BC95-G&BC68
AT Commands Manual

NB-IoT Module Series

Rev. BC95-G&BC68_AT_Commands_Manual_V1.5

Date: 2020-02-24

Status: Released

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

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

Or our local office. 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 to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

Revision History

Revision	Date	Author	Description
1.0	2017-12-28	Hayden WANG/ Gary TANG/ Arnold ZHAO	Initial
1.1	2018-03-15	Oven TAO/ Waner PAN/ Hayden WANG/ Donald TANG	1. Added the following AT commands: AT+NQSOS/ +NSOCO/+NSOSD/+NMGS/+NMGR/+NNMI/ +NSMI/+NQMGR/+NQMGS/+NMSTATUS 2. Updated parameter and/or example description for the following AT commands: AT+CGMR/+CGDCONT/+CNMA/+NUESTATS/+NSOCR/+NSOST/+NSOSTF/+NCONFIG/+NPTWEDRXS/ +QLWSREGIND/+QLEDMODE/+QLWSREGIND/+QLWULDATA/+QLWULDATAEX/+QLWULDATASTATUS/ +QLWFOTAIND/ 3. Added examples in Chapter 9.2. 4. Deleted URC +QLWULDATAEXIND.
1.2	2018-07-04	Hayden WANG/ Evan WU/ Donald TANG	1. Added the following AT commands: AT+CGCONTRDP/+CGAUTH/CNMPSD/+NCSEARFCN/+NIPINFO/+NCPCDPR/+NQPODCP/+QDNS/+QRESETDTLS/+QDTLSSTAT/+QBOOTSTRAPHOLDOFF/+QLWSERVERIP/+QSETBSPSK/ +QBSSECSWT. 2. Added the following URC: +NSOCLI. 3. Updated the parameter and example description for the following AT commands: AT +NSOCR/ +NBAND/+NFWUPD/+NPOWERCLASS. 4. Updated error values in Chapter 6. 5. Updated the display of reboot message. 6. Deleted the original Chapter 2. 7. Updated/added examples in Chapter 8.
1.3	2018-09-17	Evan WU /Hayden WANG	1. Added the following AT command: AT+QCHIPINFO/+NSONMI. 2. Updated the parameter description for the following AT commands:

			AT+CSODCP/+QLWULDATAEX/+QSECSWT. 3. Updated the note description for AT command AT+COPS.
1.4	2018-12-12	Evan WU /Hayden WANG	1. Added the following AT commands: AT+CPIN/+CPINR/+NITZ/+QCRITICALDATA/+QIDNSCFG/+QCFG. 2. Updated the description and added write command for AT command AT+CCLK. 3. Added the configuration item for the follow AT commands: AT+NCONFIG/+NUESTATS. 4. Updated the parameter description for the follow AT commands: AT+CGDCONT/+CMGS/CMGC/+NPOWERCLASS/+NSOCR/+QDNS. 5. Updated the note description for AT command AT+CGACT/+NSOST/+NSOSTF/+QREGSWT. 6. Updated the response time for AT command AT+CFUN. 7. Deleted the * mark for the following AT commands: AT+CPIN/+CSODCP/+CRTDCP. 8. Updated error values in Chapter 6.
1.5	2020-02-24	Berg LIU /Hayden WANG	1. Added the value "SIM PUK BLOCKED" for the parameter <code> of AT+CPIN command. 2. Added the value "2" for the parameter <mode> of AT+QDNS command. 3. Updated the description of the parameter <flag> of AT+NSOSTF command. 4. Updated the description of the value “0” for the parameter <cid> of AT+CGDCONT command. 5. Updated the description of the parameter <listen port> of AT+NSOCR command and the note in Chapter 4.4. 6. Added the following AT commands: AT+CGDATA/+CGCMOD/+CCIOTOPT/+CGTFT/+CGEQOS/+CCHO/+CCHC/+CGLA/+NSNPD/+NRNPDM/+NQPNPD/+CSIM/+CRSM/+NUICC/+CMMS/+NSOSTATUS/+NCIDSTATUS/+NGACTR. Deleted AT+QCHIPINFO command. 7. Added the values “2400” and “921600” for the parameter <baud_rate> of AT+NATSPEED command. 8. Added the following values for the parameter <function> of AT+NCONFIG command: “T3324_T3412_EXT_CHANGE_ REPORT”, “NON_IP_NO_SMS_ENABLE”,

		“SUPPORT_SMS”, “TWO_HARQ”, “HPPLMN_SEARCH_ENABLE”. 9. Modified the unit of “SYNC_TIME_PERIOD” value of the parameter <function> of AT+NCONFIG command from hour to minute. 10. Added error code from 533 to 537.

Contents

Table Index

1 Introduction

This document provides details of the AT commands supported by Quectel NB-IoT BC95-G and BC68 modules.

After the module is booted, the string **<CR><LF>Neul<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 of an error occurring in the use of AT commands, please refer to ***Chapter 6*** for possible **<err>** values.

If BC95-G or BC68 module is rebooted due to any abnormal power-on sequence, a message that indicates the reason of the reboot will be output before the string **<CR><LF>Neul<CR><LF>OK<CR><LF>**. Please refer to ***Chapter 7*** for more details.

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

Table 1: Unsolicited Result Code for UE Update State Indication

Unsolicited Information	Description
<CR><LF>FIRMWARE DOWNLOADING<CR><LF>	Indicates that UE 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 UE 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 information.

NOTE

During updating procedure, the module should not be operated until **FIRMWARE UPDATE OVER** is reported. For instance, executing **AT+NRB** to power off the module is not permitted, otherwise an error will occur.

1.1. Definitions

- **<CR>**: Carriage return character;
- **<LF>**: Line feed character;
- **<...>**: Parameter name. Angle brackets do not appear on command line;
- **[...]**: Optional parameter. Square brackets do not appear on command line.

1.2. AT Command Syntax

Table 1: AT Command Syntax

Test Command	**AT+<cmd>=?**	Check list or range of supported parameters for corresponding write command.
Read Command	**AT+<cmd>?**	Check current parameter values of corresponding write command.
Write Command	**AT+<cmd>=<...>**	Write command.
Execution Command	**AT+<cmd>**	Execution commands without configurable parameter

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

When entering AT commands, spaces are ignored except the following cases:

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

They can be used to make the input more human-readable. 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.

NOTE

The underlined values of AT command parameters refer to default values.

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 informational 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. 3GPP Compliance

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

2 3GPP Commands (27.007)

2.1. ATI Display Product Identification Information

The execution command returns product identification information.

ATI Display Product Identification Information	
Execution Command **ATI**	Response **Quectel** **<Object Id>** **Revision:<revision>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<Object Id>	Identifier of device type.
<revision>	Revision of software release.

Example

```
ATI
Quectel
BC95-G
Revision:BC95GJBR02A01

OK
```

2.2. ATE Set Command Echo Mode

The execution command determines whether the UE echoes characters will be received from external MCU or not during command state.

ATE Set Command Echo Mode	
Execution Command **ATE<value>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<value>	0	Echo mode OFF
	1	Echo mode ON

Example

```
ATE0
OK
ATI
Quectel
BC95-G
Revision:BC95GJBR02A01

OK
ATE1
OK
ATI
ATI
Quectel
BC95-G
Revision:BC95GJBR02A01

OK
```

2.3. AT+CGMI Request Manufacturer Identification

The execution command returns manufacturer information. By default, it will return **Quectel** on the standard platform.

AT+CGMI Request Manufacturer Identification	
Execution Command **AT+CGMI**	Response **<manufacturer>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGMI=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<manufacturer>	Manufacturer information. The message text (including line terminators) shall not exceed 2048 characters, and shall not contain the sequence **0<CR>** or **OK<CR>**.

Example

```
AT+CGMI
Quectel

OK
```

2.4. AT+CGMM Request Manufacturer Model

The execution command returns manufacturer model information.

AT+CGMM Request Manufacturer Model	
Execution Command **AT+CGMM**	Response **<model>**

	OK If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGMM=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<model>	Manufacturer model information. The message text (including line terminators) shall not exceed 2048 characters, and shall not contain the sequence **0<CR>** or **OK<CR>**.

Example

```
AT+CGMM
BC95GJB-02-STD

OK
```

2.5. AT+CGMR Request Manufacturer Revision

The execution command returns the manufacturer revision. The text is human-readable and is not intended for microcontroller parsing. By default, the firmware revision (including cores and corresponding revisions) will be returned.

The execution command returns one or more lines of information text **<mfr_revision>**.

AT+CGMR Request Manufacturer Revision	
Execution Command **AT+CGMR**	Response **<mfr_revision>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Test Command **AT+CGMR=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<mfr_revision>	Manufacturer revision. The message text (including line terminators) shall not exceed 2048 characters, and shall not contain the sequence **0<CR>** or **OK<CR>**. The format of **<mfr_revision>** may be changed over time. It should be treated as an opaque identifier.

Example

```
AT+CGMR
SSB,V150R100C10B200SP1

SECURITY_A,V150R100C20B500SP1

PROTOCOL_A,V150R100C20B500SP1

APPLICATION_A,V150R100C20B500SP1

SECURITY_B,V150R100C20B500SP1

RADIO,Hi2115_RF0

OK
AT+CGMR=?
OK
```

2.6. AT+CGSN Request Product Serial Number

The execution command returns the IMEI number and related information. For UE which does not support **<snt>**, only **OK** will be returned.

AT+CGSN Request Product Serial Number	
Execution/Write Command **AT+CGSN[=<snt>]**	Response When **<snt>=0** (or omitted) and the command is executed successfully: **<sn>**

	When **<snt>=1** and the command is executed successfully: **+CGSN:<imei>** When **<snt>=2** and the command is executed successfully: **+CGSN:<imeisv>** When **<snt>=3** and the command is executed successfully: **+CGSN:<svn>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGSN=?**	Response When UE supports **<snt>** and the command is executed successfully: **+CGSN:(**list of supported **<snt>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<snt>	Integer type. The serial number type that has been requested. 0 Returns **<sn>** 1 Returns the IMEI number 2 Returns the IMEISV number 3 Returns the SVN (Software Version Number)
<sn>	The 128-bit UUID of the UE. The message text (including line terminators) shall not exceed 2048 characters, and shall not contain the sequence **0<CR>** or **OK<CR>**.
<imei>	String type in decimal format indicating the IMEI number
<imeisv>	String type in decimal format indicating the IMEISV number
<svn>	String type in decimal format indicating the current SVN which is a part of IMEISV

Example

```
AT+CGSN=1                    //Request the IMEI number
+CGSN:490154203237511

OK
```

2.7. AT+CEREG EPS Network Registration Status

The write command controls the presentation of a URC (unsolicited result code) **+CEREG:<stat>** when **<n>=1** and there is a change in the UE's EPS network registration status in E-UTRAN, or URC **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>]]** when **<n>=2** and there is a change of the network cell in E-UTRAN. The parameters **<AcT>**, **<tac>** and **<ci>** are provided only if available. The value **<n>=3** further extends the URC with **[,<cause_type>,<reject_cause>]** if available, when the value of **<stat>** changes.

If the UE requests PSM for reducing its power consumption, the write command controls the presentation of the URC **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>], [<Periodic-TAU>]]]]**.

When **<n>=4**, the URC will provide the UE with additional information for the active time value and the extended periodic TAU value if there is a change of the network cell in E-UTRAN. The value **<n>=5** further enhances the URC with **<cause_type>** and **<reject_cause>** when the value of **<stat>** changes. The parameters **<AcT>**, **<tac>**, **<ci>**, **<cause_type>**, **<reject_cause>**, **<Active-Time>** and **<Periodic-TAU>** are provided only if available.

The read command returns the status of result code presentation and an integer **<stat>** which shows whether the network has currently indicated the registration of the UE. Location information parameters **<tac>**, **<ci>** and **<AcT>**, if available, are returned only when **<n>=2** and UE is registered on the network. The parameters **<cause_type>** and **<reject_cause>**, if available, will be returned when **<n>=3**.

The test command returns supported parameter values.

AT+CEREG EPS Network Registration Status	
Write Command **AT+CEREG=<n>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CEREG?**	Response When **<n>=0**, **1**, **2** or **3** and the command is executed

	successfully: **+CEREG:<n>,<stat>[,[<tac>],[<ci>],[<AcT>[,<cause_type>,<reject_cause>]]]** When **<n>=4** or **5** and the command is executed successfully: **+CEREG:<n>,<stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CEREG=?**	Response **+CEREG:(**list of supported **<n>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type 0 Disable network registration URC 1 Enable network registration URC: **+CEREG:<stat>** 2 Enable network registration and location information URC: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>]]** 3 Enable network registration, location information and EMM cause value information URC: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]** 4 For a UE that requests PSM, enable network registration and location information URC: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]** 5 For a UE that requests PSM, enable network registration, location information and EMM cause value information URC: **+CEREG:<stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<**

	Active-Time>],[<Periodic-TAU>]]]]
<stat>	Integer type. The EPS registration status. 0 Not registered, UE is not currently searching an operator to register to 1 Registered, home network 2 Not registered, but UE is currently trying to attach or searching an operator to register to 3 Registration denied 4 Unknown (e.g. out of E-UTRAN coverage) 5 Registered, roaming
<tac>	String type. Two bytes tracking area code in hexadecimal format (e.g. "00C3" equals 195 in decimal).
<ci>	String type. Four bytes E-UTRAN cell ID in hexadecimal format.
<AcT>	Integer type. The access technology of the serving cell. 7 E-UTRAN 9 E-UTRAN (NB-S1 mode)
<cause_type>	Integer type. The type of **<reject_cause>**. 0 Indicates that **<reject_cause>** contains an EMM cause value 1 Indicates that **<reject_cause>** contains a manufacturer-specific cause value
<reject_cause>	Integer type. Contains the cause of the failed registration. The value is of type as defined by **<cause_type>**.
<Active-Time>	String type. One byte in an 8-bit format. Indicates the active time value (T3324) allocated to the UE in E-UTRAN. The active time value is coded as one byte (octet 3) of the GPRS Timer 2 information element coded as bit format (e.g. "00100100" equals 4 minutes). For the coding and the value range, see *3GPP TS 24.008*. See also *3GPP TS 23.682* and *3GPP TS 23.401*. Bits 5 to 1 represent the binary coded timer value. Bits 6 to 8 define the timer value unit for the GPRS timer as follows: Bits 8 7 6 0 0 0 value is incremented in multiples of 2 seconds 0 0 1 value is incremented in multiples of 1 minute 0 1 0 value is incremented in multiples of decihours 1 1 1 value indicates that the timer is deactivated Other values shall be interpreted as multiples of 1 minute in this version of the protocol.
<Periodic-TAU>	String type. One byte in an 8-bit format. Indicates the extended periodic TAU value (T3412) allocated to the UE in E-UTRAN. The extended periodic TAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). For the coding and the value range, see *3GPP TS 24.008*. See also *3GPP TS 23.682* and *3GPP TS 23.401*. Bits 5 to 1 represent the binary coded timer value. Bits 6 to 8 define the timer value unit for the GPRS timer as follows: Bits 8 7 6

	0	0	0	value is incremented in multiples of 10 minutes
	0	0	1	value is incremented in multiples of 1 hour
	0	1	0	value is incremented in multiples of 10 hours
	0	1	1	value is incremented in multiples of 2 seconds
	1	0	0	value is incremented in multiples of 30 seconds
	1	0	1	value is incremented in multiples of 1 minute
	1	1	0	value is incremented in multiples of 320 hours
	1	1	1	value indicates that the timer is deactivated

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
```

2.8. AT+CSCON Signalling Connection Status

The command gives details of the radio connection status perceived by the UE (i.e. to the base station). It returns an indication of the current state. Please note that this state is only updated when radio events, such as data sending and receiving, take place. This means that the current state may be out of date. The module may not be able to use a base station due to changes in link quality even when a status of "Connected" is returned.

The write command controls the presentation of URC. If setting fails, a UE error, **+CME ERROR:<err>** is returned.

When the UE is in E-UTRAN, the mode of the UE refers to idle when no PS signalling connection and to connected mode when a PS signalling connection between UE and network is set up.

The **<state>** value indicates state of the UE when the UE is in E-UTRAN.

The read command returns the status of result code presentation and an integer **<mode>** which shows whether the UE is currently in idle mode or connected mode.

The test command returns supported values as a compound value.

AT+CSCON Signalling Connection Status	
Write Command **AT+CSCON=<n>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CSCON?**	Response **+CSCON:<n>,<mode>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CSCON=?**	Response **+CSCON:**(list of supported **<n>**s) **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. Enable/disable the URC	
	0	Disable the URC
	1	Enable the URC **+CSCON:<mode>**
<mode>	Integer type. The signalling connection status.	
	0	Idle
	1	Connected
	2-255	**<reserved for future use>**

Example

```
AT+CSCON=0
OK
```

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

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

OK
AT+CSCON=1
OK
AT+CSCON?
+CSCON:1,1

OK
```

2.9. AT+CLAC List Available Commands

The command lists the available AT commands. The execution command causes the UE to return one or more lines of AT commands. Please note that this command only returns the AT commands that are available for the user.

AT+CLAC List Available Commands	
Execution Command **AT+CLAC**	Response **<AT Command>** **[<AT Command>]** **[...]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CLAC=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<AT Command>	Defines the AT command including the prefix AT. Text shall not contain the sequence **0<CR>** or **OK<CR>**.

Example

```
AT+CLAC
AT+COPS

AT+CGATT

...

AT+NSOCR

AT+NSOST

…

OK
```

2.10. AT+CSQ Get Signal Strength Indicator

The execution command returns received signal strength indication **<rssi>** and channel bit error rate **<ber>** from the UE.

The test command returns values supported as compound values.

AT+CSQ Get Signal Strength Indicator	
Execution Command **AT+CSQ**	Response **+CSQ:<rssi>,<ber>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CSQ=?**	Response **+CSQ:(**list of supported **<rssi>**s**)**,**(**list of supported **<ber>**s**)**

	OK If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<rssi>	Integer type. Received signal strength. Unit: dBm.	
	0	-113dBm or less
	1	-111dBm
	2...30	-109dBm... -53dBm
	31	-51dBm or greater
	99	Not known or not detectable
<ber>*	Integer type. Channel bit error rate (in percent).	
	0...7	As RxQual values (please refer to 3GPP specifications)
	99	Not known or not detectable

NOTE

"*" means under development. **<ber>** will always be 99 currently.

Example

```
AT+CSQ
+CSQ:31,99

OK
```

2.11. AT+CGPADDR Show PDP Addresses

The command returns the IP address of the device.

The execution command returns a list of PDP addresses for the specified context identifiers. If no **<cid>** is specified, the addresses for all defined contexts are returned.

The test command returns a list of defined **<cid>**s. These are **<cid>**s that have been activated and may or may not have an IP address associated with them.

AT+CGPADDR Show PDP Addresses	
Execution/Write Command **AT+CGPADDR[=<cid>[,<cid>[,...]]]**	Response **+CGPADDR:<cid>[,<PDP_addr_1>[,<PDP_addr_2>]]** **[+CGPADDR:<cid>,[<PDP_addr_1>[,<PDP_addr_2>]]]** **[...]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGPADDR=?**	Response **+CGPADDR:(**list of defined **<cid>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. It specifies a particular PDP context definition (see the **AT+CGDCONT** command). **<cid>** values between 0 and 10 are supported.
<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.

NOTES

1. 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>**.
2. With AUTOCONNECT enabled, **<cid>=0** will not be listed until an IP address is acquired.

Example

```
AT+CGPADDR=0
+CGPADDR:0,101.43.5.1
OK
AT+CGPADDR=?
+CGPADDR:(0)

OK
```

2.12. AT+COPS PLMN Selection

The write command forces an attempt to select and register the EPS network operator using the USIM card installed in the currently selected card slot. **<mode>** is used to select whether the selection is done automatically by the UE or is forced by this command to operator **<oper>** (it shall be given in format **<format>**) to a certain access technology, indicated in **<AcT>**. If the selected operator is not available, no other operator shall be selected. If the selected access technology is not available, then the same operator shall be selected in other access technology. The selected operator name format shall also apply to the read command (**AT+COPS?**). **<mode>=2** forces an attempt to deregister from the network. The selected mode affects all further network registration (e.g. after **<mode>=2**, UE shall be unregistered until **<mode>=0** or 1 is selected). This command can be aborted when registration/deregistration attempt is made.

The 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.

The test command returns a set of five parameters, each representing an operator present in the network. A set consists of an integer indicating the availability of the operator **<stat>**, long and short alphanumeric format of the operator's name, numeric format representation of the operator and access technology. Any of the formats may be unavailable and should then be an empty field. The list of operators shall be in the order of: home network, networks referenced in USIM or active application in the UICC (USIM) in the following order: HPLMN selector, user controlled PLMN selector, operator controlled PLMN selector and PLMN selector (in the USIM), and other networks.

The **<AcT>** access technology selected parameters should only be used in terminals capable to register to more than one access technology. Selection of **<AcT>** does not limit the capability to cell reselections, even though an attempt is made to select an access technology, the UE may still re-select a cell in another access technology.

AT+COPS PLMN Selection	
Write Command **AT+COPS=<mode>[,<format>[,<oper>[,<AcT>]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+COPS?**	Response **+COPS:<mode>[,<format>,<oper>][,<AcT>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+COPS=?**	Response **+COPS:[**list of supported**(<stat>**s,numeric **<oper>[,<AcT>])**s**][,,(**range of supported **<mode>**s**),(**list of supported **<format>**s**)]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	630s

Parameter

<mode>	Integer type. 0 Automatic (**<format>**, **<oper>** and **<AcT>** are omitted) 1 Manual (**<oper>** shall be present, and **<AcT>** is optional) 2 Deregister from network. When **<mode>=1**, the PLMN setting will not be retained after the UE is rebooted. **<mode>=1** is only for development use and **<mode>=0** should be used in production when AUTOCONNECT is enabled.
<format>	Integer type 2 Numeric **<oper>**
<oper>	String type. **<format>** indicates if the format is numeric; numeric format is the NB-IoT network location area identification number which consists of a three BCD digit ITU-T country code coded, plus a two or three BCD digit network code, which is administration specific. **<oper>** field could not be present when **<mode>=0**.
<stat>	Integer type 0 Unknown 1 Available 2 Current 3 Forbidden
<AcT>	Integer type. Access technology selected. No **<Act>** returned for **AT+COPS?**. 7 E-UTRAN 9 E-UTRAN (NB-S1 mode)

NOTE

The test command can only be executed in the RRC-disconnected state (Idle or PSM), otherwise an error will be returned.

Example

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

OK
AT+COPS=?
+COPS:(2,,,"46000"),,(0-2),(2)

OK
```

2.13. AT+CGATT PS Attach or Detach

The write command is used to attach the UE to, or detach the UE from, the packet domain service. After the command has completed, the UE remains in V.250 command state. The command will be ignored and returned with only **OK** if the UE has already been configured with the same **<state>**. If **AT+CGATT** is in progress, further execution of this command before the attach or detach procedure is finished will return an error. If the requested state cannot be achieved, an **ERROR** or **+CME ERROR** response is returned. Active PDP contexts will be automatically deactivated when the attachment state changes to detached.

The read command returns the current packet domain service state. The test command is used for requesting information on the supported packet domain service states.

AT+CGATT PS Attach or Detach	
Write Command **AT+CGATT=<state>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CGATT?**	Response **+CGATT:<state>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGATT=?**	Response **+CGATT:(**list of supported **<state>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	1s, determined by network.

Parameter

<state>	Integer type. Indicates the state of PDP context activation.

	0 Detach 1 Attach When **<state>=1**, **AT+COPS=0** is automatically selected.

NOTE

If the initial PDP context is supported, the context with **<cid>=0** will be automatically defined at startup.

Example

```
AT+CGATT?
+CGATT:0

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

OK
```

2.14. AT+CGACT Activate or Deactivate PDP Context

The write command is used to activate or deactivate the specified PDP context(s). After the command has completed, the UE remains in V.250 command state. If any PDP context is already in the requested state, the state for that context remains unchanged. If the requested state for any specified context cannot be achieved, an **ERROR** or **+CME ERROR** response is returned. Extended error responses are enabled by the **AT+CMEE** command. If the UE is not PS attached when the activation form of the command is executed, the UE first performs a PS attach and then attempts to activate the specified contexts. If the attach fails then the UE responds with error or, if extended error responses are enabled, with the appropriate failure-to-attach error message.

For EPS, if an attempt to disconnect the last PDN connection is made, the UE will respond with **ERROR** or, if extended error responses are enabled, a **+CME ERROR**.

For EPS, the activation request for an EPS bearer resource will be answered by the network by either an EPS dedicated bearer activation or EPS bearer modification request. The request must be accepted by the UE before the PDP context can be set into established state.
The read command returns the current activation states for all the defined PDP contexts.

The test command is used for requesting information on the supported PDP context activation states.

AT+CGACT Activate or Deactivate PDP Context	
Write Command **AT+CGACT=<state>,<cid>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CGACT?**	Response **+CGACT:<cid>,<state>** **[+ CGACT:<cid>,<state>]** **[...]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGACT=?**	Response **+CGACT:(**list of supported **<state>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	1s, determined by network.

Parameter

<state>	Integer type. Indicates the activation state of PDP context. 0 Deactivated 1 Activated
<cid>	Integer type. It specifies a particular PDP context definition (see **AT+CGDCONT**). Only one **<cid>** can be activated or deactivated at a time.

NOTES

1. If the initial PDP context is supported, the context with **<cid>=0** will be automatically defined at startup.
2. The command cannot be executed during PLMN searching, attaching or detaching.
3. The command will disconnect the last PDN connection when UE and core network both support connection without PDN.

Example

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

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

OK
```

2.15. AT+CIMI Request International Mobile Subscriber Identity

The command returns International Mobile Subscriber Identity (string without double quotes).

The execution command causes the UE to return **<IMSI>**, which is intended to permit the TE to identify the individual USIM card or active application in the UICC (USIM) which is attached to UE.

AT+CIMI Request International Mobile Subscriber Identity	
Execution Command **AT+CIMI**	Response **<IMSI>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CIMI=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<IMSI>	International Mobile Subscriber Identity (string without double quotes).

NOTE

IMSI may not be displayed for a few seconds after power-on.

Example

```
AT+CIMI
460001357924680

OK
```

2.16. AT+CGDCONT Define a PDP Context

The write command specifies PDP context parameter values for a PDP context identified by **<cid>**, and the (local) context identification parameter. It also allows the TE to specify whether security protected transmission of ESM information is requested, because the PCO can include information that requires ciphering. There can be other reasons for the UE to use security protected transmission of ESM information, e.g. if the UE needs to transfer an APN. The number of PDP contexts that may be in a defined state at the same time is given by the range returned by the test command.

For EPS, the PDN connection and its associated EPS default bearer are identified herewith.

A special form of the write command, **AT+CGDCONT=<cid>** causes the values for context number **<cid>** to become undefined.

If the initial PDP context is supported, the context with **<cid>=0** is automatically defined at startup, the parameters for **<cid>=0** can be modified with **AT+CGDCONT**. If the initial PDP context is supported, **AT+CGDCONT=0** resets context number 0 to its particular default settings.

The read command returns the current settings for each defined context.

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

AT+CGDCONT Define a PDP Context	
Write Command **AT+CGDCONT=<cid>[,<PDP_type>[,<APN>[,<PDP_addr>[,<d_comp>[,<h_comp>[,<IPv4AddrAlloc>[,<request_type>[,<PCSCF_discovery>[,<IM_CN_Signalling_Flag_Ind>[,<NSLPI>[,<securePCO>]]]]]]]]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CGDCONT?**	Response **+CGDCONT:<cid>[,<PDP_type>[,<APN>[,<PDP_addr>[,<d_comp>[,<h_comp>[,<IPv4AddrAlloc>[,<request_type>[,<PCSCF_discovery>[,<IM_CN_Signalling_Flag_Ind>[,<NSLPI>[,<securePCO>]]]]]]]]]]** **[...]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGDCONT=?**	Response **+CGDCONT:(**range of supported **<cid>**s**),(**list of supported **<PDP_type>**s**),,,(**list of supported **<d_comp>**s**),(**list of supported **<h_comp>**s**),,,,,(**list of supported **<NSLPI>**s**),(**list of supported **<securePCO>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. PDP context ID. It specifies a particular PDP context definition. The parameter is local to the TE-UE interface and is used in other PDP context-related commands. The range of permitted values (the minimum value is 1 or if the initial PDP context is supported, the minimum value is 0) will be returned by the test command.

	The **<cid>**s for network-initiated PDP contexts will have values outside the ranges indicated for the **<cid>** in the test form of the **AT+CGDCONT** command. **<cid>** values of 0-10 are supported. **<cid>=0** is defined when AUTOCONNECT is enabled. **<cid>=7** cannot be set when BIP is enabled.
<PDP_type>	String type. It specifies the type of packet data protocol. IP Internet Protocol (IETF STD 5 [103]) IPV6 Internet Protocol, version 6 IPV4V6 Virtual **<PDP_type>** introduced to handle dual IP stack UE capability NONIP None IP
<APN>	String type. A logical name that is used to select the GGSN or the external packet data network. If the value is null or omitted, then the subscription value will be requested. Maximum number of **<APN>** string is 63 characters.
<d_comp>	Integer type. Controls PDP data compression 0 Off 1 On (manufacturer preferred compression) 2 V.42bis 3 V.44
<h_comp>	Integer type. Controls PDP header compression 0 Off 1 On (manufacturer preferred compression) 2 RFC 1144 [105] (applicable for SNDCP only) 3 RFC 2507 [107] 4 RFC 3095 [108] (applicable for PDCP only)
<NSLPI>	Integer type. Indicates the NAS signalling priority requested for this PDP context. 0 Indicates that this PDP context is to be activated with the value for the low priority indicator configured in the UE. 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 signalling low priority". The UE utilizes the provided NSLPI information as specified in *3GPP TS 24.301 [83]* and *3GPP TS 24.008*.
<securePCO>	integer type. Specifies if security protected transmission of PCO is requested or not (applicable for EPS only). 0 Security protected transmission of PCO is not requested 1 Security protected transmission of PCO is requested

NOTES

1. Only “IP”, “NONIP”, “IPV6” and “IPV4V6” are supported for **<PDP_type>**.
2. This command only supports **<cid>**, **<PDP_type>**, **<APN>**, **<d_comp>**, **<h_comp>**, **<NSLPI>** and **<securePCO>** parameters.
3. Only 0 is supported for the value of **<d_comp>**. Only 0 and 1 are supported for the value of **<h_comp>**.

Example

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

OK
AT+CGDCONT=1,"IP","HUAWEI.COM"
OK
AT+CGDCONT?
+CGDCONT:0,"IPV4V6",,,0,0,,,,,0,0
+CGDCONT:1,"IP","HUAWEI.COM",,0,0,,,,,0,0

OK
```

2.17. AT+CFUN Set UE Functionality

The write command selects the level of functionality in the UE. Level "full functionality" is where the highest level of power is drawn. "Minimum functionality" is where minimum power is drawn.

The read command returns the current setting of **<fun>**.

The test command returns values supported by the UE as compound values.

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

	OK If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	85s

Parameter

<fun>	Integer type. UE functionality level	
	0	Minimum functionality
	1	Full functionality. Enable UE to transmit and receive RF circuits for all supported radio access technologies. For UE supporting **AT+CSRA**, this equals the RATs indicated by the response of **AT+CSRA=?**. Currently **AT+CSRA** setting is ignored. It is not required that the transmitting and receiving RF circuits are in a disabled state when this setting takes effect. After **<fun>** is successfully configured to 1, the UE can be shut down with **AT+CFUN=0** beside other methods.
<rst>	Integer type. UE resetting	
	0	Do not reset the UE before setting it to **<fun>** power level. This is the default value when **<rst>** is omitted.
	1	Reset the UE before setting it to **<fun>** power level (not supported and will be ignored)

NOTE

The module will enter Deep Sleep when the system is inactive, but only if the PSM has been enabled by the core network.

Example

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

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

OK

2.18. 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. **ERROR** is returned normally when error is related to syntax, invalid parameters or UE functionality.

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

The test command returns values supported as a compound value.

AT+CMEE Report UE Error	
Write Command **AT+CMEE=<n>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CMEE?**	Response **+CMEE:<n>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CMEE=?**	Response **+CMEE:(**list of supported **<n>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. Error mode.	
	0	Disable **+CME ERROR:<err>** result code and use **ERROR** instead
	1	Enable **+CME ERROR:<err>** result code and use numeric **<err>** values

2.19. AT+CCLK Return Current Date and Time

The clock will be set automatically once the UE has connected to the network. Please execute **AT+NITZ=0** before using write command to set the time.

The read command returns the current setting of the clock.

AT+CCLK Return Current Date and Time	
Write Command **AT+CCLK=<time>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CCLK?**	Response **+CCLK:<time>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CCLK=?**	Response **OK**
Maximum Response Time	300ms

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; and range is -96~+96). For instance, 6th of May 1994, 22:10:00 GMT+2 hours equal

	"94/05/06,22:10:00+08".

NOTES

1. If UE does not support time zone information, then the last three characters of **<time>** are not returned by **AT+CCLK?** command.
2. No value will be returned before the core network sends "EMM INFORMATION".

Example

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

AT+CCLK?
+CCLK:18/11/09,05:36:42+32

OK
AT+CCLK=?
OK
```

2.20. AT+CPSMS Power Saving Mode Setting

The write command controls the setting of the UE's power saving mode (PSM) parameters. It can be used to configure whether to apply PSM or not. Please refer to the URCs provided by **AT+CEREG** for the active time value, and the extended periodic TAU value that are allocated to the UE by the network in E-UTRAN.

A special form of the command can be given as **AT+CPSMS=2**. In this form, the use of PSM will be disabled and data for all parameters in **AT+CPSMS** command will be removed or, if available, set to the default values.

The read command returns the current parameter values.

The test command returns the supported **<mode>**s and the value ranges for the requested extended periodic TAU value in E-UTRAN and the requested active time value as compound values.

AT+CPSMS Power Saving Mode Setting	
Write Command **AT+CPSMS=<mode>[,<Requested_Periodic-RAU>[,<Requested_GPRS READYtimer>[,<Requested_Periodi**	Response **OK** If there is any error:

cTAU>[,<Requested_Active-Time>]]]]	**ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CPSMS?**	Response **+CPSMS:<mode>[,,,<Requested_Periodic-TAU>],[<Requested_Active-Time>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CPSMS=?**	Response **+CPSMS:(**list of supported **<mode>**s**),,,(**range of supported **<Requested_Periodic-TAU>**s**),(**range of supported **<Requested_Active-Time>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<mode>	Integer type. Disable or enable the use of PSM in the UE 0 Disable the use of PSM 1 Enable the use of PSM 2 Disable the use of PSM and discard all parameters for PSM or, if available, reset to the default values.
<Requested_Periodic-TAU>	String type. One byte in an 8-bit format. Requested extended periodic TAU value (T3412) to be allocated to the UE in E-UTRAN. The requested extended periodic TAU value is coded as one byte (octet 3) of the GPRS Timer 3 information element coded as bit format (e.g. "01000111" equals 70 hours). Bits 5 to 1 represent the binary coded timer value Bits 6 to 8 define the timer value unit as follows: Bits 8 7 6 0 0 0 value is incremented in multiples of 10 minutes

	0 0 1 value is incremented in multiples of 1 hour 0 1 0 value is incremented in multiples of 10 hours 0 1 1 value is incremented in multiples of 2 seconds 1 0 0 value is incremented in multiples of 30 seconds 1 0 1 value is incremented in multiples of 1 minute 1 1 0 value is incremented in multiples of 320 hours(note) 1 1 1 value indicates that the timer is deactivated The default value is 10 hours.
<Requested_Active-Time>	String type. One byte in an 8-bit format. Requested Active Time value (T3324) to be allocated to the UE. The requested Active Time value is coded as one byte (octet 3) of the GPRS Timer 2 information element coded as bit format (e.g. "00100100" equals 4 minutes). Bits 5 to 1 represent the binary coded timer value. Bits 6 to 8 define the timer value unit for the GPRS timer as follows: Bits 8 7 6 0 0 0 value is incremented in multiples of 2 seconds 0 0 1 value is incremented in multiples of 1 minute 0 1 0 value is incremented in multiples of decihours 1 1 1 value indicates that the timer is deactivated The default value is 10 seconds.

NOTES

1. This timer value unit is only applicable to the T3412 extended value IE. If it is received in an integrity protected message, value shall be interpreted as multiples of 320 hours. Otherwise value shall be interpreted as multiples of 1 hour.
2. The read command **AT+CPSMS?** can only get value 0 and 1 of **<mode>**.
3. The **<Requested_Periodic-RAU>** and **<Requested_GPRSREADYtimer>** parameters are currently not supported. Any value entered will be ignored and no value will be output.

Example

```
AT+CPSMS=1,,,01000011,01000011
OK
AT+CPSMS?
+CPSMS:1,,,01000011,01000011

OK
AT+CPSMS=?
+CPSMS:(0,1,2),,,( 00000000-11111111), (00000000-11111111)
```

OK

2.21. AT+CEDRXS eDRX Setting

The write command controls the setting of the UE's eDRX parameters. It can be used to control whether the UE wants to apply eDRX or not, as well as the requested eDRX value for each specified type of access technology.

The write command also controls the presentation of the URC when **<mode>=2** and there is a change in the eDRX parameters provided by the network:
+CEDRXP:<AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]

A special form of the command can be given as **AT+CEDRXS=3**. In this form, eDRX will be disabled and data for all parameters in **AT+CEDRXS** command will be removed.

The read command returns the current settings for each defined value of **<AcT-type>**.

The test command returns the supported **<mode>**s and the value ranges for the access technology and the requested eDRX value as compound values.

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

	OK If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<mode> Integer type. Disable or enable the use of eDRX in the UE. This parameter is applicable to all specified types of access technology, i.e. the most recent setting of **<mode>** will take 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 the URC: **+CEDRXP:<AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]**

3 Disable the use of eDRX and discard all parameters for eDRX.

<AcT-type> Integer type. Indicates the type of access technology. **AT+CEDRXS?** is used to specify the relationship between the type of access technology and the requested eDRX value.

0 Access technology is not using eDRX. This parameter value is only used in the URC.

5 E-UTRAN (NB-S1 mode)

<Requested_eDRX_value> String type. 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. 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. 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
	1	1	1	1	40.96 seconds

NOTE

The use of eDRX in the UE is enabled by default in R01(B300SP5) and later versions.

Example

```
AT+CEDRXS=1,5,"0101"
OK
AT+CEDRXS?
+CEDRXS:5,"0101"

OK
AT+CEDRXS=?
+CEDRXS:(0,1,2,3),(5),("0000"-"1111")
```

```
OK
AT+CEDRXS=0,5
OK
```

2.22. AT+CEER Extended Error Report

The execution command causes the UE to return one or more lines of information text **<report>**, determined by the UE manufacturer, which should offer the user of the UE an extended report of the reason for the following errors:

- The failure in the last call release;
- The failure in the last unsuccessful PDP context activation;
- The failure in the PDP context deactivation.

Typically, the text will consist of a single line containing the cause information given by network in textual format.

AT+CEER Extended Error Report	
Execution Command **AT+CEER**	Response **+CEER:<report>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CEER=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<report>	Extended error report. The message text (including line terminators) shall not exceed 2041 characters. The text shall not contain the sequence **0<CR>** or **OK<CR>**.

Example

```
AT+CEER
+CEER:EMM_CAUSE_EPS_AND_NON_EPS_SERVICES_NOT_ALLOWED
```

```
OK
AT+CEER=?
OK
```

2.23. AT+CEDRXRDP eDRX Read Dynamic 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 eDRX Read Dynamic Parameters	
Execution Command **AT+CEDRXRDP**	Response **+CEDRXRDP:<AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CEDRXRDP=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<AcT-type> Integer type. The type of access technology. **AT+CEDRXS?** is used to specify the relationship between the type of access technology and the requested eDRX value.

0 Access technology is not using eDRX. This parameter value is only used in the URC

5 E-UTRAN (NB-S1 mode)

<Requested_eDRX_value> String type. 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. 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. 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

Example

```
AT+CEDRXRDP
```

```
+CEDRXRDP:5,"0010","1110","0101"

OK
AT+CEDRXRDP=?
OK
```

2.24. AT+CTZR Time Zone Reporting

The write command controls the reporting of time zone change event. If reporting is enabled, the UE returns the URC **+CTZV:<tz>, +CTZE:<tz>,<dst>,[<time>]**, or **+CTZEU:<tz>,<dst>,[<utime>]** whenever the time zone is changed. The UE also provides the time zone upon network registration if provided by the network. If setting fails in an UE error, **+CME ERROR:<err>** is returned.

The read command returns the current reporting settings in the UE.

The test command returns supported **<reporting>** values as a compound value.

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

	ERROR Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<reporting>	Integer type. Reporting status 0 Disable time zone change event reporting 1 Enable time zone change event reporting by URC **+CTZV:<tz>** 2 Enable extended time zone and local time reporting by URC **+CTZE:<tz>,<dst>,[<time>]** 3 Enable extended time zone and universal time reporting by URC **+CTZEU:<tz>,<dst>,[<utime>]**
<tz>	String type. Represents the 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-digit integer with the range -48 ~ +56. To maintain a fixed width, numbers in the range -9 ~ +9 are expressed with a leading zero, e.g. "-09", "+00" and "+09".
<dst>	Integer type. Indicates whether **<tz>** includes daylight savings adjustment 0 **<tz>** includes no adjustment for daylight saving time 1 **<tz>** includes +1 hour (equals 4 quarters in **<tz>**) adjustment for daylight saving time 2 **<tz>** includes +2 hours (equals 8 quarters in **<tz>**) adjustment for daylight saving time
<time>	String type. Represents the 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 will be present in the URC for extended time zone and local time reporting if the universal time is provided by the network.
<utime>	String type. Represents the 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 will be present in the URC for extended time zone and universal time reporting if provided by the network.

Example

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

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

OK
```

2.25. AT+CIPCA Initial PDP Context Activation

The write command controls whether the UE is attached to E-UTRAN with or without a PDN connection. The value of **<n>=3** applies to E-UTRAN RATs. Changing **<n>** will never cause a PDP context deactivation.

For **<AttachWithoutPDN>=1**, the EPS attach is performed without a PDN connection.

The read command returns current settings of the command.

The test command returns values supported as a compound value.

AT+CIPCA Initial PDP Context Activation	
Write Command **AT+CIPCA=<n>[,<AttachWithoutPDN>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CIPCA?**	Response **+CIPCA:<n>[,<AttachWithoutPDN>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CIPCA=?**	Response **+CIPCA:(**list of supported **<n>**s**),(**list of supported **<AttachWithoutPDN>**s**)** **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. Activation of PDP context upon attachment. 3 No change in current setting
<AttachWithoutPDN>	Integer type. EPS attach with or without PDN connection. If **<AttachWithoutPDN>** is omitted, default value 0 will be used. 0 EPS attach with PDN connection 1 EPS attach without PDN connection

NOTE

For this command, the term roaming corresponds to being registered to a VPLMN which is not equivalent to HPLMN or EHPLMN.

Example

```
AT+CIPCA=3
OK

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

OK
```

2.26. AT+CGAPNRC APN Rate Control

This execution command returns APN rate control parameters (see *3GPP TS 24.008 [8]*) 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.

The test command returns a list of **<cid>**s associated with secondary and non-secondary active PDP contexts.

AT+CGAPNRC APN Rate Control	
Execution/Write Command **AT+CGAPNRC[=<cid>]**	Response **[+CGAPNRC:<cid>[,<Additional_exception_reports>[,<Uplink_time_unit>[,<Maximum_uplink_rate>]]]]** **[+CGAPNRC:<cid>[,<Additional_exception_reports>[,<Uplink_time_unit>[,<Maximum_uplink_rate>]]]]** **[...]]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGAPNRC=?**	Response **+CGAPNRC:(**list of **<cid>**s associated with active contexts**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. It specifies a particular PDP context definition (refer to **AT+CGDCONT** command).
<Additional_exception_reports>	Integer type. Indicates whether to send the additional exception reports when the maximum uplink rate is reached. 0 **<Additional_exception_reports>** at maximum rate reached are not allowed to be sent. 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. 0 Unrestricted 1 Minute 2 Hour

	3 Day 4 Week
<Maximum_uplink_rate>	Integer type. It specifies the maximum number of messages the UE is restricted to send per uplink time unit. The time unit is indicated in the uplink time unit. If the uplink time unit is set to "unrestricted", the maximum uplink data volume the UE can send is not restricted.

Example

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

OK
AT+CGAPNRC=?
+CGAPNRC:(0)

OK
```

2.27. AT+CSODCP Send Originating Data via the Control Plane

The write command is used by the TE to transmit data (non-IP message) over control plane to network via UE. Context identifier **<cid>** is used to link the data to a particular context.

This command optionally indicates that the application on the UE expects that the exchange of data will be completed with this uplink data transfer; or will be completed with the next received downlink data.

This command also optionally indicates whether the data to be transmitted is an exception data or not. It causes transmission of an "ESM DATA TRANSPORT" message, as defined in *3GPP TS 24.301 [83]*.

The test command returns the maximum number of bytes of the user data container supported by the UE, supported **<RAI>**s and supported **<type_of_user_data>**s as a compound value.

AT+CSODCP Send Originating Data via the Control Plane	
Write Command **AT+CSODCP=<cid>,<cpdata_length>,<cpdata>[,<RAI>[,<type_of_user_data>[,<sequence>]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Test Command **AT+CSODCP=?**	Response **+CSODCP:(**range of supported **<cid>**s**),(**maximum number of bytes of the **<cpdata_length>),(**list of supported **<RAI>**s**),(**list of supported **<type_of_user_data>**s**),(**range of supported **<sequence>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. A numeric parameter which specifies a particular PDP context or EPS bearer context definition. The **<cid>** parameter is local to the TE-UE interface and identifies the PDP or EPS bearer contexts which have been setup via AT commands (see the **AT+CGDCONT** command).
<cpdata_length>	Integer type. Indicates the number of bytes of the **<cpdata>** information element. When there is no data to transmit, the value shall be set to 0.
<cpdata>	String of octets. Contains the user data container contents. When there is no data to transmit, the **<cpdata>** shall be an empty string (**""**). This parameter shall not be subject to conventional character conversion as per **+CSCS** (refer to *3GPP TS 27.007 [9]*). The coding format of the user data container and the maximum length of **<cpdata>** are implementation specific. Maximum data length will be 1358 bytes when non-IP is used, otherwise will be 0.
<RAI>	Integer type. Indicates the value of the release assistance indication. 0 No information available 1 The UE expects that exchange of data will be completed with the transmission of the “ESM DATA TRANSPORT” message. 2 The UE expects that exchange of data will be completed with the receipt of an “ESM DATA TRANSPORT” message.
<type_of_user_data>	Integer type. Indicates whether the user data that is transmitted is regular or exceptional. 0 Regular data 1 Exception data
<sequence>	Sequence of data. The range is 1-255. If it is omitted, data sent status will not be reported. If it is not omitted, when datagram is sent by RF or discard, the result below will be reported: **+CSODCPR:<cid>,<sequence>,<status>**. **<sequence>** is not defined by 3GPP.

<status>	The status of datagram. 0 Error 1 Sent

NOTE

Only one message will be buffered every time.

Example

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

OK
```

2.28. AT+CRTDCP Report Terminating Data via the Control Plane

When receiving a message from the CDP server, the write command is used to enable and disable reporting of data that is transmitted via the control plane in downlink direction from the network to the UE. If reporting is enabled, the UE returns the URC **+CRTDCP:<cid>,<cpdata_length>,<cpdata>** when data is received from the network.

The read command returns the current settings.

The test command returns supported values as compound values.

AT+CRTDCP Report Terminating Data via the Control Plane	
Write Command **AT+CRTDCP=<reporting>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CRTDCP?**	Response **+CRTDCP:<reporting>** **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Test Command **AT+CRTDCP=?**	Response **+CRTDCP:(**list of supported **<reporting>**s**),(**range of supported **<cid>**s**),(**maximum number of octets of user data indicated by **<cpdata_length>)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<reporting>	Integer type. Controls reporting of mobile terminated control plane data events 0 Disable reporting of UE control plane data 1 Enable reporting of UE control plane data by the URC: **+CRTDCP:<cid>,<cpdata_length>,<cpdata>**
<cid>	Integer type. A numeric parameter which specifies a particular PDP context or EPS bearer context definition. The **<cid>** parameter is local to the TE-UE interface and identifies the PDP or EPS bearer contexts which have been setup via AT command (see **AT+CGDCONT** command).
<cpdata_length>	Integer type. Indicates the number of bytes of the **<cpdata>** information element. When there is no data to transmit, the value shall be set to zero.
<cpdata>	String of octets. Contains the user data container contents. When there is no data to transmit, the **<cpdata>** shall be an empty string (**""**). This parameter shall not be subject to conventional character conversion as per **+CSCS** (refer to *3GPP TS 27.007 [9]*). The coding format of the user data container and the maximum length of **<cpdata>** are implementation specific. Maximum received data length will be 1358 bytes when non-IP is set by **AT+CGDCONT**, otherwise will be 0.

Example

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

OK
```

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

OK

+CRTDCP:0,2,"ab"          //URC
```

2.29. AT+CGCONTRDP Read PDP Context Dynamic Parameters

The execution command returns the relevant information **<DNS_prim_addr>** and **<DNS_sec_addr>** for an active non-secondary PDP context with the context identifier **<cid>**.

- If the UE indicates more than two IP addresses of P-CSCF servers or more than two IP addresses of DNS servers, multiple lines of information per **<cid>** will be returned.
- 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. If this UE with dual stack capabilities indicates more than two IP addresses of P-CSCF servers or more than two IP addresses of DNS servers, multiple of such pairs of lines will be returned.
- If the parameter **<cid>** is omitted, the relevant information for all active non-secondary PDP contexts is returned.

The test command returns a list of **<cid>**s associated with active non-secondary contexts.

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

	ERROR Or **+CME ERROR: <err>**
Maximum Response Time	300ms

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 (see the **AT+CGDCONT** command).
<bearer_id>	Integer type. Identifies the bearer, i.e. the EPS bearer in EPS.
<apn>	String type. A logical 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 (0-255) parameters in the form: **"a1.a2.a3.a4.m1.m2.m3.m4"** for IPv4 or **"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.
<DNS_prim_addr>	String type. Shows the IP address of the primary DNS server.
<DNS_sec_addr>	String type. Shows the IP address of the secondary DNS server.

NOTES

1. **<DNS_prim_addr>** and **<DNS_sec_addr>** could be displayed only when **<PDP_type>** (refer to **AT+CGDCONT** command) is “IP” or “IPv6”.
2. PDP context dynamic parameters to be read can be configured by **AT+NCPCDPR**.

Example

```
AT+CGCONTRDP
+CGCONTRDP:0,,"cmcc.MNC004.MCC460.GPRS",,,211.136.20.203,211.136.17.107

OK
AT+CGCONTRDP=?
+CGCONTRDP:(0)

OK
```

2.30. AT+CGAUTH Define PDP Context Authentication Parameters

The write command allows the TE to specify authentication parameters for a PDP context identified by the (local) context identification parameter **<cid>** used during the PDP context activation and the PDP context modification procedures. Since the **<cid>** is the same parameter that is used in the **AT+CGDCONT** command, **AT+CGAUTH** is effectively as an extension to the command.

The read command returns the current settings for each defined context.

The test command returns values supported as compound values.

AT+CGAUTH Define PDP Context Authentication Parameters	
Write Command **AT+CGAUTH=<cid>[,<auth_prot>[,<userid>[,<password>]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CGAUTH?**	Response **[+CGAUTH:<cid>,<auth_prot>,<userid>,<password>]** **[+CGAUTH:<cid>,<auth_prot>,<userid>,<password>]** **[...]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGAUTH=?**	Response **+CGAUTH:(**range of supported **<cid>**s**),(**list of supported **<auth_prot>**s**),(**range of supported **<userid>**s**),(**range of supported **<password>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Maximum Response Time	300ms

Parameter

<cid>	Integer type. It specifies a particular PDP context definition (see the **AT+CGDCONT** command).
<auth_prot>	Integer type. Authentication protocol used for this PDP context. 0 None. Used to indicate that no authentication protocol is used for this PDP context. Username and password are removed if previously specified. 1 PAP 2 CHAP
<userid>	String type. User name for access to the IP network. **<userid>** is needed when **<auth_prot>** is 1 or 2. The maximum **<userid>** string length is 60 bytes.
<password>	String type. Password for access to the IP network. The string length is 60 bytes.

Example

```
AT+CGAUTH=1,2,"1234","1234"
OK
AT+CGAUTH?
+CGAUTH:1,2,"1234","1234"

OK
AT+CGAUTH=?
+CGAUTH:(0-10),(0,1,2),(0-60),(0-60)

OK
```

2.31. AT+CNMPSD No More PS Data

The execution command indicates that no application on the UE is expected to exchange data. This command may be used in both normal and modem compatibility modes.

AT+CNMPSD No More PS Data	
Execution Command **AT+CNMPSD**	Response **OK** If there is any error: **ERROR**

	Or **+CME ERROR: <err>**
Test Command **AT+CNMPSD=?**	Response **OK**
Maximum Response Time	300ms

Example

```
AT+CNMPSD
OK
AT+CNMPSD=?
OK
```

2.32. AT+CPIN Enter PIN

The write command sends a necessary password to the UE before it can be operated (USIM PIN, USIM PUK, etc.). If the PIN needs to be entered twice, the TA shall automatically repeat the PIN. If no PIN request is pending, no action will be taken towards UE and an error message, **+CME ERROR: <err>**, will be returned to the TE.

If the PIN required is USIM PUK, the second pin, **<newpin>**, is required. This second pin is used to replace the old pin in the USIM.

The read command returns an alphanumeric string indicating whether some password is required or not.

AT+CPIN Enter PIN	
Write Command **AT+CPIN=<pin>[,<newpin>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CPIN?**	Response **+CPIN: <code>** **OK** If there is any error: **ERROR** Or

	+CME ERROR: <err>
Test Command **AT+CPIN=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<pin>	String type. Password.	
<newpin>	String type. New password required if the requested code was a PUK.	
<code>	READY	UE is not pending for any password
	SIM PIN	UE is waiting USIM PIN to be given
	SIM PUK	UE is waiting USIM PUK to be given
	SIM PUK BLOCKED	Unsolicited code when powering on if the SIM card is locked after entering wrong PUK for 10 times.

NOTES

1. Both write and read commands of **AT+CPIN** can only be executed when **AT+CFUN=1**.
2. USIM PIN and USIM PUK refer to the PIN of selected application on the UICC. For example, in an UTRAN context, the selected application on the currently selected UICC should be a USIM and the USIM PIN then represents the PIN of the selected USIM. Please refer to *3GPP TS 31.101 [65]* for further details on application selection on the UICC.

Example

```
AT+CPIN=1234
OK
AT+CPIN?
+CPIN: READY

OK
AT+CPIN=?
OK
```

2.33. AT+CPINR Query Remaining PIN Retries

The write command and execution command cause the UE to return the number of remaining PIN retries for the UE passwords with intermediate result code **+CPINR: <code>,<retries>[,<default_retries>]** for standard PINs.

AT+CPINR Query Remaining PIN Retries	
Write Command **AT+CPINR=<code>**	Response **[+CPINR: <code>,<retries>[,<default_retries>]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Execution Command **AT+CPINR**	Response **[+CPINR: <code>,<retries>[,<default_retries>]]** **[+CPINR: <code>,<retries>[,<default_retries>]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CPINR=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<retries>	Integer type. Number of remaining retries per PIN.
<default_retries>	Integer Type. Number of default/initial retries per PIN.
<code>	Type of PIN. All values listed under the description of the **AT+CPIN** command, **<code>** parameter, except 'READY'.

NOTE

The write and execution commands of **AT+CPINR** can only be executed when **AT+CFUN=1**.

Example

```
AT+CPINR="SIM PUK"
+CPINR: SIM PUK,10,10
```

```
OK
AT+CPINR
+CPINR: SIM PIN,3,3
+CPINR: SIM PUK,10,10

OK
```

2.34. AT+CCHO Open Logical Channel

The write command causes the UE to return **<sessionid>** to allow the TE to identify a channel that is being allocated by the currently selected UICC, which is attached to ME. The currently selected UICC will open a new logical channel; select the application identified by the **<dfname>** received with this command and return a session ID as the response. The ME shall restrict the communication between the TE and the UICC to this logical channel.

This **<sessionid>** is to be used when sending commands with Restricted UICC Logical Channel access **+CRLA** (refer to *3GPP TS 27.007*) or Generic UICC Logical Channel access **AT+CGLA** commands.

AT+CCHO Open Logical Channel	
Write Command **AT+CCHO=<dfname>**	Response **<sessionid>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CCHO=?**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<dfname>	All selectable applications in the UICC are referenced by a DF name coded on 1 to 16 bytes.

<sessionid>	Integer type. A session ID to be used in order to target a specific application on the smart card (e.g. (U)SIM, WIM, ISIM) using logical channels mechanism.

NOTE

The logical channel number is contained in the CLASS byte of an APDU command, thus implicitly contained in all APDU commands sent to a UICC. In this case it will be up to the MT to manage the logical channel part of the APDU CLASS byte and to ensure that the chosen logical channel is relevant to the **<sessionid>** indicated in the AT command.

Example

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

2.35. AT+CCHC Close Logical Channel

The write command asks the ME to close a communication session with the active UICC. The ME shall close the previously opened logical channel. The TE will no longer be able to send commands on this logical channel.

The UICC will close the logical channel when receiving this command.

AT+CCHC Close Logical Channel	
Write Command **AT+CCHC=<sessionid>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CCHC=?**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<sessionid>	Integer type. A session Id to be used in order to target a specific application on the smart card (e.g. (U)SIM, WIM, ISIM) using logical channels mechanism.

Example

```
AT+CCHC=1
OK
AT+CCHC=?
OK
```

2.36. AT+CGLA Generic UICC Logical Channel Access

The write command transmits the **<command>** to the MT, it then shall send as it is to the selected UICC. In the same manner the UICC **<response>** shall be sent back by the MT to the TA as it is.

This command allows a direct control of the currently selected UICC by a distant application on the TE. The TE shall then take care of processing UICC information within the frame specified by GSM/UMTS.

Although Generic UICC Logical Channel Access command **AT+CGLA** allows TE to take control over the UICC-MT interface, there are some functions of the UICC-MT interface that logically do not need to be accessed from outside the TA/MT. Moreover, for security reason the GSM network authentication should not be handled outside the TA/MT. Therefore it shall not be allowed to execute a Run GSM Algorithm command or an Authenticate command in GSM context from the TE using **AT+CGLA** at all time whether the **AT+CGLA** is locked or unlocked. This shall not forbid the TE to send Authenticate commands in other security contexts (e.g. EAP security context). For example, the TA/MT shall forbid the transfer of the Authenticate command to a USIM application when parameters P2=0 (GSM security context). See *3GPP TS 31.102 [59]* for USIM authentication command definition.

AT+CGLA Generic UICC Logical Channel Access	
Write Command **AT+CGLA=<sessionid>,<length>,<command>**	Response **+CGLA: <length>,<response>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command	Response

AT+CGLA=?	**OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<sessionid>	Integer type. This is the identifier of the session to be used in order to send the APDU commands to the UICC. It is mandatory in order to send commands to the UICC when targeting applications on the smart card using a logical channel other than the default channel (channel "0").
<length>	Integer type. Length of the characters that are sent to TE in **<command>** or **<response>** (two times the actual length of the command or response).
<command>	Command passed on by the MT to the UICC in the format as described in *3GPP TS 31.101 [65]* (hexadecimal character format; refer to **+CSCS**).
<response>	Response to the command passed on by the UICC to the MT in the format as described in *3GPP TS 31.101 [65]* (hexadecimal character format; refer **+CSCS**).

NOTE

Compared to Restricted UICC Access command **+CRLA** (refer to *3GPP TS 27.007*), the definition of **AT+CGLA** allows TE to take more control over the UICC-MT interface. The locking and unlocking of the interface may be done by a special **<command>** value or automatically by TA/MT (by interpreting **<command>** parameter). In case that TE application does not use the unlock command (or does not send a **<command>** causing automatic unlock) in a certain timeout value, MT may release the locking.

Example

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

2.37. AT+CRSM Restricted SIM Access

TE application has easier but more limited access to the SIM database with this command instead of Generic SIM Access **AT+CSIM**.

The write command transmits to the MT the SIM **<command>** and its required parameters. If a SIM installed in the currently selected card slot, the MT handles internally all SIM-MT interface locking and file

selection routines. As response to the command, MT sends the actual SIM information parameters and response data. MT error result code **+CME ERROR** may be returned when the command cannot be passed to the SIM, but failure in the execution of the command in the SIM is reported in **<sw1>** and **<sw2>** parameters.

Coordination of command requests to SIM and the ones issued by GSM/UMTS application inside the MT is implementation dependent. However the TE should be aware of the precedence of the GSM/UMTS application commands to the TE commands.

AT+CRSM Restricted SIM Access	
Write Command **AT+CRSM=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>[,<pathid>]]]]**	Response **+CRSM: <sw1>,<sw2>[,<response>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CRSM=?**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<command>	Command passed on by the MT to the 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 All other values are reserved, command 203 and 219 are not supported currently.
<fileid>	Integer type. This is the identifier of an elementary data file on SIM. Mandatory for every command except STATUS.

<P1>, <P2>, <P3>	Integer type. Parameters passed on by the MT to the SIM. These parameters are mandatory for every command, except GET RESPONSE and STATUS. The values are described in *3GPP TS 51.011 [28].*
<data>	Information which shall be written to the SIM (hexadecimal character format; please refer to **+CSCS**).
<pathid>	String type. Contain the path of an elementary file on the SIM/UICC in hexadecimal format as defined in *ETSI TS 102 221 [60]* (e.g. "7F205F70" in SIM and UICC case).The **<pathid>** shall only be used in the mode "select by path from MF" as defined in *ETSI TS 102 221 [60]*.
<sw1>, <sw2>	Integer type. Information from the SIM about the execution of the actual command. These parameters are delivered to the TE in both cases, on successful or failed execution of the command.
<response>	Response of a successful completion of the command previously issued (hexadecimal character format; please refer to **+CSCS**). 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 (please refer to *3GPP TS 51.011 [28]*). After READ BINARY, READ RECORD or RETRIEVE DATA command the requested data will be returned. **<response>** is not returned after a successful UPDATE BINARY,UPDATE RECORD or SET DATA command.

NOTES

1. The MT internally executes all commands necessary for selecting the desired file, before performing the actual command.
2. The range of valid file identifier depends on the actual SIM and is defined in *3GPP TS 51.011 [28]*. Optional files may not be present at all.
3. Since valid elementary file identifier may not be unique over all valid dedicated file identifier the **<pathid>** indicates the targeted UICC/SIM directory path in case of ambiguous file identifier. For earlier versions of this specification or if **<pathid>** is omitted, it could be implementation specific which one will be selected.
4. 203 and 219 among the value of **<command>** are not supported currently.

Example

```
AT+CRSM=242
+CRSM: 144,0

OK
AT+CRSM=?
OK
```

2.38. AT+CSIM Generic SIM Access

The write command transmits the **<command>** to the MT, it then shall send as it is to the SIM. In the same manner, the SIM **<response>** shall be sent back by the MT to the TA as it is.

This command allows a direct control of the SIM that is installed in the currently selected card slot, by an distant application on the TE. The TE shall then take care of processing SIM information within the frame specified by GSM/UMTS.

AT+CSIM Generic SIM Access	
Write Command **AT+CSIM=<length>,<command>**	Response **+CSIM: <length>,<response>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CSIM=?**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<length>	Integer type. Length of the characters that are sent to TE in **<command>** or **<response>** (two times the actual length of the command or response).
<command>	Command passed on by the MT to the SIM in the format as described in *3GPP TS 51.011 [28]* (hexadecimal character format; please refer to **+CSCS**).
<response>	Response to the command passed on by the SIM to the MT in the format as described in *3GPP TS 51.011 [28]* (hexadecimal character format; please refer to **+CSCS**).

NOTE

Compared to Restricted SIM Access command **AT+CRSM**, the definition of **AT+CSIM** allows TE to take more control over the SIM-MT interface. The locking and unlocking of the interface may be done by a

special **<command>** value or automatically by TA/MT (by interpreting **<command>** parameter). In case that TE application does not use the unlock command (or does not send a **<command>** causing automatic unlock) in a certain timeout value, MT may release the locking.

Example

```
AT+CSIM=10,"00B2010426"
+CSIM: 4,6981

OK
AT+CSIM=?
OK
```

2.39. AT+CGDATA Enter Data State

The writer command causes the MT to perform whatever actions are necessary to establish communication between the TE and the network using one or more Packet Domain PDP types. This may include performing a PS attaching and one or more PDP context activations. If the **<L2P>** parameter value is unacceptable to the MT, the MT shall return an **ERROR** or **+CME ERROR** response. Otherwise, the MT issues the intermediate result code **CONNECT** and enters V.250 online data state.

Commands following **AT+CGDATA** command in the AT command line shall not be processed by the MT. The detailed behavior after the online data state has been entered is dependent on the PDP type. It is described briefly in *3GPP TS 27.060 [34]* and in more detail in *3GPP TS 29.061 [39]* and the specifications for the relevant PDPs. PS attachment and PDP context activation procedures may take place prior to or during the PDP startup if they have not already been performed using the **AT+CGATT** and **AT+CGACT** commands. If context activation takes place during the PDP startup, one or more **<cid>**s may be specified in order to provide the information needed for the context activation request(s).

During each PDP startup procedure, the MT may have access to some or all of the following information:

- The MT may have a priori knowledge, for example, it may implement only one PDP type.
- The command may have provided an **<L2P>** parameter value.
- The TE may provide a PDP type and/or PDP address to the MT during in the PDP startup procedure.

If any of this information is in conflict, the command will fail. Any PDP type and/or PDP address present in the above information shall be compared with the PDP type and/or PDP address in any context definitions specified in the command in the order in which their **<cid>**s appear. For a context definition to match:

- The PDP type must match exactly.
- The PDP addresses are considered to match if they are identical or if either or both addresses are unspecified. For example, a PPP NCP request specifying PDP type = IP and no PDP address would

cause the MT to search through the specified context definitions for one with PDP type = IP and any PDP address.

The context shall be activated using the matched value for PDP type and a static PDP address if available, together with the other information found in the PDP context definition. If a static PDP address is not available then a dynamic address is requested.

If no **<cid>** is given or if there is no matching context definition, the MT shall attempt to activate the context with whatever information is available to the MT. The other context parameters shall be set to their default values.

If the activation is successful, data transfer may proceed. After data transfer is complete, and the layer 2 protocol termination procedure has completed successfully, the V.250 command state is re-entered and the MT returns the final result code OK.

In the event of an erroneous termination or a failure to start up, the V.250 command state is re-entered and the MT returns the final result code "NO CARRIER" or, if enabled, **+CME ERROR**. Attachment, activation and other errors may be reported.

The test command is used for requesting information on the supported layer 2 protocols. This command may be used in both normal and modem compatibility modes.

AT+CGDATA Enter Data State	
Write Command **AT+CGDATA[=<L2P>[,<cid>[,<cid>[,...]]]]**	Response **CONNECT** **ERROR** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGDATA=?**	Response **+CGDATA= (**list of supported **<L2P>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<L2P>	String type. Indicates the layer 2 protocol to be used between the TE and MT NULL None, for PDP type OSP:IHOSS (Obsolete) PPP Point-to-point protocol for a PDP such as IP PAD Character stream for X.25 character (triple X PAD) mode (Obsolete) X25 X.25 L2 (LAPB) for X.25 packet mode (Obsolete) M-xxxx Manufacturer-specific protocol (xxxx is an alphanumeric string) If the value is omitted, the layer 2 protocol is unspecified. Other values are reserved and will result in an ERROR response.
<cid>	Integer type. Specifies a particular PDP context definition (see the **AT+CGDCONT** command).

NOTES

1. If the initial PDP context is supported, the context with **<cid>=0** will be automatically defined at startup.
2. Only **<L2P>="PPP"** is supported.
3. The baud rate needs to be greater than or equal to 57600.
4. Only one **<cid>** can be specified each time.

Example

```
AT+CGDATA=?
+CGDATA=("PPP")

OK
```

2.40. AT+CCIOTOPT CIoT Optimization Configuration

The write command controls which CIoT EPS optimizations the UE indicates as supported and preferred in the ATTACH REQUEST and TRACKING AREA UPDATE REQUEST messages. The command also allows reporting of the CIoT EPS optimizations that are supported by the network. A UE supporting CIoT functionality may support control plane CIoT EPS optimization or user plane CIoT EPS optimization or both (see *3GPP TS 24.301 [83]*, *subclause 9.9.3.34*). Based on the application characteristics the UE may prefer to be registered for control plane CIoT EPS optimization or for user plane CIoT EPS optimization (see *3GPP TS 24.301 [83]*, *subclause 9.9.3.0B*).

Further, the network may support control plane CIoT EPS optimization or user plane CIoT EPS optimization or both (see *3GPP TS 24.301 [83]*, *subclause 9.9.3.12A*).

The write command is used also to control the URC **+CCIOTOPTI**. An URC **+CCIOTOPTI: <supported_Network_opt>** is used to indicate the supported CIoT EPS optimizations by the network.

AT+CCIOTOPT CIoT Optimization Configuration	
Write Command **AT+CCIOTOPT=[<n>,[<supported_UE_opt>[,<preferred_UE_opt>]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CCIOTOPT?**	Response **+CCIOTOPT:<n>,<supported_UE_opt>,<preferred_UE_opt>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CCIOTOPT=?**	Response **+CCIOTOPT:(**list of supported**<n>**s**),(**list of supported**<supported_UE_opt>**s**),(**list of supported**<preferred_UE_opt>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. Enables or disables reporting of URC **+CCIOTOPTI**. 0 Disable reporting. 1 Enable reporting. 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. 0 No support. 1 Support for control plane CIoT EPS optimization. 2 Support for user plane CIoT EPS optimization.

	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.	
	0	No preference.
	1	Preference for control plane CIoT EPS optimization.
	2	Preference for user plane CIoT EPS optimization.
<supported_Network_opt>	Integer type. Indicates the Network support for CIoT EPS optimizations.	
	0	No support.
	1	Support for control plane CIoT EPS optimization.
	2	Support for user plane CIoT EPS optimization.
	3	Support for both control plane CIoT EPS optimization and user plane CIoT EPS optimization.

NOTES

1. **<supported_UE_opt>** only supports 1 and 3.
2. **<preferred_UE_opt>** only supports 1 and 2.
3. **<preferred_UE_opt>** can only be set to 2 only when **<supported_UE_opt>=3**.

Example

```
AT+CCIOTOPT=1,3,2
OK
AT+CCIOTOPT?
+CCIOTOPT:1,3,2

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

OK
```

2.41. AT+CGCMOD Modify PDP Context

The execution command is used to modify the specified PDP context(s) with respect to QoS profiles and TFTs. After the command has completed, the MT returns to V.250 online data state. If the requested modification for any specified context cannot be achieved, an **ERROR** or **+CME ERROR** response is returned. Extended error responses are enabled by the **AT+CMEE** command.

For EPS, the modification request for an EPS bearer resource will be answered by the network by an EPS bearer modification request. The request must be accepted by the MT before the PDP context is effectively

changed. If no **<cid>** is specified, the activation form of the command modifies all active contexts. The test command returns a list of **<cid>**s associated with active contexts.

AT+CGCMOD Modify PDP Context	
Execution Command **AT+CGCMOD[=<cid>[,<cid>[,...]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGCMOD=?**	Response **+CGCMOD:(**list of **<cid>**s associated with active contexts**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. Specifies a particular PDP context definition(see the **AT+CGDCONT** command).

NOTES

1. Only one **<cid>** could be modified each time.
2. Other commands will be blocked when processing.

Example

```
AT+CGCMOD=?
+CGCOMD:(0)

OK
```

2.42. AT+CGEQOS Define EPS Quality of Service

The write command allows the TE to specify the EPS Quality of Service parameters **<cid>**, **<QCI>**, **[<DL_GBR>,<UL_GBR>]** and **[<DL_MBR>,<UL_MBR>]** for a PDP context or Traffic Flows (see *3GPP TS 24.301 [83]* and *3GPP TS 23.203 [85]*). When in UMTS/GPRS, the MT applies a mapping function to UTMS/GPRS Quality of Service.

A special form of the set command, **AT+CGEQOS= <cid>** causes the values for context number **<cid>** to become undefined.

The read command returns the current settings for each defined QoS.

AT+CGEQOS Define EPS Quality of Service	
Write Command **AT+CGEQOS=[<cid>[,<QCI>[,<DL_GBR>,<UL_GBR>[,<DL_MBR>,<UL_MBR]]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CGEQOS?**	Response **[+CGEQOS:<cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>]][<CR><LF>+CGEQOS:<cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>][...]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGEQOS=?**	Response **+CGEQOS:(**range of supported**<cid>**s**),(**list of supported**<QCI>**s**),(**list of supported**<DL_GBR>**s**),(**list of supported**<UL_GBR>**s**),(**list of supported**<DL_MBR>**s**),(**list of supported**<UL_MBR>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Maximum Response Time	300ms

Parameter

<cid>	Integer type. Specifies a particular PDP context definition (see the **AT+CGDCONT** command).
<QCI>	Integer type. Specifies a class of EPS QoS (see *3GPP TS 23.203 [85]* and *3GPP TS* 24.301 [83]). 0 QCI is selected by network [1–4] Value range for guaranteed bit rate Traffic Flows 75 Value for guaranteed bit rate Traffic Flows [5–9] Value range for non-guaranteed bit rate Traffic Flows 79 Value for non-guaranteed bit rate Traffic Flows [128–254] Value range for Operator-specific QCIs The QCI values 65, 66, 69 and 70 are not allowed to be requested by the UE. If the TE requests a QCI parameter 65, 66, 69 or 70, the MT responds with result code **+CME ERROR: 181** (unsupported QCI value).
<DL_GBR>	Integer type. Indicates DL GBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI (see *3GPP TS 24.301 [83]*).
<UL_GBR>	Integer type. Indicates UL GBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI (see *3GPP TS 24.301 [83]*).
<DL_MBR>	Integer type. Indicates DL MBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI (see *3GPP TS 24.301 [83]*).
<UL_MBR>	Integer type. Indicates UL MBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI (see *3GPP TS 24.301 [83]*).

NOTES

1. Parameter **<QCI>** can be set alone, and only 0, 5, 6, 7, 8, 9 and 79 are supported for **<QCI>**.
2. This command can only be executed when powering on the module or defining PDP context.

Example

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

OK
AT+CGEQOS=?
+CGEQOS:(0),(0,5,6,7,8,9,79)
```

OK

2.43. AT+CGTFT Traffic Flow Template

This command allows the TE to specify a Packet Filter - PF for a Traffic Flow Template - TFT that is used in the GGSN in UMTS/GPRS and Packet GW in EPS for routing of packets onto different QoS flows towards the TE. The concept is further described in the *3GPP TS 23.060 [47]*. A TFT consists of from one and up to 16 Packet Filters, each identified by a unique **<packet filter identifier>**. A Packet Filter also has an **<evaluation precedence index>** that is unique within all TFTs associated with all PDP contexts that are associated with the same PDP address.

The write command specifies a Packet Filter that is to be added to the TFT stored in the MT and used for the context identified by the (local) context identification parameter, **<cid>**. The specified TFT will be stored in the GGSN in UMTS/GPRS and Packet GW in EPS only at activation or MS-initiated modification of the related context. Since this is the same parameter that is used in the **AT+CGDCONT** command, the **AT+CGTFT** command is effectively an extension to these commands. The Packet Filters consist of a number of parameters, each of which may be set to a separate value. A special form of the set command **AT+CGTFT=<cid>** causes all of the Packet Filters in the TFT for context number **<cid>** to become undefined. At any time there may exist only one PDP context with no associated TFT amongst all PDP contexts associated to one PDP address. At an attempt to delete a TFT, which would violate this rule, an **ERROR** or **+CME ERROR** response is returned. Extended error responses are enabled by the **AT+CMEE** command.

The read command returns the current settings for all Packet Filters for each defined context.

The test command returns values supported as compound values. If the MT supports several PDP types, the parameter value ranges for each PDP type are returned on a separate line. TFTs shall be used for PDP-type IP and PPP only. For PDP-type/PPP, a TFT is applicable only when IP traffic is carried over PPP. If PPP carries header compressed IP packets, then a TFT cannot be used.

AT+CGTFT Traffic Flow Template	
Write Command **AT+CGTFT=[<cid>,[<packet filter identifier>,<evaluation precedence index>[,<remote address_and subnet mask>[,<protocol number (ipv4) / next header (ipv6)>[,<local port range>[,<remote port range>[,<ipsec security parameter index (spi)>[,<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>[,<flow label (ipv6)>[,<direction>[,<local addr**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

ess and subnet mask>]]]]]]]]]]	
Read Command **AT+CGTFT?**	Response **[+CGTFT: <cid>,<packet filter identifier>,<evaluation precedence index>,<remote address and subnet mask>,<protocol number (ipv4)/next header (ipv6)>,<local port range>,<remote port range>,<ipsec security parameter index (spi)>,<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>,<flow label (ipv6)>,<direction>,<local address and subnet mask>]** **[+CGTFT: <cid>,<packet filter identifier>,<evaluation precedence index>,<remote address and subnet mask>,<protocol number (ipv4)/next header (ipv6)>,<local port range>,<remote port range>,<ipsec security parameter index (spi)>,<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>,<flow label (ipv6)>,<direction>,<local address and subnet mask>[...]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CGTFT=?**	Response **+CGTFT: <PDP_type>,(**list of supported**<packet filter identifier>**s)**,(**list of supported **<evaluation precedence index>**s)**,(**list of supported **<remote Traffic flow template address and subnet mask>**s)**,(**list of supported **<protocol number (ipv4) /next header (ipv6)>**s)**,(**list of supported **<local port range>**s)**,(**list of supported **<remote port range>**s)**,(**list of supported **<ipsec security parameter index (spi)>**s)**,(**list of supported **<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>**s)**,(**list of supported **<flow label (ipv6)>**s)**,(**list of supported **<direction>**s)**,(**list of supported **<local address and subnet mask>**s) **[<CR><LF>+CGTFT: <PDP_type>,(**list of supported **<packet filter identifier>**s)**,(**list of supported **<evaluation precedence index>**s)**,(**list of supported **<remote address and subnet mask>**s)**,(**list of supported **<protocol number (ipv4) / next header (ipv6)>**s)**,(**list of supported **<local port range>**s)**,(**list of supported **<remote port range>**s)**,(**list of supported **<ipsec security parameter index (spi)>**s)**,(**list of supported **<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>**s)**,(**list of supported **<flow label (ipv6)>**s)**,(**list of

	supported **<direction>**s)**,(**list of supported **<local address and subnet mask>**s)**[...]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. Specifies a particular PDP context definition (refer to **AT+CGDCONT** command).
<PDP_type>	String type. Specifies the type of packet data protocol (see the **AT+CGDCONT** command).
<packet filter identifier>	Integer type. Value range is from 1 to 16.
<evaluation precedence index>	Integer type. The value range is from 0 to 255.
<remote address and subnet mask>	String type. The string is given as dot-separated numeric (0-255) parameters on the form: **"a1.a2.a3.a4.m1.m2.m3.m4"** for IPv4 or **"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. When **AT+CGPIAF**(not supported currently) is supported, its settings can influence the format of this parameter returned with the read form of **AT+CGTFT**.
<protocol number (ipv4) / next header (ipv6)>	Integer type. Value range is from 0 to 255.
<local port range>	String type. The string is given as dot-separated numeric (0-65535) parameters on the form "f.t".
<remote port range>	String type. The string is given as dot-separated numeric (0-65535) parameters on the form "f.t".
<ipsec security parameter index (spi)>	Numeric value in hexadecimal format. The value range is from 00000000 to FFFFFFFF.
<type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>	String type. The string is given as dot-separated numeric (0-255) parameters on the form "t.m".
<flow label (ipv6)>	Numeric value in hexadecimal format. The value range is from 00000 to FFFFF. Valid for IPv6 only.
<direction>	Integer type. Specifies the transmission direction in which the packet filter shall be applied. 0 Pre-Release 7 TFT filter (see *3GPP TS 24.008 [8], table*

	10.5.162) 1 Uplink 2 Downlink 3 Bidirectional (Up & Downlink)
<local address and subnet mask>	String type. The string is given as dot-separated numeric (0-255) parameters on the form: **"a1.a2.a3.a4.m1.m2.m3.m4"** for IPv4 or **"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. When **AT+CGPIAF**(not supported currently) is supported, its settings can influence the format of this parameter returned with the read form of **AT+CGTFT**. Some of the above listed attributes may coexist in a Packet Filter while others mutually exclude each other, the possible combinations are shown in *3GPP TS 23.060 [47]*.

NOTES

1. This command can only be executed when powering on the module or defining PDP context.
2. **<local address and subnet mask>** is not supported yet.

Example

```
AT+CGTFT=0,2,6,"32.1.11.160.0.0.0.0.0.0.0.0.0.0.0.0.255.255.255.255.0.0.0.0.0.0.0.0.0.0.0.0",17,"60001.60001","60350.60450",0x123456,168.252,0x12345,1

OK
AT+CGTFT?
+CGTFT:0,2,6,"2001:BA0::.FFFF:FFFF::",17,"60001.60001","60350.60450",0x123456,"168.252",0x12345,1

OK
AT+CGTFT=?
+CGTFT:"IP",(1,16),(0,255),"(0.0.0.1,255.255.255.255).(0.0.0.1,255.255.255.255)",(0,255),"(0,65535).(0,65535)","(0,65535).(0,65535)",(0x0,0xFFFFFFFF),"(0,255).(0,255)",,(0,3)

+CGTFT:"NONIP",(1,16),(0,255),,(0,255),"(0,65535).(0,65535)","(0,65535).(0,65535)",(0x0,0xFFFFFFFF),"(0,255).(0,255)",,(0,3)

+CGTFT:"IPV6",(1,16),(0,255),"(::1,FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF).(::1,FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF)",(0,255),"(0,65535).(0,65535)","(0,65535).(0,65535)",(0x0,0xFFFFFFFF),"(0,255).(0,255)",(0x0,0xFFFFF),(0,3)

+CGTFT:"IPV4V6",(1,16),(0,255),"(0.0.0.1,255.255.255.255)\(::1,FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:
```

```
FFFF:FFFF).(0.0.0.1,255.255.255.255)\(::1,FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF)",(0,255),"
(0,65535).(0,65535)","(0,65535).(0,65535)",(0x0,0xFFFFFFFF),"(0,255).(0,255)",(0x0,0xFFFFF),(0,3)

OK
```

3 3GPP Commands (27.005)

3.1. 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. If chosen service is not supported by the ME, final result code **+CME ERROR:<err>** will be returned.

The read command returns supported message types along the current service setting.

The test command returns a list of all services supported by the UE.

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

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

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)
	2...128	Reserved
<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

NOTES

1. UE will report the URC **+CMT:[<alpha>],<length><CR><LF><pdu>** (PDU mode enabled) when receiving SMS message.
2. **<alpha>** in the above URC will always be 0. For more details about the URC, please refer to **AT+CNMI** in *3GPP TS 27.005*.

Example

```
AT+CSMS=1
+CSMS:2,3,4

OK
AT+CSMS?
+CSMS:1,2,3,4

OK
```

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

OK
```

3.2. AT+CNMA New Message Acknowledgement to UE

The execution command confirms reception of a new message (SMS-DELIVER or SMS-STATUS-REPORT) which is routed directly to the UE. This acknowledgement command shall be used when the parameter **<service>** of **AT+CSMS** equals 1. In PDU mode, it is possible to send either positive (RPACK) or negative (RP-ERROR) acknowledgement to the network. Parameter **<n>** defines which one will be sent. Optionally (when **<length>** is greater than zero) an acknowledgement TPDU (SMS-DELIVER-REPORT for RPACK or RP-ERROR) may be sent to the network. The entering of PDU is done similarly as specified in command **AT+CMGS**, except that the format of **<ackpdu>** is used instead of **<pdu>** (i.e. SMSC address field is not present). PDU shall not be bounded by double quotes. UE shall not send another **+CMT** or **+CDS** result code to TE before previous one is acknowledged.

If UE does not get acknowledgement within required time (network timeout), UE should respond as specified in *3GPP TS 24.011 [6]* to the network.

If the command is executed, but no acknowledgement is expected, or some other UE related error occurs, final result code **+CME ERROR:<err>** is returned. In case that a directly routed message must be buffered in UE or AT interpreter remains too long in a state where result codes cannot be sent to TE (e.g. user is entering a message using **AT+CMGS**), acknowledgement (RP-ACK) must be sent to the network without waiting **AT+CNMA** command from TE. Later, when buffered result codes are flushed to TE, TE must send **+CNMA[=0]** acknowledgement for each result code. In this way, UE can determine if message should be placed in non-volatile memory and routing to TE disabled (**+CNMA[=0]** not received).

The test command returns a list of supported **<n>** values. If the only value supported is 0, the device does not support sending of TPDU.

AT+CNMA New Message Acknowledgement to UE	
Execution/Write Command **AT+CNMA=[<n>[,<length>[<CR>**PDU is given**]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CNMA=?**	Response **+CNMA:(**list of supported **<n>**s**)** **OK**

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type 1 Send RP-ACK (or buffered result code received correctly) 2 Send RP-ERROR (if PDU is not given, ME/TA shall send SMS-DELIVER-REPORT with *3GPP TS 23.040 [3]* TP-FCS value set to 'FF' (unspecified error cause))
<length>	Integer type. Indicates in the text mode the length of the message body **<data>** (or **<cdata>**) in characters; or in PDU mode, the length of the actual TP data unit in octets (i.e. the RP layer SMSC address octets are not counted in the length). The range is 0-232.

NOTES

1. UE will report the URC **+CMT:[<alpha>],<length><CR><LF><pdu>** (PDU mode enabled) when receiving SMS message.
2. **<alpha>** in the above URC will always be 0. For more details about the URC, please refer to **+CNMI** in *3GPP TS 27.005*.

Example

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

OK
```

3.3. AT+CSCA Service Centre Address

The write command updates the SMSC address, through which mobile originated SMS are transmitted. In text mode, setting is used by send and write commands. In PDU mode, setting is used by the same commands, but only when the length of the SMSC address coded into **<pdu>** parameter equals zero.

AT+CSCA Service Centre Address	
Write Command **AT+CSCA=<sca>[,<tosca>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CSCA?**	Response **+CSCA:<sca>[,<tosca>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CSCA=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<sca>	*3GPP TS 24.011 [6]* RP SC address Address-Value field in string format. BCD numbers (or GSM 7-bit default alphabet characters) are converted to characters of the currently selected TE character set (refer to **+CSCS** command in *3GPP TS 27.007 [9]*); type of address given by **<tosca>**.
<tosca>	*3GPP TS 24.011 [6]* RP SC address Type-of-Address octet in integer format (when first character of **<da>** is "+(IRA 43)" (refer to *3GPP TS 27.005*) default is 145, otherwise, default is 129).

Example

```
AT+CSCA=358501234567,145
OK
AT+CSCA?
+CSCA:"358501234567",145

OK
AT+CSCA=?
OK
```

3.4. AT+CMGS Send SMS Message

The write command sends message from a TE to the network (SMS-SUBMIT). Message reference value **<mr>** is returned to the TE upon successful message delivery. Optionally (when **AT+CSMS <service>** value is 1 and supported by network) **<ackpdu>** is returned. Values can be used to identify message upon unsolicited delivery status report result code. If sending fails in a network or an UE error, final result code **+CME ERROR:<err>** is returned. This command should be abortable.

- **<length>** must indicate the number of octets coded in the TP layer data unit to be given (i.e. SMSC address octets are excluded).
- The UE shall send a four-character-sequence **<CR><LF><greater_than><space>** (IRA 13, 10, 62, 32) after command line is terminated with **<CR>**; after that PDU can be given from TE to UE.
- The DCD signal shall be in ON state while PDU is given.
- The echoing of given characters back from the UE is controlled by V.25ter echo command E.
- The PDU shall be hexadecimal format (similarly as specified for **<pdu>**) and given in one line; UE converts this coding into the actual octets of PDU.
- When the length octet of the SMSC address (given in the PDU) equals zero, the SMSC address set with **AT+CSCA** command is used; in this case the SMSC Type-of-Address octet shall not be present in the PDU, i.e. TPDU starts right after SMSC length octet.

AT+CMGS Send SMS Message	
Write Command **AT+CMGS=<length>** After **>** is responded, input the PDU.	Response **+CMGS:<mr>[,<ackpdu>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CMGS=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<mr>	*3GPP TS 23.040 [3]* TP-Message-Reference in integer format.
<length>	Integer type. Indicates in the text mode (**AT+CMGF=1**) the length of the message body **<data>** (or **<cdata>**) in characters; or in PDU mode (**AT+CMGF=0**), the length of the actual TP data unit in octets (i.e. the RP layer SMSC address octets are not counted in the length). The range is 7-164.
<ackpdu>	*3GPP TS 23.040 [3]* RP-User-Data element of RP-ACK PDU. The format is same as that

	of **<pdu>** in case of SMS, but without *3GPP TS 24.011 [6]* SC address field and parameter shall be bounded by double quote characters like a normal string type parameter.

NOTES

1. **<ackpdu>** is not returned currently.
2. For address field in PDU (*3GPP TS 24.011*), the country code should be added at the beginning of it, e.g., 86 for China.

3.5. AT+CMGC Send SMS Command

The execution command sends a command message from TE to the network (SMS-COMMAND). The entering of text (*3GPP TS 23.040 [3]* TP-Command-Data) is done similarly as specified in **AT+CMGS** command, but the format is fixed to be a sequence of two IRA character long hexadecimal numbers which UE converts into 8-bit octets (refer to **AT+CMGS**). Message reference value **<mr>** is returned to the TE on successful message delivery. Optionally (when **AT+CSMS <service>** value is 1 and network supports) **<scts>** is returned. Values can be used to identify message upon unsolicited delivery status report result code. If sending fails in a network or a UE error, final result code **+CME ERROR:<err>** is returned. This command should be abortable.

AT+CMGC Send SMS Command	
Write Command **AT+CMGC=<length>** After **>** is responded, input the PDU.	Response **+CMGC:<mr>[,<ackpdu>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CMGC=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<mr>	*3GPP TS 23.040 [3]* TP-Message-Reference in integer format
<length>	Integer type. Indicates in the text mode (**AT+CMGF=1**) the length of the message body **<data>** (or **<cdata>**) in characters; or in PDU mode (**AT+CMGF=0**), the length of the actual

	TP data unit in octets (i.e. the RP layer SMSC address octets are not counted in the length). The range is 8-174.
<ackpdu>	*3GPP TS 23.040 [3]* RP-User-Data element of RP-ACK PDU. The format is same as for **<pdu>** in case of SMS, but without *3GPP TS 24.011 [6]* SC address field and parameter shall be bounded by double quote characters like a normal string type parameter.

NOTES

1. **<ackpdu>** is not returned currently.
2. For address field in PDU (*3GPP TS 24.011*), the country code should be added at the beginning of it, e.g., 86 for China.

3.6. AT+CMMS More Messages to Send

The write 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 More Messages to Send	
Write Command **AT+CMMS=[<n>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+CMMS?**	Response **+CMMS:<n>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+CMMS=?**	Response **+CMMS:(**list of supported **<n>**s**)** **OK**
Maximum Response Time	300ms

Parameter

<n>	Integer type 0 Disable (default value) 1 Keep enabled until the time between the response of the latest message sending command (**AT+CMGS**, **AT+CMSS**, etc.) and the next sending command exceeds 1-5 seconds (the exact value is up to ME implementation), then ME shall close the link and TA switches **<n>** automatically back to 0. 2 Enable (if the time between the response of the latest message sending command and the next sending command exceeds 1-5 seconds (the exact value is up to ME implementation), ME shall close the link but TA shall not switch automatically back to **<n>=0**)

Example

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

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

OK
```

4 General Commands

4.1. AT+NRB Reboot the UE

The command reboots the UE. There is a short delay before the UE reboots after the command is executed. No further AT commands will be processed.

Please note that there is no final **OK** to signal that the command line has finished processing as AT command processing terminates with this command. No confirmation messages are expected until the reboot.

AT+NRB Reboot the UE	
Execution Command **AT+NRB**	Response **REBOOTING**
Maximum Response Time	300ms

Example

```
AT+NRB
REBOOTING
```

4.2. AT+NUESTATS Query UE Statistics

The 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	
Execution Command **AT+NUESTATS**	Response **Signal power:<signal power in centibels>** **Total power:<total power in centibels>** **TX power:<current Tx power level in centibels>** **TX time:<total Tx time since last reboot in millisecond>** **RX time:<total Rx time since last reboot in millisecond>**

	Cell ID:<last cell ID> **ECL:<last ECL value>** **SNR:<last snr value>** **EARFCN:<last earfcn value>** **PCI:<last pci value>** **RSRQ:<rsrq in centibels>** **OPERATOR MODE:<operator mode>** **CURRENT BAND:<current band>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Write Command **AT+NUESTATS=RADIO**	Response **NUESTATS:RADIO,Signal power:<signal power in centi bels>** **NUESTATS:RADIO,Total power:<total power in centibel s>** **NUESTATS:RADIO,TX power:<current Tx power level i n centibels>** **NUESTATS:RADIO,TX time:<total Tx time since last re boot in millisecond>** **NUESTATS:RADIO,RX time:<total Rx time since last re boot in millisecond>** **NUESTATS:RADIO,Cell ID:<last cell ID>** **NUESTATS:RADIO,ECL:<last ECL value>** **NUESTATS:RADIO,SNR:<last snr value>** **NUESTATS:RADIO,EARFCN:<last earfcn value>** **NUESTATS:RADIO,PCI:<last pci value>** **NUESTATS:RADIO,RSRQ:<rsrq in centibels>** **NUESTATS:RADIO,OPERATOR MODE:<operator mode>**

	NUESTATS:RADIO,CURRENT BAND:<current band> **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Write Command **AT+NUESTATS=CELL**	Response **NUESTATS:CELL,<earfcn>,<physical cell id>,<primarycell>,<rsrp>,<rsrq>,<rssi>,<snr>** **[...NUESTATS:CELL,<earfcn>,<physical cell id>,<primarycell>,<rsrp>,<rsrq>,<rssi>,<snr>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Write Command **AT+NUESTATS=THP**	Response **NUESTATS:THP,<throughput_type>,<throughput>** **[...NUESTATS:THP,<throughput_type>,<throughput>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Write Command **AT+NUESTATS=BLER**	Response **NUESTATS:BLER,<block error rate type>,<block error rate>** **[...NUESTATS:BLER,<block error rate type>,<block error rate>]** **OK** If there is any error: **ERROR**

	Or **+CME ERROR: <err>**
Write Command **AT+NUESTATS=APPSMEM**	Response **NUESTATS:APPSMEM,Current Allocated:<allocated>** **NUESTATS:APPSMEM,Total Free:<free>** **NUESTATS:APPSMEM,Max Free:<max free>** **NUESTATS:APPSMEM,Num Allocs:<num allocs>** **NUESTATS:APPSMEM,Num Frees:<num frees>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Write Command **AT+NUESTATS=<type>**	Response **NUESTATS:<type>,<name/value>,<value>[,<value[,<value>[...]]]** **[...NUESTATS:<type>,<name/value>,<value>[,<value[,<value>[...]]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NUESTATS=?**	Response **NUESTATS:(**list of supported **<type>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<type>	Type of data to be displayed as an unquoted string. Supported values of **<type>** are: RADIO Radio specific information CELL Per-cell information for the top 8 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>	Signal power in centibels
<total power in centibels>	Total power in centibels
<current TX power level in centibels>	Current Tx power level in centibels
<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 SIB1 cell ID>	Last SIB1 cell ID
<last ECL value>	Last ECL value
<last snr value>	Last SNR value
<last earfcn value>	Last EARFCN value
<last pci value>	Last PCI value
<rsrq in centibels>	Reference signal received quality in centibels
<operator mode>	Operator mode for SIB1: 0 Unknown mode 1 Inband different PCI mode 2 Inband same PCI mode 3 Guardband mode 4 Standalone mode
<current band>	The band of service cell.

If **<type>=CELL**, per-cell information for the top 10 cells. Returned entries are of the form: **<earfcn>,<physical cell id>,<primary cell>,<rsrp>,<rsrq>,<rssi>,<snr>**

<earfcn>	Absolute radio-frequency channel number
<physical cell id>	Physical ID of the cell
<primary cell>	1 indicates the current serving cell
<rsrp>	Reference signal received power
<rsrq>	Reference signal received quality
<rssi>	Received signal strength indicator
<snr>	Signal to noise ratio

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

<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>	Physical layer block error rate (uplink). Integer %
<mac_dl_bler>	Physical 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**, returned entries are:	
<rlc_ul>	RLC layer throughput (uplink). Integer bps
<rlc_dl>	RLC layer throughput (downlink). Integer bps
<mac_ul>	Physical layer throughput (uplink). Integer bps
<mac_dl>	Physical layer throughput (downlink). Integer bps
If **<type>=APPSMEM**, dynamic memory usage, returned entries are:	
<allocated>	Current allocated size
<free>	Total free size
<max free>	Max free size
<num allocs>	Number of times to allocate memory
<num frees>	Number of times to free memory

NOTES

1. The response of execution command **AT+NUESTATS** shares the same information with that of the write command **AT+NUESTATS=RADIO** but without the command and variant prefix (i.e. NUESTATS:RADIO). This execution command will be removed in a future release.
2. TX time and RX time will be cleared in the three cases of UE startup, RRC deactivate, and OoS (Out of Service).

Example

```
AT+NUESTATS
Signal power:-842
Total power:-780
TX power:100
TX time:859
RX time:26543
Cell ID:137262770
ECL:0
SNR:226
EARFCN:3734
PCI:105
RSRQ:-108
```

```
OPERATOR MODE:4
CURRENT BAND:8

OK
AT+NUESTATS=CELL
NUESTATS:CELL,3734,105,1,-842,-108,-780,226

OK
AT+NUESTATS=THP
NUESTATS:THP,RLC UL,100

NUESTATS:THP,RLC DL,98

NUESTATS:THP,MAC UL,103

NUESTATS:THP,MAC DL,100

OK
AT+NUESTATS=BLER
NUESTATS:BLER,RLC UL BLER,10

NUESTATS:BLER,RLC DL BLER,5

NUESTATS:BLER,MAC UL BLER,8

NUESTATS:BLER,MAC DL BLER,3

NUESTATS:BLER,Total TX bytes,1080

NUESTATS:BLER,Total RX bytes,900

NUESTATS:BLER,Total TX blocks,80

NUESTATS:BLER,Total RX blocks,80

NUESTATS:BLER,Total RTX blocks,100

NUESTATS:BLER,Total ACK/NACK RX,100

OK
```

4.3. AT+NEARFCN Specify Search Frequencies

The write command provides a mechanism to lock to a specific E-UTRAN Absolute Radio Frequency Channel Number (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. It is not persistent over reboots. If the specified EARFCN is not present, the UE will enter out of service mode. If the specified PCI is not present, the UE will enter out of service mode.

AT+NEARFCN Specify Search Frequencies	
Write Command **AT+NEARFCN=<search_mode>,<earfcn>[,<pci>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NEARFCN=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<search_mode>	Integer type. It specifies the type of search and defines the supplied parameters 0 Lock to a specific EARFCN
<earfcn>	Integer type. A number in the range of 1-65535 representing the EARFCN to search.
<pci>	String type. E-UTRAN physical cell ID in hexadecimal format. Valid range: 0-1F7.

Example

```
AT+NEARFCN=0,2506,AB
OK
```

4.4. AT+NSOCR Create a Socket

The command creates a socket on the UE and associates with specified protocol. If the port is set, receiving is enabled and **+NSONMI** unsolicited messages will appear for any message that is received on that port.

If a socket has already been created for a protocol or port combination, then **AT+NSOCR** will fail if

requested again.

AT+NSOCR Create a Socket	
Write Command **AT+NSOCR=<type>,<protocol>,<listen port>[,<receive control>[,<af_type>[,<ip address>]]]**	Response **<socket>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<type>	Socket type. DGRAM UDP STREAM TCP
<protocol>	Integer type. Standard internet protocol definition. For example, UDP is 17, and TCP is 6.
<listen port>	Integer type. A number in the range of 0-65535. This is the local port that will be included in sent messages and on which messages will be received. If it is 0 (When creating a TCP socket), the module will assign a random **<listen port>** for this socket.
<socket>	This is a reference to the created socket. It is an integer greater than or equal to 0. A maximum of 7 sockets are supported, but other services may reduce this number.
<receive control>	Set to 1 if incoming messages should be received, 0 if incoming messages should be ignored. The default value is 1 (messages will be received).
<af_type>	String type. Address family type. The default is **"AF_INET"**. AF_INET IPv4 AF_INET6 IPv6.
<ip address>	IP address. The IP address of the network assigned to UE.

NOTES

1. When BIP (Bearer Independent Protocol) is enabled, **<listen port>** cannot be set into 20000 for TCP.
2. A maximum of 7 sockets are supported, but other services such as MQTT, CoAP, etc. may reduce this number.
3. For the parameter **<listen port>**, ports 5683, 5684, 56830, 56831 and 56833 are reserved ports

and are not recommended.

Example

```
AT+NSOCR=DGRAM,17,4587,1,AF_INET
1

OK
AT+NSOCR=DGRAM,17,1234,0
1

OK
```

4.5. AT+NSOST SendTo Command (UDP Only)

The 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+NSOST** will indicate how much of the data was successfully sent.

AT+NSOST SendTo Command (UDP Only)	
Write Command **AT+NSOST=<socket>,<remote_addr>,<remote_port>,<length>,<data>[,<sequence>]**	Response **<socket>,<length>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<socket>	Integer type. Socket number returned by **AT+NSOCR**.
<remote_addr>	IP address which can be specified in decimal, octal or hexadecimal notation.
<remote_port>	Integer type. A number in the range of 0-65535. This is the remote port on which messages will be received. Decimal and hexadecimal formats are supported.
<length>	Integer type. Decimal length of data to be sent.

<data>	Data to be transmitted in hex string format. The default maximum length of sent data is 1358 bytes, which actually depends on the MTU value sent by the network.
<sequence>	Sequence of data. Range: 1-255. If it is omitted, data sent status will not be reported. If specified, when datagram is sent over RF or is discarded, then the result will be reported: **+NSOSTR:<socket>,<sequence>,<status>**
<status>	The status of datagram. 0 Error 1 Sent

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+NSOST=1,192.158.5.1,1024,2,AB30,1
1,2

OK
```

4.6. AT+NSOSTF SendTo Command with Flags (UDP Only)

The 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+NSOSTF** return value will indicate how much of the data was successfully sent.

AT+NSOSTF SendTo Command with Flags (UDP Only)	
Write Command **AT+NSOSTF=<socket>,<remote_addr>,<remote_port>,<flag>,<length>,<data>[,<sequence>]**	Response **<socket>,<length>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Maximum Response Time	300ms

Parameter

<socket>	Integer type. Socket number returned by **AT+NSOCR**.
<remote_addr>	A dot-decimal notation IPv4 address. IP addresses can be specified in decimal, octal or hexadecimal notation.
<remote_port>	Integer type. A number in the range of 0-65535. This is the remote port on which messages will be received. Decimal and hexadecimal formats are supported.
<flag>	Integer type. It specifies the type of message transmission. Values of this argument are in hex format and are formed by logically OR'ing zero or more of the following flags: 0x100 Exception Message: Send message with high priority 0x200 Release Indicator: indicate release after this message 0x400 Release Indicator: indicate release after this message has been replied to If no flag is set, a value of 0 should be provided.
<length>	Integer type. Decimal length of data to be sent.
<data>	Data to be transmitted in hex string format. The default maximum length of sent data is 1358 bytes, which actually depends on the MTU value sent by the network.
<sequence>	Sequence of data. Range: 1-255. If it is omitted, data sending status will not be reported. If specified, when datagram is sent over RF or is discarded, then the result will be reported: **+NSOSTR:<socket>,<sequence>,<status>**
<status>	The status of datagram. 0 Error 1 Sent

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+NSOSTF=1,192.158.5.1,1024,0x100,2,AB30,1
1,2

OK
```

4.7. AT+NQSOS Query the List of Pending Socket Message

This command queries the list of the pending upstream message by UE.

AT+NQSOS Query the List of Pending Socket Message	
Write Command **AT+NQSOS=<socket>[,<socket>[,<socket>[…]]]**	Response **[+NQSOS:<socket>,<sequence>]** **[+NQSOS:<socket>,<sequence>]** **[…]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NQSOS?**	Response **[+NQSOS:<socket>,<sequence>]** **[+NQSOS:<socket>,<sequence>]** **[…]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<socket>	Socket number.
<sequence>	The sequence of pending upstream message. Range: 1-255.

Example

```
AT+NQSOS=1,2
+NQSOS:1,2
+NQSOS:2,3

OK
AT+NQSOS?
```

```
+NQSOS:1,2
+NQSOS:2,3

OK
```

4.8. AT+NSORF Receive Command

The command is used to receive data on a socket.

The command can read up to **<req_length>** characters of data from **<socket>**, and returned length is the actual number of characters returned. When data arrives, a **+NSONMI** response will be generated to indicate the socket the message was received on and also the amount of data. The **AT+NSORF** command takes a length, which is the maximum amount of data that will be returned.

If the requested length is larger than the actual size of the returned data, only the length of returned data is provided, and the remaining length is returned as 0. If the requested length is less than the amount of data returned, only the requested amount of data will be returned, plus an indication of the number of bytes remaining. Once a message has been fully read, a new **+NSONMI** notification will be sent if there is another message to process.

If messages arrive faster than they are read, and the internal message buffer is full, the most recent message will be discarded.

AT+NSORF Receive Command	
Write Command **AT+NSORF=<socket>,<req_length>**	Response **<socket>,<ip_addr>,<port>,<length>,<data>,<remaining_length>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<socket>	Integer type. Socket number returned by **AT+NSOCR**.
<req_length>	Maximum amount of data to be returned as a decimal byte length. The maximum

	request data length is 1358 bytes.
<ip_addr>	Address of system sending the message A dot-decimal notation IPv4 address. IP addresses can be specified in decimal, octal or hexadecimal notation. Only IPv4 is supported.
<port>	Port in the range of 0-65535. This is the remote port that messages were sent from.
<length>	Amount of data returned as a decimal byte length.
<remaining_length>	Amount of data remained to read for this message as a decimal byte length. Remaining length is always 0. The remaining data is readable.
<data>	Data received in hex string format. Maximum length of received data is 1358 bytes.

Example

```
AT+NSORF=1,10
1,192.168.5.1,1024,2,ABAB,0

OK
```

4.9. AT+NSOCO Connect Command (TCP Only)

The command connects a TCP server to the specified host:port.

AT+NSOCO Connect Command (TCP Only)	
Write Command **AT+NSOCO=<socket>,<remote_addr>,<remote_port>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<socket>	Integer type. Socket number returned by **AT+NSOCR**.
<remote_addr>	Address of system sending the message. IP addresses can be specified in decimal, octal or hexadecimal notation.
<remote port>	A number in the range of 0-65535. This is the remote port to be connected to. Decimal and hexadecimal formats are supported.

Example

```
AT+NSOCO=1,192.158.5.1,1024
OK
```

4.10. AT+NSOSD Sending Command (TCP Only)

The 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. If the amount of data is larger than the largest datagram that can be sent, then **AT+NSOSD** return value will indicate how much the data was successfully sent.

If **<sequence>** is specified, when datagram is asked for by the server or is discarded by UE, the result will be reported.

AT+NSOSD Sending Command (TCP Only)	
Write Command **AT+NSOSD=<socket>,<length>,<data>[,<flag>[,<sequence>]]**	Response **<socket>,<length>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<socket>	Integer type. Socket number returned by **AT+NSOCR**.
<length>	Integer type. Decimal length of data to be sent.
<data>	Data to be transmitted in hex string format. The default maximum length of send data is 1358 bytes, which actually depends on the MTU value sent by the network.
<flag>	Integer type. It specifies the type of message transmission. Values of this argument are in hex format and are formed by logically OR'ing zero or more of the following flags: 0x100 Exception Message: Send message with high priority 0x200 Release Indicator: indicate release after this message 0x400 Release Indicator: indicate release after this message has been replied to If no flags are set, a value of 0 should be provided.
<sequence>	Sequence of data. Range 1-255. If it is omitted, data sent status will not be

	reported. If specified, when datagram is acked by server or is discarded by UE, the result will be reported: **+NSOSTR:<socket>,<sequence>,<status>**
<status>	The status of datagram. 0 Error 1 Sent

NOTE

<flag> could not work currently.

Example

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

OK
AT+NSOSD=1,2,AB30,0x100
1,2

OK
AT+NSOSD=1,2,AB30,0x100,255
1,2

OK
```

4.11. AT+NSOCL Close a Socket

The command is used to close the specified socket. If there are pending messages to be read, they will be dropped. No further unsolicited **+NSONMI** notification will be generated. If the socket has already been closed, or was never created, an error will be returned.

AT+NSOCL Close a Socket	
Write Command **AT+NSOCL=<socket>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Maximum Response Time	300ms

Parameter

<socket>	Integer type. Socket number returned by **AT+NSOCR**.

Example

```
AT+NSOCL=1
OK
```

4.12. AT+NSONMI Indication of Arrived Socket Message

The write command is used to set the URC **+NSONMI** to indicate arrived socket messages:

The read command returns the current setting of the command.

AT+NSONMI Indication of Arrived Socket Message	
Write Command **AT+NSONMI=<mode>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NSONMI?**	Response **+NSONMI:<mode>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NSONMI=?**	Response **+NSONMI: (**list of supported **<mode>**s**)** **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<mode>	Integer type. Controlling downlink data format. 0 Disable the URC to indicate messages. 1 Enable URC to indicate messages: **+NSONMI:<socket>,<length>**. 2 Enable the URC below to indicate messages, server addresses and user data: **+NSONMI: <socket>,<remote_addr>,<remote_port>,<length>,<data>** 3 Enable the URC below to indicate messages and user data: **+NSONMI: <socket>,<length>,<data>**
<socket>	Integer type. Socket on which data is received. Decimal number returned by **AT+NSOCR**.
<length>	Integer type. Number of bytes of data in the first message.
<remote_addr>	Address of system sending the message.
<remote_port>	Integer type. A number in the range of 0-65535. This is the remote port on which messages will be received.
<data>	Data received in hex string format. Maximum length of received data is 1358 bytes.

Example

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

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

OK
```

NOTE

The URC **+NSONMI:<socket>,<length>** may occur at any point if it indicates a new message with no messages buffered. If there are buffered messages, it will be reported after preceding messages have been completely read by **AT+NSORF** command.

4.13. +NSOCLI Socket Close Indicator (Response Only)

This is an unsolicited message to notify that a socket has been closed by LwIP internally. It returns the socket number.

+NSOCLI Socket Close Indicator (Response Only)	
URC Format: **+NSOCLI: <socket>**	

Parameter

<socket>	Integer type. The socket on which data is received. It is a decimal number returned by **AT+NSOCR**.

NOTE

The URC will be reported under TCP only.

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

The command sends an ICMP packet to the specified host address.

AT+NPING initiates the sending of a PING packet to the specified address. This will either cause a packet to be returned if the remote system is connected and responding to PING packets or no response will be received. A maximum of 1 ping attempts will be tried. If none of the packets receive a response within the timeout period, an error will be returned.

If a response is received, the unsolicited **+NPING** message will be returned. If no response is received the **+NPINGERR** unsolicited response will be returned with an error value.

AT+NPING Test IP Network Connectivity to a Remote Host	
Write Command **AT+NPING=<remote_address>[,<p_size>[,<timeout>]]**	Response **OK** **+NPING:<remote_address>,<ttl>,<rtt>** If there is any error: **ERROR** Or

	+NPINGERR:<err>
Maximum Response Time	300ms

Parameter

<remote_address>	Address of system sending the message A dot-decimal notation IPv4 address. IP addresses can be specified in decimal, octal or hexadecimal notation. Only IPv4 is supported.
<p_size>	Integer type. Size in bytes of echo packet payload. The range is 12-1500, and the default value is 12.
<timeout>	Integer type. Maximum time in ms to wait for an echo reply response. The range is 10-600000, and the default value is 10000.
<ttl>	Integer type. TTL received in the response packet.
<rtt>	Integer type. Elapsed time in msec from packet sent to response received.
<err>	Integer type. An integer value to provide some information on why the PING request failed. 1 No response from remote host within timeout period 2 Failed to send ping request

4.15. AT+NBAND Set Supported Bands

The command sets the supported bands.

AT+NBAND Set Supported Bands	
Write Command **AT+NBAND=<n>[,<n>[,<n>[…]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NBAND?**	Response **+NBAND:<n>[,<n>[,<n>[…]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Test Command **AT+NBAND=?**	Response Returns the list of bands supported by the hardware. **+NBAND:(**list of supported **<n>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. Band in a decimal number.

NOTE

AT+NBAND=<n> must be executed when the radio is inactive (**AT+CFUN=0** will force the module to enter this state).

Example

```
AT+NBAND?
+NBAND:5,8,3,28,20,1

OK
```

4.16. AT+NLOGLEVEL Set Debug Logging Level

The command sets the logging level. Logging level is not persistent. This value will not be reset after reboot.

AT+NLOGLEVEL Set Debug Logging Level	
Write Command **AT+NLOGLEVEL=<core>,<level>**	Response **OK** If there is any error: **ERROR** Or

	+CME ERROR: <err>
Read Command **AT+NLOGLEVEL?**	Response **+NLOGLEVEL:<core>,<level>** **[…]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NLOGLEVEL=?**	Response **+NLOGLEVEL:(**list of supported **<core>**s**),(**list of supported **<level>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<core>	String type. Core required. PROTOCOL APPLICATION SECURITY
<level>	String type. Logging level required VERBOSE NORMAL WARNING ERROR NONE

Example

```
AT+NLOGLEVEL?
+NLOGLEVEL:SECURITY,NORMAL

+NLOGLEVEL:PROTOCOL,NORMAL
```

+NLOGLEVEL:APPLICATION,NORMAL

OK

4.17. AT+NCONFIG Configure UE Behaviour

The 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	
Write Command **AT+NCONFIG=<function>,<value>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NCONFIG?**	Response **+NCONFIG:<function>,<value>** **[+NCONFIG:<function>,<value>]** **[…]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NCONFIG=?**	Response **+NCONFIG:(<function>,(<value1>,<value2>[,<value3>[,…]]))** **[+NCONFIG:(<function>,(<value1>,<value2>[,<value3>[,…]]))]** **[…]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Maximum Response Time	300ms

Parameter

<function>	UE function to be configured	
	AUTOCONNECT	Control whether the module 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. And it will use the APN provided by the network.
	COMBINE_ATTACH	Enable/disable combine attach
	CELL_RESELECTION	Enable 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.
	BARRING_RELEASE_DELAY	Time (in seconds) to delay release from barring. Range: 0-1800.
	RELEASE_VERSION	Release version. Only supported in release 13 and 14. Could be set to 13 only when UE does not use power class 6.
	RPM	Enable/disable RPM. If RPM files present on USIM, the USIM settings will have precedence.
	SYNC_TIME_PERIOD	The synchronization time period from eNB. Unit: minute. Range: 0-65535. 0 means to close time synchronization.
	IPV6_GET_PREFIX_TIME	Set the maximum time of getting IPv6 prefix. Unit: second. Range: 0-65535. 0 means to get IPv6 prefix immediately, and 65535 means not to get IPv6 prefix.
	NB_CATEGORY	Configure the NB-IoT category. Only support 1 and 2 now.
	RAI	Enable/Disable RAI.
	HEAD_COMPRESS	Enable/Disable head compress.
	RLF_UPDATE	"FALSE" represents the occurrence of radio link failure, and NAS will TAU immediately. And "TRUE" representative when occurred radio link failure, ACK is expected before TAU.
	CONNECTION_REESTABLISHMENT	Enable/Disable connection re-establishment.
	TWO_HARQ	Enable/Disable two HARQ.
	PCO_IE_TYPE	PCO IE type. Support "PCO" and "EPCO"
	T3324_T3412_EXT_CHANGE_REPORT	

		Enable/Disable display unsolicited AT result for **+CEREG=5** when NAS timer for **<Active-Time>** and/or **<Periodic-TAU>** is changed.
	NON_IP_NO_SMS_ENABLE	"TRUE" represents when **<PDP_type>** is NONIP, will not support SMS.
	SUPPORT_SMS	Enable/Disable SMS.
	HPPLMN_SEARCH_ENABLE	Enable/Disable HPPLMN search.
<value>	Bool type.	
	TRUE	
	FALSE	

NOTES

1. "CR_" function is temporary and will be discarded when no longer required.
2. The following can only be set when the radio is inactive (**AT+CFUN=0** will force the module to enter this state):
   "MULTITONE", "NAS_SIM_POWER_SAVING_ENABLE", "RELEASE_VERSION", "SYNC_TIME_PERIOD", "NB_CATEGORY", "RAI", "HEAD_COMPRESS", "TWO_HARQ", "PCO_IE_TYPE", "HPPLMN_SEARCH_ENABLE" and "SUPPORT_SMS"

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:RPM,FALSE
+NCONFIG:SYNC_TIME_PERIOD,0
+NCONFIG:IPV6_GET_PREFIX_TIME,15
+NCONFIG:NB_CATEGORY,2
+NCONFIG:RAI, TRUE
+NCONFIG:HEAD_COMPRESS,FALSE
+NCONFIG:RLF_UPDATE,TRUE
+NCONFIG:CONNECTION_REESTABLISHMENT, TRUE
+NCONFIG:TWO_HARQ, TRUE
+NCONFIG:PCO_IE_TYPE,EPCO
+NCONFIG:T3324_T3412_EXT_CHANGE_REPORT,FALSE
```

```
+NCONFIG:NON_IP_NO_SMS_ENABLE,FALSE
+NCONFIG:SUPPORT_SMS, FALSE
+NCONFIG:HPPLMN_SEARCH_ENABLE,TRUE

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,(FALSE,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:(RPM,(FALSE,TRUE))
+NCONFIG:(SYNC_TIME_PERIOD,(0-65535))
+NCONFIG:(IPV6_GET_PREFIX_TIME,(0-65535))
+NCONFIG:(NB_CATEGORY,(1,2))
+NCONFIG:(RAI,(FALSE,TRUE))
+NCONFIG:(HEAD_COMPRESS,(FALSE,TRUE))
+NCONFIG:(RLF_UPDATE,(FALSE,TRUE))
+NCONFIG:(CONNECTION_REESTABLISHMENT,(FALSE,TRUE))
+NCONFIG:(TWO_HARQ,(FALSE,TRUE))
+NCONFIG:(PCO_IE_TYPE,(PCO,EPCO))
+NCONFIG:(T3324_T3412_EXT_CHANGE_REPORT,(FALSE,TRUE))
+NCONFIG:(NON_IP_NO_SMS_ENABLE,(FALSE,TRUE))
+NCONFIG:(SUPPORT_SMS,(FALSE,TRUE))
+NCONFIG:(HPPLMN_SEARCH_ENABLE,(FALSE,TRUE))

OK
```

4.18. AT+NATSPEED Configure Baud Rate of UART Port

The command is used to configure baud rate of UART port.

AT+NATSPEED Configure Baud Rate of UART Port	
Write Command **AT+NATSPEED=<baud_rate>,<timeout>,<store>,<sync_mode>[,<stopbits>[,<parity>[,<xonxoff>]]]**	Response **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Read Command **AT+NATSPEED?**	Response **+NATSPEED:<baud_rate>,<sync_mode>,<stopbits>,<parity>[,<xonxoff>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NATSPEED=?**	Response **+NATSPEED:(**list of supported **<baud_rate>**s**),(**range of supported **<timeout>**s**),(**list of supported **<store>**s**),(**range of supported **<sync_mode>**s**),(**list of supported **<stopbits>**s**),(**range of supported **<parity>**s**),(**list of supported **<xonxoff>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<baud_rate>	Integer type indicating the baud rate of UART port that has been requested. If the configured **<baud_rate>** value is higher than the fastest speed supported by the low power UART, the low power operation of Deep Sleep mode will be disabled, and the URC **+NATSPEED:DISABLE_DEEP_SLEEP** will be returned. For the supported baud rate range, please refer to the test command (**AT+NATSPEED=?**).
<timeout>	Integer type indicating the time to wait for communication before switching back to the original speed. Unit: second. Default value: 3. Maximum value: 30. 0 means using default value.
<store>	Integer type indicating whether to store **<baud_rate>** and **<sync_mode>** to NVM or not. 0 Do not store to NVM, need to configure again after reboot 1 Store to NVM, while there is interaction before timeout
<sync_mode>	Integer type. The LP UART synchronizes to each start bit that it detects and uses

	this to configure its optimum sampling point for each subsequent bit in a data word. The Sync Mode field allows this sampling point to be modified if required. **<sync mode>** do not support 3 when **<baud rate>** is 2400bps, 4800bps and 57600bps. This parameter may be removed in a future release. 0 Sample normally 1 Sample later 2 Sample earlier 3 Sample even earlier
<stopbits>	Integer type. The LP UART stop bits. 1 1 stop bit 2 2 stop bits
<parity>	Integer type. The AT UART parity. 0 No parity enabled 1 Odd parity 2 Even parity
<xonxoff>	AT UART Software (XON/XOFF) Flow Control 0 Software flow control disabled 1 Software flow control enabled

NOTE

When modifying **<baud_rate>** parameter, the new baud rate should be used to communicate with the module before the **<timeout>** timeout.

Example

```
AT+NATSPEED=9600,3,1,2,1
OK
AT+NATSPEED?
+NATSPEED:9600,2,1,0,0

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

OK
```

4.19. 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	
Execution Command **AT+NCCID**	Response **+NCCID:<ICCID>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NCCID?**	Response **+NCCID:<ICCID>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NCCID=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<ICCID>	USIM Card Identification Number

Example

```
AT+NCCID
+NCCID:44123456789012345678

OK
AT+NCCID?
+NCCID:44123456789012345678

OK
```

4.20. AT+NFWUPD Firmware Update via UART

This command supports firmware updating. It allows package zone erasing, package downloading, package's last validation result checking, package name and version querying, and firmware upgrading.

Before upgrading firmware, it needs to erase package zone and download package first. After downloading finished, it sends firmware upgrading command. Then the system will reboot and validate the package, and if the package is legal, it will start to upgrade the firmware to a new version, otherwise it will reboot and not to upgrade, then the user can check the package validation error information with corresponding AT command.

AT+NFWUPD Firmware Update via UART	
Write Command **AT+NFWUPD=<cmd>[,<sn>,<len>,<data>,<crc>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NFWUPD=?**	Response **+NFWUPD:(**range of supported **<cmd>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cmd>	Integer type. Package process command	
	0	Erase package zone in internal flash
	1	**<sn>,<len>,<data>,<crc>**. Download a package segment. Download a DFOTA package. Package segments are continuous segments of the DFOTA package. Segments can be of any length, but must be provided in order.
	2	Get the package validation result of last updating
	3	Get the package name
	4	Get the package version
	5	Upgrade firmware

<sn>	Integer type. Sequence number. It starts with 0, and increments by one for each package segment.
<len>	Integer type. Data length in bytes of data.
<data>	Data to be transmitted in hex string format.
<crc>	Integer type. An XOR8 of each byte in the package segment. It is sent as hex string.

NOTE

<cmd> values **3** and **4** are supported in R01(B300) and later baseline versions.

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

This command sets the mapping for band and power class. The read command lists all mapping of bands and power classes.

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

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<band>	Integer type. Band as key to map
<power class>	Integer type. Power class value for band, only 3, 5 and 6 are supported currently. The parameter can be set only when the radio is inactive (**AT+CFUN=0** will force the module to enter this state). Power class 3 5 6 dBm 23 20 14

NOTE

<power class> can be set to 6 only when **<function>** "RELEASE_VERSION" in **AT+NCONFIG** is 14.

4.22. AT+NPSMR Power Saving Mode Status Report

The write command controls the presentation of a URC **+NPSMR**. If **<n>=1**, **+NPSMR:<mode>** is sent from the UE when the power mode of the UE is changed.

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

AT+NPSMR Power Saving Mode Status Report	
Write Command **AT+NPSMR=<n>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NPSMR?**	Response **+NPSMR:<n>[,<mode>]** **OK**

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NPSMR=?**	Response **+NPSMR:(**list of supported **<n>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. Enable/disable URC. 0 Disable the URC 1 Enable the URC **+NPSMR:<mode>**
<mode>	Integer type. Indicates the power mode of UE. 0 Normal mode 1 Power saving mode

4.23. AT+NPTWEDRXS Paging Time Window Value and eDRX Setting

The write command controls the setting of the UE paging time window value and eDRX parameters. The command controls whether the UE wants to apply paging time window and eDRX or not, as well as the requested paging time window and eDRX value for each specified type of access technology.

A special form of the command can be given as **AT+NPTWEDRXS=3**. In this form, paging time window and eDRX will be disabled and data for all parameters in the command will be removed.

The read command returns the current settings for each defined value of **<AcT-type>**. The test command returns the supported **<mode>**s and the value ranges for the access technology and the requested paging time window and requested eDRX value as compound values.

AT+NPTWEDRXS Paging Time Window Value and EDRX Setting	
Write Command **AT+NPTWEDRXS=<mode>,<Act-t**	Response **OK**

ype>[,<Requested_Paging_time_window>[,<Requested_eDRX_value>]]	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NPTWEDRXS?**	Response **+NPTWEDRXS:<Act-type>,<Requested_Paging_time_window>,<Requested_eDRX_value>[,<NW_provided_eDRX_value>[,<Paging_time_window>]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NPTWEDRXS=?**	Response **+NPTWEDRXS:(**list of supported **<mode>**s**),(**list of supported **<Act-type>**s**),(**range of supported **<Requested_Paging_time_window>**s**),(**range of supported **<Requested_eDRX_value>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<mode>	Integer type. Indicates to disable or enable the use of requested paging time window and eDRX in the UE. This parameter is applicable to all specified types of access technology, i.e. the most recent setting of **<mode>** will take effect for all specified values of **<AcT>**. When set **<mode>** to 0 and do not have parameter **<Requested_Paging_time_window>** or **<Requested_eDRX_value>**, it will set omit value to invalid value like 0x00. 0 Disable the use of requested paging time window and eDRX 1 Enable the use of requested paging time window and eDRX 2 Enable the use of requested paging time window and eDRX, and enable the URC: **+NPTWEDRXP:<AcT-type>[,<Requested_Paging_time_window>[,<Requ**

ested_eDRX_value>[,<NW_provided_eDRX_value>[,<Paging_time_window>]]]]

3 Disable the use of eDRX and discard all parameters for eDRX.

<Act-type> Integer type. Indicates the type of access technology. This parameter is used to specify the relationship between the type of access technology and the requested eDRX value.

5 E-UTRAN (NB-S1 mode)

<Requested_Paging_time_window> String type. 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

<Requested_eDRX_value> String type. 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. 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. 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

4.24. AT+NPIN PIN Operator

The write command is used to verify, change, enable, disable, or unblock PIN. An unsolicited result will be received when execute pin command is completed.

USIM PIN, USIM PUK refer to the PIN of the selected application on the UICC. For example, in an E-UTRAN context, the selected application on the currently selected UICC should be a USIM and the USIM PIN then represents the PIN of the selected USIM. See *3GPP TS 31.101 [65]* for further details on application selection on the UICC.

AT+NPIN PIN Operator	
Write Command **AT+NPIN=<command>,<parameter1>[,<parameter2>]**	Response **OK** **+NPIN:<npin result>**

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<command>	Integer type. PIN operation mode. **<command>** **<parameter1>** **<parameter2>** PIN Operation Mode 0 **<pin>** Verify PIN 1 **<old pin>** **<new pin>** Change PIN 2 **<pin>** Enable PIN 3 **<pin>** Disable PIN 4 **<puk>** **<pin>** Unblock PIN
<parameter1>	A parameter varies with the setting of **<command>**. When **<command>=0**, **2**, and **3**, it is **<pin>**. When **<command>=1**, it is **<old pin>**. When **<command>=4**, it is **<puk>**.
<parameter2>	A parameter varies with the setting of **<command>**. When **<command>=0**, **2**, and **3**, it is omitted. When **<command>=1**, it is **<new pin>**. When **<command>=4**, it is **<pin>**.
<pin>,<old pin>,<new pin>	Decimal string format. **<old pin>** shall be the same as password specified for the facility from the MT user interface or with command **AT+NPIN=1,<old pin>,<new pin>**. **<new pin>** is the new password. The minimum length is 4, and the maximum length is 8. Unit: byte.
<puk>	Decimal string format. The string size is 8.
<npin result>	String type. Result. "OK" "ERROR PIN disabled" "ERROR PIN blocked" "ERROR wrong PIN **<pin retries remaining>**" "ERROR wrong format" "ERROR"
<pin retries remaining>	Decimal type. Remained PIN retries. If no retries left, then PIN is blocked. **AT+NPIN=4,<puk>,<pin>** needs to be entered before the PIN can be used again.

Example

```
AT+NPIN=0,29102394
```

```
OK

+NPIN:OK
```

4.25. AT+NCSEARFCN Clear Stored EARFCN

This command is used to clear stored EARFCN.

AT+NCSEARFCN Clear Stored EARFCN	
Execution Command **AT+NCSEARFCN**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

NOTE

This command must be executed when the radio is inactive (**AT+CFUN=0** will force the module to enter into this state).

Example

```
AT+NCSEARFCN
OK
```

4.26. AT+NIPINFO IP Address Information Report

This command is used to report 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>**.

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

AT+NIPINFO IP Address Information Report	
Write Command **AT+NIPINFO=<n>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NIPINFO?**	Response **+NIPINFO:<n>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NIPINFO=?**	Response **+NIPINFO:(**list of supported **<n>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. 0 Disable the URC 1 Enable the URC: **+NIPINFO:<cid>,<IP_type>,[<IP_addr>][,<failure_cause>]**
<cid>	Integer type. It specifies a particular PDP context definition (see **AT+CGDCONT**).
<IP_type>	Integer type. It specifies the type of IP address. IP Internet Protocol (*IETF STD 5 [103]*) IPV6 Internet Protocol, version 6 IPV4V6 Virtual **<PDP_type>** introduced to handle dual IP stack UE capability.
<IP_addr>	String type. IP address. The IPv4 address is expressed in decimal numbers and dots, such as 100.1.0.26. The IPv6 address is expressed in hexadecimal numbers and colon, such as 108F: 0: 0: 0: 8: 800: 200C: 417A. This parameter is not displayed

	when the address acquisition failed.	
<failure_cause>	Integer type. Causes for the failure to get an IP address.	
	1	PDN type IPv4 only allowed
	2	PDN type IPv6 only allowed
	3	Single address bearers only allowed
	4	IPv6 RA timeout
	5	Unspecified

Example

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

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

OK
```

4.27. 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	
Write Command **AT+NCPCDPR=<parameter>,<state>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NCPCDPR?**	Response **+NCPCDPR:<parameter>,<state>** **[+NCPCDPR:<parameter>,<state>]** **[...]** **OK**

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NCPCDPR=?**	Response **+NCPCDPR:(**list of supported **<parameter>**s**),(**list of supported **<state>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<parameter>	Unsigned char format. Dynamic parameters to be read. 0 Get DNS Server IPv4 Address request 1 Get DNS Server IPv6 Address request
<state>	Integer type. Read state of PDP context dynamic parameters. 0 Disable to Read 1 Enable to Read

NOTE

The command can be set only when the radio is inactive (**AT+CFUN=0** will force the module to enter this state).

Example

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

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

```
OK
```

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

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

AT+NQPODCP Query Pending Originating Data List via the Control Plane	
Execution Command **AT+NQPODCP=<cid>**	Response **[+NQPODCP:[<sequence>[<sequence>…]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NQPODCP=?**	Response **+NQPODCP:(**range of supported **<cid>s)** **OK**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. Specifies a particular PDP context definition (see the **AT+CGDCONT** command).
<sequence>	Integer type. The sequence number of pending originating message. Range: 1-255.

NOTE

<cid> values of 0-10 are supported.

Example

```
AT+NQPODCP=1
+NQPODCP:1,2,3

OK
```

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

OK
```

4.29. AT+NITZ Set Time Update Mode

The write command sets time update mode.

The read command returns the current time update mode in the UE.

If setting fails due to a UE error, **+CME ERROR: <err>** will be returned.

AT+NITZ Set Time Update Mode	
Write Command **AT+NITZ=<mode>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NITZ?**	Response **+NITZ:<mode>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NITZ=?**	Response **+NITZ:(**list of supported **<mode>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<mode>	Integer type value indicating time update mode.	
	0	Update time via local.
	1	Update time via NITZ.

Example

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

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

OK
```

4.30. AT+QLEDMODE Set NETLIGHT Function Mode

The write command is used to set the function mode of NETLIGHT.

AT+QLEDMODE Set NETLIGHT Function Mode	
Write Command **AT+QLEDMODE=<ledmode>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QLEDMODE?**	Response **+QLEDMODE:<ledmode>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+QLEDMODE=?**	Response **+QLEDMODE:**(list of supported **<ledmode>**s)

	OK If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<ledmode>	Integer type. Function mode of NETLIGHT LED. 0 Disable NETLIGHT LED function, and the NETLIGHT pin outputs a low level. 1 Enable NETLIGHT LED function, and the NETLIGHT pin outputs PWM signal, different high-level duration and low-level duration of which indicate different network status. High-level duration / Low-level duration / Network Status 64ms / 800ms / Network searching 64ms / 2000ms / Connected status		

NOTE

If NETLIGHT LED is enabled, it may cause the module to consume more power.

Example

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

4.31. AT+QIDNSCFG Configure DNS Server Address

The command is used to configure and query the DNS server address.

The read command returns the currently used DNS server address.

AT+QIDNSCFG Configure DNS Server Address	
Write Command **AT+QIDNSCFG=<pri_dns>[,<sec_dns>]**	Response **OK**

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QIDNSCFG?**	Response **PrimaryDns: <pri_dns>** **SecondaryDns: <sec_dns>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+QIDNSCFG=?**	Response **OK**
Maximum Response Time	300ms

Parameter

<pri_dns>	String parameter. Indicates the primary domain name server address.
<sec_dns>	String parameter. Indicates the second domain name server address.

NOTES

1. It is not allowed to configure the DNS server address during the execution of the **AT+QDNS** command.
2. If the DNS server address is not configured, the DNS server address delivered by the base station will be used by default.

Example

```
AT+QIDNSCFG=?
OK

AT+QIDNSCFG?
PrimaryDns: 218.4.4.4
SecondaryDns: 208.67.222.222

OK

AT+QIDNSCFG=8.8.8.8,8.8.4.4
```

```
OK

AT+QIDNSCFG?
PrimaryDns: 8.8.8.8
SecondaryDns: 8.8.4.4

OK
```

4.32. AT+QDNS Trigger DNS Domain Name Resolution

This command is used to trigger 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**.

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

Parameter

<mode>	Integer type.	
	0	DNS domain name resolution, hostname cannot be NULL.
	1	If the hostname is not NULL, then clear the hostname parsing data in memory and KV. If the hostname is NULL, all parsing data will be cleared.
	2	DNS domain name resolution, the host name cannot be NULL. The module does not cache parsing results.
<hostname>	Domain name.	
<result>	The result of domain name resolution.	
	IP	Domain name resolution completes the corresponding IP address.
	FAIL	Domain name resolution failed

Example

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

+QDNS:111.13.100.91
```

4.33. AT+QCFG Extended Configuration Settings

The command is used to query and configure various settings of UE.

AT+QCFG Extended Configuration Settings	
Test Command **AT+QCFG=?**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QCFG?**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

4.33.1. AT+QCFG="LWM2M/Lifetime" Configure OceanConnect IoT Platform Registration Lifetime

The command is used to configure lifetime value of OceanConnect IoT platform. The UE's update registration time is approximately 0.9*lifetime.

AT+QCFG="LWM2M/Lifetime" Configure OceanConnect IoT Platform Registration Lifetime	
Write Command **AT+QCFG="LWM2M/Lifetime"[,<lifetime>]**	Response If **<lifetime>** is present, configure the lifetime value of OceanConnect IoT Platform: **OK**

	If **<lifetime>** is omitted, query the lifetime value of OceanConnect IoT Platform: **+QCFG: "LWM2M/Lifetime",<lifetime>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<lifetime>	Integer type. Configure the lifetime of the LwM2M protocol. Range: 0-30x86400. Unit: second. The default lifetime value is 86400.	
	0	Disable lifetime function. UE will not send update registration packet to the OceanConnect IoT platform at regular intervals.
	1-30x86400	Enable lifetime function. The minimum effective value of **<lifetime>** is 900 even if the UE sets the lifetime value to 1-899.

Example

```
AT+QCFG="LWM2M/lifetime"
+QCFG: "LWM2M/Lifetime",0

OK
AT+QCFG="LWM2M/lifetime",900
OK
```

4.33.2. AT+QCFG="LWM2M/EndpointName" Configure the EndpointName When Accessing IoT Platform

This command is used to configure the EndpointName when accessing the IoT platform. If it is not configured, the IMEI number is used as the default value.

AT+QCFG="LWM2M/EndpointName" Configure the EndpointName When Accessing IoT Platform	
Write Command **AT+QCFG="LWM2M/Lifetime"[,<endpointname>]**	Response If **<endpointname>** is specified, configure the EndpointName of the IoT platform: **OK**

	If **<endpointname>** is omitted, query the access IoT platform EndpointName: **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<endpointname>	String type. Identifies the LwM2M Client on one LwM2M Server (including LwM2M Bootstrap-Server). Maximum number of **<endpointname>** string is 255 characters.

Example

```
AT+QCFG="LWM2M/EndpointName","1234567890"
OK

AT+QCFG="LWM2M/EndpointName"
+QCFG: "LWM2M/EndpointName","1234567890"

OK
```

4.33.3. AT+QCFG="LWM2M/BindingMode" Configure Binding Mode When Accessing IoT Platform

This command is used to configure the binding mode when accessing the IoT platform. The default is UDP mode.

AT+QCFG="LWM2M/BindingMode" Configure Binding Mode When Accessing IoT Platform	
Write Command **AT+QCFG="LWM2M/BindingMode"[,<bindingmode>]**	Response If **<bindingmode>** is specified, configure the binding mode when accessing IoT platform: **OK** If **<bindingmode>** is omitted, query the binding mode when

	accessing IoT platform: **+QCFG: "LWM2M/BindingMode",<bindingmode>** **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<bindingmode>	Integer. Binding mode in the LwM2M protocol. The default binding mode is UDP. 1 UDP mode 2 UDP-queue mode

Example

```
AT+QCFG="LWM2M/ BindingMode "
+QCFG: "LWM2M/BindingMode",1

OK
AT+QCFG="LWM2M/BindingMode",2
OK
AT+QCFG="LWM2M/BindingMode"
+QCFG: "LWM2M/BindingMode",2

OK
```

4.34. AT+NSNPD Sending Non-IP Data

The write command is used by the TE to transmit Non-IP data over control plane or user plane to network via MT. Context identifier **<cid>** is used to link the data to particular context. This command optionally indicates that the application on the MT expects that the exchange of data will be completed with this uplink data transfer; or will be completed with the next received downlink data. This command also optionally indicates whether or not the data to be transmitted is an exception data.

This command causes transmission of an ESM DATA TRANSPORT message, as defined in *3GPP TS 24.301 [83]*.

AT+NSNPD Sending Non-IP Data	
Write Command **AT+NSNPD=<cid>,<non_ip_data_length>,<non_ip_data>[,<RAI>[,<type_of_user_data>[,<sequence>]]]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NSNPD=?**	Response **+NSNPD:(**range 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**),(**range of supported **<sequence>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. A numeric parameter which specifies a particular PDP context or EPS bearer context definition. The **<cid>** parameter is local to the TE-MT interface and identifies the PDP or EPS bearer contexts which have been setup via AT command (see the **AT+CGDCONT** command).
<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.
<non_ip_data>	String of octets. Contains the user data container contents (please refer to *3GPP TS 24.301 [83] subclause 9.9.4.24*). When there is no data to transmit, the **<non_ip_data>** shall be an empty string (**""**). This parameter shall not be subject to conventional character conversion as per **+CSCS**. The coding format of the user data container and the maximum length of **<non_ip_data>** are implementation specific.
<RAI>	Integer type. Indicates the value of the release assistance indication, please refer to *3GPP TS 24.301 [83] subclause 9.9.4.25*. 0 No information available. 1 The MT expects that exchange of data will be completed with the transmission of the ESM DATA TRANSPORT message.

	2 The MT expects that exchange of data will be completed with the receipt of an ESM DATA RANSPORT message.
<type_of_user_data>	Integer type. Indicates whether the user data that is transmitted is regular or exceptional. 0 Regular data 1 Exception data
<sequence>	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>	The status of datagram. 0 Error 1 Sent

NOTES

1. Maximum data length will be 1358 bytes when Non-IP is used, otherwise, it will be 0.
2. Only one message will be buffered at any one time.

Example

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

OK
```

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

The write command is used to enable and disable reporting of Non-IP data from the network to the MT that is transmitted via the control plane or user plane in downlink direction. If reporting is enabled, the MT returns the unsolicited result code **+NRNPDM:<cid>,<cpdata_length>,<cpdata>** when data is received from the network.

Read command returns the current settings.

AT+NRNPDM Set Reporting of Non-IP Data Mode	
Write Command **AT+NRNPDM=<reporting>**	Response **OK** If there is any error: **ERROR**

	Or **+CME ERROR: <err>**
Read Command **AT+NRNPDM?**	Response **+NRNPDM:<reporting>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NRNPDM=?**	**Response** **+NRNPDM:(**range of supported **<reporting>**s**),(**range of supported **<cid>**s**),(**maximum number of octets of user data indicated by **<non_ip_data_length>)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<reporting>	Integer type, controlling reporting of mobile terminated Non-IP data events 0 Disable reporting of MT Non-IP data 1 Enable reporting of MT Non-IP data by the unsolicited result code **+NRNPDM**
<cid>	Integer type. A numeric parameter which specifies a particular PDP context or EPS bearer context definition. The **<cid>** parameter is local to the TE-MT interface and identifies the PDP or EPS bearer contexts which have been setup via AT command (see **AT+CGDCONT** command).
<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.
<non_ip_data>	String of octets. Contains the user data container contents (please refer to *3GPP TS 24.301 [83] subclause 9.9.4.24*). When there is no data to transmit, the **<non_ip_data>** shall be an empty string (**""**). This parameter shall not be subject to conventional character conversion as per **+CSCS**. The coding format of the user data container and the

	maximum length of **<non_ip_data>** are implementation specific.

NOTES

1. Maximum data length will be 1358 bytes when Non-IP is used, otherwise, it will be 0.
2. Only one message will be buffered at any one time.
3. Downlink data will be discarded before the reporting is enabled.

Example

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

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

OK
```

4.36. 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	
Write Command **AT+NQPNPD=<cid>**	Response **[+NQPNPD:[<sequence>[<sequence>…]]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NQPNPD=?**	Response **+NQPNDP:(**range of supported **<cid>**s**)** **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. Specifies a particular PDP context definition (see the **AT+CGDCONT** command).
<sequence>	The sequence of pending sending Non-IP data message. Range 1-255.

Example

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

OK
```

NOTE

<cid> values of 0-10 are supported.

4.37. AT+NSOSTATUS Get Socket Status

The execution command returns a list of all supported socket status.

AT+NSOSTATUS Get Socket Status	
Write Command **AT+NSOSTATUS=<socket id>**	Response **+NSOSTATUS:<socket id>[<status>,[<backoff value>]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Execution Command **AT+NSOSTATUS**	Response **[+NSOSTATUS:<socket id>[<status>,[<backoff value>]]**

	[+NSOSTATUS:<socket id>[<status>,[<backoff value>]] **[...]** **+NSOSTATUS:(**list of supported **<socket id>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NSOSTATUS=?**	Response **+NSOSTATUS:(**list of supported **<socket id>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<socket id>	Socket number.
<status>	Integer type. Status. 0 Available 1 Not exist. This status is also returned if the socket is not bound to the correct IP address NW assigned 2 Flow control 3 Back off
<backoff value>	Remainder time of back off timer/T3396 timer (in seconds). Only shows when the status is back off (**<status>=3**).

NOTE

<socket id>=0 is used for CoAP, and will be created while booting up if CoAP is supported.

Example

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

```
OK
AT+NSOSTATUS
+NSOSTATUS:0,0
+NSOSTATUS:1,0
+NSOSTATUS:2,1
+NSOSTATUS:3,1
+NSOSTATUS:4,1
+NSOSTATUS:5,1
+NSOSTATUS:6,1

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

OK
```

4.38. AT+NCIDSTATUS Get <cid> Status

The execution command returns a list of all defined PDP context status.

AT+NCIDSTATUS Get <cid> Status	
Write Command **AT+NCIDSTATUS=<cid>**	Response **+NCIDSTATUS:<cid>[<status>,[<backoff value>]]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Execution Command **AT+NCIDSTATUS**	Response **[+NCIDSTATUS:<cid>[<status>,[<backoff value>]]** **[+NCIDSTATUS:<cid>[<status>,[<backoff value>]]** **[...]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Test Command **AT+NCIDSTATUS=?**	Response **+NCIDSTATUS:(**range of supported **<cid>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<cid>	Integer type. specifies a particular PDP context definition (see the **AT+CGDCONT** command).
<status>	Integer type. Status. 0 Available 1 Not exist. This status will also be returned if PDP context is inactive. 2 Flow control 3 Back off
<backoff value>	Remainder time of back off timer/T3396 timer (in seconds). Only shows when the status is back off (**<status>=3**).

Example

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

OK
AT+NCIDSTATUS
+NCIDSTATUS:0,0

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

OK
```

4.39. AT+NGACTR Report PDP Context Activation/Deactivation Result

The write command controls the presentation of an URC **+NGACTR**.

If **<n>=1**, **+NGACTR:<n>** is sent from the MT when the PDP context activation or deactivation completes.

AT+NGACTR Report PDP Context Activation/Deactivation Result	
Write Command **AT+NGACTR=<n>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Execution Command **AT+NGACTR?**	Response **+NGACTR:<n>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NGACTR=?**	Response **+NGACTR:(**list of supported **<n>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<n>	Integer type. 0 Disable unsolicited result code 1 Enable the URC **+NGACTR:<cid>,<state>,<result>**
<cid>	Integer type. Specifies a particular PDP context definition (see the **AT+CGDCONT** command).

<state>	Integer type. Indicates the state of PDP context activation. The default value is manufacturer specific. 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

Example

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

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

OK
```

4.40. AT+NUICC Set UICC Power Mode

This command is used to power on/off UICC. Used only when the configuration item NAS_SIM_POWER_SAVING_ENABLE of **+NCONFIG** is true, and is needed when **AT+CSIM** or **AT+CRSM** is used. It increases the power consumption when power on.

AT+NUICC Set UICC Power Mode	
Write Command **AT+NUICC=<mode>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Test Command **AT+NUICC=?**	Response **+NUICC:(**list of supported **<mode>**s**)** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<mode>	Integer type. UICC power mode. 0 Power off 1 Power on

Example

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

5 LwM2M IoT Platform Commands

This chapter describes the AT commands related to LwM2M IoT platforms.

5.1. AT+NCDP Configure and Query CDP Server Settings

The command is used to set and query the server IP address and port for the CDP server. It is used when there is a HiSilicon CDP or OceanConnect IoT platform acting as gateway to network server applications.

This command is available after the IMEI number has been set.

AT+NCDP Configure and Query CDP Server Settings	
Write Command **AT+NCDP=<ip_addr>[,<port>]**	Response Update the CDP server configuration from the supplied parameters. **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NCDP?**	Response Return the current CDP server IP address and port. **+NCDP:<ip_addr>,<port>** **OK** If CDP server is not set: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<ip_addr>	IoT platform address. Support IPv4, IPv6 and domain name. Maximum length does

	not exceed 256 bytes
<port>	Integer type. Unsigned integer. The range is 0-65535. If port 0 is provided, the default port (5683) will be used. If no port is specified, the previously set port will be used. If no port is specified, and no port was previously set, the default port will be used.

NOTES

1. The write command is available after the IMEI number has been set.
2. The changes will take effect before successful network attachment.

Example

```
AT+NCDP=192.168.5.1,5683
OK
AT+NCDP?
+NCDP:192.168.5.1,5683

OK
```

5.2. AT+QSECSWT Set Data Encryption Mode

This command is used to set the encryption mode of the data and standard DTLS session timeout renegotiation interval.

AT+QSECSWT Set Data Encryption Mode	
Write Command **AT+QSECSWT=<type>[,<NAT type>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QSECSWT?**	Response **+QSECSWT:<type>[,<NAT type>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Maximum Response Time	300ms

Parameter

<type>	Encryption mode type.	
	0	None encryption
	1	Standard DTLS encryption mode
	2	DTLS+ encryption mode
<NAT type>	NAT type that is only valid in standard DTLS encryption mode	
	0	NAT is enabled and the renegotiate time is 30s
	1	NAT and renegotiation are disabled

NOTE

The parameter **<NAT type>** is only valid in standard DTLS encryption mode (**<type>=1**).

Example

```
AT+QSECSWT=0
OK
AT+QSECSWT=1,1
OK
AT+QSECSWT?
+NSECSWT:1,1

OK
```

5.3. AT+QSETPSK Set PSK ID and PSK

This command is used to configure PSK ID and PSK. The command will take effect after rebooting the module.

AT+QSETPSK Set PSK ID and PSK	
Write Command **AT+QSETPSK=<pskid>,<psk>**	Response **OK** If there is any error: **ERROR** Or

	+CME ERROR:<err>
Read Command **AT+QSETPSK?**	Response **+QSETPSK: <pskid>,<psk>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	2s

Parameter

<pskid>	PSK index. The fixed length is 15 decimal digits, the value must be the same as the device IMEI. In addition, this parameter must also be set to the same value on the IoT platform. When **<pskid>** is set to 0, the IMEI is used as the PSK ID.
<psk>	Indicates the PSK. This parameter must be set to a 16-digit hexadecimal number. In addition, it must also be set to the same value on OceanConnect IoT platform.

Example

```
AT+QSETPSK= 867724030000225,E8E19CC16CE1F388E8E19CC16CE1F388
OK
AT+QSETPSK?
+QSETPSK: 867724030000225,***

OK
```

5.4. AT+QLWSREGIND Registration Control

The command is used to control module to launch registration, deregistration or update to the IoT platform. It will give an **<err>** code and description as an intermediate message if the message cannot be sent.

AT+QLWSREGIND Registration Control	
Write Command **AT+QLWSREGIND=<type>**	Response **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<type>	Integer type. Operation type. 0 Trigger register operation 1 Trigger deregister operation

Example

```
AT+QLWSREGIND=0
OK
```

5.5. AT+QLWULDATA Send Data

The command is used to send data to IoT platforms with LwM2M protocol. It will give an **<err>** code and description as an intermediate message if the message cannot be sent. Before the module registered to the IoT platform, executing the command will trigger registration operation and discard the data.

AT+QLWULDATA Send Data	
Write Command **AT+QLWULDATA=<length>,<data>[,<seq_num>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<length>	Integer type. Length of data sent.
<data>	Hexadecimal format string. Maximum length of data to be sent is 1024 bytes.
<seq_num>	Sequence number. Range: 0-255. If non-zero **<seq_num>** is used to send CoAP data and there is CON or NON CoAP data with the same **<seq_num>** which has not been sent completely, the data to be sent will be discarded and an error will be

returned.

Example

```
AT+QLWULDATA=3,AA34BB
OK
```

5.6. AT+QLWULDATAEX Send CON/NON Message

This command is used to send confirmable (CON) or non-confirmable (NON) message with the identifier of RAI (Release Assistant Indication) to IoT platforms with the LwM2M protocol. After sending CON data, the sending result will be automatically notified to the TE. TE can also use the command **AT+QLWULDATASTATUS?** to query the status of the CON data that has been sent. If sending CON data, it must acquire the state (fail/timeout/success/got reset message) of sending CON data before sending the next CON or NON data. If the module is not registered to NB-IoT platform, this command will get the module to initiate registration only, and the module will discard the sent data.

AT+QLWULDATAEX Send CON/NON Message	
Write Command **AT+QLWULDATAEX=<length>,<data>,<mode>[,<seq_num>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<length>	Integer type. Length of data sent.
<data>	Hexadecimal format string. Maximum length of data to be sent is 1024 bytes.
<mode>	Message sending mode. 0x0000 Send NON message 0x0001 Send NON message with RELEASE auxiliary instructions 0x0010 Send NON message with RELEASE_AFTER_REPLY auxiliary instructions 0x0100 Send CON message 0x0101 Send CON message with RELEASE_AFTER_REPLY auxiliary instructions
<seq_num>	Sequence number. Range: 0-255. If non-zero **<seq_num>** is used to send CoAP data and there is CON or NON CoAP data with the same **<seq_num>** which has not been sent completely, the data to be sent will be discarded and an error will be returned. If **<seq_num>** is 0, it will be ignored.

Example

```
AT+QLWULDATAEX=3,AA34BB,0x0001
OK
```

5.7. AT+QLWULDATASTATUS Query Delivery Status of CON Messages

This command queries the status of the sending CON data to NB-IoT platform. This command only queries the status of the CON data that has been sent.

When non-zero **<seq_num>** is used to send CON CoAP data, the response of this command is **+QLWULDATASTATUS: <status>[,<seq_num>]**. When there is no **<seq_num>** in the AT command to send CON CoAP data or the **<seq_num>** is zero, the response of this command will be **+QLWULDATASTATUS: <status>**.

AT+QLWULDATASTATUS Query Delivery Status of CON Messages	
Read Command **AT+QLWULDATASTATUS?**	Response **+QLWULDATASTATUS:<status>[,<seq_num>]** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<status>	Integer type. Status of CON data sending. 0 Have not been sent 1 Sent, waiting response of IoT platform 2 Sent failed 3 Timeout 4 Success 5 Got reset message
<seq_num>	Sequence number. Range: 0-255. If **<seq_num>** is 0, it will be omitted.

Example

```
AT+QLWULDATASTATUS?
```

```
+QLWULDATASTATUS:4

OK
```

5.8. AT+QLWFOTAIND Set DFOTA Mode

This command is used to set DFOTA to either of the following modes:

- Automatic mode: DFOTA-based download and upgrade are automatically completed without command instruction from the MCU.
- Controlled mode: the MCU uses this command to control whether to download or upgrade firmware.

In controlled mode, when URC **+QLWEVTIND:6** (i.e. received "put package URI" message) or URC **+QLWEVTIND:7** (i.e. received "update" message) has been received, the MCU uses this command to control whether to download or upgrade firmware.

AT+QLWFOTAIND Set DFOTA Mode	
Write Command **AT+QLWFOTAIND=<type>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<type>	Integer type. DFOTA mode.	
	0	The automatic DFOTA mode is used.
	1	The controlled DFOTA mode is used
	2	MCU notifies module to start downloading the version file.
	3	MCU notifies module to cancel version file download.
	4	MCU notifies module to start update.
	5	MCU notifies module to cancel update.

NOTE

When **<type>** is set to 0 or 1, it needs to be restarted to take effect, and it will be saved in the flash.
When **<type>** is set to 2, 3, 4, or 5, it will take effect immediately and will not be saved.

Example

```
AT+QLWFOTAIND=1
OK
```

5.9. AT+QREGSWT Set Registration Mode

The command is used to set registration mode after the module reboot.

- If **<type>** is set to 0, after being rebooted and attached to network, the module will send REGISTERNOTIFY message to the device, then the device triggers registration by command **AT+QLWSREGIND**.
- If **<type>** is set to 1, after being rebooted and attached to network, the module will trigger automatic registration.
- If **<type>** is set to 2, after being rebooted and attached to network, the module will not trigger registration.

AT+QREGSWT Set Registration Mode	
Write Command **AT+QREGSWT=<type>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QREGSWT?**	Response **+QREGSWT:<type>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<type>	Integer type. Registration mode.	
	0	Manual registration mode
	1	Automatic registration mode

2	Disable registration

NOTES

1. If there is no need for IoT platforms, the registration has to be disabled with **AT+QREGSWT=2**, which will take effect only after rebooting the UE with command **AT+NRB**, otherwise, it will detach the UE from network, which may cause failure of related services (e.g. TCP/UDP).
2. For the ONT version of single Band 8, the IoT platform function is disabled by default since the version R01(B300SP7). That is, 2 is the default value of **<type>**. For details, please refer to the Release Notes of corresponding versions.

Example

```
AT+QREGSWT=1
OK

AT+QREGSWT?
+QREGSWT:1

OK
```

5.10. AT+NMGS Send a Message

The command is used to send data to IoT platforms. It will give an **<err>** code and description as an intermediate message if the message cannot be sent. Before the module is registered to the IoT platform, executing the command will trigger registration operation and discard the data.

AT+NMGS Send a Message	
Write Command **AT+NMGS=<length>,<data>[,<seq_num>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<length>	Decimal length of message.
<data>	Data to be transmitted in hex string format. The maximum length of data to be sent is 1024 bytes.
<seq_num>	Sequence number. Range: 0-255. If non-zero **<seq_num>** is used to send CoAP data and there is NON CoAP data with the same **<seq_num>** which has not been sent completely, the data to be sent will be discarded and an error will be returned.

Example

```
AT+NMGS=3,AA11BB
OK
```

5.11. AT+NMGR Receive a Message

The command is used to receive a message from IoT platforms with LwM2M protocol.

The command returns the oldest buffered message and deletes it from the buffer. If there are no messages, then no command response will be given. If new message indications (**AT+NNMI**) are turned on, then received messages will not be available via this command.

AT+NMGR Receive a Message	
Execution Command **AT+NMGR**	Response **<length>,<data>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<length>	Decimal length of message. Unit: bytes. Maximum value: 1024.
<data>	Data received in hex string format. The maximum length of received data is 1024 bytes.

Example

```
AT+NMGR
5,48656C6C6F

OK
```

5.12. AT+NNMI New Message Indications

The command sets or gets new message indications that are sent. New message indications can be sent when a downstream message is received by the UE from IoT platforms.

When new message indications and messages (**AT+NNMI=1**) are enabled, all currently buffered messages will be returned in the format of **+NNMI:<length>,<data>**. For example: **+NNMI:5,48656C6C6F**.

If "indication only" (**AT+NNMI=2**) is turned on, each newly received message triggers an indication that a new datagram is waiting using the unsolicited informational response. The buffered messages can be collected using **AT+NMGR**. The format of response is **+NNMI**. The **<status>** will restore to the default value (1 by default) after reboot.

AT+NNMI New Message Indications	
Write Command **AT+NNMI=<status>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NNMI?**	Response **+NNMI:<status>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<status>	Integer type. Status of new message indication. 0 No indications 1 Indications and message 2 Indications only
<length>	Decimal length of message.
<data>	Data to be transmitted in hex string format.

Example

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

OK
```

5.13. AT+NSMI Sent Message Indications

The command sets or gets indications that are sent when an upstream message is sent to IoT platforms.

If sent message indications are turned on, the unsolicited informational response **+NSMI:<status>[,<seq_num>]** (e.g. **+NSMI:SENT**) will be issued when a new message is sent into NB-IoT stack.

Only when there is non-zero **<seq_num>** in the AT command to send data, the notification message sent by the module to the MCU contains **<seq_num>**, and the value of **<seq_num>** is the same as the value of **<seq_num>** carried by the AT command.

When there is non-zero **<seq_num>** in the AT command to send CoAP data and the CoAP data is sent to NB-IoT platform successfully, the value of **<status>** in notification message sent by the module to the MCU is SENT_TO_AIR_INTERFACE. When there is no **<seq_num>** or the value of the **<seq_num>** is zero in the AT command to send CoAP data and the CoAP data is sent to NB-IoT platform successfully, the value of **<status>** in notification message sent by the module to the MCU is SENT.

For CON data, SENT and SENT_TO_AIR_INTERFACE indicate the results of sending data to the NB-IoT platform.

For NON data, SENT indicates that data is sent to the C core. SENT_TO_AIR_INTERFACE indicates that data is sent to the air interface of the base station.

AT+NSMI Sent Message Indications	
Write Command **AT+NSMI=<indications>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+NSMI?**	Response **+NSMI:<indications>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<indications>	0 No indications 1 Indications will be sent
<status>	SENT SENT_TO_AIR_INTERFACE DISCARDED
<seq_num>	Sequence number. Range: 1-255. Indicates the non-zero **<seq_num>** carried in the AT command to send data to IoT platform.

Example

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

OK
```

5.14. AT+NQMGR Query the Status of Messages Received

The command queries the status of the downstream messages received from IoT platforms.

AT+NQMGR Query the Status of Messages Received	
Execution Command **AT+NQMGR**	Response **BUFFERED=<buffered>,RECEIVED=<received>,DROPPED=<dropped>** **OK** If there is any error: **ERROR** Or **+CME ERROR:<err>**
Maximum Response Time	300ms

Parameter

<buffered>	The number of messages waiting to be read in the downstream buffer.
<received>	The total number of messages received by the UE since UE boots.
<dropped>	The number of messages dropped by the UE since UE boots.

Example

```
AT+NQMGR
BUFFERED=0,RECEIVED=34,DROPPED=2

OK
```

5.15. AT+NQMGS Query the Status of Messages Sent

The command queries the status of the upstream messages sent to IoT platforms.

AT+NQMGS Query the Status of Messages Sent	
Execution Command **AT+NQMGS**	Response **PENDING=<pending>,SENT=<sent>,ERROR=<error>** **OK**

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<pending>	The number of messages waiting to be sent in the upstream buffer.
<sent>	The total number of uplink messages sent into the NB-IoT stack since UE boot.
<error>	The number of messages that could not be sent by the UE due to an error since UE boot.

NOTES

1. For CON CoAP data, the statistics are modified only after timeout, RST, sent_fail, or sent_success.
2. For NON CoAP data, if a non-zero **<seq_num>** is carried during data transmission, the statistics are modified only when the data is sent to the air interface. If the sent data does not carry **<seq_num>** or the **<seq_num>** is zero, the statistics are modified when the data is sent to the protocol core.

Example

```
AT+NQMGS
PENDING=1,SENT=34,ERROR=0

OK
```

5.16. AT+NMSTATUS Message Registration Status

The command reports the current registration status when connected to the CDP server. When LwM2M is in the status of "MO_DATA_ENABLED", the UE can send data.

AT+NMSTATUS Message Registration Status	
Test Command **AT+NMSTATUS=?**	Response (list of supported **<registration_status>**s) **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Read Command **AT+NMSTATUS?**	Response **+NMSTATUS:<registration_status>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<registration_status>	Current registration status. UNINITIALISED MISSING_CONFIG INIITIALISING INIITIALISED INIT_FAILED REGISTERING REGISTERED DEREGISTERED MO_DATA_ENABLED NO_UE_IP REJECTED_BY_SERVER TIMEOUT_AND_RETRYING REG_FAILED DEREG_FAILED

5.17. +QLWEVTIND LwM2M Event Report (Response Only)

The module reports the LwM2M event to the device.

+QLWEVTIND LwM2M Event Report (Response Only)	
URC Format **+QLWEVTIND:<type>**	

Parameter

<type>	0	Register completed
	1	Deregister completed
	2	Registration status updated
	3	Object 19/0/0 observe completed
	4	Bootstrap completed
	5	5/0/3 resource observe completed
	6	Notify the device to receive update package URL
	7	Notify the device download has been completed
	9	Cancel object 19/0/0 observe

5.18. AT+QRESETDTLS Reset DTLS Mode

The command is used to reset DTLS mode.

- If DTLS has completed the handshake or renegotiation, the DTLS state can be set to initialization state through this AT command, and the handshake process will run when the data is sent next time.
- If the current state of DTLS is shaking hands or during renegotiation, the command will return **OK** directly. The next data sending will not trigger a handshake.

AT+QRESETDTLS Reset DTLS Mode	
Execution Command **AT+QRESETDTLS**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Example

```
AT+QRESETDTLS
OK
```

5.19. AT+QDTLSSTAT Query the State of DTLS

This command is used to query the current link state of DTLS. In every 8 seconds of sending uplink data, if DTLS is not negotiated, then the DTLS state will not be able to be queried.

- If DTLS negotiation can be completed within 8 seconds, DTLS status can be queried after completion of negotiation.
- Whether it is register by the module power on or manually register by the AT command, the IoT platform registration process is triggered by LwM2M and will not block any AT command.

After DTLS negotiation or handshaking is completed, the module initiatively sends the **+QDTLSSTAT:0/3** message to the terminal.

AT+QDTLSSTAT Query the State of DTLS	
Read Command **AT+QDTLSSTAT?**	Response **+QDTLSSTAT:<type>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<type>	0	DTLS negotiation or handshake completed
	1	Not start negotiation or handshake
	2	In the negotiation or handshake state
	3	Negotiation or handshake failed

Example

```
AT+QDTLSSTAT?
+QDTLSSTAT:0

OK
```

5.20. AT+QBOOTSTRAPHOLDOFF Bootstrap Hold off Time

The command is used to set and query client hold off time (relevant only for bootstrap server). The default value is 600 seconds, and the maximum time is recommended not exceeding 3000 seconds.

- If the time is configured too long (>3000s), the terminal's bootstrap sequence time will be delayed, then affecting the power consumption of terminal access.
- If the time is configured too short (<600s), the time of factory bootstrap and server-initiated bootstrap will be too short, which will affect the success rate of factory bootstrap and server-initiated bootstrap. Therefore, if the module needs to execute the Server Initiated bootstrap process (without configuring bootstrap server and LwM2M server addresses), or the LwM2M server address has been configured, the client's hold off time value is not recommended to be less than 600s.

When the bootstrap server address is configured, the client's hold off time can be set as 0. In such case, the module will go directly into the client-initiated bootstrap, thus shortening the client-initiated bootstrap time.

- If the bootstrap server address is not configured, **AT+QBOOTSTRAPHOLDOFF=0** will return **ERROR**.
- If bootstrap server does not exist, the client's hold off time cannot be set to 0. If the client's hold off time is 0, the bootstrap server IP cannot be erased.

AT+QBOOTSTRAPHOLDOFF Bootstrap Hold off Time	
Write Command **AT+QBOOTSTRAPHOLDOFF=<ClientHoldOffTime>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QBOOTSTRAPHOLDOFF?**	Response **+QBOOTSTRAPHOLDOFF:<ClientHoldOffTime>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<ClientHoldOffTime>	Integer type. Client hold off time. Default set to 600 seconds. Recommended not exceeding 3000 seconds.

Example

```
AT+QBOOTSTRAPHOLDOFF=600
OK
AT+QBOOTSTRAPHOLDOFF?
+QBOOTSTRAPHOLDOFF:600

OK
```

5.21. AT+QLWSERVERIP Set/Delete Bootstrap/LwM2M Server IP

The command is used to set or delete the IP address of bootstrap/IoT server. The KV used in **AT+QLWSERVERIP** and **AT+NCDP** command is the same.

- **AT+QLWSERVERIP=DEL,<ip_addr>,<port>** will delete the IP information when the IP address and the port can match the KV.
- **AT+QLWSERVERIP=DEL,<ip_addr>** will delete the IP information when the IP address can match the KV.

AT+QLWSERVERIP Set/Delete Bootstrap/LwM2M Server IP	
Write Command **AT+QLWSERVERIP=<type>,<ip_addr>[,<port>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QLWSERVERIP?**	Response **+QLWSERVERIP: <type>,<ip_addr>,<port>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**

Maximum Response Time	300ms

Parameter

<type>	Bootstrap/LwM2M Server IP type. LWM2M IoT Server IP BS Bootstrap Server IP DEL Delete the IP information
<ip_addr>	Domain name or IP address. IP address format: decimal format. IPv4 address is a1.a2.a3.a4. The range is 0-255.
<port>	Integer type. Port number. The range is 0-65535. The default value is 5683.

Example

```
AT+QLWSERVERIP=BS,180.101.147.115,5683
OK
AT+QLWSERVERIP=LWM2M,180.101.147.115,5683
OK
AT+QLWSERVERIP?
+QLWSERVERIP:BS,180.101.147.115,5683
+QLWSERVERIP:LWM2M,180.101.147.115,5683

OK
AT+QLWSERVERIP=DEL,180.101.147.115,5683
OK
```

5.22. AT+QCRITICALDATA Enable Emergency Data Sending

The write command is used to enable emergency data sending in the DFOTA process, and if the sending of emergency data is enabled, the current DFOTA process will be stopped to give priority to data transmission. The upgrade result of DFOTA relies on the timeout mechanism.

In the process of DFOTA upgrade, if there is an request to send emergency data, this AT command can be sent first; if the response is **OK**, the emergency data will be sent, and if the response is **ERROR**, the emergency data will not be sent.

AT+QCRITICALDATA Enable Emergency Data Sending	
Write Command **AT+QCRITICALDATA=<state>**	Response **OK**

	If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<state>	Integer type. The value can only be set to 1 to enable emergency data sending.

Example

```
AT+QCRITICALDATA=1
OK
```

5.23. AT+QSETBSPSK Configure PSK ID and PSK to Bootstrap

This command is used to configure the PSK ID and PSK which are used to set DTLS connection with bootstrap Server. The command will take effect after rebooting the module

AT+QSETBSPSK Configure PSK ID and PSK to Bootstrap	
Write Command **AT+QSETBSPSK=<pskid>,<psk>**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QSETBSPSK?**	Response **+QSETBSPSK:<pskid>,<psk>** **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Maximum Response Time	2s

Parameter

<pskid>	PSK index. Fixed length of 15-bit decimal digits number for indexing PSK. The value needs to be the same with the IMEI number of the device, and it needs to be consistent with the IoT platform. When **<pskid>** is set to 0, the IMEI number is used as the PSK ID.
<psk>	Pre-shared Key. Fixed length of 16 bytes hexadecimal number, which need to be consistent with that of the IoT platform.

Example

```
AT+QSETBSPSK=201703230000024,0123456789ABCDEF0123456789ABCDEF
OK
AT+QSETBSPSK?
+QSETBSPSK:201703230000024,***

OK
```

5.24. AT+QBSSECSWT Configure Bootstrap Server Connection Switches

This command is used to configure DTLS connection switches with bootstrap Server and standard DTLS negotiation time. Start the timer when the DTLS negotiated at the first time, and restart the timer when received the DTLS message from bootstrap Server. When the uplink message is sent and the timer has expired, it is necessary to trigger a renegotiation.

AT+QBSSECSWT Configure Bootstrap Server Connection Switches	
Write Command **AT+QBSSECSWT=<type>[,<NAT type>]**	Response **OK** If there is any error: **ERROR** Or **+CME ERROR: <err>**
Read Command **AT+QBSSECSWT?**	Response **+QBSSECSWT:<type>[,<NAT type>]** **OK** If there is any error:

	ERROR Or **+CME ERROR: <err>**
Maximum Response Time	300ms

Parameter

<type>	Encryption mode type.	
	0	None encryption
	1	Standard DTLS encryption mode
	2	DTLS+ encryption mode
<NAT type>	NAT type that is only valid in standard DTLS encryption mode	
	0	NAT is enabled and the renegotiate time is 30s
	1	NAT and renegotiation are disabled

NOTE

The parameter **<NAT type>** is only valid in standard DTLS encryption mode (**<type>=1**).

Example

```
AT+QBSSECSWT=1,1
OK
AT+QBSSECSWT?
+QBSSECSWT:1,1

OK
```

6 Error Values

This chapter introduces the error values related to BC95-G and BC68 modules.

The error codes listed in the following two tables are compliant with the 3GPP specifications. Please refer to sub-clause 9.2 of *3GPP TS 27.007 V13.5.0* for all possible **<err>** values.

Table 2: General Errors (27.007)

Code of <err>	Description
3	Operation not allowed
4	Operation not supported
5	Need to enter 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 3: General Errors (27.005)

Code of <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 following error codes are specific ones for BC95-G and BC68 modules. Error code 512 was previously error code 256. Error code 513 was previously error code 257.

Table 4: Specific Error Codes

Code of <err>	Description
512	Required parameter not configured
513	TUP not registered
514	AT internal error
515	CID is active
516	Incorrect state for command
517	CID is invalid
518	CID is not active
520	Deactivate the last active CID
521	CID is not defined
522	UART parity error
523	UART frame error
524	UE is in minimal function mode (**AT+CFUN=0**)
525	AT command aborted: in processing
526	AT command aborted: error
527	Command interrupted
528	Configuration conflicts
529	During FOTA updating
530	Not the AT allocated socket
531	USIM PIN is blocked
532	USIM PUK is blocked
533	Not mipi module

534	File not found
535	Conditions of use not satisfied
536	AT UART buffer error
537	Back off timer is running

NOTE

AT+CMEE=<n> command disables (**<n>=0**) or enables (**<n>=1**) the use of final result code **+CME ERROR:<err>**. When **<n>=1**, a limited set of error codes will be returned.

7 Reboot Reasons

If the module is rebooted for any reason apart from either being power cycled or being externally reset, it will return a message before the **<CR><LF>Neul<CR><LF>OK<CR><LF>** message to indicate the reboot reason.

Table 5: Reboot Messages Displayed

No.	Reboot Message Displayed
1	REBOOT_CAUSE_SECURITY_RESET_UNKNOWN
2	REBOOT_CAUSE_SECURITY_SYSRESETREQ
3	REBOOT_CAUSE_SECURITY_WATCHDOG
4	REBOOT_CAUSE_SECURITY_SELF
5	REBOOT_CAUSE_SECURITY_ALTBOOT
6	REBOOT_CAUSE_SECURITY_REG_0
7	REBOOT_CAUSE_SECURITY_REG_3
8	REBOOT_CAUSE_SECURITY_STANDARD_CHIP_WATCHDOG
9	REBOOT_CAUSE_SECURITY_UPDATER_CHIP_WATCHDOG
10	REBOOT_CAUSE_SEUCRITY_SCAN_ENTER_EXIT
11	REBOOT_CAUSE_SECURITY_PMU_POWER_ON_RESET
12	REBOOT_CAUSE_SECURITY_RESET_PIN
13	REBOOT_CAUSE_SECURITY_REGIONS_UPDATED
14	REBOOT_CAUSE_SECURITY_FOTA_UPGRADE
15	REBOOT_CAUSE_PROTOCOL_SYSRESETREQ
16	REBOOT_CAUSE_PROTOCOL_WATCHDOG

17	REBOOT_CAUSE_PROTOCOL_MONITOR_REBOOT_REQ
18	REBOOT_CAUSE_PROTOCOL_RPC_TIMEOUT
19	REBOOT_CAUSE_APPLICATION_SYSRESETREQ
20	REBOOT_CAUSE_APPLICATION_WATCHDOG
21	REBOOT_CAUSE_APPLICATION_AT
22	REBOOT_CAUSE_APPLICATION_RPC_TIMEOUT
23	REBOOT_CAUSE_PROTOCOL_IMSI_UPDATE
24	REBOOT_CAUSE_UNKNOWN

8 Examples

8.1. Network Attachment

- Through **AT+NCONFIG=AUTOCONNECT,TRUE** command, the module supports automatic network attachment.
- If manual network attachment is desired, then please disable automatic network attachment through **AT+NCONFIG=AUTOCONNECT,FALSE**. The setting will be saved to NVM and take effect after rebooting the module by **AT+NRB**.

8.1.1. Automatic Network Attachment

The following shows a simple example for automatic network attachment. Users only need to query whether the module has attached on network by the following commands:

```
REBOOT_CAUSE_SECURITY_PMU_POWER_ON_RESET
Neul
OK                          //Power on the module

AT+CFUN?
+CFUN:1

OK
AT+CIMI                     //Query the IMSI number.
460111174590523

OK
AT+CEREG?                   //Query the network registration status: 1 means registered on network,
                            while 2 means searching the network.
+CEREG:0,1

OK
AT+CGATT?                   //Query whether the network has been activated.
+CGATT:1                    //The network has been activated successfully. Sometimes, there might
                            be a need to wait for 30s.
```

```
OK
AT+CGPADDR                    //Query the IP address of the module
+CGPADDR:0,10.169.241.248
OK
```

8.1.2. 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
REBOOTING

REBOOT_CAUSE_APPLICATION_AT     //Reboot cause
Neul
OK
AT+NBAND=5                      //Specify the band to be searched
OK
AT+CFUN=1                       //Set the UE into full functionality mode
OK
AT+CIMI                         //Query the IMSI number
460111174590523

OK
AT+CGATT=1                      //Trigger network attachment
OK
AT+NUESTATS                     //Query the module status
Signal power:-827
Total power:-684
TX power:230
TX time:2272
RX time:9354
Cell ID:110258001
ECL:1
SNR:-30
EARFCN:2506
PCI:100
RSRQ:-157
OPERATOR MODE:4

OK
```

```
AT+CEREG?                    //Query the network registration status: 1 means registered on network, while 2 means searching the network.
+CEREG:0,1

OK
AT+CGATT?                    //Query whether the network has been activated.
+CGATT:1                     //The network has been activated successfully. Sometimes, there might be a need to wait for 30s.

OK
AT+CGPADDR                   //Query the IP address of the module
+CGPADDR:0,10.3.42.109
OK
```

8.2. Send/Receive/Read a UDP Message

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

```
AT+QREGSWT=2                 //Disable registration function of IoT platform
OK
…                            //Connect to network

AT+CGPADDR                   //Query the IP address of the module
+CGPADDR:0,10.3.42.109
OK
AT+NSOCR=DGRAM,17,0,1        //Create a socket
1

OK
AT+NSOST=1,220.180.239.212,8012,5,1245783132,100    //Send a message
1,5

OK

+NSOSTR:1,100,1              //Datagram is sent by RF

+NSONMI:1,5                  //Received the message
AT+NSORF=1,5                 //Read the message
1,220.180.239.212,8012,5,1245783132,0
```

```
OK
AT+NSOCL=1                          //Close the socket
OK
```

8.3. Send/Receive/Read a TCP Message

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

```
AT+QREGSWT=2                        //Disable registration function of IoT platform
OK
…                                   //Attach on network

AT+CGPADDR                          //Query the IP address of the module
+CGPADDR:0,10.3.42.79
OK
AT+NSOCR=STREAM,6,0,1               //Create a socket
1

OK
AT+NSOCO=1,220.180.239.212,8009     //Connect to the server
OK
AT+NSOSD=1,4,01020304,0x100,101     //Send the messages
1,4

OK

+NSOSTR:1,101,1                     //Datagram has been confirmed to be received by the server

+NSONMI:1,4                         //Received the message
AT+NSORF=1,4                        //Read the message
1,220.180.239.212,8009,4,01020304,0

OK
AT+NSOCL=1                          //Close the socket
OK
```

8.4. Examples Relating to OceanConnect IoT Platform

8.4.1. Register to OceanConnect IoT Platform

8.4.1.1. Automatic Registration Mode

AT+QREGSWT?	
+QREGSWT:1	//Automatic registration mode (default mode)
OK	
AT+NCDP=180.101.147.115,5683	//Set IP address and port for the IoT platform
OK	
AT+NRB	//Reboot the module
REBOOTING	
REBOOT_CAUSE_APPLICATION_AT	//Reboot cause
Neul	
OK	
AT+CGPADDR	//Query the IP address of the module
+CGPADDR:0,10.3.42.109	
OK	
+QLWEVTIND:0	//Successful registration indication.
+QLWEVTIND:3	//IoT platform has observed the data object 19. When the module reports this message, the data can be sent to the IoT platform.

8.4.1.2. Manual Registration Mode

AT+CGATT?	//Query the PS service attachment status
+CGATT:1	//Attached to the PS service
OK	
AT+NCDP=180.101.147.115,5683	//Set IP address and port for the IoT platform
OK	
AT+QREGSWT?	//Query the registration mode
+QREGSWT:0	//Manual registration mode

```
OK
AT+QLWSREGIND=0                    //Start to register to the IoT platform
OK

+QLWEVTIND:0                       //Successful registration indication

+QLWEVTIND:3                       //IoT platform has observed the data object 19. When
                                   the module reports this message, the data can be sent
                                   to the IoT platform.
AT+QLWSREGIND=1                    //Deregister from the IoT platform
OK

+QLWEVTIND:1                       //Successful deregistration indication
```

8.4.2. Send/Receive Data with OceanConnect IoT Platform

```
+QLWEVTIND:0                       //Successful registration indication

+QLWEVTIND:3                       //IoT platform has observed the data object 19. When
                                   the module reports this message, the data can be sent
                                   to the IoT platform.
AT+QLWULDATA=3,313233              //Send data in NON mode. It does not need the IoT
                                   platform to ACK this data message.
OK

+NNMI:4,AAAA0000                   //The module has received data from the IoT platform
AT+QLWULDATAEX=3,313233,0X0100     //Send CON message. The IoT platform needs to
                                   ACK this data message.
OK

+QLWULDATASTATUS:4                 //Successful message sending

+NNMI:4,AAAA0000                   //The module has received data from the IoT platform
AT+QLWULDATASTATUS?                //Query the data sending status
+QLWULDATASTATUS:4                 //Successful message sending

OK
```

8.4.3. Upgrade Firmware on OceanConnect IoT Platform

8.4.3.1. Automatic Firmware Upgrade via DFOTA

```
//Create a firmware upgrade task on the IoT platform.

+QLWEVTIND:0                          //Successful registration indication

+QLWEVTIND:3                          //IoT platform has observed the data object 19

AT+QLWULDATA=3,313233                 //Send any packet of upstream data and trigger an online firmware upgrade

OK

+QLWEVTIND:5                          //IoT platform has observed the firmware object 5

FIRMWARE DOWNLOADING

FIRMWARE DOWNLOADED

FIRMWARE UPDATING

//The module is undergoing local updates, which can result in multiple reboots and thus will take some time.

REBOOT_CAUSE_SECURITY_FOTA_UPGRADE    //The module reboots
Neul
OK

FIRMWARE UPDATE SUCCESS

+QLWEVTIND:0                          //Successful registration indication

+QLWEVTIND:3                          //IoT platform has observed the data object 19

FIRMWARE UPDATE OVER
```

8.4.3.2. Manual Firmware Upgrade via DFOTA

```
//Create a firmware upgrade task on the IoT platform.

AT+QLWFOTAIND?                        //Query the firmware upgrade mode
+QLWFOTAIND:1                         //Manual mode
```

```
OK
AT+QLWSREGIND=0                          //Start to register to the IoT platform
OK

+QLWEVTIND:0                             //Successful registration indication

+QLWEVTIND:3                             //IoT platform has observed the data object 19

+QLWEVTIND:5                             //IoT platform has observed the firmware object 5

FIRMWARE DOWNLOADING

+QLWEVTIND:6                             //Notify the device to receive update package URL
AT+QLWFOTAIND=2                          //Device notifies the module to start downloading
OK


FIRMWARE DOWNLOADED

+QLWEVTIND:7                             //Notify the device that download has been completed
AT+QLWFOTAIND=4                          //Device notifies the module to start updating
OK

FIRMWARE UPDATING

REBOOT_CAUSE_SECURITY_FOTA_UPGRADE       //The module reboots
Neul
OK

FIRMWARE UPDATE SUCCESS

+QLWEVTIND:0                             //Successful registration indication

+QLWEVTIND:3                             //IoT platform has observed the data object 19

FIRMWARE UPDATE OVER
```

8.4.4. Register to OceanConnect IoT Platform with DTLS

```
AT+CGATT?                                //Query the PS service attach status
+CGATT:1                                 //Attached to the PS service

OK
```

```
AT+NCDP= 180.101.147.115,5684          //Set IoT platform IP address and port. The port is 5684.
OK
AT+QSECSWT=1                           //Encryption using standard DTLS
OK
AT+QSETPSK=201703230000024,0123456789ABCDEF0123456789ABCDEF
OK

AT+NRB                                 //Reboot the module
REBOOTING

REBOOT_CAUSE_APPLICATION_AT            //Reboot cause
Neul
OK
                                       //Set PSK ID and PSK
AT+QREGSWT?                            //Query the registration mode
+QREGSWT:0                             //Manual registration mode

OK
AT+QLWSREGIND=0                        //Start to register to the IoT platform
OK

+QLWEVTIND:0                           //Successful registration indication

+QLWEVTIND:3                           //IoT platform has observed the data object 19
```

8.5. 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 Plane CIoT EPS optimization, where UE may include RAI in non-access stratum (NAS) signaling to indicate that no further uplink or downlink data transmission and no further uplink data transmission subsequent to the uplink data transmission is expected, thus helping the network to decide if the connection can be released.

Release 14 introduces RAI for access stratum (AS) for both Control and User Plane CIoT EPS optimization. When AS RAI is configured, UE may trigger a buffer status report (BSR) with zero byte size, indicating to eNB that no further data is expected in UL or DL in the near future and the connection may be released.

8.5.1. RAI Function Operation Example for R13

8.5.1.1. Send UDP Data with Flag

Run the **AT+CSCON** command to enable the URC reporting function. Based on the URC reporting status, users can determine whether the flag is valid.

```
[13:38:08:105]AT+CSCON=1                                          //Enable URC reporting
[13:38:08:121]OK
[13:38:08:344]
[13:38:08:344]+CSCON:0
[13:38:24:470]AT+NSOCR=DGRAM,17,1234,1                            //Create UDP socket
[13:38:24:502]0

[13:38:24:518]OK
[13:38:28:717]AT+NSOSTF=0,220.180.239.212,8052,0x400,2,AB30       //Send data with 0x400 flag
[13:38:28:765]0,2

[13:38:28:765]OK
[13:38:29:340]
[13:38:29:340]+CSCON:1
[13:38:29:611]
[13:38:29:611]+NSONMI:0,2                                         //Received downlink data
[13:38:29:931]
[13:38:29:931]+CSCON:0                                            //Release RRC connection immediately
[13:38:37:244]AT+NSORF=0,512
[13:38:37:276]0,220.180.239.212,8052,2,AB30,0                     //Read downlink data

[13:38:37:307]OK
[13:38:56:275]AT+NSOSTF=0,220.180.239.212,8052,0x200,2,AB30       //Send data with 0x200 flag
[13:38:56:323]0,2

[13:38:56:339]OK
[13:38:58:782]
[13:38:58:782]+CSCON:1                                            //Establish RRC connection and start sending data
[13:38:59:086]
[13:38:59:086]+CSCON:0                                            //There is no downlink data, and release RRC
                                                                  connection immediately
```

8.5.1.2. Send LwM2M CON/NON Data

Run the **AT+CSCON** command to enable the URC reporting function. Based on the URC reporting status, users can determine whether the flag is valid.

```
[14:17:35:307]AT+CSCON=1                                    //Enable URC reporting
[14:17:35:322]OK
[14:17:45:876]
[14:17:45:876]+CSCON:0
[14:17:53:301]AT+NCDP?
[14:17:53:301]+NCDP:180.101.147.115,5683                    //IoT platform server address

[14:17:53:333]OK
[14:18:03:920]AT+QLWULDATAEX=3,AA34BB,0x0001                //Send NON data with 0x0001 flag
[14:18:05:404]OK
[14:18:05:532]
[14:18:05:532]+CSCON:1                                      //Establish RRC connection and start sending data
[14:18:05:947]
[14:18:05:947]+CSCON:0                                      //Release RRC connection immediately after data transmission

[14:21:06:880]AT+QLWULDATAEX=3,AA34BB,0x0101                //Send CON data with 0x0101 flag
[14:21:06:911]OK
[14:21:07:933]
[14:21:07:933]+CSCON:1                                      //Establish RRC connection and start sending data
[14:21:08:221]
[14:21:08:221]+QLWULDATASTATUS:4                            //CON data received by IoT platform
[14:21:09:179]
[14:21:09:179]+CSCON:0                                      //Release RRC connection immediately after data transmission
```

8.5.2. RAI Function Operation Example for R14

In the following examples, the RAI function of R14 has been enabled.

```
//Check whether RAI is enabled in R14
AT+NCONFIG?
……
+NCONFIG:RELEASE_VERSION,14        //Enable R14
+NCONFIG:RAI,TRUE                  //Enable RAI
……

OK
```

8.5.2.1. Send UDP data

Run the **AT+CSCON** command to enable the URC reporting function. Based on the URC reporting status, users can determine whether the flag is valid.

```
[13:56:40:879]AT+CSCON=1                                    //Enable URC reporting
[13:56:40:893]OK
[13:56:49:346]AT+NSOCR=DGRAM,17,5690,1                      //Create UDP socket
[13:56:49:359]0

[13:56:49:359]OK
[13:56:56:079]
[13:56:56:079]+CSCON:0                                      //RRC connection release
[13:56:59:217]AT+NSOST=0,220.180.239.212,8426,2,1122        //Send data
[13:56:59:259]0,2

[13:56:59:259]OK
[13:56:59:644]
[13:56:59:644]+CSCON:1                                      //Establish RRC connection and start sending data
[13:57:01:153]AT+CNMPSD                                     //Notify network of no data exchange
[13:57:01:162]OK
[13:57:01:365]
[13:57:01:365]+CSCON:0                                      //RRC connection is released immediately
```

8.5.2.2. Send LwM2M CON/NON Data

Run the **AT+CSCON** command to enable the URC reporting function. Based on the URC reporting status, users can determine whether the flag is valid.

```
//The module is connected to the IoT platform
[14:07:36:307]AT+CSCON=1                                    //Enable URC reporting
[14:07:36:322]OK
[14:07:36:876]
[14:07:41:273]AT+QLWULDATAEX=5,1122334455,0x0000,2          //Send NON data with 0x0000 flag
[14:07:41:309]OK
[14:07:41:720]
[14:07:42:866]+CSCON:1                                      //Establish RRC connection and start sending data
[14:07:44:296]AT+CNMPSD                                     //Notify network of no data exchange
[14:07:44:305]OK
[14:07:44:516]
[14:07:44:516]+CSCON:0                                      //RRC connection is released immediately
[14:08:41:954]AT+QLWULDATAEX=5,1122334455,0x0100            //Send CON data with 0x0100 flag
[14:08:41:979]OK
```

```
[14:08:42:392]
[14:08:43:725]+CSCON:1                    //Establish RRC connection and start sending data
[14:08:43:863]
[14:08:43:863]+QLWULDATASTATUS:4
[14:08:49:328]AT+CNMPSD                   //Notify network of no data exchange
[14:08:49:328]OK
[14:08:49:542]
[14:08:49:542]+CSCON:0                    //RRC connection is released immediately
```

9 Appendix A Reference

Table 6: Terms and Abbreviations

Abbreviation	Description
APDU	Application Protocol Data Unit
APN	Access Point Name
AS	Access Stratum
BIP	Bearer Independent Protocol
BS	Bootstrap
BSR	Buffer Status Report
CDP	Connected Device Platform
CHAP	Challenge-Handshake Authentication Protocol
CS	Circuit Switched
DCE	Data Communication Equipment
DFOTA	Delta Firmware Over-The-Air
DNS	Domain Name Server
DTLS	Datagram Transport Layer Security
EARFCN	E-UTRA Absolute Radio Frequency Channel Number
ECL	Enhanced Coverage Level
E-UTRAN	Evolved Universal Terrestrial Radio Access Network
eDRX	Extended Discontinuous Reception
EGPRS	Enhanced General Packet Radio Service

EHPLMN	Equivalent Home Public Land Mobile Network
eNB	Evolved Node B
EPS	Evolved Packet System
ESM	EPS Session Management
GBR	Guaranteed Bit Rate
GERAN	GSM/EDGE Radio Access Network
GGSN	Gateway GPRS Support Node
GMT	Greenwich Mean Time
GPRS	General Packet Radio Service
HPLMN	Home Public Land Mobile Network
HSDPA	High Speed Downlink Packet Access
HSUPA	High-Speed Uplink Packet Access
ICCID	Integrated Circuit Card Identifier
ICMP	Internet Control Messages Protocol
ID	Identifier
IMEI	International Mobile Equipment Identity
IMEISV	International Mobile Equipment Identity and Software Version
IP	Internet Protocol
ISIM	IM (IP Multimedia) Service Identity Module
KV	Key Value
LPWA	Low-Power Wide-Area
LwIP	Lightweight IP
LwM2M	Lightweight Machine to Machine
MS	Mobile Station
NAS	Non-Access Stratum

NAT	Network Address Translation
NB-IoT	Narrow Band Internet of Things
NITZ	Network Informed Time Zone
NSLPI	NAS Signalling Low Priority Indication
NVM	Non-Volatile Memory
OoS	Out of Service
PAP	Password Authentication Protocol
PCI	Physical Cell Identity
PCO	Protocol Configuration Options
PDP	Packet Data Protocol
PDU	Packet Data Unit
PLMN	Public Land Mobile Network
PSK	Pre-Shared Key
PWM	Pulse Width Modulation
QCI	QoS Class Identifier
QoS	Quality of Service
RAI	Release Assistance Indication
RP	Relay Protocol
RPM	RPM Package Manager
RRC	Radio Resource Control
RTC	Real Time Clock
RxQual	Receive Quality
SMSC	Short Message Service Center
SIB	System Information Block
SNR	Signal-to-Noise Ratio

SVN	Software Version Number
TA	Terminal Adapter
TAU	Tracking Area Update
TCP	Transmission Control Protocol
TE	Terminal Equipment
TFT	Traffic Flow Template
TP	Transfer Layer Protocol
TPDU	Transport Protocol Data Unit
TTL	Time to Live
UDP	User Datagram Protocol
UE	User Equipment
UICC	Universal Integrated Circuit Card
URC	Unsolicited Result Code
UTRAN	Universal Terrestrial Radio Access Network
USIM	Universal Subscriber Idneity Module
UUID	Universally Unique Identifier
VPLMN	Visiting Public Land Mobile Network
WIM	Wireless Identity Module
XOFF	Transmit off
XON	Transmit on