br>0 Disable network registration URC<br>1 Enable network registration URC: **+CEREG:<stat>**<br>2 Enable network registration and location information URC: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>]]**<br>3 Enable network registration, location information and EMM cause value information URC: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]**<br>4 For a UE that requests PSM, enable the URC of network registration, location information and T3412 and T3324 timer duration information: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]**<br>5 For a UE that requests PSM, enable the URC of network registration, location information, EMM cause value information and T3412 and T3324 timer duration information: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]]** |
| **<stat>** | Integer type. EPS registration status.<br>0 Not registered. UE is not currently searching an operator to register to<br>1 Registered, home network<br>2 Not registered, but UE is currently trying to attach or searching an operator to register to<br>3 Registration denied<br>4 Unknown (e.g. out of E-UTRAN coverage)<br>5 Registered, roaming<br>6 Register to SMS ONLY, local network (currently not supported)<br>7 Register to SMS ONLY, roaming network (currently not supported) |

| | |
|---|---|
| | 8 Only attach to emergency bearer service (currently not supported)<br>9 Register to a non-preferred CSFB local network (currently not supported)<br>10 Register to a non-preferred CSFB roaming network (currently not supported) |
| **<tac>** | String type. Two-byte tracking area code in hexadecimal format (e.g., 00C3 equals 195 in decimal). |
| **<ci>** | String type. Four-byte E-UTRAN cell ID in hexadecimal format. |
| **<AcT>** | Integer type. Access technology of the serving cell.<br>9 E-UTRAN (NB-S1 mode) |
| **<cause_type>** | Integer type. Type of **<reject_cause>**.<br>0 **<reject_cause>** contains an EMM cause value<br>1 **<reject_cause>** contains a manufacturer-specific cause value |
| **<reject_cause>** | Integer type. Contains the cause of the registration failure. The value type as defined by **<cause_type>**. |
| **<Active-Time>** | String type. One byte in an 8-bit format. Active time value (T3324). The coding method is:<br>Bits 5 to 1 represents the binary coded timer value.<br>Bits 8 to 6 defines the timer value unit for the GPRS timer as follows:<br>Bits<br>8 7 6<br>0 0 0 Value is incremented in multiples of 2 seconds<br>0 0 1 Value is incremented in multiples of 1 minute<br>0 1 0 Value is incremented in multiples of 6 minutes<br>1 1 1 Value indicates that the timer is deactivated<br>Other value is incremented in multiples of 1 minute |
| **<Periodic-TAU>** | String type. One byte in an 8-bit format. Indicates the extended periodic TAU value (T3412). The coding method is:<br>Bits 5 to 1 represents the binary coded timer value.<br>Bits 8 to 6 defines the timer value unit for the GPRS timer as follows:<br>Bits<br>8 7 6<br>0 0 0 Value is incremented in multiples of 10 minutes<br>0 0 1 Value is incremented in multiples of 1 hour<br>0 1 0 Value is incremented in multiples of 10 hours<br>0 1 1 Value is incremented in multiples of 2 seconds<br>1 0 0 Value is incremented in multiples of 30 seconds<br>1 0 1 Value is incremented in multiples of 1 minute<br>1 1 0 Value is incremented in multiples of 320 hours<br>1 1 1 Value indicates that the timer is deactivated |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CEREG=1                    //Enable network registration URC
OK
```

```
AT+CEREG?
+CEREG:1,1

OK
AT+CEREG=?
+CEREG:(0,1,2,3,4,5)

OK
```

## 5.5. AT+CTZR Time Zone Reporting

This command controls the time zone change event reporting. If reporting is enabled, the MT returns the unsolicited result code **+CTZV: <tz>**, **+CTZE: <tz>,<dst>,[<time>]** or **+CTZEU:<tz>,<dst>,[<utime>]** whenever the time zone is changed.

| AT+CTZR Time Zone Reporting | |
|---|---|
| Test Command<br>**AT+CTZR=?** | Response<br>**+CTZR:(**list of supported **<reporting>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CTZR?** | Response<br>**+CTZR:<reporting>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CTZR=<reporting>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |

| | |
|---|---|
| Maximum Response Time | 1 s |
| Characteristics | The command takes effect immediately.<br>The configurations are not saved. |

## Parameter

| | |
|---|---|
| **<reporting>** | Integer type. Enable or disable time zone change event reporting<br>0 Disable time zone change event reporting<br>1 Enable time zone change event reporting by URC **+CTZV:<tz>**<br>2 Enable extended time zone and local time reporting by unsolicited result code **+CTZE:<tz>,<dst>,[<time>]**<br>3 Enable extended time zone and universal time reporting by unsolicited result code **+CTZEU:<tz>,<dst>,[<utime>]** |
| **<tz>** | String type. Sum of the local time zone (difference between the local time and GMT expressed in quarters of an hour) plus daylight saving time. The format is “±zz", expressed as a fixed width, two digits integer with the range of -47 to +56. To maintain a fixed width, numbers in the range of -9 to +9 are expressed with a leading zero, e.g. “-09", “+00", and “+09". |
| **<dst>** | Integer type. Whether **<tz>** includes daylight savings adjustment or not.<br>0 **<tz>** includes no adjustment for daylight saving time<br>1 **<tz>** includes +1 hour (equals 4 quarters in **<tz>**) adjustment for daylight saving time<br>2 **<tz>** includes +2 hours (equals 8 quarters in **<tz>**) adjustment for daylight saving time |
| **<time>** | String Type. Local time. The format is "YYYY/MM/DD,HH:mm:SS" expressed as integers representing year (YYYY), month (MM), date (DD), hour (HH), minute (MM) and second (SS). The local time can be derived by the UE from information provided by the network at the time of delivering time zone information and is present in the unsolicited result code for extended time zone and local time reporting if the universal time is provided by the network. |
| **<utime>** | String Type. Universal time. The format is "YYYY/MM/DD,HH:MM:SS", expressed as integers representing year (YYYY), month (MM), date (DD), hour (HH), minute (MM) and second (SS). The universal time can be provided by the network at the time of delivering time zone information and is present in the unsolicited result code for extended time zone and universal time reporting if provided by the network. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CTZR=0
OK
AT+CTZR?
+CTZR:0
```

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

OK
```

## 5.6. AT+CSQ Signal Quality Report

This command returns the received signal strength level **<rssi>** and the channel bit error rate **<ber>**.

| AT+CSQ Signal Quality Report | |
|---|---|
| Test Command<br>**AT+CSQ=?** | Response<br>**+CSQ:(**list of supported **<rssi>**s**),(**list of supported **<ber>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>or<br>**+CME ERROR: <err>** |
| Execution Command<br>**AT+CSQ** | Response<br>**+CSQ:<rssi>,<ber>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | | |
|---|---|---|
| **<rssi>** | Integer type. Received signal strength level. | |
| | 0 | -113 dBm or less |
| | 1 | -111 dBm |
| | 2–30 | -109 to -53 dBm |
| | 31 | -51 dBm or greater |

| | | |
|---|---|---|
| | 99 | Not known or not detectable |
| **<ber>** | Integer type. Channel bit error rate (in percent). It is unavailable in NB network. The value is fixed to 99. | |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

### Example

```
AT+CSQ
+CSQ:21,99

OK
```

## 5.7. AT+NPTWEDRXS PTW and eDRX Setting

This command controls the setting of the UE's PTW and eDRX parameters. It controls whether the UE applies eDRX and PTW or not, as well as the requested PTW and eDRX values for each specified type of access technology.

| **AT+NPTWEDRXS PTW and eDRX Setting** | |
|---|---|
| Test Command<br>**AT+NPTWEDRXS=?** | Response<br>**+NPTWEDRXS:(**list of supported **<mode>**s**),(**list of supported **<AcT-type>**s**),(**list of supported **<Requested_Paging_Time_Window>**s**),(**list of supported **<Requested_eDRX_value>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NPTWEDRXS?** | Response<br>**+NPTWEDRXS:<AcT-type>,<Requested_Paging_Time_Window>,<Requested_eDRX_value>[,<NW_Provided_eDRX_value>[,<Paging_time_window>]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |

| | |
|---|---|
| Write Command<br>**AT+NPTWEDRXS=<mode>,<AcT-type>[,<Requested_Paging_Time_Window>[,<Requested_eDRX_value>]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | The command takes effect immediately.<br>The configurations are saved automatically. |

## Parameter

**<mode>**

Integer type. Disable or enable the use of PTW and eDRX. This parameter is applicable to all specified types of access technologies, i.e. the most recent setting of **<mode>** takes effect for all specified values of **<AcT-type>**.

0 Disable the use of PTW and eDRX

1 Enable the use of PTW and eDRX

2 Enable the use of PTW and eDRX and enable the following URC when network parameters change:
**+NPTWEDRXP:<AcT-type>[,<Requested_Paging_Time_Window>[,<Requested_eDRX_value>[,<NW_Provided_eDRX_value>[,<Paging_time_window>]]]]**

3 Disable the use of PTW and eDRX and discard all other parameters in the Write Command

**<AcT-type>**

Integer type. The type of access technology.

0 The currently access network does not use eDRX (terminal or network does not support eDRX). This value is only used to actively report URC **+CEDRXP**. When the value is 0, other parameters are omitted

5 E-UTRAN (NB-S1 mode)

**<Requested_Paging_Time_Window>**

String type. Requested PTW. Half a byte in a 4-bit format. NB-S1 mode.

Bits

| 4 | 3 | 2 | 1 | PTW 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 |

**<Requested_eDRX_value>**

String type. The eDRX cycle provided by the network. Half a byte in a 4-bit format. NB-S1 mode.

Bits

| 4 | 3 | 2 | 1 | E-UTRAN eDRX cycle length duration |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 20.48 seconds |
| 0 | 0 | 1 | 1 | 40.96 seconds |
| 0 | 1 | 0 | 1 | 81.92 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 |

**<NW_Provided_eDRX_value>**

String type. The eDRX cycle provided by the network. Half a byte in a 4-bit format. NB-S1 mode.

Bits

| 4 | 3 | 2 | 1 | E-UTRAN eDRX cycle length |
|---|---|---|---|---|
| 0 | 0 | 1 | 0 | 20.48 seconds |
| 0 | 0 | 1 | 1 | 40.96 seconds |
| 0 | 1 | 0 | 1 | 81.92 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 type. Paging time window provided by the network. Half a byte in a 4-bit format. NB-S1 mode.

Bits

| 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 |
| **<err>** | Error code. See ***Chapter 10*** for details. | | | | |

## 5.8. AT+CESQ Extended Signal Quality

This Execution Command returns received signal quality parameters.

The terminal will provide a current signal strength indicator of 0 to 99, where a larger number indicates better signal quality. This information is based on a single measurement, so the information can be expected to change greatly over short periods of time.

| **AT+CESQ Extended Signal Quality** | |
|---|---|
| Test Command<br>**AT+CESQ=?** | Response<br>**+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**)**<br><br>**OK** |
| Execution Command<br>**AT+CESQ** | Response<br>**+CESQ:<rxlev>,<ber>,<rscp>,<ecno>,<rsrq>,<rsrp>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 300 ms |

| Characteristics | - |
|---|---|

## Parameter

| | | |
|---|---|---|
| **<rxlev>** | Integer type. Received signal strength level (RSSI). | |
| | 0 | RSSI < -110 dBm |
| | 1 | -110 dBm ≤ RSSI < -109 dBm |
| | 2 | -109 dBm ≤ RSSI < -108 dBm |
| | … | |
| | 61 | -50 dBm ≤ RSSI < -49 dBm |
| | 62 | -49 dBm ≤ RSSI < -48 dBm |
| | 63 | -48 dBm ≤ RSSI |
| | 99 | Not known or not detectable |
| **<ber>** | Integer type. Channel bit error rate (in percent). | |
| | 0–7 | RxQual values. *RXQUAL_0* to *RXQUAL_7* as defined in *3GPP TS 45.008*. |
| | 99 | Not known or not detectable |
| **<rscp>** | Integer type. Received signal code power (RSCP, see *3GPP 25.133* and *3GPP 25.123*). | |
| | 0 | RSCP < -120 dBm |
| | 1 | -120 dBm ≤ RSCP < -119 dBm |
| | 2 | -119 dBm ≤ RSCP < -118 dBm |
| | … | |
| | 94 | -27 dBm ≤ RSCP < -26 dBm |
| | 95 | -26 dBm ≤ RSCP < -25 dBm |
| | 96 | -25 dBm ≤ RSCP |
| | 255 | Not known or not detectable |
| **<ecno>** | Integer type, Ec/No (see *3GPP 25.133*) | |
| | 0 | Ec/No < -24 dBm or less |
| | 1 | -24 dBm ≤ Ec/No < -23.5 dBm |
| | 2 | -23.5 dBm ≤ Ec/No < -23 dBm |
| | … | |
| | 47 | -1 dBm ≤ Ec/No < -0.5 dBm |
| | 48 | -0.5 dBm ≤ Ec/No < 0 dBm |
| | 49 | 0 dBm ≤ Ec/No |
| | 255 | Not known or not detectable |
| **<rsrq>** | Integer type. Reference signal received quality (RSRQ, see *3GPP 36.133*) | |
| | 0 | RSRQ < -19.5 dB or less |
| | 1 | -19.5 dB ≤ RSRQ < -19 dB |
| | 2 | -19 dB ≤ RSRQ < -18.5 dB |
| | … | |
| | 32 | -4 dB ≤ RSRQ < -3.5 dB |
| | 33 | -3.5 dB ≤ RSRQ < -3 dB |
| | 34 | -3 dB ≤ RSRQ |
| | 255 | Not known or not detectable |

| | |
|---|---|
| **<rsrp>** | Integer type. Reference signal received power (RSRP, see *3GPP 36.133*).<br>0 RSRP < -140 dBm or less<br>1 -140 dBm ≤ RSRP < -139 dBm<br>2 -139 dBm ≤ RSRP < -138 dBm<br>…<br>95 -46 dBm ≤ RSRP < -45 dBm<br>96 -45 dBm ≤ RSRP < -44 dBm<br>97 -44 dBm ≤ RSRP<br>255 Not known or not detectable |
| **<err>** | Error code. Please refer to ***Chapter 10*** for details. |

## Example

```
AT+CESQ
+CESQ:99,99,255,255,25,30

OK
```

**NOTE**

Currently, **<rxlev>**, **<ber>**, **<rscp>** and **<ecno>** are not applicable for BC680Z-EU module, so the return value is fixed to **<rxlev>**=99, **<ber>**=99, **<rscp>**=255, and **<ecno>**=255.

# 6 PS Commands

## 6.1. AT+CGDCONT Define PDP Context

This command specifies PDP context parameters for a PDP context identified by **<cid>**. As a special form of the Write Command, **AT+CGDCONT=<cid>** indicates that deleting **<cid>** corresponding PDP context parameters to make it undefined, and the last context parameter cannot be turned into undefined.

If the initial PDP context is enabled, the module automatically defines the PDP context of **<cid>**=0, and supports the configuration of the PDP context of **<cid>** with this command. **AT+CGDCONT=0** resets the parameters of the PDP context of **<cid>**=0 to default values.

This Test Command returns values supported as a compound value. If the UE supports several PDP types, **<PDP_type>**, the parameter value ranges for each **<PDP_type>** are returned on a separate line.

This Read Command returns the current settings for each defined PDP context.

| **AT+CGDCONT Define PDP Context** | |
|---|---|
| Test Command<br>**AT+CGDCONT=?** | Response<br>**+CGDCONT:(**list of supported **<cid>**s)**,<PDP_type>,,,,(**list of supported **<h_comp>**s)**,,,,,(**list of supported **<NSLPI>**s)**,(**list of supported **<securePCO>**s)**<br>**[+CGDCONT:(**list of supported **<cid>**s)**,<PDP_type>,,,,(**list of supported **<h_comp>**s)**,,,,,(**list of supported **<NSLPI>**s)**,(**list of supported **<securePCO>**s)]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CGDCONT?** | Response<br>**+CGDCONT:<cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>[,[,[,[,[,<NSLPI>[,<secu** |

| | |
|---|---|
| | **rePCO>]]]]]**<br>**[...]]**<br><br>**OK** |
| Write Command<br>**AT+CGDCONT=<cid>[,<PDP_type>[,<APN>[,<PDP_addr>[,[,<h_comp>[,[,[,[,[,<NSLPI>[,<securePCO>]]]]]]]]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 2 s |
| Characteristics | The command takes effect immediately.<br>When **<cid>**=0, the configurations are saved automatically. |

## Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter that specifies a particular PDP context definition.<br>Range: 0–10. |
| **<PDP_type>** | String type. A string parameter which specifies the type of packet data protocol.<br>"IP" Internet Protocol version 4 ( *IETF STD 5 [103]*)<br>"IPV6" Internet Protocol version 6<br>"IPV4V6" Internet Protocol version 6 and version 4 to handle dual-IP-stack UE capability<br>"NonIP" None IP |
| **<APN>** | String type. A logical name that is used to select the GGSN or the external packet data network. |
| **<PDP_addr>** | String type. MS address. If the value is omitted, it may be provided by the TE during the PDP startup procedure or, failing that, a dynamic address will be requested. |
| **<h_comp>** | Integer type. Controls PDP header compression. Currently only 0 and 1 are supported.<br>0 Off<br>1 On<br>2 RFC 1144 (applicable for SNDCP only) (currently not supported)<br>3 RFC 2507 (currently not supported)<br>4 RFC 3095 (applicable for PDCP only) (currently not supported) |
| **<NSLPI>** | Integer type. The NAS signaling priority requested for this PDP context. MT utilizes the provided NSLPI information as specified in *3GPP TS 24.301* and *3GPP TS 24.008*.<br>0 Indicates that this PDP context is to be activated with the value for the low priority indicator<br>1 Indicates that this PDP context is to be activated with the value for the low priority indicator set to "MS is not configured for NAS signaling low priority" |
| **<securePCO>** | Integer type. Specifies whether security protected transmission of PCO is requested or |

| | |
|---|---|
| | not (applicable for EPS only).<br>0 Security protected transmission of PCO is not requested<br>1 Security protected transmission of PCO is requested |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

1. **<cid>** defined and activated or been activating are not allowed to be re-defined.
2. Currently only one PDP context (**<cid>**) is supported to be activated at a time.

**Example**

```
AT+CGDCONT=?
+CGDCONT:(0-10),("IP","IPV6","IPV4V6","NONIP"),,,,(0,1),,,,,(0,1),(0,1)

OK
AT+CGDCONT=0,"IP","ABC.COM"
OK
AT+CGDCONT?
+CGDCONT:0,"IP","ABC.COM",,,0,,,,,0,0

OK
```

## 6.2. AT+CGATT PS Attach or Detach

This command attaches the UE to, or detach the UE from, the packet domain service. This command is an asynchronous command. The module responds immediately after receiving the command. The response content only represents the analysis and feedback of the command, and does not reflect the specific attachment/detachment result.

| **AT+CGATT PS Attach or Detach** | |
|---|---|
| Test Command<br>**AT+CGATT=?** | Response<br>**+CGATT:(**list of supported **<state>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command | Response |

| | |
|---|---|
| **AT+CGATT?** | **+CGATT:<state>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CGATT=<state>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 60 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<state>** | Integer type. Indicates the state of PS attachment.<br>0 Detached<br>1 Attached |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

When the module is in deep sleep, executing the Read Command **AT+CGATT?** synchronizes the network status. If the signal environment becomes worse, the synchronization time may be extended, and the command will return to 0. At this time, you can execute **AT+CGATT?** again to return to actual state.

## Example

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

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

**OK**

## 6.3. AT+CGACT PDP Context Activate/Deactivate

This command activates or deactivates the specified PDP context(s). After the command has completed, the MT remains in V.25ter command state. The state of a PDP context already in the requested state remains unchanged. If the requested state for any specified context cannot be achieved, an **ERROR** or **+CME ERROR** response is returned.

If the MT is not PS attached when the activation form of the command is executed, the MT first performs a PS attachment and then attempts to activate the specified contexts. If the attachment fails then the MT responds with an error or, if extended error responses are enabled, with the appropriate failure-to-attach error message.

| **AT+CGACT PDP Context Activate/Deactivate** | |
|---|---|
| Test Command<br>**AT+CGACT=?** | Response<br>**+CGACT:(**list of supported **<state>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CGACT?** | Response<br>**+CGACT:<cid>,<state>**<br>**[+CGACT:<cid>,<state>]**<br>**[...]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CGACT=<state>,<cid>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |

| | |
|---|---|
| Maximum Response Time | 60 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<state>** | Integer type. The state of PDP context activation.<br>0 Deactivated<br>1 Activated |
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). Currently only one PDP context (**<cid>**) is supported to be activated at a time. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

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

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

OK
```

## 6.4. AT+NCPCDPR Configure PDP Context Dynamic Parameters to be Read

This command allows configuring PDP context dynamic parameters to be read.

| **AT+NCPCDPR Configure PDP Context Dynamic Parameters to be Read** | |
|---|---|
| Test Command<br>**AT+NCPCDPR=?** | Response<br>**+NCPCDPR:(**list of supported **<parameter>**s**),(**list of supported **<state>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or |

| | |
|---|---|
| | **+CME ERROR: <err>** |
| Read Command<br>**AT+NCPCDPR?** | Response<br>**+NCPCDPR:<parameter>,<state>**<br>**+NCPCDPR:<parameter>,<state>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NCPCDPR=<parameter>,<state>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect after the module is rebooted.<br>The configurations are saved automatically. |

## Parameter

| | |
|---|---|
| **<parameter>** | Integer type. Dynamic parameters to be read.<br>0 Get DNS server IPv4 Address request<br>1 Get DNS server IPv6 Address request |
| **<state>** | Integer type. Read state of PDP context dynamic parameters.<br>0 Disable to read<br>1 Enable to read |
| **<err>** | Error code. See ***Chapter 10*** for details |

## Example

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

OK
```

## 6.5. AT+CSODCP Sending Originating Data via Control Plane

This command transmits data over control plane to network.

| AT+CSODCP Send Originating Data via Control Plane | |
|---|---|
| Test Command<br>**AT+CSODCP=?** | Response<br>**+CSODCP:(**list of supported **<cid>**s)**,(**list of supported **<cpdata_length>**s)**,(**list of supported **<RAI>**s)**,(**list of supported **<type_of_user_data>**s)**,(**list of supported **<sequence>**s)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CSODCP=<cid>,<cpdata_length>,<cpdata>[,<RAI>[,<type_of_user_data>[,<sequence]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<cid>** | Integer type. Specifies a particular PDP context definition (see **AT+CGDCONT**) |
| **<cpdata_length>** | Integer type. The number of octets of the **<cpdata>** information element. It should be set to 0 when no data is transmitting. |
| **<cpdata>** | String type. The data to be sent. It shall be an empty string ("") when no data is transmitting. |
| **<RAI>** | Integer type. Indicates the value of the release assistance indication.<br>0 No information available<br>1 The MT expects that exchange of data will be completed with the transmission of the uplink data<br>2 The MT expects that exchange of data will be completed with the receipt of the downlink response |

| | |
|---|---|
| **<type_of_user_data>** | Integer type. Indicates whether the user data that is transmitted is regular or exceptional.<br>0 Regular data<br>1 Exception data |
| **<sequence>** | Integer type. Data sequence. Range: 1–255. If it is omitted, the data sending status will not be reported. Otherwise, when the data packet is sent by RF or abandoned, report the following result code:<br>**+CSODCPR:<cid>,<sequence>,<status>** |
| **<status>** | Integer type. Indicated the data status.<br>0 Error<br>1 Data sent |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**Example**

```
AT+CSODCP=?
+CSODCP:(0-10),(0-1600),(0-2),(0,1),(1-255)

OK
```

## 6.6. AT+NQPODCP Query Pending Originating Data List via the Control Plane

This command queries the pending originating data list via the control plane.

| **AT+NQPODCP Query Pending Originating Data List via the Control Plane** | |
|---|---|
| Test Command<br>**AT+NQPODCP=?** | Response<br>**+NQPODCP:(**list of supported **<cid>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NQPODCP=<cid>** | Response<br>**[+NQPODCP:[<sequence>[,<sequence>...]]]**<br><br>**OK** |

| | |
|---|---|
| | If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<sequence>** | Integer type. The sequence number of pending originating message. Range: 1–255. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NQPODCP=?
+NQPODCP:(0-10)

OK
```

## 6.7. AT+CRTDCP Report CP Data

This command controls whether downlink data is actively reported. If the data is actively reported, the URC **+CRTDCP:<cid>,<cpdata_length>,<cpdata>** is reported.

| **AT+CRTDCP Report CP Data** | |
|---|---|
| Test Command<br>**AT+CRTDCP=?** | Response<br>**+CRTDCP:(**list of supported **<reporting>**s**),(**list of supported **<cid>**s**),(**maximum number of octets of user data indicated by **<cpdata_length>)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CRTDCP?** | Response<br>**+CRTDCP:<reporting>** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CRTDCP=<reporting>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately<br>The configurations are saved automatically. |

## Parameter

| | |
|---|---|
| **<reporting>** | Integer type. Control whether the terminal reports CP data.<br>0 Disable reporting CP data<br>1 Enable reporting CP data by the URC:<br>**+CRTDCP:<cid>,<cpdata_length>,<cpdata>** |
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<cpdata_length>** | Integer type. The number of bytes of **<cpdata>**. Maximum length: 65535 bytes. When there is no data to transmit, the value shall be set to zero. |
| **<cpdata>** | Octets string. CP data. When there is no data to transmit, the **<cpdata>** shall be an empty string (**""**). |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+CRTDCP=1
OK
AT+CRTDCP?
+CRTDCP:1

OK
AT+CRTDCP=?
+CRTDCP:(0,1),(0-10),(65535)
```

**OK**

## 6.8. AT+CSCON Signalling Connection Status

This command controls whether to report the signaling connection status URC **+CSCON** and queries the current configuration and signaling connection status. When active reporting of the URC is set, the signaling connection status URC is automatically reported after MT is in idle mode or connected mode.

| **AT+CSCON Signalling Connection Status** | |
|---|---|
| Test Command<br>**AT+CSCON=?** | Response<br>**+CSCON:(**list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CSCON?** | Response<br>**+CSCON:<n>,<mode>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CSCON=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configuration is not saved. |

### Parameter

| | |
|---|---|
| **<n>** | Integer type. Enable/disable the URC.<br>0 Disable the URC<br>1 Enable the URC **+CSCON:<mode>** |
| **<mode>** | Integer type. The signaling connection status.<br>0 Idle<br>1 Connected |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CSCON=0
OK
AT+CSCON?
+CSCON:0,0

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

OK
```

## 6.9. AT+NIPINFO IP Address Information Report

This command reports IP address information. The Write Command controls the presentation of a URC:

- If the IP address is obtained successfully, **<failure_cause>** will be omitted. The URC will be reported in the format of **+NIPINFO:<cid>,<IP_type>,<IP_addr>**.
- If it fails to obtain an IP address, **<IP_addr>** will be omitted. The URC will be reported in the format of **+NIPINFO:<cid>,<IP_type>,<failure_cause>**.

| **AT+NIPINFO IP Address Information Report** | |
|---|---|
| Test Command<br>**AT+NIPINFO=?** | Response<br>**+NIPINFO:(**list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or |

| | |
|---|---|
| | **+CME ERROR: <err>** |
| Read Command<br>**AT+NIPINFO?** | Response<br>**+NIPINFO:<n>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NIPINFO=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configuration is not saved. |

## Parameter

| | |
|---|---|
| **<n>** | Integer type.<br>0 Disable the URC<br>1 Enable the URC:<br>**+NIPINFO:<cid>,<IP_type>[,<IP_addr>][,<failure_cause>]** |
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<IP_type>** | String type. It specifies the type of IP address.<br>IP Internet Protocol, version 4<br>IPV6 Internet Protocol, version 6<br>IPV4V6 Internet Protocol (*IETF STD 5[103]*) |
| **<IP_addr>** | String type. IP address. The IPv4 address is expressed in decimal numbers and dots, such as 192.0.2.2. The IPv6 address is expressed in hexadecimal numbers and colon, such as 2001:db8:1111:2222:3333:4444:5555:6666. This parameter is not displayed when the address acquisition failed. |
| **<failure_cause>** | Integer type. Causes for the failure to get an IP address.<br>1 PDN type IPv4 only allowed<br>2 PDN type IPv6 only allowed<br>3 Single address bearers only allowed |

| | | |
|---|---|---|
| | 4 | IPv6 RA timeout |
| | 5 | Unspecified |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

### Example

```
AT+NIPINFO?
+NIPINFO:0

OK
```

## 6.10. AT+CGPADDR Show PDP Addresses

This command returns the IP address of the device. The Write Command returns a list of PDP addresses that specify **<cid>**. If no **<cid>** is specified, the addresses for all defined contexts are returned. The Test Command returns a list of defined **<cid>**s.

| **AT+CGPADDR Show PDP Addresses** | |
|---|---|
| Test Command<br>**AT+CGPADDR=?** | Response<br>**+CGPADDR:(**list of defined **<cid>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution/Write Command<br>**AT+CGPADDR[=<cid>[,<cid>[,...]]]** | Response<br>**+CGPADDR:<cid>[,<PDP_addr_1>[,<PDP_addr_2>]]**<br>**[+CGPADDR:<cid>,[<PDP_addr_1>[,<PDP_addr_2>]]]**<br>**[...]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<PDP_addr_1>** and **<PDP_addr_2>** | String type. Identify the UE in the address space applicable to the PDP. The address may be static or dynamic. For a static address, it will be the one set by the **AT+CGDCONT** command when the context was defined. For a dynamic address it will be the one assigned during the last PDP context activation that used the context definition referred to by **<cid>**. Both **<PDP_addr_1>** and **<PDP_addr_2>** are omitted if none is available. **<PDP_addr_1>** and **<PDP_addr_2>** are included when both IPv4 and IPv6 addresses are specified, with **<PDP_addr_1>** referring to the IPv4 address and **<PDP_addr_2>** referring to the IPv6 address.<br>The format of the string is as follows: (Each dot is separated by a value ranging from 0 to 255).<br>IPv4: a1.a2.a3.a4<br>IPv6:<br>a1.a2.a3.a4.a5.a6.a7.a8.a9.a10.a11.a12.a13.a14.a15.a16 |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

In dual-stack terminals (**<PDP_type>=IPv4v6**), the IPv6 address will be provided in **<PDP_addr_2>**. For terminals with a single IPv6 stack (**<PDP_type>=IPv6**) or due to backwards compatibility, the IPv6 address can be provided in parameter **<PDP_addr_1>**.

### Example

```
AT+CGPADDR=0
+CGPADDR:0,192.0.2.2

OK
```

## 6.11. AT+CGAPNRC APN Rate Control

The Execution Command returns APN rate control parameters associated with the provided context identifier **<cid>**. If the parameter **<cid>** is omitted, the APN rate control parameters for all active PDP contexts will be returned. If the specified CID is not activated, an **ERROR** is returned.

| AT+CGAPNRC APN Rate Control | |
|---|---|
| Test Command<br>**AT+CGAPNRC=?** | Response<br>**+CGAPNRC:(**list of **<cid>**s associated with active contexts**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution/Write Command<br>**AT+CGAPNRC[=<cid>]** | Response<br>**[+CGAPNRC:<cid>[,<Additional_exception_reports>[,<Uplink_time_unit>[,<Maximum_uplink_rate>]]]**<br>**[+CGAPNRC:<cid>[,<Additional_exception_reports>[,<Uplink_time_unit>[,<Maximum_uplink_rate>]]]**<br>**[...]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<Additional_exception_reports>** | Integer type. Indicates whether to send the additional exception reports when the maximum uplink rate is reached.<br>0 **<Additional_exception_reports>** at maximum rate reached are not allowed to be sent.<br>1 **<Additional_exception_reports>** at maximum rate reached are allowed to be sent. |
| **<Uplink_time_unit>** | Integer type. It specifies the time unit to be used for the maximum uplink rate.<br>0 Unrestricted<br>1 Minute<br>2 Hour |

| | | |
|---|---|---|
| | 3 | Day |
| | 4 | Week |
| **<Maximum_uplink_rate>** | Integer type. Maximum uplink rate. | |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

### Example

```
AT+CGAPNRC
+CGAPNRC:0,0,0

OK
```

## 6.12. AT+CEDRXRDP Read Dynamic eDRX Parameters

The Execution Command returns the parameters **<AcT-type>**, **<Requested_eDRX_value>**, **<NW-provided_eDRX_value>** and **<Paging_time_window>** if eDRX is used for the cell that the UE is currently registered to.

If the cell that the UE is currently registered to is not using eDRX, **<AcT-type>=0** will be returned.

| **AT+CEDRXRDP Read Dynamic eDRX Parameters** | |
|---|---|
| Test Command<br>**AT+CEDRXRDP=?** | Response<br>**OK** |
| Execution Command<br>**AT+CEDRXRDP** | Response<br>**+CEDRXRDP:<AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

**<AcT-type>** Integer type. The type of access technology.

0 Access technology is not using eDRX (the terminal or network does not support eDRX)
If the value is 0, other parameters are omitted.

5 E-UTRAN (NB-S1 mode)

**<Requested_eDRX_value>** String type. Requested eDRX cycle. Half a byte in a 4-bit format. NB-S1 mode.

| Bit | | | | |
|---|---|---|---|---|
| 4 | 3 | 2 | 1 | E-UTRAN eDRX cycle length duration |
| 0 | 0 | 1 | 0 | 20.48 seconds |
| 0 | 0 | 1 | 1 | 40.96 seconds |
| 0 | 1 | 0 | 1 | 81.92 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 |

**<NW-provided_eDRX_value>** String type. eDRX cycle issued by the network. Half a byte in a 4-bit format. NB-S1 mode.

| Bit | | | | |
|---|---|---|---|---|
| 4 | 3 | 2 | 1 | E-UTRAN eDRX cycle length duration |
| 0 | 0 | 1 | 0 | 20.48 seconds |
| 0 | 0 | 1 | 1 | 40.96 seconds |
| 0 | 1 | 0 | 1 | 81.92 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 type. Paging Time Window issued by the network. Half a byte in a 4-bit format. NB-S1 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 |
| **<err>** | Error code. See ***Chapter 10*** for details. | | | | |

**Example**

```
AT+CEDRXRDP=?
OK
```

## 6.13. AT+CCIOTOPT CIoT Optimization Configuration

The Write Command controls which CIoT EPS optimizations the UE indicates as supported and preferred.

The Write Command also controls the URC **+CCIOTOPTI:<supported_Network_opt>**, which is used to indicate the supported CIoT EPS optimizations by the network.

| **AT+CCIOTOPT CIoT Optimization Configuration** | |
|---|---|
| Test Command<br>**AT+CCIOTOPT=?** | Response<br>**+CCIOTOPT:(**list of supported**<n>**s**),(**list of supported **<supported_UE_opt>**s**),(**list of supported **<preferred_UE_opt>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CCIOTOPT?** | Response<br>**+CCIOTOPT:<n>,<supported_UE_opt>,<preferred_UE_opt>** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CCIOTOPT=<n>,[<supported_UE_opt>[,<preferred_UE_opt>]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configurations are saved.<br>The parameter **<n>** is not saved after the module is rebooted. |

## Parameter

| | |
|---|---|
| **<n>** | Integer type. Enables or disables reporting of URC **+CCIOTOPTI**.<br>0 Disable URC<br>1 Enable URC<br>3 Disable reporting and reset the parameters for CIoT EPS optimization to the default values. |
| **<supported_UE_opt>** | Integer type. Indicates the UE's support for CIoT EPS optimizations.<br>1 Support for control plane CIoT EPS optimization.<br>3 Support for both control plane CIoT EPS optimization and user plane CIoT EPS optimization. |
| **<preferred_UE_opt>** | Integer type. Indicates the UE's preference for CIoT EPS optimizations.<br>1 Preference for control plane CIoT EPS optimization.<br>2 Preference for user plane CIoT EPS optimization. |
| **<supported_Network_opt>** | Integer type. Indicates the network's support for CIoT EPS optimizations.<br>0 Not support<br>1 Support control plane CIoT EPS optimization<br>2 Support user plane CIoT EPS optimization<br>3 Support both control plane and user plane CIoT EPS optimizations |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

1. **<preferred_UE_opt>**: Preferred optimization. It cannot be set to zero (indicating no optimization level)
2. The Write Command can also be used to enable or disable reporting **+CCIOTPTI**. The proactively report of URC **+CCIOTOPTI:<supported_Network_opt>** indicates whether the network supports CIoT EPS optimizations.

**Example**

```
AT+CCIOTOPT=0,1,1
OK
AT+CCIOTOPT?
+CCIOTOPT:0,1,1

OK
```

## 6.14. AT+CGCONTRDP Read PDP Context Dynamic Parameters

The Write Command returns the relevant information **<bearer_id>**, **<apn>**, **<local_addr and subnet_mask>**, **<gw_addr>**, **<DNS_prim_addr>**, **<DNS_sec_addr>**, **<IPv4_MTU>**, **<NonIP_MTU>** and **<Serving_PLMN_rate_control_value>** for an active non-secondary PDP context.

If the UE has dual stack capabilities, at least one pair of lines with information is returned per **<cid>**: one line with IPv4 parameters followed by one line with IPv6 parameters.

| AT+CGCONTRDP Read PDP Context Dynamic Parameters | |
|---|---|
| Test Command<br>**AT+CGCONTRDP=?** | Response<br>**+CGCONTRDP:(**list of **<cid>**s associated with active contexts**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution/Write Command<br>**AT+CGCONTRDP[=<cid>]** | Response<br>**+CGCONTRDP:<cid>,<bearer_id>,<apn>[,<local_addr and subnet_mask>[,<gw_addr>[,<DNS_prim_addr>[,<DNS_sec_addr>[,[,[,[,[,<IPv4_MTU>[,[,[,<NonIP_MTU>[,<Servi** |

| | |
|---|---|
| | **ng_PLMN_rate_control_value>]]]]]]]]]]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<cid>** | Integer type. It specifies a particular non-secondary PDP context definition. The parameter is local to the TE-MT interface and is used in other PDP context-related commands. |
| **<bearer_id>** | Integer type. Bearer identification, i.e. the EPS bearer in EPS. |
| **<apn>** | String type. A local name that was used to select the GGSN or the external packet data network. |
| **<local_addr and subnet_mask>** | String type. Shows the IP address and subnet mask of the MT. The string is given as dot-separated numeric parameters in the form:<br>**"a1.a2.a3.a4.m1.m2.m3.m4"** for IPv4 or<br>**"a1.a2.a3.a4.a5.a6.a7.a8.a9.a10.a11.a12.a13.a14.a15.a16.m1.m2.m3.m4.m5.m6.m7.m8.m9.m10.m11.m12.m13.m14.m15.m16"** for IPv6. |
| **<gw_addr>** | String type. Shows the Gateway Address of the MT. The string is given as dot-separated numeric (0–255) parameters. This parameter is omitted when Non-IP is used. |
| **<DNS_prim_addr>** | String type. Shows the IP address of the primary DNS server. This parameter is omitted when Non-IP is used. |
| **<DNS_sec_addr>** | String type. Shows the IP address of the secondary DNS server. This parameter is omitted when Non-IP is used. |
| **<IPv4_MTU>** | Integer type. IPv4 maximum transmission unit size in bytes. This parameter is omitted when Non-IP is used. |
| **<Non-IP_MTU>** | Integer type. Non-IP maximum transmission unit size in bytes. This parameter exists when Non-IP is used. |
| **<Serving_PLMN_rate_control_value>** | Integer type. Maximum number of uplink messages that UE allows to send within 6 minutes. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**Example**

```
AT+CGCONTRDP=0
+CGCONTRDP:0,,"ctnb.mnc011.mcc460.gprs",,,192.0.2.2,192.0.2.0

OK
```

## 6.15. AT+CIPCA Initial PDP Context Activation

The Write Command sets whether the UE establishes a PDN connection during EPS attachment.

| AT+CIPCA Initial PDP Context Activation | |
|---|---|
| Test Command<br>**AT+CIPCA=?** | Response<br>**+CIPCA:(**list of supported **<n>**s**),(**list of supported **<AttachWithoutPDN>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CIPCA?** | Response<br>**+CIPCA:<n>[,<AttachWithoutPDN>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CIPCA=[<n>[,<AttachWithoutPDN>]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configurations are saved automatically. |

### Parameter

| | |
|---|---|
| **<n>** | Integer type. Activation of PDP context upon attachment.<br>3 No change in current setting |
| **<AttachWithoutPDN>** | Integer type. EPS attach with or without PDN connection.<br>0 EPS attach with PDN connection<br>1 EPS attach without PDN connection |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

The **<AttachWithoutPDN>=1** configuration takes effect only if the network supports AttachWithoutPDN.

### Example

```
AT+CIPCA=3,0
OK
AT+CIPCA?
+CIPCA:3,0

OK
AT+CIPCA=?
+CIPCA:(3),(0,1)

OK
```

## 6.16. AT+CGAUTH Define PDP Context Authentication Parameters

This Write Command sets authentication parameters for a PDP context identified by **<cid>** used during the PDP context activation and the PDP context modification procedures.

| **AT+CGAUTH Define PDP Context Authentication Parameters** | |
|---|---|
| Test Command<br>**AT+CGAUTH=?** | Response<br>**+CGAUTH:(**list of supported **<cid>**s**),(**list of supported **<auth_prot>**s**),(**list of supported **<userid>**s**),(**list of supported **<password>**s**)**<br><br>**OK**<br><br>If there is any error: |

| | |
|---|---|
| | **ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+CGAUTH?** | Response<br>**[+CGAUTH:<cid>,<auth_prot>,<userid>,<password>]**<br>**[+CGAUTH:<cid>,<auth_prot>,<userid>,<password>]**<br>**[...]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+CGAUTH=<cid>[,<auth_prot>[,<userid>[,<password>]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configurations are not saved. |

## Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<auth_prot>** | Integer type. Authentication protocol used for this PDP context.<br>0 None. Used to indicate that no authentication protocol is used for this PDP context. Username and password are removed if previously specified<br>1 PAP<br>2 CHAP (not supported) |
| **<userid>** | String type. User name for access to the IP network. The maximum length is 16 bytes. |
| **<password>** | String type. Password for access to the IP network. The maximum length is 16 bytes. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

Currently, **<auth_prot>** cannot be configured to CHAP. If **<auth_prot>** is set to 2, an error is returned.

### Example

```
AT+CGAUTH=0,1,user1234,pass1234
OK
AT+CGAUTH?
+CGAUTH:0,1,"user1234","pass1234"

OK
AT+CGAUTH=?
+CGAUTH:(0-10),(0,1),(0-16),(0-16)

OK
```

## 6.17. AT+CNMPSD No More PS Data

The Execution Command indicates that no application on the UE is expected to exchange data. It is used to request the network to quickly release the RRC connection after data interaction is complete.

| AT+CNMPSD No More PS Data | |
|---|---|
| Test Command<br>**AT+CNMPSD=?** | Response<br>**OK** |
| Execution Command<br>**AT+CNMPSD** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+CNMPSD
OK
```

## 6.18. AT+NCIDSTATUS Get <cid> Status

The Write Command returns the PDP context state of the specified **<cid>**. The Execution Command returns a list of all defined PDP context status.

| AT+NCIDSTATUS Get <cid> Status | |
|---|---|
| Test Command<br>**AT+NCIDSTATUS=?** | Response<br>**+NCIDSTATUS:(**list of supported **<cid>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command/Execution Command<br>**AT+NCIDSTATUS[=<cid>]** | Response<br>**+NCIDSTATUS:<cid>[<status>,[<backoff value>]]**<br>**[+NCIDSTATUS:<cid>[<status>,[<backoff value>]]]**<br>**[...]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<status>** | Integer type.<br>0 Available<br>1 Not exist. This status will also be returned if PDP context is inactive<br>2 Flow control<br>3 Back off |
| **<backoff value>** | Integer type. Remainder time of back off timer/T3396 timer. Only shows when the status is back off. Unit: second. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**Example**

```
AT+NCIDSTATUS
+NCIDSTATUS:0,0

OK
AT+NCIDSTATUS=0
+NCIDSTATUS:0,0

OK
```

## 6.19. AT+NGACTR Report PDP Context Activation/Deactivation Result

The Write Command controls the presentation of an URC **+NGACTR:<cid>,<state>,<result>** when activating or deactivating a PDP context.

| AT+NGACTR Report PDP Context Activation/Deactivation Result | |
|---|---|
| Test Command<br>**AT+NGACTR=?** | Response<br>**+NGACTR:(**list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NGACTR?** | Response<br>**+NGACTR:<n>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NGACTR=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |

| | |
|---|---|
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configurations are saved automatically. |

## Parameter

| | | |
|---|---|---|
| **<n>** | Integer type. | |
| | 0 | Disable URC |
| | 1 | Enable the URC **+NGACTR:<cid>,<state>,<result>** |
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). | |
| **<state>** | Integer type. Indicates the state of PDP context activation. | |
| | 0 | Deactivated |
| | 1 | Activated |
| **<result>** | Integer type. Indicates the result of PDP context activation or deactivation. | |
| | 0 | Success |
| | 1 | Failed reason: no context defined |
| | 2 | Failed reason: context already active |
| | 3 | Failed reason: context not active |
| | 4 | Failed reason: resource error |
| | 5 | Failed reason: local reject |
| | 6 | Failed reason: APN error |
| | 7 | Failed reason: APN backoff running |
| | 8 | Failed reason: IPv4 only |
| | 9 | Failed reason: IPv6 only |
| | 10 | Failed reason: IP only |
| | 11 | Failed reason: Non-IP only |
| | 12 | Failed reason: single IP only |
| | 13 | Failed reason: service error |
| | 14 | Failed reason: max connect reached |
| | 15 | Failed reason: reactivation requested |
| | 16 | Failed reason: last PDN disconnect not allowed |
| | 17 | Failed reason: unknown |
| | 18 | Failed reason: NSLPI override not allowed |
| | 19 | Failed reason: attaching |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

## Example

```
AT+NGACTR=0
OK
AT+NGACTR?
+NGACTR:0
```

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

OK
```

## 6.20. AT+NSNPD Send Non-IP Data

This command is used by the UE to transmit Non-IP data over control plane or user plane to network.

| AT+NSNPD Send Non-IP Data | |
|---|---|
| Test Command<br>**AT+NSNPD=?** | Response<br>**+NSNPD:(**list of supported **<cid>**s**),(**maximum number of bytes of the **<non_ip_data_length>),(**list of supported **<RAI>**s**),(**list of supported **<type_of_user_data>**s**),(**list of supported **<sequence>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSNPD=<cid>,<non_ip_data_length>,<non_ip_data>[,<RAI>[,<type_of_user_data>[,sequence]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<non_ip_data_length>** | Integer type. Indicates the number of bytes of **<non_ip_data>** information element. This parameter indicates the maximum number of bytes to be sent. |

| | |
|---|---|
| | When there is no data to transmit, the value shall be set to zero. The maximun number is 1358. |
| **<non_ip_data>** | Octets string. It contains the user data contents. When there is no data to transmit, the **<non_ip_data>** shall be an empty string (**""**). |
| **<RAI>** | Integer type. Indicates the value of the release assistance indication.<br>0 No indication<br>1 Expects that exchange of data will be completed when the uplink data transmit completes<br>2 Expects that exchange of data will be completed when the downlink responds |
| **<type_of_user_data>** | Integer type. Indicates whether the transmitted user data is regular or exceptional.<br>0 Regular data<br>1 Exception data |
| **<sequence>** | Integer type. Sequence of data. Range: 1–255. If it is omitted, data sending status will not be reported. If it is not omitted, when datagram is acknowledged by server or discard by UE, the result **+NSNPDR:<cid>,<sequence>,<status>** will be reported. |
| **<status>** | Integer type. The status of datagram.<br>0 Error<br>1 Sent |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NSNPD=?
+NSNPD:(0-10),(1358),(0-2),(0,1),(1-255)

OK
```

## 6.21. AT+NRNPDM Set Reporting of Non-IP Data Mode

The Write Command enables or disables reporting of Non-IP data from the network to the UE that is transmitted via the control plane or user plane in downlink direction. If reporting is enabled, the UE returns the URC **+NRNPDM:<cid>,<cpdata_length>,<cpdata>** when data is received from the network.

| **AT+NRNPDM Set Reporting of Non-IP Data Mode** | |
|---|---|
| Test Command<br>**AT+NRNPDM=?** | Response<br>**+NRNPDM:(**list of supported **<reporting>**s**),(**list of supported **<cid>**s**),(**maximum number of sent bytes of **<non_ip_data_length>)** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NRNPDM?** | Response<br>**+NRNPDM:<reporting>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NRNPDM=<reporting>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configurations are saved automatically. |

## Parameter

| | |
|---|---|
| **<reporting>** | Integer type. Control reporting of mobile terminated (MT) Non-IP data events.<br>0 Disable reporting of MT Non-IP data<br>1 Enable reporting of MT Non-IP data by the URC:<br>**+NRNPDM:<cid>,<cpdata_length>,<cpdata>** |
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<non_ip_data_length>** | Integer type. Indicates the number of bytes of **<non_ip_data>** information element. When there is no data to transmit, the value shall be set to zero. Range: 0–1358. |
| **<non_ip_data>** | Octets string. Contains the user data contents. When there is no data to transmit, the **<non_ip_data>** shall be an empty string (**""**). |
| **<cpdata_length>** | Integer type. Indicates the number of bytes of **<cpdata>**. When there is no |

| | |
|---|---|
| | data to transmit, the value shall be set to zero. |
| **<cpdata>** | String type. Control plane data. When there is no data to transmit, the **<cpdata>** shall be an empty string (**""**). |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NRNPDM=?
+NRNPDM:(0,1),(0-10),(1358)

OK
```

## 6.22. AT+NQPNPD Query Pending Non-IP Data List

This command queries the pending Non-IP data list via the control plane or user plane by MT.

| **AT+NQPNPD Query Pending Non-IP Data List** | |
|---|---|
| Test Command<br>**AT+NQPNPD=?** | Response<br>**+NQPNPD:(**list of supported **<cid>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NQPNPD=<cid>** | Response<br>**[+NQPNPD:[<sequence>[,<sequence>…]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<cid>** | Integer type. A numeric parameter which specifies a particular PDP context definition (see **AT+CGDCONT**). |
| **<sequence>** | Integer type. The sequence of pending sending Non-IP data message. Range: 1–255. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NQPNPD=?
+NQPNPD:(0-10)

OK
```

# 7 Socket Commands

## 7.1. AT+NSOCR Create a Socket

This command creates an UDP/TCP Socket. For TCP Socket creation, there is no connection established when executing **AT+NSOCR**, and you need to execute **AT+NSOCO** to establish the TCP connection.

| AT+NSOCR Create a Socket | |
|---|---|
| Write Command<br>**AT+NSOCR=<type>,<protocol>,<listen_port>[,<receive_control>[,<af_type>[,<ip_address>]]]** | Response<br>**<socket>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<type>** | String type. Socket type.<br>DGRAM UDP<br>STREAM TCP |
| **<protocol>** | Integer type. Standard internet protocol definition. For example, UDP is 17, and TCP is 6. |
| **<listen_port>** | Integer type. Local port. Range: 0–65535. If it is 0, the module will assign a random **<listen_port>** for the socket. |
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<receive_control>** | Integer type. Receive control.<br>0 Ignore incoming downlink messages<br>1 Receive incoming downlink messages |
| **<af_type>** | String type. Address family type. |

| | | |
|---|---|---|
| | AF_INET6 | IPv6 |
| | AF_INET | IPv4 |
| **<ip_address>** | String type. IP address. | |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

**NOTE**

When a TCP Socket is created, the module cannot enter deep sleep mode. Disable PSM mode by executing **AT+CPSMS=0** when TCP Socket is used.

**Example**

```
AT+NSOCR=DGRAM,17,10004,1
0

OK
```

## 7.2. AT+NSOST Send Data (UDP Only)

This command is used to send data requests to the remote server through the NB protocol stack.

| **AT+NSOST Send Data (UDP Only)** | |
|---|---|
| Test Command<br>**AT+NSOST=?** | Response<br>**+NSOST:(**list of supported **<socket>**s)**,,(**list of supported **<remote_port>**s)**,(**list of supported **<length>**s)**,,(**list of supported **<sequence>**s)**,(**list of supported **<flag>**s)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOST=<socket>,<remote_addr>,<remote_port>,<length>,<data>[,<sequence>[,<flag>]]** | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>**[+NSOSTR:<socket>,<sequence>,<status>]**<br><br>If there is any error: |

| | |
|---|---|
| | **ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<remote_addr>** | IP address which can be expressed in decimal numbers and dots/hexadecimal numbers and colon. |
| **<remote_port>** | Integer type. Remote port. Range: 0–65535. |
| **<length>** | Integer type. The length of **<data>**. The maximum length is 1358. Unit: byte. |
| **<data>** | Hex string. Data to be transmitted. |
| **<sequence>** | Integer type. Sequence of data. Indicates whether to confirm the data sent status.<br>0 or NULL No confirmation required<br>1–255 Require confirmation. If the URC: **+NSOSTR:<socket>,<sequence>,<status>** is not reported, it indicates that the data is still being sent, and you should not resend commands with the same **<socket>** and **<sequence>** at this time. |
| **<status>** | Integer type. The status of datagram.<br>0 Error<br>1 Sent |
| **<flag>** | Integer type. Specifies the message transmission type. The value of this parameter is in hexadecimal format.<br>0 No flag<br>0x200 Release indication: release after the message is sent<br>0x400 Release indication: release after receiving a reply to the message<br>The parameter is set to zero if **<flag>** is not specified. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+NSOCR=DGRAM,17,10004,1
0

OK
AT+NSOST=0,192.0.2.2,10004,2,AB30
0,2

OK
AT+NSOST=0,192.0.2.2,10004,2,AB30,5
0,2
```

**OK**

**+NSOSTR:0,5,1**

## 7.3. AT+NSOSTF Send Messages with Flags (UDP Only)

The MCU sends a data request, which is ultimately transmitted to the remote server through the NB protocol stack.

| **AT+NSOSTF Send Messages with Flags (UDP Only)** | |
|---|---|
| Test Command<br>**AT+NSOSTF=?** | Response<br>**+NSOSTF:(**list of supported **<socket>**s)**,,(**list of supported **<remote_port>**s)**,(**list of supported **<flag>**s)**,(**list of supported **<length>**s)**,,(**list of supported **<sequence>**s)<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTF=<socket>,<remote_addr>,<remote_port>,<flag>,<length>,<data>[,<sequence>]** | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>**[+NSOSTR:<socket>,<sequence>,<status>]**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<remote_addr>** | IP address which can be expressed in decimal numbers and dots/hexadecimal numbers and colon. |

| | |
|---|---|
| **<remote_port>** | Integer type. Remote port. Range: 0–65535. |
| **<flag>** | Hex string. Specifies the message transmission type. Optional for the user to fill or leave blank.<br>0 No flag<br>0x200 Release indication: release after the message is sent<br>0x400 Release indication: release after receiving a reply to the message |
| **<length>** | Integer type. The length of **<data>**. The maximum length is 1358. Unit: byte. |
| **<data>** | Hex string. Data to be transmitted. |
| **<sequence>** | Integer type. Sequence of data. Indicates whether to confirm the data sent status.<br>0 or NULL No confirmation required<br>1–255 Require confirmation. If the URC: **+NSOSTR:<socket>,<sequence>,<status>** is not reported, it indicates that the data is still being sent, and you should not resend commands with the same **<socket>** and **<sequence>** at this time. |
| **<status>** | Integer type. The status of datagram.<br>0 Error<br>1 Sent |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NSOSTF=0,192.0.2.2,10005,0x200,2,AB30,1
0,2

OK

+NSOSTR:0,1,1
```

## 7.4. AT+NQSOS Query the List of Pending Socket Message

This command queries the Socket data that is being sent and the status of packets that are being sent and do not respond **<sequence>**.

| **AT+NQSOS Query the List of Pending Socket Message** | |
|---|---|
| Test Command<br>**AT+NQSOS=?** | Response<br>**+NQSOS:(**list of supported **<socket>**s**),(**list of supported **<socket>**s**),(**list of supported **<socket>**s**),(**list of supported **<socket>**s**),(**list of supported **<socket>**s**)**<br><br>**OK**<br>If there is any error: |

| | |
|---|---|
| | **ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NQSOS?** | Response<br>**[+NQSOS:<socket>,<sequence>]**<br>**[+NQSOS:<socket>,<sequence>]**<br>**[…]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NQSOS=<socket>[,<socket>[,<socket>[…]]]** | Response<br>**[+NQSOS:<socket>,<sequence>]**<br>**[+NQSOS:<socket>,<sequence>]**<br>**[…]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<sequence>** | Integer type. The sequence of data being sent. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

To check the sending status of data, you should include **<sequence>** in the data being sent. If there is no data being sent, it will return **OK** directly.

## Example

```
AT+NQSOS=0
```

```
OK
AT+NQSOS?
OK
```

## 7.5. AT+NSONMI Set Reporting Mode of Downlink Data

This command sets the reporting mode of downlink data.

| AT+NSONMI Set Reporting Mode of Downlink Data | |
|---|---|
| Test Command<br>**AT+NSONMI=?** | Response<br>**+NSONMI:(**list of supported **<mode>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NSONMI?** | Response<br>**+NSONMI:<mode>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSONMI=<mode>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<mode>** | Integer type. Configure the URC indicator of downlink data. |

| | | |
|---|---|---|
| | 0 | Disable the URC to indicate messages |
| | 1 | Enable URC below to indicate only message length:<br>**+NSONMI:<socket>,<length>** |
| | 2 | Enable the URC below to indicate system address, remote port, data length and data content of the sent message<br>**+NSONMI:<socket>,<remote_addr>,<remote_port>,<length>,<data>** |
| | 3 | Enable the URC below to indicate data length and data content of the message:<br>**+NSONMI:<socket>,<length>,<data>** |
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. | |
| **<remote_addr>** | String type. Address of system sending the message. | |
| **<remote_port>** | Integer type. Remote port on which messages will be received. Range: 0–65535. | |
| **<length>** | Integer type. Number of bytes of data in the first message. | |
| **<data>** | Data received in hex string format. Maximum length of received data is 1358 bytes. | |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

## Example

```
AT+NSONMI=1
OK
AT+NSONMI?
+NSONMI:1

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

OK
```

**NOTE**

1. When creating a socket via **AT+NSOCR** and **<receive_control>** is set to 0, the URC **+NSONMI** will not be reported, and downlink data will be cached.
2. Only when the cache is empty, the module will respond the URC **+NSONMI** to indicate the reception of a new message. If there are messages in the cache, the module will not respond the URC **+NSONMI**. After reading all cached data via **AT+NSORF**, it will continue to report URC **+NSONMI**.
3. In addition to the first point mentioned, when **<mode>** is set to 1, downlink data will also be cached, but the total cached data cannot exceed 2048 bytes, and the total number of cached downlink messages cannot exceed 8. If either of these conditions is exceeded, downlink data will be discarded.

## 7.6. AT+NSORF Receive Data

This command actively reads received downlink data messages, which are processed by the baseboard MCU. This command operates on data nodes within the data and reports the information of the next data node only after the current node has been processed. This process continues until there are no more data nodes (NULL).

| AT+NSORF Receive Data | |
|---|---|
| Test Command<br>**AT+NSORF=?** | Response<br>**+NSORF:(**list of supported **<socket>**s**),(**list of supported **<req_length>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSORF=<socket>,<req_length>** | Response<br>**<socket>,<ip_addr>,<port>,<length>,<data>,<remaining_length>**<br><br>**OK**<br><br>**[+NSONMI:<socket>,<length_next>]**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<req_length>** | The read data in decimal length. Unit: byte. Maximum request data length: 1358 bytes. It reads data corresponding to **<socket>**. If **<req_length>** is greater than or equal to the stored data length, it will read all of it. Otherwise, it will only read the specified length of data and store the remaining data. |
| **<ip_addr>** | Remote IP address. |

| | |
|---|---|
| **<port>** | Remote port. Range: 0–65535. |
| **<length>** | Decimal type. The length of the returned data. Unit: byte. |
| **<remaining_length>** | Decimal type. The remaining length of unread message data. |
| **<data>** | Hex string type. The received data. |
| **<length_next>** | The data length of the next data node. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## 7.7. AT+NSOCO Connect Command (TCP Only)

This command establishes a TCP connection to the specified host and port and notify that the socket is connected. It returns the socket number through the URC **+NSOCO: <socket>**. When disconnecting, please refer to **+NSOCLI** (Response only).

| **AT+NSOCO Connect Command (TCP Only)** | |
|---|---|
| Test Command<br>**AT+NSOCO=?** | Response<br>**+NSOCO:(**list of supported **<socket>**s)**,,(**list of supported **<remote_port>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOCO=<socket>,<remote_addr>,<remote_port>** | Response<br>**OK**<br><br>**+NSOCO: <socket>**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 300 ms |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<remote_addr>** | String type. Address of system sending the message expressed in decimal |

| | |
|---|---|
| | numbers and dots/hexadecimal numbers and colon. |
| **<remote_port>** | Integer type. The remote port to be connected to. Range: 0–65535. It supports both decimal and hexadecimal formats. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NSOCO=1,192.0.2.2,1024
OK

+NSOCO: 1
AT+NSOCO=?
+NSOCO:(0-6),,(0-65535)

OK
```

## 7.8. AT+NSOSD Send Message (TCP Only)

This command sends a TCP datagram to the TCP server. It will return with the socket that it was sent on, and the number of bytes of data sent. The return value of **AT+NSOSD** indicates how much the data was successfully sent.

| **AT+NSOSD Send Message (TCP Only)** | |
|---|---|
| Test Command<br>**AT+NSOSD=?** | Response<br>**+NSOSD:(**list of supported **<socket>**s**),(**list of supported **<length>**s**),,(**list of supported **<flag>**s**),(**list of supported **<sequence>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSD=<socket>,<length>,<data>[,<flag>[,<sequence>]]** | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or |

| | +CME ERROR: <err> |
|---|---|
| Maximum Response Time | 300 ms |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<length>** | Integer type. Decimal length of data to be sent. The default maximum length of send data is 1358 bytes. |
| **<data>** | Hex string type. Data to be transmitted. |
| **<flag>** | Integer type. It specifies the type of message transmission. Values of this parameter are in hex format.<br>0 No flag<br>0x200 Release indication: release after the message is sent<br>0x400 Release indication: release after receiving a reply to the message<br>Release indication is not supported in UP mode. The parameter value will be set to 0 if **<flag>** is not specified. |
| **<sequence>** | Integer type. Sequence of data. Range 0–255. If it is omitted, data sent status will not be reported. If it is specified, when datagram is acknowledged by server or is discarded by UE, the result will be reported:<br>**+NSOSTR:<socket>,<sequence>,<status>** |
| **<status>** | Integer type. The status of datagram.<br>0 Error<br>1 Sent |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

The parameter **<flag>** of this command is currently not effective.

## Example

```
AT+NSOSD=1,2,AB30
1,2

OK
AT+NSOSD=1,2,AB30,0
1,2

OK
AT+NSOSD=1,2,AB30,0,255
```

```
1,2

OK

+NSOSTR:1,255,1
AT+NSOSD=?
+NSOSD:(0-6),(1-1358),,(0,0x200,0x400),(0-255)

OK
```

## 7.9. AT+NSOCL Close a Socket

This command closes the specified socket. If the socket has already been closed, or was never created, an error will be returned.

| AT+NSOCL Close a Socket | |
|---|---|
| Test Command<br>**AT+NSOCL=?** | Response<br>**+NSOCL:(**list of supported **<socket>**s**),(**list of supported **<mode>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOCL=<socket>[,<mode>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 300 ms |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<mode>** | Integer type. Socket closing mode. Applicable only to TCP. If the type of the Socket to |

| | |
|---|---|
| | be closed is UDP, this parameter will be omitted.<br>0 Normally close the TCP Socket and wait for the interaction with the TCP Server to complete<br>1 Forcefully close the TCP Socket immediately |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NSOCL=1
OK
AT+NSOCL=?
+NSOCL:(0-6),(0-1)

OK
```

**NOTE**

After TCP Socket is closed, there will be a URC indication: **+NSOCLI: <socket>**.

## 7.10. AT+NSOSTATUS Get Socket Status

The Write Command returns the Socket status information associated with the specified **<socket>**. The Execution Command returns a list of all supported Socket status.

| **AT+NSOSTATUS Get Socket Status** | |
|---|---|
| Test Command<br>**AT+NSOSTATUS=?** | Response<br>**+NSOSTATUS:(**list of supported **<socket>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTATUS=<socket>** | Response<br>**+NSOSTATUS:<socket><status>**<br><br>**OK**<br><br>If there is any error: |

| | |
|---|---|
| | **ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution Command<br>**AT+NSOSTATUS** | Response<br>**+NSOSTATUS:<socket>,<status>**<br>**[+NSOSTATUS:<socket>,<status>]**<br>**[...]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number. The module supports two Sockets. Range: 0–6. |
| **<status>** | Integer type. Socket status.<br>0 Available<br>1 Not exist. This status is also returned if the socket is not bound to the correct IP address NW assigned<br>2 Flow control (Not supported currently)<br>3 Back off (Not supported currently) |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+NSOSTATUS=0
+NSOSTATUS:0,1

OK
AT+NSOSTATUS=?
+NSOSTATUS:(0-6)

OK
```

## 7.11. AT+QIDNSCFG Configure DNS

This command configures the DNS server address. **<pri_dns>** and **<sec_dns>** are already terminated with the null character "\0".

| AT+QIDNSCFG Configure DNS | |
|---|---|
| Test Command<br>**AT+QIDNSCFG=?** | Response<br>**OK** |
| Read Command<br>**AT+QIDNSCFG?** | Response<br>**PrimaryDns: <pri_dns>**<br>**SecondaryDns: <sec_dns>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+QIDNSCFG=<pri_dns>[,<sec_dns>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configuration is not saved. |

### Parameter

| | |
|---|---|
| **<pri_dns>** | String type. The primary domain name server address. The IPv4 address is expressed in decimal numbers and dots and the IPv6 address is expressed in hexadecimal numbers and colon. |
| **<sec_dns>** | String type. The second domain name server address. The IPv4 address is expressed in decimal numbers and dots and the IPv6 address is expressed in hexadecimal numbers and colon. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

1. It is not allowed to configure the DNS server address during the execution of the **AT+QDNS** command.
2. When the DNS server address network has not been issued by the network, **<pri_dns>** and **<sec_dns>** will return 0.0.0.0 when reading command. If DNS servers are not configured at this point, it may lead to failures in related services that rely on domain name resolution, such as DM and LwM2M functionality.

**Example**

```
AT+QIDNSCFG=?
OK
AT+QIDNSCFG?
PrimaryDns: 192.0.2.2
SecondaryDns: 192.0.2.0

OK
```

## 7.12. AT+QDNS Trigger DNS Domain Name Resolution

This command triggers the DNS domain name resolution. When the domain name resolution is completed, the UE will display the result of domain name resolution: **+QDNS:<result>**. If the result of the domain name resolution is not returned, and then execute the Write Command, it will report **ERROR**.

Each DNS IP is retried three times, with a default interval of 100 seconds between each retry.

| AT+QDNS Trigger DNS Domain Name Resolution | |
|---|---|
| Write Command<br>**AT+QDNS=<mode>[,<hostname>][,<retry>][,<timeout>]]** | Response<br>**OK**<br><br>**+QDNS:<result>**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 300 ms |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<mode>** | Integer type. DNS resolution mode value.<br>0 DNS domain name resolution. **<hostname>** must be specified<br>1 Clear the parsing data. If **<hostname>** is specified, then clear the hostname parsing data in memory and KV. If **<hostname>** is not specified, all parsing data will be cleared |
| **<hostname>** | String type. Domain name. |
| **<retry>** | Integer type. Retry times. Range: 1–3. |
| **<timeout>** | Integer type. Timeout. Unit: second. Range: 10–100. |
| **<result>** | String type. The result of domain name resolution.<br>IP Domain name resolution completes the corresponding IP address.<br>FAIL Domain name resolution failed |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

The module does not come with pre-configured DNS server addresses by default. DNS resolution needs to be carried out using the DNS server addresses provided by the network. If the network does not issue DNS server addresses, you will need to use **AT+QIDNSCFG** to configure DNS server addresses first, or else DNS resolution will not be possible.

### Example

```
AT+QDNS=0,www.example.com
OK

+QDNS:192.0.2.2
```

## 7.13. AT+NSOSTEX Send Messages (UDP Only)

This command sends a UDP datagram containing length bytes of data to the specified host:port. It will return with the socket which it was sent on, and the number of bytes of data sent. If the amount of data is larger than the largest datagram that can be sent, return value of **AT+NSOSTEX** will indicate how much of the data was successfully sent.

| **AT+NSOSTEX Send Messages (UDP Only)** | |
|---|---|
| Test Command<br>**AT+NSOSTEX=?** | Response<br>**+NSOSTEX:(**list of supported **<socket>**s)**,,**(list of supported **<remote_port>**s)**,(**list of supported **<sequence>**s)**,(**list of supported **<length>**s)** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTEX=<socket>,<remote_addr>,<remote_port>,<sequence>,<length>,<data>** | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTEX=<socket>,<remote_addr>,<remote_port>,<sequence>**<br>After **">"** is responded, input the **<data>** to be sent. Tap **"CTRL+Z"** to send, and tap **"ESC"** to cancel the operation. | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTEX=<socket>,<remote_addr>,<remote_port>,<sequence>,<length>**<br>After **">"** is responded, input the **<data>** to be sent. Automatically send when the length reaches **<length>**. | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<remote_addr>** | String type. IP address expressed in decimal numbers and dots/hexadecimal numbers and colon. |

| | |
|---|---|
| **<remote_port>** | Integer type. The remote port on which messages will be received. Range: 0–65535. Decimal and hexadecimal formats are supported. |
| **<length>** | Integer type. Decimal length of data to be sent. The default maximum length of send data is 1358 bytes. |
| **<data>** | Hex string type. Data to be transmitted. |
| **<sequence>** | Integer type. Sequence of data. Range: 0–255. If it set to zero, data sent status will not be reported. If it is set to a non-zero value, when datagram is sent over RF or is discarded, then the result will be reported:<br>**+NSOSTR:<socket>,<sequence>,<status>** |
| **<status>** | Integer type. The status of datagram.<br>0 Error<br>1 Sent |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

The same **<sequence>** cannot be used until the data is sent over RF or discarded, otherwise the AT command will return **ERROR**.

### Example

```
AT+NSOSTEX=1,192.0.2.2,1024,1,2,AB30
1,2

OK

+NSOSTR:1,1,1
AT+NSOSTEX=1,192.0.2.2,1024,1,4
>
AB30
1,4

OK

+NSOSTR:1,1,1
AT+NSOSTEX=1,192.0.2.2,1024,1
>
AB30                //Input "Ctrl"+"Z" or "1A"
1,4

OK

+NSOSTR:1,1,1
```

## 7.14. AT+NSOSTFEX Send Messages with Flags (UDP Only)

This command sends a UDP datagram to the specified host:port and allows meta-data flags to be set. It will return with the socket which it was sent on, and the number of bytes of data sent. If the amount of data is larger than the largest datagram that can be sent, the **AT+NSOSTFEX** return value will indicate how much of the data was successfully sent.

| **AT+NSOSTFEX Send Messages with Flags (UDP Only)** | |
|---|---|
| Test Command<br>**AT+NSOSTFEX=?** | Response<br>**+NSOSTFEX:(**list of supported **<socket>**s**),,(**list of supported **<remote_port>**s**),(**list of supported **<flag>**s**),(**list of supported **<sequence>**s**),(**list of supported **<length>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTFEX=<socket>,<remote_addr>,<remote_port>,<flag>,<sequence>,<length>,<data>** | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTFEX=<socket>,<remote_addr>,<remote_port>,<flag>,<sequence>**<br>After **">"** is responded, input the **<data>** to be sent. Tap **"CTRL+Z"** to send, and tap **"ESC"** to cancel the operation. | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSTFEX=<socket>,<remote_addr>,<remote_port>,<flag>,<sequence>,<length>**<br>After **">"** is responded, input the **<data>** to be sent. Automatically send when the | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error: |

| | |
|---|---|
| length reaches **<length>**. | **ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<remote_addr>** | String type. IP address expressed in decimal numbers and dots/hexadecimal numbers and colon. |
| **<remote_port>** | Integer type. The remote port on which messages will be received. Range: 0–65535. Decimal and hexadecimal formats are supported. |
| **<flag>** | Integer type. It specifies the type of message transmission. Values of this parameter are in hex format<br>0 No flag<br>0x200 Release indication: release after the message is sent<br>0x400 Release indication: release after receiving a reply to the message<br>Release indication is not supported in UP mode. The parameter value will be set to 0 if **<flag>** is not specified. |
| **<sequence>** | Integer type. Sequence of data. Range: 0–255. If it is set to zero, data sending status will not be reported. If it is set to a non-zero value, when datagram is sent over RF or is discarded, then the result will be reported:<br>**+NSOSTR:<socket>,<sequence>,<status>** |
| **<length>** | Integer type. Decimal length of data to be sent. The default maximum length of send data is 1358 bytes, |
| **<data>** | Hex string type. Data to be transmitted. |
| **<status>** | Integer type. The status of datagram.<br>0 Error<br>1 Sent |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

The same **<sequence>** cannot be used until the data is sent over RF or discarded, otherwise the AT command will return **ERROR**.

**Example**

```
AT+NSOSTFEX=1,192.0.2.2,1024,0x200,1,2,AB30
1,2

OK

+NSOSTR:1,1,1
AT+NSOSTFEX=1,192.0.2.2,1024,0x200,1,4
>
AB30
1,4

OK

+NSOSTR:1,1,1
AT+NSOSTFEX=1,192.0.2.2,1024,0x200,1
>
AB30          //Input “Ctrl"+"Z” or “1A”
1,4

OK

+NSOSTR:1,1,1
```

## 7.15. AT+NSOSDEX Send Messages (TCP Only)

This command sends TCP datagram to a TCP server. It returns the socket ID used to send data and length of sent data. If the data size is larger than the allowed maximum datagram, the return value of **AT+NSOSTFEX** informs the length of data actually sent out.

If **<sequence>** is specified, the result will be reported when the server requests a data packet or the UE discards a data packet.

| **AT+NSOSDEX Send Messages (TCP Only)** | |
|---|---|
| Test Command<br>**AT+NSOSDEX=?** | Response<br>**+NSOSDEX:(**list of supported **<socket>**s**),(**list of supported **<flag>**s**),(**list of supported **<sequence>**s**),(**list of supported **<length>**s**)**<br><br>**OK** |

| | |
|---|---|
| | If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSDEX=<socket>,<flag>,<sequece>,<length>,<data>** | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSDEX=<socket>,<flag>,<sequence>**<br>After **">"** is responded, input the **<data>** to be sent. Tap "**CTRL**"+"**Z**" to send, and tap **"ESC"** to cancel the operation. | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSOSDEX=<socket>,<flag>,<sequence>,<length>**<br>After **">"** is responded, input the **<data>** to be sent. Automatically send when the length reaches **<length>**. | Response<br>**<socket>,<length>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<socket>** | Integer type. Socket number returned by **AT+NSOCR**. Range: 0–6. |
| **<flag>** | Integer type. It specifies the type of message transmission. Values of this parameter are in hex format<br>0 No flag<br>0x200 Release indication: release after the message is sent<br>0x400 Release indication: release after receiving a reply to the message |

| | |
|---|---|
| | Release indication is not supported in UP mode. The parameter value will be set to 0 if **<flag>** is not specified. |
| **<sequence>** | Integer type. Sequence of data. Range: 0–255. If it is set to zero, data sending status will not be reported. If it is set to a non-zero value, when datagram is sent over RF or is discarded, then the result will be reported: **+NSOSTR:<socket>,<sequence>,<status>** |
| **<length>** | Integer type. Decimal length of data to be sent. The default maximum length of send data is 1358 bytes. |
| **<data>** | Hex string type. Data to be transmitted. |
| **<status>** | Integer type. The status of datagram.<br>0 Error<br>1 Sent |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

The parameter **<flag>** of this command is currently not effective.

### Example

```
AT+NSOSDEX=1,0,1,2,AB30
1,2

OK

+NSOSTR:1,1,1
AT+NSOSDEX=1,0,1
>
AB30          //Input “Ctrl"+"Z” or “1A”

1,6

OK

+NSOSTR:1,1,1
AT+NSOSDEX=1,0,1,4
>
AB30
1,4

OK

+NSOSTR:1,1,1
```

## 7.16. Matters Needing Attention

- It takes a certain time to release Socket resources in the module. It is recommended to perform the Socket closing operation after completing UDP/TCP Socket business to avoid rapidly creating, sending, and closing Sockets in a short period. This can prevent Socket resource exhaustion, which may lead to a failure in creating a new Socket with **AT+NSOCR**.

- When entering data continuously and at a speed higher than the under layer sending speed, it may result in sending errors due to the send buffer becoming full. It is recommended to wait for a few seconds before sending data again.

- For extended AT commands related to data transmission, it is currently not recommended for users to use ASCII mode for content delivery. This is because special characters within ASCII strings, such as quotes (""), are not controlled and may lead to module exceptions. It is recommended that users use hexadecimal ASCII codes to describe ASCII strings.

# 8 Protocol extension Command

## 8.1. AT+NEARFCN Specify EARFCN

The Write Command provides a mechanism to lock to a specific EARFCN and, if desired, physical cell ID. All actions will be locked to this carrier until either the lock is removed or the UE is rebooted. If the specified EARFCN or PCI is not present, the UE will enter out of service mode.

| **AT+NEARFCN Specify EARFCN** | |
|---|---|
| Test Command<br>**AT+NEARFCN=?** | Response<br>**+NEARFCN:(**list of supported **<search_mode>**s**),(**list of supported **<EARFCN>**s**)[,(**list of supported **<PCI>**s**)]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NEARFCN?** | Response<br>**+NEARFCN:<lock_mode>[,<EARFCN>[,<PCI>]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NEARFCN=<search_mode>,<earfcn>[,<PCI>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |

| | |
|---|---|
| Characteristics | The command takes effect immediately.<br>The configurations are not saved. |

### Parameter

| | |
|---|---|
| **<search_mode>** | Integer type. It specifies the type of search and defines the supplied parameters.<br>0 Lock to a specific EARFCN and PCI |
| **<EARFCN>** | Integer type. EARFCN to be searched. Range: 0–262143.<br>When **<EARFCN>** is 0, remove the locked EARFCN and physical cell ID.<br>The configuration should be consistent with the BAND supported by module. |
| **<PCI>** | Hexadecimal integer type. E-UTRAN physical cell ID. Range: 0–1F7. |
| **<lock_mode>** | Integer type. Report the current lock mode.<br>0 EARFCN and physical cell ID are not locked<br>**+NEARFCN:<lock_mode>** returns after the read command execution<br>1 Lock EARFCN. **+NEARFCN:<lock_mode>,<EARFCN>** returns after the read command execution<br>2 Lock physical cell ID.**+NEARFCN:<lock_mode>,<EARFCN>,<PCI>** returns after the read command execution |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NEARFCN=0,2508
OK
```

## 8.2. AT+NCSEARFCN Clear Stored EARFCN

This command clears stored EARFCN, which must be executed after **AT+CFUN=0** execution.

| **AT+NCSEARFCN Clear Stored EARFCN** | |
|---|---|
| Execution Command<br>**AT+NCSEARFCN** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NCSEARFCN
OK
```

## 8.3. AT+NBAND Set Supported Bands

This command sets the supported bands, which needs to be within the UE capability range, otherwise the configuration will not take effect.

| **AT+NBAND Set Supported Bands** | |
|---|---|
| Test Command<br>**AT+NBAND=?** | Response<br>**+NBAND:(** list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NBAND?** | Response<br>**+NBAND:<n>[,<n>[,<n>[…]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NBAND=<n>[,<n>[,<n>[…]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |

| | |
|---|---|
| Characteristics | This command takes effect after the module is rebooted.<br>The configuration is saved automatically. |

### Parameter

| | |
|---|---|
| **<n>** | Integer type. Band in a decimal number.<br>The actual configuration needs to be based on the capabilities supported by the terminal.<br>When the configured band exceeds the capabilities, **ERROR** returns.<br>You can query the frequency band information supported by the current module through **AT+NBAND=?**. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NBAND=?
+NBAND:(5,8,3,20,28)

OK
AT+NBAND=5
OK
```

## 8.4. AT+NCONFIG Configure UE Behaviour

This command allows configuring certain aspects of UE behavior. It takes a function and a value that controls operation of that function.

| **AT+NCONFIG Configure UE Behaviour** | |
|---|---|
| Test Command<br>**AT+NCONFIG=?** | Response<br>**+NCONFIG:(<function>,(**list of supported **<value>**s**))**<br>**+NCONFIG:(<function>,(**list of supported **<value>**s**))**<br>**[…]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NCONFIG?** | Response<br>**+NCONFIG:<function>,<value>**<br>**[+NCONFIG:<function>,<value>]** |

| | |
|---|---|
| | **[…]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NCONFIG=<function>,<value>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | | |
|---|---|---|
| **<function>** | String type. UE function to be configured. | |
| | AUTOCONNECT | Control whether UE will automatically attempt to connect to the network after power-on or reboot. When enabled, it will set **AT+CFUN=1** and read the PLMN from the USIM |
| | CR_0354_0338_SCRAMBLING | Enable/Disable scrambling mode defined in CR(0354_0338) |
| | CR_0859_SI_AVOID | Enable/Disable scrambling mode defined in CR(0859) |
| | COMBINE_ATTACH | Enable/disable combine attach<br>Not supported currently |
| | CELL_RESELECTION | Enable/disable support for RRC cell reselection |
| | ENABLE_BIP | Enable/disable BIP |
| | MULTITONE | Enable/disable multitone |
| | NAS_SIM_POWER_SAVING_ENABLE | Enable/disable USIM card power saving mode.<br>The value is fixed to TRUE |
| | BARRING_RELEASE_DELAY | Time (in seconds) from barring to timeout.<br>Range: 0-1800. |
| | RELEASE_VERSION | 3GPP protocol version number |
| | SYNC_TIME_PERIOD | The circle to read SIB 16. Unit: second.<br>Range: 0-65535. |

| | | |
|---|---|---|
| | IPV6_GET_PREFIX_TIME | Set the maximum time of getting IPv6 prefix. Unit: second. Range: 0-65535. |
| | PCO_IE_TYPE | Supported PCO type. |
| | T3324_T3412_EXT_CHANGE_REPORT | Enable/Disable the active reporting when T3324/T3412 changes. (Only reports is supported) |
| | NON_IP_NO_SMS_ENABLE | Enable/Disable SMS prohibition in Non-IP. |
| | RAI | Enable/Disable RAI. UP mode is not supported. |
| | NPLMNS_OOS_IND_ENABLE | Enable/Disable the OOS timer status URC **+QPLMNS:<state>[,<OOS_time_step>]** report. See ***Chapter 8.9*** for details about URC. |

**<value>** The value of configured UE function.

The corresponding values of <function> are as follows:

| **<function>** | **<value>** | |
|---|---|---|
| AUTOCONNECT | Bool type. | |
| | TRUE | Enable |
| | FALSE | Disable |
| CR_0354_0338_SCRAMBLING | Bool type. | |
| | TRUE | Enable |
| | FALSE | Disable |
| CR_0859_SI_AVOID | Bool type. | |
| | TRUE | Enable |
| | FALSE | Disable |
| COMBINE_ATTACH | Bool type. The function is not supported currently. | |
| | TRUE | Enable |
| | FALSE | Disable |
| CELL_RESELECTION | Bool type. | |
| | TRUE | Enable |
| | FALSE | Disable |
| ENABLE_BIP | Bool type. | |
| | TRUE | Enable |
| | FALSE | Disable |
| MULTITONE | Bool type. | |
| | TRUE | Enable |
| | FALSE | Disable |
| NAS_SIM_POWER_SAVING_ENABLE | Bool type. | |
| | TRUE | Enable |
| | FALSE | Disable |
| BARRING_RELEASE_DELAY | Integer type. Range: 0–1800. Unit: second. | |
| RELEASE_VERSION | Integer type. | |
| | 13 | Rel13 |
| | 14 | Rel14 |
| SYNC_TIME_PERIOD | Integer type. 0 Indicates disable time synchronization. Range: 0–65535. Unit: minute. | |

| | | |
|---|---|---|
| | IPV6_GET_PREFIX_TIME | Integer type. The maximum time to obtain IPv6. Range: 0–65535. Unit: second. |
| | PCO_IE_TYPE | String type. PCO and EPCO are supported. |
| | T3324_T3412_EXT_CHANGE_REPORT | Bool type.<br>TRUE Enable<br>FALSE Disable |
| | NON_IP_NO_SMS_ENABLE | Bool type.<br>TRUE Enable<br>FALSE Disable |
| | RAI | Bool type.<br>TRUE Enable<br>FALSE Disable |
| | NPLMNS_OOS_IND_ENABLE | Bool type.<br>TRUE Enable<br>FALSE Disable |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

**NOTE**

**<function>** can be configured as CR_0354_0338_SCRAMBLING and CR_0859_SI_AVOID, but the command does not take effect.

### Example

```
AT+NCONFIG?
+NCONFIG:AUTOCONNECT,TRUE
+NCONFIG:CR_0354_0338_SCRAMBLING,TRUE
+NCONFIG:CR_0859_SI_AVOID,TRUE
+NCONFIG:COMBINE_ATTACH,FALSE
+NCONFIG:CELL_RESELECTION,TRUE
+NCONFIG:ENABLE_BIP,FALSE
+NCONFIG:MULTITONE,TRUE
+NCONFIG:NAS_SIM_POWER_SAVING_ENABLE,TRUE
+NCONFIG:BARRING_RELEASE_DELAY,64
+NCONFIG:RELEASE_VERSION,14
+NCONFIG:SYNC_TIME_PERIOD,0
+NCONFIG:IPV6_GET_PREFIX_TIME,15
+NCONFIG:PCO_IE_TYPE,EPCO
+NCONFIG:T3324_T3412_EXT_CHANGE_REPORT,FALSE
+NCONFIG:NON_IP_NO_SMS_ENABLE,FALSE
+NCONFIG:RAI,TRUE
+NCONFIG:NPLMNS_OOS_IND_ENABLE,FALSE

OK
```

```
AT+NCONFIG=?
+NCONFIG:(AUTOCONNECT,(FALSE,TRUE))
+NCONFIG:(CR_0354_0338_SCRAMBLING,(FALSE,TRUE))
+NCONFIG:(CR_0859_SI_AVOID,(FALSE,TRUE))
+NCONFIG:(COMBINE_ATTACH,(FALSE,TRUE))
+NCONFIG:(CELL_RESELECTION,(FALSE,TRUE))
+NCONFIG:(ENABLE_BIP,(TRUE))
+NCONFIG:(MULTITONE,(FALSE,TRUE))
+NCONFIG:(NAS_SIM_POWER_SAVING_ENABLE,(FALSE,TRUE))
+NCONFIG:(BARRING_RELEASE_DELAY,(0-1800))
+NCONFIG:(RELEASE_VERSION,(13,14))
+NCONFIG:(SYNC_TIME_PERIOD,(0-65535))
+NCONFIG:(PCO_IE_TYPE,(PCO,EPCO))
+NCONFIG:(NON_IP_NO_SMS_ENABLE,(FALSE,TRUE))
+NCONFIG:(T3324_T3412_EXT_CHANGE_REPORT,(FALSE,TRUE))
+NCONFIG:(IPV6_GET_PREFIX_TIME,(0-65535))
+NCONFIG:(RAI,(FALSE,TRUE))
+NCONFIG:(NPLMNS_OOS_IND_ENABLE,(FALSE,TRUE))

OK
```

## 8.5. AT+NUESTATS Query UE Statistics

This command fetches the most recent operational statistics. It can take an optional parameter that allows different sets of statistics to be displayed. The **<type>=RADIO** provides the default set of values, and **<type>=ALL** will print all data.

| AT+NUESTATS Query UE Statistics | |
|---|---|
| Test Command<br>**AT+NUESTATS=?** | Response<br>**NUESTATS:(**list of supported **<type>s)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NUESTATS=RADIO**<br>Query specific wireless information | Response<br>**NUESTATS:RADIO,Signal power:<signal_power_in_centibels>**<br><br>**NUESTATS:RADIO,Total power:<total_power_in_centibels>** |

| | |
|---|---|
| | **NUESTATS:RADIO,TX power:<current_Tx_power_level_in_centibels>**<br><br>**NUESTATS:RADIO,TX time:<total_Tx_time_since_last_reboot_in_millisecond>**<br><br>**NUESTATS:RADIO,RX time:<total_Rx_time_since_last_reboot_in_millisecond>**<br><br>**NUESTATS:RADIO,Cell ID:<last_cell_ID>**<br><br>**NUESTATS:RADIO,ECL:<last_ECL_value>**<br><br>**NUESTATS:RADIO,SNR:<last_snr_value>**<br><br>**NUESTATS:RADIO,EARFCN:<last_earfcn_value>**<br><br>**NUESTATS:RADIO,PCI:<last_pci_value>**<br><br>**NUESTATS:RADIO,RSRQ:<rsrq_in_centibels>**<br><br>**NUESTATS:RADIO,OPERATOR MODE:<operator_mode>**<br><br>**NUESTATS:RADIO,CURRENT BAND:<current_band>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NUESTATS=CELL**<br>Query the information of the top 5 cells | Response<br>**NUESTATS:CELL,<earfcn>,<physical_cell_id>,<primarycell>, <RSRP>,<RSRQ>,<RSSI>,<SNR>**<br><br>**[...NUESTATS:CELL,<earfcn>,<physical_cell_id>,<primarycell>,<RSRP>,<RSRQ>,<RSSI>,<SNR>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or |

| | |
|---|---|
| | **+CME ERROR: <err>** |
| Write Command<br>**AT+NUESTATS=THP**<br>Query throughput | Response<br>**NUESTATS:THP,<throughput_type>,<throughput>**<br><br>**[...NUESTATS:THP,<throughput_type>,<throughput>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NUESTATS=BLER**<br>Query block error rate information | Response<br>**NUESTATS:BLER,<block_error_rate_type>,<block_error_rate>**<br><br>**[...NUESTATS:BLER,<block_error_rate_type>,<block_error_rate>]**<br><br>**OK**<br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NUESTATS=APPSMEM**<br>Query dynamic access usage | Response<br>**+NUESTATS:APPSMEM,Heap Size:<heap_size>**<br><br>**+NUESTATS:APPSMEM,Heap Peak Used:<used_size>**<br><br>**+NUESTATS:APPSMEM,Current Allocated:<allocated>**<br><br>**+NUESTATS:APPSMEM,Total Free:<free>**<br><br>**+NUESTATS:APPSMEM,Max Free:<max_free>**<br><br>**+NUESTATS:APPSMEM,Num Allocs:<num_allocs>**<br><br>**+NUESTATS:APPSMEM,Num Frees:<num_frees>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR** |

| | |
|---|---|
| | Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NUESTATS=<type>** | Response<br>**NUESTATS:<type>,<name/value>,<value>[,<value[,<value>[...]]]**<br><br>**[...NUESTATS:<type>,<name/value>,<value>[,<value[,<value>[...]]]]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution Command<br>**AT+NUESTATS** | Response<br>**Signal power:<signal_power_in_centibels>**<br>**Total power:<total_power_in_centibels>**<br>**TX power:<current_Tx_power_level_in_centibels>**<br>**TX time:<total_Tx_time_since_last_reboot_in_millisecond>**<br>**RX time:<total_Rx_time_since_last_reboot_in_millisecond>**<br>**Cell ID:<last_cell_ID>**<br>**ECL:<last_ECL_value>**<br>**SNR:<last_snr_value>**<br>**EARFCN:<last_earfcn_value>**<br>**PCI:<last_pci_value>**<br>**RSRQ:<rsrq_in_centibels>**<br>**OPERATOR MODE:<operator_mode>**<br>**CURRENT BAND:<current_band>**<br><br>**OK**<br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<type>** | Type of data to be displayed as an unquoted string. Supported values of **<type>** are:<br>RADIO Radio specific information. |

When no cell is camped on, this value is invalid

| | |
|---|---|
| CELL | Cell information for the top 5 cells |
| BLER | Block error rate information |
| THP | Throughput |
| APPSMEM | Dynamic memory usage |
| ALL | All information. The value of **<type>** output is the correct one for each data type |

If **<type>=RADIO**, return the default set of values:

| | |
|---|---|
| **<signal_power_in_centibels>** | Reference signal received power. Unit: 0.1 dBm |
| **<total_power_in_centibels>** | Total received signal strength. Unit: 0.1 dBm |
| **<current_Tx_power_level_in_centibels>** | The latest current Tx power level. Unit: 0.1 dBm. |
| **<total_Tx_time_since_last_reboot_in_millisecond>** | Total Tx time since last reboot in millisecond |
| **<total_Rx_time_since_last_reboot_in_millisecond>** | Total Rx time since last reboot in millisecond |
| **<last_cell_ID>** | Last SIB1 cell ID in 28 bits.<br>If the value length is less than 4 bytes, the high bits will be filled with 0 |
| **<last_ECL_value>** | Last ECL. Range: 0–2. |
| **<last_SNR_value>** | Last SNR. Unit: 0.1 dBm. |
| **<last_EARFCN_value>** | Last EARFCN. Range: 0–262143. |
| **<last_PCI_value>** | Last PCI. Range: 0–503. |
| **<RSRQ_in_centibels>** | Reference signal received quality. Unit: 0.1 dBm. |
| **<operator_mode>** | Operator mode for serving cell:<br>0 Unknown mode<br>1 Inband different PCI mode<br>2 Inband same PCI mode<br>3 Guardband mode<br>4 Standalone mode |
| **<current_band>** | The band of service cell. |

If **<type>=CELL**, per-cell information for the top 5 cells is returned in the form: **<earfcn>,<physical_cell_ID>,<primary_cell>,<RSRP>,<RSRQ>[,<RSSI>,<SNR>]** (**<RSSI>** and **<SNR>** are not output in neighbour cell).

| | |
|---|---|
| **<earfcn>** | Absolute radio-frequency channel number. |
| **<physical_cell_ID>** | Physical ID of the cell. |
| **<primary_cell>** | Indicates the current serving cell or neighbor cell.<br>0 the searched neighbor cell<br>1 the current serving cell |
| **<RSRP>** | Reference signal received power. |
| **<RSRQ>** | Reference signal received quality. |
| **<RSSI>** | Received signal strength indicator. |
| **<SNR>** | Signal to noise ratio. |

When **<type>**=BLER, the corresponding relationship between the returned **<block_error_rate_type>** and **<block_error_rate>** is as follows:

| **<block_error_rate_type>** | **<block_error_rate>** |
|---|---|
| **<rlc_ul_bler>** | RLC layer block error rate (uplink). Integer % |
| **<rlc_dl_bler>** | RLC layer block error rate (downlink). Integer % |

| | |
|---|---|
| **<mac_ul_bler>** | MAC layer block error rate (uplink). Integer % |
| **<mac_dl_bler>** | MAC layer block error rate (downlink). Integer % |
| **<total_bytes_transmitted>** | Total bytes transmitted |
| **<total_bytes_received>** | Total bytes received |
| **<transport_blocks_sent>** | Transport blocks sent |
| **<transport_blocks_received>** | Transport blocks received |
| **<transport_blocks_retransmitted>** | Transport blocks retransmitted |
| **<total_ack/nack_messages_received>** | Total ACK/NACK messages received |

If **<type>**=THP, the corresponding relationship between the returned **<throughput_type>** and **<throughput>** is as follows:

| **<throughput_type>** | **<throughput>** |
|---|---|
| **<rlc_ul>** | RLC layer throughput (uplink). Integer bps |
| **<rlc_dl>** | RLC layer throughput (downlink). Integer bps |
| **<mac_ul>** | MAC layer throughput (uplink). Integer bps |
| **<mac_dl>** | MAC layer throughput (downlink). Integer bps |

If **<type>**=APPSMEM, dynamic memory usage, returned entries are:

| | |
|---|---|
| **<heap_size>** | Total stack size |
| **<used_size>** | Used size peak |
| **<allocated>** | Current allocated size. Unit: byte |
| **<free>** | Total free size. Unit: byte |
| **<max_free>** | Max free size. Unit: byte |
| **<num_frees>** | Number of times to free memory |

If **<type>**=BLER, returned entries are:

| | |
|---|---|
| **<band>** | The band number corresponding to the serving cell EARFCN. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+NUESTATS
Signal power:-570
Total power:-462
TX power:-150
TX time:574
RX time:45709
Cell ID:138954162
ECL:0
SNR:225
EARFCN:3686
PCI:207
RSRQ:-108
OPERATOR MODE:4
CURRENT BAND:8

OK
```

```
AT+NUESTATS=CELL
NUESTATS:CELL,3686,207,1,-570,-108,-462,225

OK
AT+NUESTATS=THP
NUESTATS:THP,RLC UL,1053

NUESTATS:THP,RLC DL,202

NUESTATS:THP,MAC UL,1780

NUESTATS:THP,MAC DL,759

OK
```

## 8.6. AT+NCCID USIM Card Identification

Both the execution and read commands read the ICCID of the USIM card. If no USIM card is present, or the USIM card is unreadable, no data is returned.

| AT+NCCID USIM Card Identification | |
|---|---|
| Test Command<br>**AT+NCCID=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NCCID?** | Response<br>**+NCCID:<ICCID>**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution Command<br>**AT+NCCID** | Response<br>**+NCCID:<ICCID>**<br><br>**OK** |

| | |
|---|---|
| | If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<ICCID>** | String type. USIM Card Identification Number. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NCCID
+NCCID:89861118283000384362

OK
AT+NCCID?
+NCCID:89861118283000384362

OK
```

## 8.7. AT+NSET Extend the Configuration

This command extends the configuration. The command functions are different depending on the value of **<function>**.

| **AT+NSET Extend the Configuration** | |
|---|---|
| Write Command<br>**AT+NSET=<function>,<n1>[,<n2>]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NSET=<function>** | Response<br>**[+NSET:<function>,<n1>]** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect immediately.<br>The configuration is saved automatically. |

## Paramete

**<function>** String type. Command function. The meaning and the values are as shown below:

**"**OOS_TIMER" Configure the OOS network search interval.

Command format: **+NSET="OOS_TIMER",<n1>**.

**<n1>** OOS network search interval in decimal. Unit: second.
Range: 0–65535. Default value: 0.
When **<n1>** is 0, it means 60 seconds.

**"**UPREQ" Configure whether the terminal expects data to use UP optimized transmission mode.

Command format: **+NSET="UPREQ",<n1>**

**<n1>** Whether UP optimized transmission mode is expected by the terminal for data to use.
0 Not expected
1 Expected

"NOSIMST" Configure the duration for the module entering deep sleep when there is no card. When the module is booted without a card, it will not be allowed to enter deep sleep until this time has elapsed. The default is infinite (that is, not entering deep sleep mode).

Command format: **+NSET="NOSIMST",<n1>**

**<n1>** Sleep delay duration. Range: 0–127. Unit: second.
127 Infinite
0~126 Specific delay length

"PSRPT" Configure whether to turn on AT to output PS DEBUG printing information.

Command format: **+NSET="PSRPT",<n1>**

**<n1>** whether to turn on AT to output PS DEBUG printing information.
0 Turn off
1 Turn on

After AT port is turned on, the printing information is actively reported: **+PS:<EventType>[,<Para1>[,<Para2>.....]]**.

| | | |
|---|---|---|
| | "INITUSIM" | Configure whether to perform card initialization when the module is waked up from sleep.<br>Command format: **+NSET="INITUSIM",<n1>**<br>**<n1>** Whether to perform card initialization when the module is waked up from sleep.<br>0 Card initialization is not performed when the module is waked from sleep; actions such as card replacement, card insertion and removal are not supported during deep sleep.<br>1 Perform card initialization when the module is waked up from sleep |
| **<err>** | Error code. See ***Chapter 10*** for details. | |

**Example**

```
AT+NSET="OOS_TIMER",125
OK
AT+NSET="OOS_TIMER"
+NSET:"OOS_TIMER",125

OK
```

## 8.8. AT+NPOWERCLASS Set the Mapping for Band and Power Class

This command sets the power class for specified band.

| **AT+NPOWERCLASS Set the Mapping for Band and Power Class** | |
|---|---|
| Test Command<br>**AT+NPOWERCLASS=?** | Response<br>**+NPOWERCLASS:(**list of supported **<band>**s**),(**list of supported **<power_class>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NPOWERCLASS?** | Response<br>**+NPOWERCLASS:<band>,<power_class>**<br>**[…]** |

| | |
|---|---|
| | **OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NPOWERCLASS=<band>,<power class>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | This command takes effect after the module is rebooted.<br>The configuration is saved automatically. |

## Parameter

| | |
|---|---|
| **<band>** | Integer type. The specified band. The actual configuration needs to be based on the capabilities supported by the terminal. If the value of band exceeds the capabilities, **ERROR** returns. |
| **<power_class>** | Integer type. Power class value for band, only 3, 5 and 6 are supported currently.<br>Power class 3 5 6<br>Maximum transmit power 23 20 14 |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

When **<power_class>**=6, setting power for band separately is not supported. If **<power_class>**=6, the power level of all bands needs to be configured to 6; if it is changed from **<poewrclass>**=6 to other values, specify the frequency band. The power level of the specified band is the set value, and the power level of other bands is set to 3.

## Example

```
AT+NPOWERCLASS=3,5
OK
AT+NPOWERCLASS=?
+NPOWERCLASS:(5,8,3,20,28),(3,5,6)

OK
```

## 8.9. AT+QPLMNS Search PLMN

The Write Command starts searching PLMN when UE is out of service. If UE is in service status, the command returns **+CME ERROR: <err>**.

The Read Command returns the current PLMN search status and the remaining time of PLMN search timer.

| AT+QPLMNS Search PLMN | |
|---|---|
| Test Command<br>**AT+QPLMNS=?** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+QPLMNS?** | Response<br>**+QPLMNS:<state>[,<OOS_time_step>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Execution Command<br>**AT+QPLMNS** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 5 s |
| Characteristics | - |

### Parameter

| | |
|---|---|
| **<state>** | Integer type. PLMN status in NAS.<br>0 PLMN searching is inactivated. PLMN is not searched<br>1 PLMN is being searched<br>2 PLMN is selected |

| | |
|---|---|
| | 3 UE is in OOS status. The PLMN search timer is started |
| **<OOS_time_step>** | Integer type. Remaining time of PLMN search timer. Unit: second. Only valid when **<state>**=3. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

# 9 Platform Extended Commands

## 9.1. AT+NPSMR Report Power Saving Mode Status

The Write Command controls the report of **+NPSMR**. If **<n>**=1, **+NPSMR:<mode>** is reported when the power mode is changed.

The Read Command returns **+NPSMR:<n>** when **<n>** is 0, and returns **+NPSMR:<n>,<mode>** when **<n>** is 1.

| AT+NPSMR Report Power Saving Mode Status | |
|---|---|
| Test Command<br>**AT+NPSMR=?** | Response<br>**+NPSMR:(**list of supported **<n>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NPSMR?** | Response<br>**+NPSMR:<n>[,<mode>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NPSMR=<n>** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |

| | |
|---|---|
| Maximum Response Time | 1 s |
| Characteristics | The command takes effect immediately.<br>The configuration is not saved. |

### Parameter

| | |
|---|---|
| **<n>** | Integer type. Enable/disable URC report.<br>0 Disable URC report.<br>1 Enable URC **+NPSMR:<mode>** |
| **<mode>** | Integer type. Power mode of UE.<br>0 Normal mode<br>1 Power saving mode |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NPSMR=1
OK
```

## 9.2. AT+QCHIPINFO Read System Information

This command reads system information, including temperature and battery voltage.

| **AT+QCHIPINFO Read System Information** | |
|---|---|
| Test Command<br>**AT+QCHIPINFO=?** | Response<br>**+QCHIPINFO:(**list of supported **<cmd>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+QCHIPINFO=<cmd>** | Response<br>**+QCHIPINFO:<cmd>,<result>**<br>**[+QCHIPINFO:<cmd>,<result>]**<br><br>**OK**<br><br>If there is any error: |

| | |
|---|---|
| | **ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | The command takes effect immediately.<br>The configuration is not saved. |

**Parameter**

| | |
|---|---|
| **<cmd>** | Integer type. System information to be read.<br>ALL Return all data<br>TEMP Current temperature of the module in degree centigrade<br>VBAT Battery voltage in mV |
| **<result>** | Values corresponding to parameter **<cmd>**. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## 9.3. AT+NPING Test IP Network Connectivity to a Remote Host

This command tests the IP network connectivity to a remote host. After the command is executed, the MCU sends a data request, and finally the data will be sent to the remote server through the NB protocol stack.

| **AT+NPING Test IP Network Connectivity to a Remote Host** | |
|---|---|
| Test Command<br>**AT+NPING=?** | Response<br>**+NPING(IPV4):(**peer address **<remote_address>),(**list of supported **<p_size>**s**),(**list of supported **<timeout>**s**),(**list of supported **<cid>**s**),(**list of supported **<count>**s**),(**list of supported **<flag>**s**)**<br>**+NPING(IPV6):(**peer address**<remote_address>),(**list of supported **<p_size>**s**),(**list of supported **<timeout>**s**),(**list of supported **<cid>**s**),(**list of supported **<count>**s**),(**list of supported **<flag>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command | Response |

| | |
|---|---|
| **AT+NPING=<remote_address>[,<p_size>][,<timeout>[,<cid>[,<count>[,<flag>]]]]** | **OK**<br><br>**+NPING:<remote_address>,<ttl>,<rtt>**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+NPINGERR:<err>[,<err_code>]** |
| Maximum Response Time | 1 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<remote_address>** | String type. The peer address to send messages. Support IPv4 and IPv6 addresses. IP addresses can be specified in dotted decimal or colon hexadecimal. |
| **<p_size>** | Integer type. ICMP data length. Range of 12–1480 for IPv4 and range of 12–1460 for IPv6. Unit: byte. |
| **<timeout>** | Integer type. Maximum time in ms to wait for an echo reply response. It will be determined as no response after the timeout. If it is 0, it means that the Ping service will not be initiated and an error will be returned. |
| **<cid>** | Integer type. Context ID used for Ping. (See **AT+CGDCONT** for details). Range: 0–10. |
| **<count>** | Integer type. The number of Pings sent. Range: 1–5. |
| **<ttl>** | Integer type. TTL received in the response packet. |
| **<rtt>** | Integer type. Elapsed time from packet sent to response received. Unit: ms. |
| **<err>** | Integer type. An integer value to provide some information on why the Ping request failed.<br>1 No response from remote host within timeout period |
| **<err_code>** | If **<err>**=2, more specific failure cause is given in **<err_code>** on why Ping request failed. See ***Chapter 10*** for details. |
| **<flag>** | Integer type. It specifies the type of message transmission.<br>0 Same as normal data transmission<br>0x200 Release Indicator: indicate release after this message has been sent<br>0x400 Release Indicator: indicate release after this message has been replied to |
| **<err>** | Error code. See ***Chapter 10*** for details. |

**NOTE**

1. You can execute **AT+NPINGSTOP** to stop sending Ping packet if needed.
2. Before the Ping packet is sent, the Ping packet service cannot be initiated again. If the current

Ping packet is stopped by **AT+NPINGSTOP**, you can execute **AT+NPING** to continue initiating the next Ping packet service.

3. If the module does not obtain an IP address, it will not send data (you can use **AT+CGPADDR** to query whether the module obtains an IP address).

## 9.4. AT+NPINGSTOP Stop Sending Ping Packets

This command stops sending Ping packets.

| AT+NPINGSTOP Stop Sending Ping Packets | |
|---|---|
| Execution Command<br>**AT+NPINGSTOP** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR:<err>** |
| Maximum Response Time | 1 s |
| Characteristics | - |

**Parameter**

| | |
|---|---|
| **<err>** | Error code. See ***Chapter 10*** for details. |

## 9.5. AT+NATSPEED Configure Baud Rate of UART Port

This command configures baud rate of UART port.

| AT+NATSPEED Configure Baud Rate of UART Port | |
|---|---|
| Test Command<br>**AT+NATSPEED=?** | Response<br>**+NATSPEED:(**list of supported **<baud_rate>**s**),(**list of supported **<timeout>**s**),(**list of supported **<store>**s**),(**list of supported **<sync_mode>**s**),(**list of supported **<stopbits>**s**),(**list of supported **<parity>**s**),(**list of supported **<xonxoff>**s**)**<br><br>**OK** |

| | |
|---|---|
| | If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Read Command<br>**AT+NATSPEED?** | Response<br>**+NATSPEED:<baud_rate>,<sync_mode>,<stopbits>,<parity>[,<xonxoff>]**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NATSPEED=<baud_rate>,<timeout>,<store>,<sync_mode>[,<stopbits>[,<parity>[,<xonxoff>]]]** | Response<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | Refer to the description of **<store>**, the communication activation needs to be performed within the timeout period. |

## Parameter

| | |
|---|---|
| **<baud_rate>** | Integer type. Baud rate of UART port that has been requested. The supported values are 1200, 2400, 4800, 9600, 57600, 115200, 230400, 460800 and 921600. Default value: 9600 |
| **<timeout>** | Integer type. Time of waiting for communication before switching back to the original baud rate. Default value: 3. Maximum value: 30. 0 means using default value. Unit: second. |
| **<store>** | Integer type. Whether to store the configurations to NVM. Stored in NVM by default.<br>0 Do not store to NVM, and the configuration takes effect immediately after setting and re-configuration is needed after rebooting<br>1 Store to NVM, while there is interaction before timeout |
| **<sync_mode>** | Synchronization mode. It is not currently supported to configure this parameter. Default value: 0. |
| **<stopbits>** | Stop bit. Only 2 stop bits (default) are supported. The module with 2-bit stop bit supports communication with external devices with 1, 1.5, and 2 stop bits. |

| | |
|---|---|
| **<parity>** | Integer type. AT UART odd/even parity check.<br>0 No parity check enabled<br>1 Odd parity check<br>2 Even parity check |
| **<xonxoff>** | Integer type. AT UART does not support software (XON/XOFF) flow control. Only 0 is supported and this parameter is not effective actually.<br>0 Software flow control is disabled<br>1 Software flow control is enabled |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+NATSPEED=?
+NATSPEED:(1200,2400,4800,9600,57600,115200,230400,460800,921600),(0-30),(0,1),(0-3),(2),(0-2),(0)

OK
```

**NOTE**

Currently, if the baud rate of AT UART port is greater than 9600 bps, the first wake-up AT is at risk of being lost. It is not recommended to configure a baud rate greater than 9600 bps in low-power usage scenarios.

## 9.6. AT+NPIN PIN Operations

This command verifies, unblocks, changes, enables or disables PIN. A URC is reported when a PIN related command is executed.

| **AT+NPIN PIN Operations** | |
|---|---|
| Test Command<br>**AT+NPIN=?** | Response<br>**+NPIN: (**list of supported **<command>**s**)**<br><br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Write Command<br>**AT+NPIN=<command>,<parameter1>** | Response<br>**OK** |

| | |
|---|---|
| **[,<parameter2>]** | **+NPIN: <npin_result>**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 10 s |
| Characteristics | - |

## Parameter

| | |
|---|---|
| **<command>** | Integer type. PIN operation mode.<br>**<command>** **<parameter1>** **<parameter2>** PIN Operation Mode<br>0 **<pin>** Verify PIN<br>1 **<old_pin>** **<new_pin>** Change PIN<br>2 **<pin>** Enable PIN<br>3 **<pin>** Disable PIN<br>4 **<puk>** **<pin>** Unblock PIN |
| **<parameter1>** | A parameter varies with the setting of **<command>**.<br>When **<command>**=0, 2, and 3, it is **<pin>**.<br>When **<command>**=1, it is **<old_pin>**.<br>When **<command>**=4, it is **<puk>**. |
| **<parameter2>** | A parameter varies with the setting of **<command>**.<br>When **<command>**=0, 2, and 3, it is omitted.<br>When **<command>**=1, it is **<new_pin>**.<br>When **<command>**=4, it is **<pin>**. |
| **<pin>,<old_pin>** | String type in decimal format. PIN. |
| **<new_pin>** | String type in decimal format. New PIN. Length: 4–8. Unit: byte. |
| **<puk>** | String type in decimal format. PUK. The string length is 8 bytes. |
| **<npin_result>** | String type. The result as follows.<br>**OK**<br>**ERROR wrong PIN <pin retries remaining>**<br>**ERROR** |
| **<PIN_retries_remaining>** | Integer type in decimal format. Remained PIN retries. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

## Example

```
AT+NPIN=?
+NPIN: (0,1,2,3,4)
```

OK

## 9.7. AT+QCFG Extended Configuration Settings

This command configures various extended settings of UE.

### 9.7.1. AT+QCFG="slplocktimes" Configure Countdown to Entering Sleep Mode

This command configures the countdown for the UE to enter sleep mode. If you do not perform AT interaction and apply for the lock before the timeout, the lock will be released after the timeout and the module prepares to enter deep sleep.

| AT+QCFG="slplocktimes" Configure Countdown to Entering Sleep Mode | |
|---|---|
| Write Command<br>**AT+QCFG="slplocktimes"[,<slplock_time>]** | Response<br>If the optional parameter is omitted, query the current configuration:<br>**+QCFG: "slplocktimes",<slplock_time>**<br><br>**OK**<br><br>If the optional parameter is specified, configure the countdown to entering sleep mode:<br>**OK**<br><br>If there is any error:<br>**ERROR**<br>Or<br>**+CME ERROR: <err>** |
| Maximum Response Time | 1 s |
| Characteristics | The command takes effect after the module is rebooted. The configuration is saved after the module enters deep sleep mode or is rebooted by **AT+NRB**. |

**Parameter**

| | |
|---|---|
| **<slplock_time>** | Integer type. The countdown for the module to enter sleep mode. Range: 3–60. Default value: 3. Unit: second. |
| **<err>** | Error code. See ***Chapter 10*** for details. |

### Example

```
AT+QCFG="slplocktimes"
+QCFG: "slplocktimes",10

OK
AT+QCFG="slplocktimes",20
OK
AT+QCFG="slplocktimes"
+QCFG: "slplocktimes",20

OK
```

# 10 Error Codes

This chapter introduces the error codes related to BC680Z-EU module, including 3GPP standard AT command error codes, application platform-level extended error codes and customer-defined extended error codes.

The error code judgment of AT commands is not yet complete, and the error code format needs to be specified as follows:

**+CME ERROR: <err>**
**ERROR**

**NOTE**

Characters other than **\r\n** are not allowed before the error code. Otherwise, it will be judged as a non-AT command error code, causing channel 8007 to be busy.

**Table 3: General Error Codes (27.007)**

| <err> | Description |
|---|---|
| 3 | Operation not allowed |
| 4 | Operation not supported |
| 5 | Need to input PIN |
| 23 | Memory failure |
| 30 | No network service |
| 50 | Incorrect parameters |
| 51 | Command implemented but currently disabled |
| 52 | Command aborted by user |
| 100 | Unknown |
| 159 | Uplink busy/flow control |

**Table 4: General Error Codes (27.005)**

| <err> | Description |
|---|---|
| 300 | ME failure |
| 301 | SMS service of ME reserved |
| 302 | Operation not allowed |
| 303 | Operation not supported |
| 304 | Invalid PDU mode parameter |
| 305 | Invalid text mode parameter |
| 310 | USIM not inserted |
| 311 | USIM PIN required |
| 312 | PH-USIM PIN required |
| 313 | USIM failure |
| 314 | USIM busy |
| 315 | USIM wrong |
| 316 | USIM PUK required |
| 317 | USIM PIN2 required |
| 318 | USIM PUK2 required |
| 320 | Memory failure |
| 321 | Invalid memory index |
| 322 | Memory full |
| 330 | SMSC address unknown |
| 331 | No network service |
| 332 | Network timeout |
| 340 | No +CNMA acknowledgement expected |
| 500 | Unknown error |

The error codes listed in the following table are specific error codes for BC680Z-EU module.

**Table 5: Platform-level Extended Error Codes**

| <err> | Description |
|---|---|
| 512 | The inputted parameter of AT command is invalid |
| 514 | Internal error |
| 518 | NB network is abnormal, TCP/IP network is busy |
| 523 | AT command is lost or characters are dirty, which is commonly caused by BBPLL_REF_CLK clock instability when standby wakes up |
| 524 | Currently in single-core mode, only the M3 core works normally, and the incorrect transmission is an AT command that needs to be processed by the DSP core |
| 525 | The AT channel is busy, the last AT request has not been processed yet |
| 526 | Need to reboot the device |
| 529 | FOTA upgrade is currently in progress and no other operation control is accepted |
| 535 | Operation is not allowed, usually the state machine does not match |

**Table 6: Error Codes in +CME ERROR**

| <err> | Description |
|---|---|
| 0 | Phone failure |
| 1 | Can not connect to the phone |
| 2 | Phone adapter connection reserved |
| 3 | Operation not allowed |
| 4 | Operation not supported |
| 5 | PH-SIM PIN code is required |
| 6 | PH-FSIM PIN code is required |
| 7 | PH-FSIM PUK code is required |

| | |
|---|---|
| 10 | SIM card not inserted |
| 11 | PIN code of SIM card is required |
| 12 | PUK code of SIM card is required |
| 13 | SIM card check failure, initialization failure, etc. |
| 14 | SIM card is busy |
| 15 | SIM card error |
| 16 | Password error |
| 17 | PIN2 code of SIM card is required |
| 18 | PUK2 code of SIM card is required |
| 20 | Memory is full |
| 21 | Invalid index |
| 22 | Not found |
| 23 | Memory failure |
| 24 | The text string is too long |
| 25 | There are illegal characters in the text string |
| 26 | The dial number is too long |
| 27 | There are invalid characters in the dial number |
| 30 | No network service |
| 31 | Network timeout |
| 32 | The network is not allowed, only for emergency calls |
| 40 | Network personalization PIN required |
| 41 | Network personalization PUK required |
| 42 | Network subset personalization PIN required |
| 43 | Network subset personalization PUK required |
| 44 | Service provider personalization PIN required |

| | |
|---|---|
| 45 | Service provider personalization PUK required |
| 46 | Corporate personalization PIN required |
| 47 | Corporate personalization PUK required |
| 48 | Secret key is request (Note: This key is required when entering the Contact) |
| 50 | Incorrect parameters |
| 100 | Unknown error |
| 101 | The terminal is not ready |
| 114 | PDP is busy |
| 115 | Cancel PDP activation |
| 132 | This business does not support |
| 133 | This business option is not in the scope of the contract |
| 134 | Network abnormal |
| 148 | Unknown error is related to GPRS business |
| 149 | Authentication failed |
| 171 | The last PDN is not allowed to connect |
| 181 | QCI value is not supported |

# 11 Reboot Reasons

This unsolicited report will be reported when the system is powered on, instructing the system to power on. After the system starts, it reports the startup reason to external users. Under normal case, after the user receives this unsolicited report, it indicates that the module has completed the SIM card initialization and the 3GPP related commands can be used.

For the three cases of soft reboot, watchdog reboot and unknown abnormal reboot, after the external MCU receives the actively reported value during operation, it indicates that the module has rebooted. At this time, the MCU and module status are no longer consistent, and the specific reboot reason needs to be determined, or reboot and reset the whole device. The printed reboot reason is for reference only and cannot be used as the only reason to judge the module reboot.

**Table 7: Reboot Reasons**

| Reboot Reason | Description |
|---|---|
| REBOOT_CAUSE_SECURITY_PMU_POWER_ON_RESET | Normal power-on operation |
| REBOOT_CAUSE_SECURITY_RESET_PIN | PIN reset, for external to perform hard reset of the chip. |
| REBOOT_CAUSE_APPLICATION_AT | Soft reboot. **AT+NRB** triggers a soft reboot. FOTA reboot and assertion reboot will also report this reboot reason. |
| REBOOT_CAUSE_APPLICATION_RTC | RTC wake-up operation |
| REBOOT_CAUSE_SECURITY_EXTERNAL_PIN | External PIN wake-up, often bound to data communication |
| REBOOT_CAUSE_SECURITY_WATCHDOG | Hardware watchdog reboots |

# 12 Examples

## 12.1. Network Attachment

After executing **AT+NCONFIG=AUTOCONNECT,TRUE** and rebooting the module, the module tries to attach the network automatically.

If manual network attachment is required, please disable automatic network attachment through **AT+NCONFIG=AUTOCONNECT,FALSE**. The setting will be saved to NVM and take effect after the module is rebooted by **AT+NRB**.

### 12.1.1. Automatic Network Attachment

The following shows a simple example of automatic network attachment. You can query whether the module has been attached to network by the following commands:

```
REBOOT_CAUSE_APPLICATION_AT
OK
AT+CFUN?                          //Query CFUN status
+CFUN:1

OK
AT+CIMI                           //Query the IMSI number
460113028919256

OK
AT+CEREG?                         //Query the network registration status
+CEREG:0,1

OK
AT+CGPADDR                        //Query the IP address of the module
+CGPADDR:0,192.0.2.2

OK
```

### 12.1.2. Manual Network Attachment

Full-band searching may take more time. Therefore, it is recommended to specify the band(s) to be searched when manual network attachment is selected.

```
AT+NCONFIG=AUTOCONNECT,FALSE          //Disable automatic network attachment
OK
AT+NRB                                //Reboot the module
REBOOT_CAUSE_APPLICATION_AT
OK
AT+NBAND=8                            //Specify the band B5 to be searched
OK
AT+CFUN=1                             //Switch UE to full functionality
OK
AT+CFUN?                              //Query CFUN status
+CFUN:1

OK
AT+CIMI                               //Query the IMSI number
460113028919256

OK
AT+CGATT=1                            //Trigger network attachment
OK
AT+NUESTATS                           //Query the module status
Signal power:-566
Total power:-458
TX power:-140
TX time:319
RX time:10019
Cell ID:138954162
ECL:0
SNR:219
EARFCN:3686
PCI:207
RSRQ:-108
OPERATOR MODE:4
CURRENT BAND:8

OK
AT+CEREG?                             //Query the network registration status
+CEREG:0,1

OK
AT+CGATT?                             //Query the network attachment status
```

```
+CGATT:1

OK
AT+CGPADDR                          //Query the IP address of the module
+CGPADDR:0,192.0.2.2

OK
```

## 12.2. Send/Receive/Read a UDP Message

The following shows a simple example of sending, receiving and reading a UDP message. Once the socket is closed, no response will be received. When LwM2M IoT platform is not used, the registration function of LwM2M IoT platform needs to be disabled before the module is connected to the network.

```
…                                   //Connect to network
AT+CGPADDR                          //Query the IP address of the module
+CGPADDR:0,192.0.2.2

OK
AT+NSOCR=DGRAM,17,10005             //Create a UDP socket and transmit the message
0

OK
AT+NSOST=0,192.0.2.0,8029,5,1234567890      //Send a message
0,5                                 //The data message sent

OK

+NSONMI:0,5                         //Receive the message
AT+NSORF=0,5                        //Read 5 bytes message on Socket 0
0,192.0.2.0,8029,5,1234567890,0

OK
AT+NSOCL=0                          //Close the Socket
OK
```

## 12.3. Release Assistance Indication（RAI）

When the UE has no more data to transmit, it waits for the network to release the connection to enter Idle mode. In order for the network to release the UE to Idle mode quickly to save power, release assistance

indication (RAI) is introduced in Release 13 for control panel CIoT EPS optimization, where UE may include RAI in non-access stratum (NAS) signaling to indicate that no further uplink or downlink data transmission is subsequent to the uplink data transmission is expected, thus helping the network to confirm if the connection can be released.

Release 14 introduces RAI for access stratum (AS) for both control and user panels CIoT EPS optimization.

When AS RAI is configured, UE may trigger a buffer status report (BSR) sized zero byte, indicating to eNB that no further data is expected in UL or DL, so the connection may be released.

The following examples of R13 RAI.

**RAI function operation example of R13**

Execute **AT+CSCON=1** to enable URC reporting function. Based on the URC reporting status, you can determine whether the flag is valid.

```
AT+CSCON=1                                  //Enable URC reporting
OK

+CSCON:0
AT+NSOCR=DGRAM,17,10005                     //Create UDP socket
0

OK
AT+NSOSTF=0,192.0.2.2,8029,0x400,2,AB30     //Send data with 0x400 flag
0,2

OK

+CSCON:1

+NSONMI:0,2                                 //Receive downlink data

+CSCON:0                                    //Release RRC connection immediately
AT+NSORF=0,3                                //Read downlink data
0,192.0.2.2,8029,2,AAAA,0

OK
AT+NSOSTF=0,192.0.2.2,8029,0x200,2,AB30     //Send data with 0x200 flag
0,2

OK

+CSCON:1                          //Establish RRC connection and start sending data
```

**+CSCON:0** //Release RRC connection immediately after data transmission

# 13 Appendix A AT Command Delay Response Description

1. The response time of **AT+CFUN=0** may exceed 5 seconds, but its maximum response time is no more than 10 seconds.

2. **AT+CGACT=1** may keep trying to activate when the network is lost. The maximum response time of this command is 1 minute due to the 1-minute protection timer of the module.

3. The maximum response time for sending a short message command **AT+CMGS=<length><CR>PDU is given<ctrl-Z/ESC>** is 1 minute.

4. The response time of the PIN related command will not exceed 10 seconds due to the protection timer of the module.

5. The response time of other commands is within 1 second.

To sum up:

- Commands with a maximum response time of 10 seconds:
  1. PIN related command: **AT+CPIN**
  2. **AT+CFUN=0**

- Commands with a maximum response time of 1 minute:
  1. **AT+CGACT=1**
  2. **AT+CMGS**

- The response time of other commands is no more than 1 second.

# 14 Appendix B Terms and Abbreviations

**Table 8: Terms and Abbreviations**

| Abbreviation | Description |
|---|---|
| 3GPP | 3rd Generation Partnership Project |
| APDU | Application Protocol Data Unit |
| API | Application Programming Interface |
| APN | Access Point Name |
| BCD | Binary-Coded Decimal |
| BIP | Bearer Independent Protocol |
| BS | Bootstrap |
| BSR | Buffer Status Report |
| CHAP | Challenge-Handshake Authentication Protocol |
| CIoT | Cellular Internet of Things |
| CoAP | Constrained Application Protocol |
| CP | Control Plane |
| CRC | Cyclic Redundancy Check |
| CS | Circuit Switched |
| DCE | Data Communication Equipment |
| DF | Dedicated File |
| DFOTA | Delta Firmware Upgrade Over-The-Air |

| | |
|---|---|
| DL | Downlink |
| DNS | Domain Name Server |
| DRX | Discontinuous Reception |
| DTLS | Datagram Transport Layer Security |
| EARFCN | E-UTRA Absolute Radio Frequency Channel Number |
| ECL | Enhanced Coverage Level |
| eDRX | Extended Discontinuous Reception |
| EGPRS | Enhanced General Packet Radio Service |
| EHPLMN | Equivalent Home PLMN (Public Land Mobile Network) |
| EMM | EPS Mobility Management |
| EMS | EPS Session Management |
| eNB | Evolved Node B |
| EPS | Evolved Packet System |
| ESM | EPS Session Management |
| E-UTRA(N) | Evolved Universal Terrestrial Radio Access (Network) |
| FOTA | Firmware Over-The-Air |
| FSM | Finite State Machine |
| GBR | Guaranteed Bit Rate |
| GGSN | Gateway GPRS Support Node |
| GMT | Greenwich Mean Time |
| GPRS | General Packet Radio Service |
| GSM | Global System for Mobile Communications |
| HARQ | Hybrid Automatic Repeat reQuest |
| HPLMN | Home Public Land Mobile Network |
| ICCID | Integrated Circuit Card Identifier |

| | |
|---|---|
| ICMP | Internet Control Messages Protocol |
| ID | Identifier |
| IE | Information Element |
| IMEI | International Mobile Equipment Identity |
| IMEISV | International Mobile Equipment Identity and Software Version |
| IMSI | International Mobile Subscriber Identity |
| IoT | Internet of Things |
| IP | Internet Protocol |
| IPv4 | Internet Protocol version 4 |
| IPv6 | Internet Protocol version 6 |
| IRA | International Reference Alphabet |
| ISIM | IM (IP Multimedia) Service Identity Module |
| LED | Light Emitting Diode |
| LwIP | Lightweight IP (Internet Protocol) |
| LwM2M | Lightweight Machine to Machine |
| MAC | Medium Access Control |
| MBR | Maximum Bit Rate |
| MCU | Microprogrammed Control Unit |
| MQTT | Message Queuing Telemetry Transport |
| MS | Mobile Station |
| MSL | Maximum Segment Lifetime |
| MT | Mobile Termination |
| MTU | Maximum Transfer Unit |
| NAS | Non-access stratum |
| NAT | Network Address Translation |

| | |
|---|---|
| NB-IoT | Narrow Band Internet of Thing |
| NITZ | Network Identity and Time Zone |
| NSLPI | NAS Signaling Low Priority Indication |
| NTP | Network Time Protocol |
| NVM | Non-volatile Memory |
| Oos | Out of Service |
| OTDOA | Observed Time Difference of Arrival |
| PAD | Packet Assembler Disassembler |
| PAP | Password Authentication Protocol |
| PCI | Physical Cell Identity |
| PCO | Protocol Configuration Options |
| P-CSCF | Proxy-Call Session Control Function |
| PDCP | Packet Data Convergence Protocol |
| PDN | Packet Data Network |
| PDP | Packet Data Protocol |
| PDU | Packet Data Unit |
| PF | Packet Filter |
| PIN | Personal Identification Number |
| PLMN | Public Land Mobile Network |
| PPP | Point-to-Point Protocol |
| PS | Packet Switch |
| PSM | Power Saving Mode |
| PSK | Pre-shared Key |
| PTW | Paging Time Window |
| PUK | PIN Unlock Key |

| | |
|---|---|
| PWM | Pulse Width Modulation |
| QCI | QoS Class Identifier |
| QoS | Quality of Service |
| RAI | Release Assistant Indication |
| RAT | Radio Access Technology |
| RF | Radio Frequency |
| RFC | Request for Comments |
| RLC | Radio Link Control |
| RP | Relay Protocol |
| RPM | RPM Package Manager (originally Red Hat Package Manager) |
| RRC | Radio Resource Control |
| RTC | Real Time Clock |
| RX | Receive |
| RxQual | Receive Quality |
| SC | Service Center |
| SIB | System Information Block |
| SMS | Short Message Service |
| SMSC | Short Message Service Center |
| SN | Serial Number |
| SNDCP | Sub-Network Dependent Convergence Protocol |
| SNR | Signal-to-Noise Ratio |
| SNTP | Simple Network Time Protocol |
| SVN | Software Version Number |
| TA | Terminal Adapter |
| TAC | Tracking Area Code |

| | |
|---|---|
| TAU | Tracking Area Update |
| TCP | Transmission Control Protocol |
| TE | Terminal Equipment |
| TFT | Traffic Flow Template |
| TP | Transport |
| TPDU | Transport Protocol Data Unit |
| TTL | Time to Live |
| TUP | Telephone User Part |
| TX | Transmit |
| UART | Universal Asynchronous Receiver/Transmitter |
| UDP | User Datagram Protocol |
| UE | User Equipment |
| UICC | Universal Integrated Circuit Card |
| UL | Uplink |
| UMTS | Universal Mobile Telecommunications System |
| UP | User Plane |
| URC | Unsolicited Result Code |
| URL | Uniform Resource Locator |
| USIM | Universal Subscriber Identity Module |
| UTC | Coordinated Universal Time |
| UUID | Universally Unique Identifier |
| VPLMN | Visiting Public Land Mobile Network |
| WDT | Watchdog Timer |
| WIM | Wireless Identity Module |
| XOFF | Transmit off |

| | |
|---|---|
| XON | Transmit on |
| XOR | Exclusive-OR (gate) |