HT MICRON SEMICONDUTORES S.A.
Av. Unisinos, 1550 | 93022-750 | São Leopoldo – RS | Brasil
www.htmicron.com.br

HT MICRON semicondutores

iMCP HTNB32L-XXX AT Commands

NB-IoT RF System-in-Package AT Commands Manual

Classification: PUBLIC

Doc. Type: USER DOCUMENTATION

Revision: v.01

Date: 14/04/2023

Code: ATC-HTNB32L-XXX

SUMMARY

DOCUMENT INFO

This document provides technical information about the AT Commands for iMCP HTNB32L-XXX. It is intended to contribute only the necessary information to run the AT Commands firmware. More specific details can be found at HTNB32L GitHub Page.

1. GENERAL DESCRIPTION

The iMCP HTNB32L-XXX is a highly compact and low-power wireless communication MCO/SiP featuring Qualcomm QCX-212 LTE IoT Modem supporting single-mode 3GPP Release 14 Cat. NB2 IoT connectivity. Its SDK (Software Development Kit) provides OpenCPU solutions based on a FreeRTOS system, where users can embed their own IoT application, as well as AT Commands, used in a master-slave models.

2. INTRODUCTION

In this section will be explained the AT Commands definitions and syntax.

2.1. DEFINITIONS

The following syntactical definitions are used in this document:

- <CR>: Carriage return character.
- <LF>: Linefeed character.
- <...>: Name enclosed in angle brackets is a syntactical element. Brackets do not appear in the command line.
- [...]: Optional sub parameter of a command or an optional part of TA information response is enclosed in square brackets. Brackets do not appear in the command line. When a sub parameter is not provided in parameter type commands, the new value equals to its previous value. In action type commands, action must be performed based on the recommended default setting of the sub parameter.
- NO_SAVE: The parameter of the current AT command is lost if the module is rebooted or current AT command does not have a parameter.
- AUTO_SAVE: The parameter of the current AT command is kept in NVRAM automatically and takes effect immediately. It is not lost if the module is rebooted.
- AUTO_SAVE_REBOOT: The parameter of the current AT command is kept in NVRAM automatically and takes effect after reboot. It is not lost if the module is rebooted.
- '-': Indicates that the AT command does not depend on the parameter saving mode.

2.2. AT COMMAND SYNTAX

2.2.1. AT command type

Table 2.1: AT command type

Type	Format	Description
Test command	AT+<cmd>=?	Check possible sub parameter values
Read command	AT+<cmd>?	Check current sub parameter values
Set command	AT+<cmd>=p1[,p2[,p3[…]]]	Set command
Execution command	AT+<cmd>	Execution command

2.2.2. Command line

The commands in this specification use syntax rules of extended commands. Every extended command has a Test command (`trailing=?`) to test the existence of the command and to give information about the type of its sub parameters. Parameter type commands also have a Read command (`trailing?`) to check the current values of sub parameters. Action type commands do not store the values of any of their possible sub parameters, and therefore do not have a Read command.

Every command must be sent followed by semicolon "**;**" and appended with <CR><LF> characters. For example, AT+<cmd>=p1[,p2[,p3[…]]];<CR><LF>, for Set command type, AT+<cmd>=?;<CR><LF>, for Test command type, AT+<cmd>?;<CR><LF>, for Read command type and AT+<cmd>;<CR><LF>, for execution command type.

In Set commands, use commas to separate arguments. For example, AT+<cmd>=p1,p2,p3;<CR><LF>. Optional sub parameters can be omitted.

If all commands in a command line are successfully run, the result code <CR><LF>OK<CR><LF> is sent from the TA to the TE.

If sub parameter values of a command are not accepted by the TA (or command itself is invalid, or command cannot be performed for some reason), result code <CR><LF>ERROR<CR><LF> is sent to the TE and no subsequent commands in the command line are processed. ERROR response may be replaced by +CME ERROR: <err> (refer clause 4) when the command was not processed due to an error related to the MT operation.

2.3. 3GPP COMPLIANCE

Basic commands are compiled with ITU-T V.250(07/2003).
3GPP commands are complied with the V16.0.0 (2019-03) of AT command set for User Equipment (UE) (3GPP TS 27.007) and V15.0.0 (2018-06) of Use of Data Terminal Equipment - Data Circuit terminating Equipment (DTE - DCE) interface for Short Message Service (SMS) and Cell Broadcast Service (CBS) (3GPP TS 27.005).

3. GENERAL CONTROL COMMANDS

3.1. BASIC COMMANDS (ITU-T V.250)

3.1.1. ATE command echo

The setting of this parameter determines whether the DCE echoes characters received from the DTE during command state and online command state.

Table 3.1: ATE

ATE	Response

Set command
`ATE<value>`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Maximum response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<value>	Integer type	
	0	DCE does not echo characters during command state and online command state
	1	DCE echoes characters during command state and online command state
	The default value is 1.	

Example

```
ATE0
OK
```

3.1.2. ATQ result code suppression

The setting of this parameter determines whether the DCE transmits unsolicited result codes to the DTE. When result codes are being suppressed, unsolicited result is not transmitted.

NOTE

Currently, this command is not fully implemented, as defined in ITU-T V.250.

Table 3.2: ATQ

ATQ	Response
Set command ATQ<value>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<value>	Integer type	
	0	DCE transmits unsolicited result codes
	1	Unsolicited result codes are suppressed and not transmitted.
	The default value is 0.	

Example

```
ATQ0
OK
```

NOTE

If set to "1", all unsolicited result codes are all suppressed, including: PING/IPERF/LWM2M unsolicited result codes;

NOTE

If set to "1", only suppress the unsolicited result codes; And AT response/result codes are not suppressed;

3.2. 3GPP COMMANDS (27.007)

3.2.1. AT+CFUN set phone functionality

The Set command selects the level of functionality in the MT. 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 MT as compound values.

Table 3.3: AT+CFUN

AT+CFUN	Response
Set command AT+CFUN=<fun>[,<rst>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CFUN?	Response +CFUN: <fun> OK
Test command AT+CFUN=?	Response +CFUN: (list of supported <fun>s), (list of supported <rst>s) OK
Maximum Response Time	25 s
Parameter Saving Mode	NO_SAVE

Parameter

<fun>	Integer type	
	0	Minimum functionality
	1	Full functionality
	4	Turn off RF
<rst>	Integer type	
	0	Do not reset the MT before setting it to <fun> power level. This is always defaulted when <rst> is not given.
	1	Reset the MT before setting it to <fun> power level (currently not supported).

Example

```
AT+CFUN=?
+CFUN:(0,1,4),(0)
OK

AT+CFUN?
+CFUN:1
OK

AT+CFUN=1
OK
```

3.2.2. AT+CGSN request product serial number

The Execution command returns the International Mobile Equipment Identity (IMEI) number and related information.

The Test command returns values supported as a compound value.

Table 3.4: AT+CGSN

AT+CGSN	Response
Set command AT+CGSN =<snt>	Response When <snt>=0 and command successful: + CGSN: <sn> OK When <snt>=1 and command successful: + CGSN: <imei> OK When <snt>=2 and command successful: + CGSN: <imeisv> OK When <snt>=3 and command successful: + CGSN: <svn> OK If there is an error, the response is as follows: +CME ERROR: <err>
Execution command AT+CGSN?	Response <sn> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGSN=?	Response + CGSN: (list of supported <snt>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<snt>	Integer type	
	0	Returns <sn>
	1	Returns IMEI (International Mobile station Equipment Identity)
	2	Returns IMEISV (International Mobile station Equipment Identity and Software Version number)
	3	Returns SVN (Software Version Number)
<sn>	One or more lines of information text determined by the MT manufacturer.	
<imei>	String type. The IMEI in decimal format.	
<imeisv>	String type. The IMEISV in decimal format.	
<svn>	String type. The current SVN in decimal format, it is a part of IMEISV	

Example

```
AT+CGSN=1
+CGSN: "788596633100008"
OK

AT+CGSN=2
+CGSN: "7885966331000001"
OK
```

```
AT+CGSN=3
+CGSN: ”01”
OK

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

3.2.3. AT+CGMR request manufacturer revision

The Execution command returns the manufacturer revision. It also returns the firmware revision and build time.

Table 3.5: AT+CGMR

AT+CGMR	Response
Execution command AT+CGMR	Response +CGMR: <”Board Version && SDK Version && EVB Version && Compiled Time”>
Test command AT+CGMR=?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Example

```
AT+CGMR
+CGMR:
-- Board: QCX212_EVK –
-- SDK Version: QCX212_SW_V001.000.xxx –
-- EVB Version: QCX212_HW_V1.0 –
-- Compiled: Jul 23 2019 20:50:16 –
OK

AT+CGMR=?
OK
```

3.2.4. AT+CMEE report mobile termination 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 MT. When enabled, MT-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 TA functionality.

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

The Test command returns values supported as a compound value.

Table 3.6: AT+CMEE

AT+CMEE	Response
Set command AT+CMEE=<n>	Response OK
Read command AT+CMEE?	Response +CMEE: <n> OK
Test command AT+CMEE=?	Response +CMEE: (list of supported <n>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	Integer type	
<n>	0	Disable +CME ERROR: <err> result code and use ERROR instead
	1	Enable +CME ERROR: <err> result code and use numeric <err> values
	2	Enable +CME ERROR: <err> result code and use verbose <err> values

Example

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

AT+CMEE?
+CMEE: 1
OK

AT+CMEE=2
OK
```

3.2.5. AT+COPS PLMN selection

The Set command forces an attempt to select and register the network operator using the USIM card installed in the currently selected card slot. <mode> is used to select whether the selection is performed automatically by the MT or is forced by this command to operator <oper> (it is given in format <format>) to a certain access technology, indicated in <AcT>. If the selected operator is not available, no other operator is selected (except <mode>=4). If the selected access technology is not available, then the same operator is selected in other access technology. The selected operator name format also applies to further Read commands (AT+COPS?). <mode>=2 forces an attempt to deregister from the network. The selected mode affects all further network registration (for example, after <mode>=2, MT is unregistered until <mode>=0 or 1 is selected). This command must be abortable 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 must then be an empty field. The list of operators is in order: home network, networks referenced in USIM or active application in the UICC (GSM or USIM) in the following order: HPLMN selector, user controlled PLMN selector, operator controlled PLMN selector and PLMN selector (in the USIM or GSM application), and other networks. After the operator list MT returns lists of supported <mode>s and <format>s. These lists are delimited from the operator list by two commas.

Table 3.7: AT+COPS

AT+COPS	Response
Set command AT+COPS=<mode>[,<format>[,<oper>[,AcT]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+COPS?	Response +COPS: <mode>[,<format>,<oper>][,AcT] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COPS=?	Response +COPS: [list of supported (<stat>, long alphanumeric <oper>, short alphanumeric <oper>,numeric <oper>[, <AcT>])s],, (list of supported <mode>s), (list of supported <format>s) OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	245 s
Parameter Saving Mode	AUTO_SAVE Only when <mode>=0/1/4, the configuration will be saved to NVRAM automatically.

Parameter

<mode>	Integer type	
	0	Automatic mode (<oper> field is ignored)
	1	Manual operator selection (<oper> field is present, and <AcT> is optional)
	2	Manually deregister from network
	3	Set <format> not shown in read command response
	4	Manual/automatic selected. If manual selection fails, automatic mode (<mode>=0) is entered
<format>	Integer type	
	0	Long format alphanumeric <oper>
	1	Short format alphanumeric <oper>
	2	Numeric <oper>

<oper>	String Type	
	<format> indicates if the format is alphanumeric or numeric. Long alphanumeric format can be up to 16 characters long and short format up to 8 characters. Numeric format is the GSM 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.	
<stat>	Integer Type	
	0	Unknown
	1	Operator available
	2	Operator currently selected
	3	Operator forbidden to be selected
<AcT>	Integer Type	
	9	NB-IoT

Example

```
AT+COPS=1,2,"46000"
OK

AT+COPS?
+COPS: 0,2,"46000",9
OK

AT+COPS=?
+COPS: (2,"CHINA MOBILE","CMCC","46000",9),(1,"CHINA
TELECOM","CTCC","46011",9),(3,"CHINA UNICOMM","CUCC","46001",9),,(0-
4),(0-2)
OK
```

3.2.6. AT+CREG network registration

The Set command controls the presentation of an unsolicited result code +CREG: <stat> when <n>=1 and there is a change in the circuit mode network registration status of the MT in GERAN/UTRAN/E-UTRAN, or unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>]] when <n>=2 and there is a change of the network cell in GERAN/UTRAN/E-UTRAN. The parameters <AcT>, <lac> and <ci> are sent only if available. The value <n>=3 further extends the unsolicited result code with [,<cause_type>,<reject_cause>], when available, when the value of <stat> changes.

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 MT. Location information elements <lac>, <ci> and <AcT>, if available, are returned only when <n>=2 and MT is registered in the network. The parameters [,<cause_type>,<reject_cause>], if available, are returned when <n>=3.

Test command returns values supported as a compound value.

Table 3.8: AT+CREG

AT+CREG	Response
Set command AT+CREG=[<n>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CREG?	Response +CREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>] [,<cause_type>,<reject_cause>]] OK
Test command AT+CREG=?	Response +CREG: (list of supported <n>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<n>	Integer type	
	0	Disable network registration unsolicited result code
	1	Enable network registration unsolicited result code +CREG: <stat>
	2	Enable network registration and location information unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>]
	3	Enable network registration, location information and cause value information unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]
<stat>	Integer type	
	0	Not registered, MT is not currently searching a new operator to register to
	1	Registered, home network
	2	Not registered, but MT is searching a new operator to register to
	3	Registration denied
	4	Unknown (for example, out of GERAN/UTRAN/E-UTRAN coverage)
	5	Registered, roaming
	6	Registered for "SMS only", home network (applicable only when <AcT> indicates E-UTRAN)
	7	Registered for "SMS only", roaming (applicable only when <AcT> indicates E-UTRAN)
<tac>	String type	
	Tracking area code in hexadecimal format, two bytes	
<ci>	String type	
	Four-byte E-UTRAN Cell ID in hexadecimal format	
<AcT>	Integer type	
	9	E-UTRAN (NB-S1 mode)
<cause_type>	Integer type	
	0	<reject_cause> contains an EMM cause value (see 3GPP TS 24.008 Annex G)
	1	<reject_cause> packet contains a manufacturer specific cause

<reject_cause>	Integer type
	Contains the cause of the failed registration. The value is of type as defined by <cause_type> (see 3GPP TS 24.301)

Example

```
AT+CREG?
+CREG: 3,0
OK
```

3.2.7. AT+CEREG EPS network registration status

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

If the UE requests PSM for reducing its power consumption, the Set command controls the presentation of an unsolicited result code:

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

When <n>=4, the unsolicited result code provides 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 unsolicited result code 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 MT. Location information elements <tac>, <ci> and <AcT>, if available, are returned only when <n>=2 and MT is registered in the network. The parameters [,<cause_type>,<reject_cause>], if available, are returned when <n>=3. The Test command returns values supported as a compound value.

Table 3.9: AT+CEREG

AT+CEREG	Response
Set command AT+CEREG=[<n>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CEREG?	When <n>=0,1,2 or 3 and command successful: +CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>[,<cause_type>,<reject_cause>]]] OK When <n>=4 or 5 and command successful:

	+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>],[<cause_type>,<reject_cause>[,[<Active_Time>],[<Periodic_TAU>]]]] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CEREG=?	Response +CEREG: (list of supported <n>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<n>	Integer type	
	0	Disable network registration unsolicited result code
	1	Enable network registration unsolicited result code +CREG: <stat>
	2	Enable network registration and location information unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>]
	3	Enable network registration, location information and cause value information unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]
	4	For a UE that wants to apply PSM, enable network registration and location information unsolicited result code +CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<ActiveTime>],[<Periodic-TAU>]]]
	5	For a UE that wants to apply PSM, enable network registration, location information, and EMM cause value information unsolicited result code +CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<Active-Time>],[<Periodic-TAU>]]]
<stat>	Integer type	
	0	Not registered, MT is not currently searching a new operator to register to
	1	Registered, home network
	2	Not registered, but MT is searching a new operator to register to
	3	Registration denied
	4	Unknown (for example, out of GERAN/UTRAN/E-UTRAN coverage)
	5	Registered, roaming
<tac>	String type	
	Tracking area code in hexadecimal format, two bytes	
<ci>	String type	
	Four-byte E-UTRAN Cell ID in hexadecimal format	
<AcT>	Integer type	
	9	E-UTRAN (NB-S1 mode)
<cause_type>	Integer type	
	0	<reject_cause> contains an EMM cause value (see 3GPP TS 24.008 Annex G)

	1	<reject_cause> packet contains a manufacturer specific cause
<reject_cause>	Integer type	
	Contains the cause of the failed registration. The value is of type as defined by <cause_type> (see 3GPP TS 24.301)	
<Active_Time>	String type	
	One byte in an 8-bit format. 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 the GPRS Timer 2 IE in 3GPP TS 24.008 Table 10.5.163/3Gpp TS 24.008, 3GPP TS 23.682 and 3GPP TS 23.401. Bits 5 to 1 represents the binary coded timer value. Bits 8 to 6 defines the timer value unit for the GPRS timer as follows: Bits 8 7 6 0 0 0 value is incremented in multiples of 2 seconds 0 0 1 value is incremented in multiples of 1 minute 0 1 0 value is incremented in multiples of 6 minutes 1 1 1 value indicates that the timer is deactivated	
<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 GPRS Timers 3 IE in 3GPP TS 24.008 Table 10.5.163a/3GPP TS 24.008, 3GPP TS 23.682 and 3GPP TS 23.401. Bits 5 to 1 represents the binary coded timer value. Bits 8 to 6 defines the timer value increment 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=5
OK

AT+CEREG?
+CEREG: 5,1,”5b49”,”0190271a”,9
OK

AT+CEREG=?
```

```
+CEREG: (0,1,2,3,4,5)
OK
```

3.2.8. AT+CSQ get signal quality

The Execution command returns received signal quality <rssi> and channel bit error rate <ber> from the MT.

The Test command returns values supported as compound values.

Table 3.10: AT+CSQ

AT+CSQ	Response
Execution command AT+CSQ	Response +CSQ: <rssi>,<ber> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CSQ=?	Response +CSQ: (list of supported <rssi>s), (list of supported <ber>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<rssi>	Integer type	
	0	-113dBm or less
	1	-111dBm
	2 to 30	-109dBm to -53dBm
	31	-51 dBm or greater
	99	Not known or not detectable
<ber>	Integer type	
	0 to 7	As RXQUAL values in the table in subclause 8.2.4 of GSM/EDGE Radio Subsystem Link Control (3GPP TS 45.008)
	99	Not known or not detectable

Example

```
AT+CSQ
+CSQ: 27,0
OK
```

3.2.9. AT+CESQ get extended signal quality

The Execution command returns received signal quality parameters. As it only supports NB-IoT <rxlev> and <ber> are set to value 99, <rscp> and <ecno> are set to 255.

The Test command returns values supported as compound values.

Table 3.11: AT+CESQ

AT+CESQ	Response
Execution command AT+CESQ	Response +CESQ: <rxlev>,<ber>,<rscp>,<ecno>,<rsrq>,<rsrp> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CESQ=?	Response +CESQ: (list of supported <rxlev>s),(list of supported <ber>s), (list of supported <rscp>s),(list of supported <ecno>s), (list of supported <rsrq>s),(list of supported <rsrp>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<rxlev>	Integer type	
	99	Not known or not detectable
<ber>	Integer type	
	99	Not known or not detectable
<rscp>	Integer type	
	255	Not known or not detectable
<ecno>	Integer type	
	255	Not known or not detectable
<rsrq>	Integer type	
	0	RSRQ < -19.5dB
	1	-19.5dB <= RSRQ < -19dB
	2	-19dB <= RSRQ < -18.5dB
	…	
	32	-4dB <= RSRQ < -3.5dB
	32	-3.5dB <= RSRQ < -3dB
	32	-3dB <= RSRQ
	255	Not known or not detectable
<rsrp>	Integer type	
	0	RSRP < -149dBm
	1	-140dBm <= RSRP < -139dBm
	2	-139dBm <= RSRP < -138dBm
	…	
	95	-46dBm <= RSRP < -45dBm
	96	-45dBm <= RSRP < -44dBm
	97	-44dBm <= RSRP
	255	Not known or not detectable

Example

```
AT+CESQ
+CESQ: 99,99,255,255,26,56
OK
AT+CESQ=?
+CESQ: (99),(99),(255),(255),(0-34,255),(0-97,255)
OK
```

3.2.10. AT+CPSMS power-saving mode setting

The Set command controls the setting of the power saving mode (PSM) parameters of the UE. The command controls whether the UE wants to apply PSM or not. See unsolicited result codes 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 is disabled and data for all parameters in the command +CPSMS is removed.

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.

Table 3.12: AT+CPSMS

AT+CPSMS	Response
Set command AT+CPSMS=<mode> [,<Requested_Periodic-RAU> [,<Requested_GPRS-READY-timer> [,<Requested_Periodic-TAU> [,<Requested_Active-Time>]]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CPSMS?	Response +CPSMS: <mode>, [<Requested_Periodic-RAU>], [<Requested_GPRS-READY-timer>], [<Requested_Periodic-TAU>], [<Requested_Active-Time>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CPSMS=?	Response +CPSMS: (list of supported <n>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<mode>	Integer type	
	0	Disable the use of PSM

	1	Enable the use of PSM
	2	Disable the use of PSM and discard all parameters for PSM
<Requested_Periodic-RAU>	String Type	
	Not supported by NB-IoT	
<Requested_GPRS-READY-timer>	String Type	
	Not supported by NB-IoT	
<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. (e.g. "01000111" equals 70 hours). Bits 5 to 1 represents the binary coded timer value. Bits 8 to 6 defines 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 1 1 1 value indicates that the timer is deactivated The default value is 20 hours	
<Requested_Active-Time>	String Type	
	one byte in an 8-bit format. Requested Active Time Value (T3324) to be allocated to the UE. (for example, "00100100" equals 4 minutes). Bits 5 to 1 represent the binary coded timer value. Bits 6 to 8 defines the timer value unit for the GPRS timer as follows: Bits 8 7 6 0 0 0 value is incremented in multiples of 2 seconds 0 0 1 value is incremented in multiples of 1 minute 0 1 0 value is incremented in multiples of 6 minutes 1 1 1 value indicates that the timer is deactivated The default value is 5 minutes	

Example

```
AT+CPSMS=1,,,,”00100010”
OK

AT+CPSMS?
+CPSMS: 1,,,,”00100010”
OK

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

```
OK
```

3.2.11. AT+CEDRXS eDRX setting

The Set 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 NB-IoT.

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

Table 3.13: AT+CEDRXS

AT+CEDRXS	Response

Set command
`AT+CEDRXS=<mode>,<AcTtype>[,<Requested_eDRX_value>]`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Read command
`AT+CEDRXS?`
	Response
	`+CEDRXS: <AcT-type>,<Requested_eDRX_value>`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Test command `AT+CEDRXS=?`	Response `+CEDRXS:` (list of supported `<mode>`s), (list of supported `<AcT-type>`s), (list of supported `<Requested_eDRX_value>`s) `OK`
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	Integer type	
<mode>	Disable or enable the use of eDRX in the UE. This parameter is applicable to all specified types of access technologies, i.e. the most recent setting of `<mode>` takes effect for all specified values of `<AcT_type>`.	
	0	Disable the use of eDRX
	1	Enable the use of eDRX
	2	Enable the use of eDRX and enable the unsolicited result code: `+CEDRXP: <AcTtype>[,<Requested_eDRX_value>[,<NWprovided_eDRX_value>[,<Paging_time_window>]]]`

	3 | Disable the use of eDRX and discard all parameters for eDRX.
<AcT_type>	Integer Type
	The type of access technology. `AT+CEDRXS?` specifies the relationship between the type of access technology and the requested eDRX value.
	5 | E-UTRAN (NB-S1 mode)
<requested_eDRX_value>	String Type
	Half a byte in a 4-bit format. NB-S1 mode.
	Bits 4 3 2 1 E-UTRAN eDRX cycle length duration 0 0 1 0 20.48 seconds 0 0 1 1 40.96 seconds 0 1 0 1 81.92 seconds 1 0 0 1 163.84 seconds 1 0 1 0 327.68 seconds 1 0 1 1 655.36 seconds 1 1 0 0 1310.72 seconds 1 1 0 1 2621.44 seconds 1 1 1 0 5242.88 seconds 1 1 1 1 10485.76 seconds
<NW_provided_eDRX_value>	String Type
	Half a byte in a 4-bit format. NB-S1 mode.
	Bits 4 3 2 1 E-UTRAN eDRX cycle length duration 0 0 1 0 20.48 seconds 0 0 1 1 40.96 seconds 0 1 0 1 81.92 seconds 1 0 0 1 163.84 seconds 1 0 1 0 327.68 seconds 1 0 1 1 655.36 seconds 1 1 0 0 1310.72 seconds 1 1 0 1 2621.44 seconds 1 1 1 0 5242.88 seconds 1 1 1 1 10485.76 seconds
<paging_time_window>	String Type
	Half a byte in a 4-bit format. NB-S1 mode.
	Bits 4 3 2 1 Paging Time Window length 0 0 0 0 2.56 seconds 0 0 0 1 5.12 seconds 0 0 1 0 7.68 seconds 0 0 1 1 10.24 seconds 0 1 0 0 12.8 seconds 0 1 0 1 15.36 seconds 0 1 1 0 17.92 seconds 0 1 1 1 20.48 seconds 1 0 0 0 23.04 seconds

	1 0 0 1 25.6 seconds 1 0 1 0 28.16 seconds 1 0 1 1 30.72 seconds 1 1 0 0 33.28 seconds 1 1 0 1 35.84 seconds 1 1 1 0 38.4 seconds 1 1 1 1 40.96 seconds

Example

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

AT+CEDRXS?
+CEDRXS: 5,"0010"
OK

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

3.2.12. AT+CEDRXRDP eDRX read dynamic parameters

The Execution command returns <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 is returned.

Table 3.14: AT+CEDRXRDP

AT+CEDRXRDP	Response
Execution command AT+CEDRXRDP	Response +CEDRXRDP: <AcTtype>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]] If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CEDRXRDP=?	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<AcT_type>	Integer Type	
	The type of access technology. AT+CEDRXS? specifies the relationship between the type of access technology and the requested eDRX value.	
	0	Access technology not supporting eDRX.

	5 | E-UTRAN (NB-S1 mode)
<requested_eDRX_value>	String Type
	Half a byte in a 4-bit format. NB-S1 mode.
	Bits 4 3 2 1 E-UTRAN eDRX cycle length duration 0 0 1 0 20.48 seconds 0 0 1 1 40.96 seconds 0 1 0 1 81.92 seconds 1 0 0 1 163.84 seconds 1 0 1 0 327.68 seconds 1 0 1 1 655.36 seconds 1 1 0 0 1310.72 seconds 1 1 0 1 2621.44 seconds 1 1 1 0 5242.88 seconds 1 1 1 1 10485.76 seconds
<NW_provided_eDRX_value>	String Type
	Half a byte in a 4-bit format. NB-S1 mode.
	Bits 4 3 2 1 E-UTRAN eDRX cycle length duration 0 0 1 0 20.48 seconds 0 0 1 1 40.96 seconds 0 1 0 1 81.92 seconds 1 0 0 1 163.84 seconds 1 0 1 0 327.68 seconds 1 0 1 1 655.36 seconds 1 1 0 0 1310.72 seconds 1 1 0 1 2621.44 seconds 1 1 1 0 5242.88 seconds 1 1 1 1 10485.76 seconds
<paging_time_window>	String Type
	Half a byte in a 4-bit format. NB-S1 mode.
	Bits 4 3 2 1 Paging Time Window length 0 0 0 0 2.56 seconds 0 0 0 1 5.12 seconds 0 0 1 0 7.68 seconds 0 0 1 1 10.24 seconds 0 1 0 0 12.8 seconds 0 1 0 1 15.36 seconds 0 1 1 0 17.92 seconds 0 1 1 1 20.48 seconds 1 0 0 0 23.04 seconds 1 0 0 1 25.6 seconds 1 0 1 0 28.16 seconds 1 0 1 1 30.72 seconds 1 1 0 0 33.28 seconds 1 1 0 1 35.84 seconds

	1 1 1 0 38.4 seconds 1 1 1 1 40.96 seconds

Example

```
AT+CEDRXRDP
+CEDRXRDP: 5,”0010”,”1101”,”0100”
OK

AT+CEDRXRDP=?
OK
```

3.2.13. AT+CCIOTOPT CIoT optimization configuration

The Set 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.

The Set command is used also to control the unsolicited result code `+CCIOTOPTI`. An unsolicited result code `+CCIOTOPTI: <supported_Network_opt>` is used to indicate the supported CIoT EPS optimizations by the network.

The Read command returns the current settings for supported and preferred CIoT EPS optimizations and the current status of unsolicited result code `+CCIOTOPTI`.

The Test command returns values supported as compound values.

Table 3.15: AT+CCIOTOPT

AT+CCIOTOPT	Response

Set command
`AT+CCIOTOPT=<n>`
`[,<support_UE_opt>`
`[,<preferred_UE_opt>]]`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Read command
`AT+CCIOTOPT?`
	Response
	`+CCIOTOPT: <n>,<support_UE_opt>,`
	`<preferred_UE_opt>`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Test command `AT+CCIOTOPT=?`	Response `+CCIOTOPT:` (list of supported `<n>`s), (list of supported `<support_UE_opt>`s), (list of supported `<preferred_UE_opt>`s) `OK`
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<n>	Integer type	
	Enable/disable 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.	
	1	Support control plane CIoT EPS optimization
	3	Support both control plane and user plane CIoT EPS optimizations
<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's support for CIoT EPS optimizations.	
	0	Not support
	1	Support control plane CIoT EPS optimization
	2	Support user plane CIoT EPS optimization
	3	Support both control plane and user plane CIoT EPS optimizations

Example

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

AT+CCIOTOPT?
+CCIOTOPT: 0,3,1
OK
```

3.2.14. AT+CGCMOD PDP context modify

The Set command is used to modify the specified PDP context with request to QoS profiles and TFTs. If the requested modification for any specified context cannot be achieved, an ERROR or +CME ERROR response is returned.

The Test command returns a list of <cid>s associated with active contexts.

Table 3.16: AT+CGCMOD

AT+CGCMOD	Response
Set command AT+CGCMOD=<cid>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>

Test command AT+CGCMOD=?	Response +CGCMOD: (list of <cid>s associated with active contexts) OK
Maximum Response Time	70 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.

Example

```
AT+CGCMOD=?
+CGCMOD: (5)
OK
```

3.2.15. AT+CGATT PS attach or detach

The Set command is used to attach the MT to, or detach the MT from, the Packet Domain service. If the MT is already in the requested state, the command is ignored, and the OK response is returned. If the requested state cannot be achieved, +CME ERROR response is returned. Any active PDP contexts are 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.

Table 3.17: AT+CGATT

AT+CGATT	Response
Set command AT+CGATT=<state>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CGATT?	Response +CGATT: <state> OK
Test command AT+CGATT=?	Response +CGATT: (list of supported <state>s) OK
Maximum Response Time	20 s
Parameter Saving Mode	NO_SAVE

Parameter

<state>	Integer type	
	Indicates the state of PS attachment.	
	0	Detached
	1	Attached

Example

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

3.2.16. AT+CGACT PDP context activate or deactivate

The Set command is used to activate or deactivate the specified PDP context. 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 `+CME ERROR` response is returned. If the MT is not PS attached when the activation form of the command is executed, the MT first performs a PS attach and then attempts to activate the specified contexts. If the attach fail, then the MT responds with `+CME ERROR`.

For EPS, if an attempt is made to disconnect the last PDN connection, the MT responds with a `+CME ERROR`.

For EPS, the activation request for an EPS bearer resource is answered by the network by either an EPS dedicated bearer activation or EPS bearer modification request. The request must be accepted by the MT before the PDP context can be set in to 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.

Table 3.18: AT+CGACT

AT+CGACT	Response

Set command
`AT+CGACT=<state>, <cid>`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Read command
`AT+CGACT?`
	Response
	`[+CGACT: <cid>,<state>]`
	`OK`

Test command `AT+CGACT=?`	Response `+CGACT:` (list of supported `<state>`s) `OK`
Maximum Response Time	70 s
Parameter Saving Mode	NO_SAVE

Parameter

<state>	Integer type	
	Indicates the state of PDP context activation.	
	0	Detached
	1	Attached
<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.

Example

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

AT+CGACT?
+CGACT: 5,1
OK
```

3.2.17. AT+CGDATA enter data state

The Set command causes the MT to perform whatever actions are necessary to establish communication between the TE and the network using one Packet Domain PDP types. This may include performing a PS attach and one PDP context activation. <cid> must be specified (see the +CGDCONT) to provide the information needed for the context activation request.

The Test command is used for requesting information on the supported <L2P> protocols.

Table 3.19: AT+CGDATA

AT+CGDATA	Response
Set command AT+CGDATA=<L2P>,<cid>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGDATA=?	Response +CGDATA: (list of supported <L2P>s) OK
Maximum Response Time	70 s
Parameter Saving Mode	NO_SAVE

Parameter

<L2P>	String type	
	Indicates the layer 2 protocol to be used between the TE and MT.	
	M-PT	Specified protocol – PDP Type, such as IP/IPV6/IPV4V6/Non-IP
<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.

Example

```
AT+CGDATA=?
+CGDATA: "M-PT"
OK
```

3.2.18. AT+CGDCONT define a PDP context

The Set command specifies the PDP context parameter values for a PDP context identified by the (local) context identification parameter, <cid>. 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, for example, 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 is identified herewith.

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

The Read command returns the current settings for each defined context.
The Test command returns values supported as compound values.

Table 3.20: AT+CGDCONT

AT+CGDCONT	Response
Set command AT+CGDCONT=<cid> [,<PDP_type>[,<APN>[,<PDP_addr> [,<d_comp>[,<h_comp> [,<IPv4AddrAlloc>[,<request_type> [,<P-CSCF_discovery> [,<IM_CN_Signalling_Flag_Ind> [,<NSLPI>[,<securePCO> [,<IPv4_MTU_discovery> [,<Local_Addr_Ind> [,<NonIP_MTU_discovery]]]]]]]]]]]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CGDCONT?	Response +CGDCONT: <cid>,<PDP_type>,<APN>, <PDP_addr>,<d_comp>,<h_comp> [,<IPv4AddrAlloc>[,<request_type> [,<P-CSCF_discovery> [,<IM_CN_Signalling_Flag_Ind> [,<NSLPI>[,<securePCO> [,<IPv4_MTU_discovery> [,<Local_Addr_Ind> [,<Non-IP_MTU_discovery>]]]]]]]]]
Test command AT+CGDCONT=?	Response +CGDCONT: (list of supported <cid>s), <PDP_type>,,,(list of supported <d_comp>s), (list of supported <h_comp>s), (list of supported <IPv4AddrAlloc>s), (list of supported <request_type>s), (list of supported <P-CSCF_discovery>s), (list of supported <IM_CN_Signalling_Flag_Ind>s),

	(list of supported <NSLPI>s), (list of supported <securePCO>s), (list of supported <IPv4_MTU_discovery>s), (list of supported <Local_Addr_Ind>s), (list of supported <NonIP_MTU_discovery>s), (list of supported <Reliable_Data_Service>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<PDP_type>	String type	
	Specifies the type of packet data protocol.	
	"IP"	Internet Protocol
	"IPV6"	Internet Protocol, version 6
	"IPV4V6"	Virtual <PDP_type> introduced to handle dual IP stack UE capacity
	"Non-IP"	Transfer of Non-IP data to external packet data network
<APN>	String type	
	A logical name that is used to select the GGSN or the external packet data network. The maximum configurable APN length is 99 bytes. If the value is null or omitted, then the subscription value will be requested.	
<PDP_addr>	String type	
	A string parameter thar identifies the MT in the address space applicable to the PDP.	
<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
	2	RFC 1144 (applicable for SNDCP only)
	3	RFC 2507
	4	RFC 3095[ROHC] (applicable for PDCP only)
<IPv4AddrAlloc>	Integer type	
	Controls how the MT/TA requests to get the IPv4 address information.	
	0	IPv4 address allocation through NAS signaling
	1	IPv4 address allocated through DHCP (not support)

<request_type>	Integer type	
	Indicates the type of PDP context activation request for the PDP context.	
	0	PDP context is for new PDP context establishment or for handover from a non-3GPP access network
	1	PDP context is for emergency bearer services
	2	PDP context is for new PDP context establishment
	3	PDP context is for handover from a non-3GPP access network
<P-CSCF_discovery>	Integer type	
	Influences how the MT/TA requests to get the P-CSCF address.	
	0	Preference of P-CSCF address discovery not influenced by AT+CGDCONT
	1	Preference of P-CSCF address discovery through NAS signaling
	2	Preference of P-CSCF address discovery through DHCP
<IM_CN_Signalling_Flag_Ind>	Integer type	
	Indicates to the network whether the PDP context is for IM CN subsystem-related signaling only or not.	
<NSLPI>	Integer type	
	Indicates the NAS signaling 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 MT.
	1	Indicates that this PDP context is to be activated with the value for the low-priority indicator set to "MS is not configured for NAS signaling low priority"
<securePCO>	Integer type	
	Specifies if security protected transmission of PCO is requested or not.	
	0	Security protected transmission of PCO is not requested
	1	Security protected transmission of PCO is requested
<IPv4_MTU_discovery>	Integer type	
	Influences how the MT/TA requests to get the IPv4 MTU size.	
	0	Preference of IPv4 MTU size discovery not influenced by +CGDCONT
	1	Preference of IPv4 MTU size discovery through NAS signaling
<Local_Addr_Ind>	Integer type	
	Indicates to the network whether the MS supports local IP address in TFTs	
	0	Indicates that the MS does not support local IP address in TFTs
	1	Indicates that the MS supports local IP address in TFTs
<Non-IP_MTU_discovery>	Integer type	
	Influences how the MT/TA requests to get the Non-IP MTU size	
	0	Preference of Non-IP MTU size discovery not influenced by +CGDCONT

	1	Preference of Non-IP MTU size discovery through NAS signaling
<Reliable_Data_Service>	Integer type	
	Indicates whether the UE is using Reliable Data Service for a PDN	
	0	Reliable Data Service is not being used for the PDN connection
	1	Reliable Data Service is being used for the PDN connection

Example

```
AT+CGDCONT=5,"IP","CMNbiot.mnc004.mcc460.gprs",,1,1,0,0,0,0,0,0,1,0,1
OK

AT+CGDCONT?
+CGDCONT: 5,"IP","snbiot.mnc006.mcc460.gprs","10.212.162.96",0,0
OK

AT+CGDCONT=1,"ipv4v6"
OK

AT+CGDCONT?
+CGDCONT: 5,"IP","snbiot.mnc006.mcc460.gprs","10.212.154.7",0,0
+CGDCONT: 1,"IPV4V6",,,0,0
OK

AT+CGDCONT=?
+CGDCONT: (1-11),"IP",,,(0),(0),(0),(0,2),(0),(0),(0,1),(0),(0,1),(0),
(0),(0)
+CGDCONT: (1-11),"IPV6",,,(0),(0),(0),(0,2),(0),(0),(0,1),(0),(0),(0),
(0),(0)
+CGDCONT: (1-11),"IPV4V6",,,(0),(0),(0),(0,2),(0),(0),(0,1),(0),(0,1),
(0),(0),(0)
+CGDCONT: (1-11),"NonIP",,,(0),(0),(0),(0,2),(0),(0),(0,1),(0),(0),(0),
(0,1),(0)
OK
```

3.2.19. AT+CGCONTRDP PDP context read dynamic parameters

The Execution command returns the relevant information for an active non-secondary PDP context with the context identifier <cid>. If the MT has dual stack capabilities, at least one pair of lines with information is returned per <cid>. First one line with the IPv4 parameters followed by one line with the IPv6 parameters. If this MT with dual stack capabilities indicates more than more than two IP addresses of DNS servers, multiple of such pairs of lines are 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.

Table 3.21: AT+CGCONTRDP

AT+CGCONTRDP	Response
Set command AT+CGCONTRDP=<cid>	Response [+CGCONTRDP: <cid>,<bearer_id>,<apn> [,<local_addr and subnet_mask>[,<gw_addr> [,<DNS_prim_addr>[,<DNS_sec_addr> [,<PCSCF_prim_addr>[,<PCSCF_sec_addr> [,<IM_CN_Signalling_Flag>[,<LIPA_indication> [,<IPv4_MTU>[,<WLAN_Offload>[,<Local_Addr_Ind> [,<NonIP_MTU> [,<Serving_PLMN_rate_control_value>]]]]]]]]]]]]] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGCONTRDP=?	Response +CGCONTRDP: (list of <cid>s associated with active contexts) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type
	Specifies a particular PDP context definition.
	1 … 11 | Supported PDP context.
<bearer_id>	Integer type
	Specifies a particular PDP context definition.
<apn>	String type
	A logical name that is used to select the GGSN or the external packet data network. The maximum configurable APN length is 99 bytes. If the value is null or omitted, then the subscription value will be requested.
<local_addr_and_subnet_mask>	String type
	The IP address and subnet mask of the MT.
<gw_addr>	String type
	The IP address of gateway.
<DNS_prim_addr>	String type
	The IP address of the primary DNS server.
<DNS_sec_addr>	String type
	The IP address of the primary DNS server.
<P_CSCF_prim_addr>	String type
	The IP address of the primary P-CSCF server.
<P_CSCF_sec_addr>	String type
	The IP address of the secondary P-CSCF server.
<IM_CN_Signalling_Flag>	Integer type
	Shows whether the PDP context is for IM CN subsystem-related signaling only or not.
<LIPA_indication>	Integer type
	Indicates that the PDP context provides connectivity using a LIPA PDN connection.

<IPv4_MTU>	Integer type
	Shows the IPv4 MTU size in octets.
<WLAN_Offload>	Integer type
	Indicates whether traffic can be offloaded using the specified PDN connection through a WLAN or not.
<Local_Addr_Ind>	Integer type
	Indicates whether the MS and the network support local IP address in TFTs
<NonIP_MTU>	Integer type
	Shows the Non-IP MTU size in octets.
<Serving_PLMN_rate_control_value>	Integer type
	Indicates the maximum number of uplink messages the UE is allowed to send in a 6 minute interval.

Example

```
AT+CGCONTRDP=5
+CGCONTRDP: 5,5,"CMNbiot.mnc004.mcc460.gprs","100.115.240.198.255.255.
255.0","211.136.20.203","211.136.17.107"
OK
```

3.2.20. AT+CGEQOS define EPS quality of service

The Set command allows the TE to specify the EPS Quality of Service parameters <cid>, <QCI>, [<DL_GBR> and <UL_GBR>] and [<DL_MBR> and <UL_MBR>] for a PDP context or Traffic Flows (see Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3 (3GPP TS 24.301) [83] and Policy And Charging Control Architecture (3GPP TS 23.203) [85]).

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

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

The Test command returns the ranges of the supported parameters as compound values.

Table 3.22: AT+CGEQOS

AT+CGEQOS	Response
Set command AT+CGEQOS=<cid>[,<QCI>[,<DL_GBR>,<UL_GBR>[,<DL_MBR>,<UL_MBR>]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CGEQOS?	Response [+CGEQOS:<cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>]] OK If there is an error, the response is as follows: +CME ERROR: <err>

Test command AT+CGEQOS=?	Response +CGEQOS: (range of supported <cid>s),(list of supported <QCI>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<QCI>	Integer type	
	Specifies a class of EPS QoS.	
	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 nonguaranteed bit rate Traffic Flows
	79	Value for nonguaranteed bit rate Traffic Flows
<DL_GBR>	Integer type	
	Indicates DL GBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	
<UL_GBR>	Integer type	
	Indicates UL GBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	
<DL_MBR>	Integer type	
	Indicates DL MBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	
<UL_MBR>	Integer type	
	Indicates UL MBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	

Example

```
AT+CGEQOS=5,9,64,64,64,64
OK
```

3.2.21. AT+CGEQOSRDP EPS quality of service read dynamic parameters

The Execution command returns the quality of service parameters <QCI>,[<DL_GBR> and <UL_GBR>] and [<DL_MBR> and <UL_MBR>] of the active secondary or non-secondary PDP context associated to the provided context identifier <cid>.

If the parameter <cid> is omitted, the quality of service parameters for all secondary and non-secondary active PDP contexts are returned.

The Test command returns a list of <cid>s associated with the secondary or non-secondary active PDP contexts.

Table 3.23: AT+CGEQOSRDP

AT+CGEQOSRDP	Response
Set command AT+CGEQOSRDP=[<cid>]	Response +CGEQOSRDP: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>],[<DL_AMBR>,<UL_AMBR>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGEQOSRDP=?	Response +CGEQOSRDP: (list of <cid>s associated with active contexts) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<QCI>	Integer type	
	Specifies a class of EPS QoS.	
	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 nonguaranteed bit rate Traffic Flows
	79	Value for nonguaranteed bit rate Traffic Flows
	128 … 254	Value range for operator specific QCIs
<DL_GBR>	Integer type	
	Indicates DL GBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	
<UL_GBR>	Integer type	
	Indicates UL GBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	
<DL_MBR>	Integer type	
	Indicates DL MBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	
<UL_MBR>	Integer type	
	Indicates UL MBR if there is GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI.	
<DL_AMBR>	Integer type	
	Indicates DL APN aggregate MBR. The value is in kbit/s.	
<UL_AMBR>	Integer type	
	Indicates UL APN aggregate MBR. The value is in kbit/s.	

Example

```
AT+CGEQOSRDP
+CGEQOSRDP: 5,9
OK
```

3.2.22. 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. 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 Set 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>.

A special form of the Set command, +CGTFT=<cid>, causes all of the PFs in the TFT for context number <cid> to become undefined. At any time, there exists 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 +CME ERROR response is returned.

The Read command returns the current settings for all PFs 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 are used for PDP-type IP only.

Table 3.24: AT+CGTFT

AT+CGTFT	Response
Set command AT+CGTFT=<cid>, [<packet filter identifier>, <evaluation procedure 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> [,<flow label (ipv6)> [,<direction>]]]]]]]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
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>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGTFT=?	Response +CGTFT: (list of supported <cid>s), (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) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context
<packet filter identifier>	Integer type	
	1 … 16	Value range is from 1 to 16
<evaluation precedence index>	Integer type	
	0 … 255	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)	
<protocol number (ipv4) / next header (ipv6)>	Integer type	
	0 … 255	The value range is from 0 to 255
<local port range>	String type	
	The string is given as dot-separated numeric (0-65535)	
<remote port range>	String type	
	The string is given as dot-separated numeric (0-65535)	
<ipsec security parameter index>	Integer type	
	Numeric value in hexadecimal format	

<type of service (ipv4) and mask / traffic class (ipv6) and mask>	String type	
	The string is given as dot-separated numeric (0-255)	
<flow label (ipv6)>	Integer type	
	Numeric value in hexadecimal format	
<direction>	Integer type	
	Specifies the transmission direction in which the PF is applied.	
	0	Prerelease 7 TFT filter
	1	Uplink
	2	Downlink
	3	Directional (Up & downlink)

Example

```
AT+CGTFT=5,2,6,"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",,168.252,,1
OK
```

3.2.23. AT+CSODCP sending of originating data through the control plane

The Set command is used by the TE to transmit data over control plane to network through 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:

- is completed with this uplink data transfer; or
- is completed with the next received downlink data.

This command also optionally indicates whether the data to be transmitted is an exception data.

This command causes transmission of an ESM DATA TRANSPORT message, as defined in Non-Access-Stratum (NAS) Protocol for Evolved Packet System (EPS); Stage 3 (3GPP TS 24.301).

Table 3.25: AT+CSODCP

AT+CSODCP	Response
Set command AT+CSODCP=<cid>,<cpdata_length>,<cpdata>[,<RAI>[,<type_of_user_data>]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CSODCP=?	Response +CSODCP: (range of supported <cid>s),(maximum number of octets of user data indicated by <cpdata_length>),(list of supported <RAI>s),(list of supported <type_of_user_data>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<cpdata_length>	Integer type	
	Indicates the number of octets of the <cpdata> information element. When there is no data to transmit, the value is set to zero.	
	0 … 950	Number of octets of <cpdata>
<cpdata>	String type	
	String of octets. Contains the user data container contents. When there is no data to transmit, the <cpdata> is an empty string ("").	
<RAI>	Integer type	
	Indicates the value of the release assistance indication.	
	0	No information available
	1	The MT expects that exchange of data is completed with the transmission of the ESM DATA TRANSPORT message.
	2	The MT expects that exchange of data is 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

Example

```
AT+CSODCP=0,20,"A1B2C3E4F50011223344A1B2C3E4F50011223344",0,0
OK
```

3.2.24. AT+CRTDCP reporting of terminating data through the control plane

The Set command is used to enable and disable reporting of data from the network to the MT that is transmitted through the control plane in downlink direction. If reporting is enabled, the MT returns the unsolicited result code +CRTDCP: <cid>,<cpdata_length>,<cpdata> when data is received from the network.

Table 3.26: AT+CRTDCP

AT+CRTDCP	Response
Set command AT+CRTDCP=[<reporting>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CRTDCP?	Response +CRTDCP: <reporting> OK If there is an error, the response is as follows: +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
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<reporting>	Integer type	
	Controlling reporting of mobile terminated control plane data events.	
	0	Disable reporting of MT control plane data.
	1	Enable reporting of MT control plane data by the unsolicited result code +CRTDCP.
<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<cpdata_length>	Integer type	
	Indicates the number of octets of the <cpdata> information element. When there is no data to transmit, the value is set to zero.	
	0 … 950	Number of octets of <cpdata>
<cpdata>	String type	
	String of octets. Contains the user data container contents. When there is no data to transmit, the <cpdata> is an empty string ("").	

Example

```
AT+CRTDCP=0
OK

AT+CRTDCP?
+CRTDCP: 0
OK

AT+CRTDCP=1
OK

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

3.2.25. AT+CGAPNRC APN rate control

This Execution command returns the APN rate control parameters (see Mobile Radio Interface Layer 3 Specification; Core Network Protocols; Stage 3 (3GPP TS 24.008) [8]) associated to the provided context identifier <cid>. If the parameter <cid> is omitted, the APN rate control parameters for all active PDP contexts are returned.

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

Table 3.27: AT+CGAPNRC

AT+CGAPNRC	Response
Set command AT+CGAPNRC[=<cid>]	Response +CGAPNRC: <cid>[,<Additional_exception_reports>[,<Uplink_time_unit>[,<Maximum_uplink_rate>]]] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGAPNRC=?	Response +CGAPNRC: (list of <cid>s associated with active contexts) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<Additional_exception_reports>	Integer type	
	Indicates whether additional exception reports are allowed to be sent when the maximum uplink rate is reached. This refers to bit 4 of octet 1 of the APN rate control parameters IE as specified in Mobile Radio Interface Layer 3 Specification; Core Network Protocols; Stage 3 (3GPP TS 24.008) [8] subclause 10.5.6.3.2.	
	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	
	Specifies the time unit to be used for the maximum uplink rate. This refers to bits 1 to 3 of octet 1 of the APN rate control parameters IE as specified in Mobile Radio Interface Layer 3 Specification; Core Network Protocols; Stage 3 (3GPP TS 24.008) [8] subclause 10.5.6.3.2.	
	0	Unrestricted
	1	Minute
	2	Hour
	3	Day
	4	Week
<Maximum_uplink_rate>	Integer type	
	Specifies the maximum number of messages the UE is restricted to send per uplink time unit. This refers to octet 2 to 4 of the APN rate control parameters IE as specified in Mobile Radio Interface Layer 3 Specification; Core Network Protocols; Stage 3 (3GPP TS 24.008) [8] subclause 10.5.6.3.2.	

Example

```
AT+CGAPNRC=?
+CGAPNRC: (5)
OK
```

3.2.26. AT+CGEREP Packet domain event reporting

Set command enables or disables sending of unsolicited result codes, +CGEV: XXX from MT to TE in the case of certain events occurring in the Packet Domain MT or the network. <mode> controls the processing of unsolicited result codes specified within this command. <bfr> controls the effect on buffered codes when <mode> 1 or 2 is entered. If a setting is not supported by the MT, ERROR or +CME ERROR: is returned.

Read command returns the current mode and buffer settings.

Test command returns the modes and buffer settings supported by the MT as compound values.

Table 3.28: AT+CGEREP

AT+CGEREP	Response
Set command AT+CGEREP=mode[,<bfr>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CGEREP?	Response +CGEREP: <mode>,<bfr> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGEREP=?	Response +CGEREP: (list of supported <mode>s), (list of supported <bfr>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type	
	0	Buffer unsolicited result codes in the MT. If MT result code buffer is full, the oldest ones can be discarded. No codes are forwarded to the TE.
	1	Discard unsolicited result codes when MT-TE link is reserved (for example, in on-line data mode); otherwise forward them directly to the TE.
<bfr>	Integer type	
	0	MT buffer of unsolicited result codes defined within this command is cleared when <mode> = 1.

Example

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

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

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

3.2.27. +CGEV used to indicate EPS PDN connection and bearer resources operations status

This is an unsolicited message to indicate EPS PDN connection and bearer resources operations status.

Table 3.29: +CGEV

Message	Status
+CGEV: NW PDN DEACT <cid>	The network has forced a context deactivation.
+CGEV: ME PDN DEACT <cid>	The mobile termination has forced a context deactivation.
+CGEV: ME PDN ACT <cid> [,<pdnReason>]	The ME has activated a context.
+CGEV: NW MODIFY <cid>, <change_reason>,<event_type>	The network has modified a context.
+CGEV: ME MODIFY <cid>, <change_reason>,<event_type>	The mobile termination has modified a context.

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<pdnReason>	Integer type	
	0	IPV4 only allowed
	1	IPV6 only allowed
	2	Single address bearer only allowed
	3	
	4	Single address bearer only allowed and active second bearer failed
<bearerType>	Integer type	
	0	NULL
	1	Default
	2	Dedicated
<change_reason>	Integer type	
	A bit map that indicates what kind of change occurred. The value is determined by summing all the applicable bits.	
	Bit 1	TFT changed
	Bit 2	Qos changed
	Bit 3	WLAN Offload changed

Example

```
+CGEV: ME PDN ACT 5,0
```

3.2.28. AT+CGPADDR show PDP addresses

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.

Table 3.30: AT+CGPADDR

AT+CGAPNRC	Response
Set command AT+CGPADDR[=<cid>]	Response +CGPADDR: <cid>[,<PDP_addr_1>[,<PDP_addr_2>]] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGPADDR=?	Response +CGPADDR: (list of defined <cid>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<PDP_addr_1>	String type	
	Identifies the MT in the address space applicable to the PDP. Contains the IPv4 address. Omitted if not available.	
	The string is given as dot-separated numeric (0-255) parameter	
<PDP_addr_2>	String type	
	Identifies the MT in the address space applicable to the PDP. Contains the IPv6 address. Omitted if not available.	
	The string is given as dot-separated numeric (0-255) parameter	

Example

```
AT+CGPADDR
+CGPADDR: 5,"100.120.44.90"
OK
```

3.2.29. AT+CSCON signaling connection status

The Set command controls the presentation of an unsolicited result code +CSCON. If <n>=1, +CSCON: <mode> is sent from the MT when the connection mode of the MT is changed.

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

Test command returns supported values as a compound value.

Table 3.31: AT+CSCON

AT+CSCON	Response
Set command AT+CSCON=<n>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CSCON?	Response +CSCON: <n>,<mode> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CSCON=?	Response +CSCON: (list of supported <n>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<n>	Integer type	
	0	Disable unsolicited result code
	1	Enable unsolicited result code +CSCON:<mode>
<mode>	Integer type	
	Indicates the signaling connection status.	
	0	Idle
	1	Connected

Example

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

AT+CSCON=1
OK

AT+CSCON?
+CSCON: 1,0
OK

AT+CSCON=0
OK

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

3.2.30. AT+CCLK return current date and time

Set command sets the real-time clock of the MT.

The Read command returns the current setting of the clock.

Table 3.32: AT+CCLK

AT+CCLK	Response
Set command AT+CCLK=<n>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CCLK?	Response +CCLK: <n>,<mode> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CCLK=?	Response +CCLK: (list of supported <n>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	String type
<time>	The format is "yyyy/mm/dd, hh:mm:ss±zz", where characters indicate year, 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 -47 ~ +48). For instance, 6th of May 2014, 22:10:00 GMT+2 hours equal "2014/05/06,22:10:00+08".

Example

```
AT+CCLK="2018/07/25,02:22:22+00"
OK

AT+CCLK?
+CCLK: "2018/07/25,02:22:30+00"
OK

AT+CCLK=?
OK
```

3.2.31. AT+CIMI request international mobile subscriber identity

Execution command causes the TA to return <IMSI>, which is intended to permit the TE to identify the individual SIM card or active application in the UICC which is attached to MT.

Table 3.33: AT+CIMI

AT+CIMI	Response
Set command AT+CIMI	Response <IMSI> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CIMI=?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

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

Example

```
AT+CIMI=?
OK

AT+CIMI
460043263600043
OK
```

3.2.32. AT+CPIN enter PIN

Set command sends a password to the MT, which is necessary before it can be operated (SIM PIN, SIM PUK, etc.). If the PIN is to be entered twice, the TA automatically repeats the PIN. If no PIN request is pending, no action is taken towards MT and an error message, +CME ERROR, is returned to TE. If the PIN required is SIM PUK or SIM PUK2, the second pin is required. This second pin, <newpin>, is used to replace the old pin in the SIM.

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

Table 3.34: AT+CPIN

AT+CPIN	Response
Set command AT+CPIN=<pin>[,<newpin>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CPIN?	Response +CPIN: <code> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CPIN=?	Response +CCLK: (list of supported <n>s) OK

Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<pin>	String type	
	Password if entered alone. Old pin if entered alongside <newpin>	
<newpin>	String type	
	New pin to replace <pin>	
<code>	String type	
	READY	MT is not pending for any password
	SIM PIN	MT is waiting SIM PIN to be given
	SIM PUK	MT is waiting SIM PUK to be given

Example

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

3.2.33. AT+CLCK facility lock

Execute command is used to lock, unlock, or interrogate a MT or a network facility <fac>. Password is normally needed to do such actions. When querying the status of a network service (<mode>=2) the response line for 'not active' case (<status>=0) must be returned only if service is not active for any <class>. This command must be abortable when network facilities are set or interrogated.

Test command returns facility values supported as a compound value.

Table 3.35: AT+CLCK

AT+CLCK	Response
Set command AT+CLCK=<fac>,<mode>[,<passwd>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err> When <mode>=2 and command successful: +CLCK: <status>
Test command AT+CLCK=?	Response +CLCK: (list of supported <fac>s) OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fac>	String type	
	"SC"	SIM (lock SIM/UICC card installed in the currently selected card slot) (SIM/UICC asks password in MT power-up and when this lock command issued)

<mode>	Integer type	
	0	Unlock
	1	Lock
	2	Query status
<status>	Integer type	
	0	Not active
	1	Active
<passwd>	String type	
	Is the same as password specified for the facility from the MT user interface or with command Change Password +CPWD	

Example

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

3.2.34. AT+CPWD change password

Command sets a new password for the facility lock function defined by command Facility Lock +CLCK. Test command returns a list of pairs which present the available facilities and the maximum length of their password.

Table 3.36: AT+CPWD

AT+CPWD	Response
Set command AT+CPWD=<fac>,<oldpwd>,<newpwd>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CPWD=?	Response +CPWD: (list of supported <fac>,<pwdlength>s) OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fac>	String type	
	"SC"	SIM (lock SIM/UICC card installed in the currently selected card slot) (SIM/UICC asks password in MT power-up and when this lock command issued)
<oldpwd>	String type	
	Is the same as password specified for the facility from the MT user interface or with command Change Password +CPWD.	
<newpwd>	String type	
	Is the new password, maximum length of password can be determined with <pwdlength>	
<pwdlength>	String type	
	Maximum length of the password for the facility	

Example

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

3.2.35. AT+CSIM generic SIM access

Set command transmits to the MT the <command> it then sends as it is to the SIM. In the same manner, the SIM <response> is 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 a distant application on the TE. The TE then takes care of processing SIM information within the frame specified by GSM/UMTS.

Table 3.37: AT+CSIM

AT+CSIM	Response
Set command AT+CSIM=<length>,<command>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CSIM=?	Response +CSIM: (list of supported <fac>,<pwdlength>s) OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

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>	String type
	Command passed on by the MT to the SIM in the format as described in Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface (3GPP TS 51.011) [28] (hexadecimal character format)
<response>	String type
	Response to the command passed on by the SIM to the MT in the format as described in Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface (3GPP TS 51.011) [28] (hexadecimal character format)

Example

```
AT+CSIM=?
OK

AT+CSIM=14,"00A4000C023F00"
+CSIM: 4, "9000"
OK
```

3.2.36. AT+CRSM restricted SIM

By using this command instead of Generic SIM Access +CSIM TE application has easier but more limited access to the SIM database. Set 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 must be aware of the precedence of the GSM/UMTS application commands to the TE commands.

Table 3.38: AT+CRSM

AT+CRSM	Response
Set command AT+CRSM=<command>[,<fileid> [,<P1>,<P2>,<P3>[,<data>[,<pathid>]]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CRSM=?	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<command>	Integer type	
	Command passed on by the MT to the SIM; refer Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) Interface (3GPP TS 51.011) [28]	
	176	READ BINARY
	178	READ RECORD
	192	GET RESPONSE
	214	UPDATE BINARY
	220	UPDATE RECORD
	242	STATUS
<fileid>	Integer type	
	This is the identifier of an elementary datafile on SIM. Mandatory for every command except STATUS. The range of valid file identifiers depends on the actual SIM and is defined in Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) Interface (3GPP TS 51.011) [28]. Optional files may not be present at all.	
<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 Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) Interface (3GPP TS 51.011) [28].	
<data>	String type	
	Information that is written to the SIM (hexadecimal character format).	

<pathid>	String type
	Contains the path of an elementary file on the SIM/UICC in hexadecimal format as defined in ETSI TS 102 221 [60] (for example, "7F205F70" in SIM and UICC case). The <pathid> is only 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>	String type
	Response of a successful completion of the command previously issued (hexadecimal character format). STATUS and GET RESPONSE return data, which gives information about the current elementary data field. This information includes the type of file and its size (refer Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface (3GPP TS 51.011) [28]). After READ BINARY, READ RECORD or RETRIEVE DATA command the requested data is returned. <response> is not returned after a successful UPDATE BINARY, UPDATE RECORD or SET DATA command.

Example

```
AT+CRSM=176,28423,0,0,18
+CRSM: 144, 0, "08490660"
OK
```

3.2.37. AT+CCHO open logical channel

Execution of the command causes the MT 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 opens 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 restricts 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 or Generic UICC Logical Channel access +CGLA commands.

Table 3.39: AT+CCHO

AT+CCHO	Response
Set command AT+CCHO=<dfname>	Response <sessionid> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CCHO=?	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s

Parameter Saving Mode	NO_SAVE

Parameter

<dfname>	String type
	All selectable applications in the UICC are referenced by a DF name coded on 1 to 16 bytes.
<sessionid>	Integer type
	A session ID to be used to target a specific application on the Smart Card (for example (U)SIM, WIM, ISIM) using logical channels mechanism.

Example

```
AT+CCHO="A0000000437450617373E496F54"
1
OK
```

3.2.38. AT+CCHC close logical channel

This command asks the ME to close a communication session with the active UICC. The ME closes the previously opened logical channel. The TE can no longer be able to send commands on this logical channel. The UICC closes the logical channel when receiving this command.

Table 3.40: AT+CCHC

AT+CCHC	Response
Set command AT+CCHC=<dfname>	Response +CCHC OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CCHC=?	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<sessionid>	Integer type
	A session ID to be used to target a specific application on the Smart Card (for example (U)SIM, WIM, ISIM) using logical channels mechanism.

Example

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

3.2.39. AT+CGLA generic UICC logical channel access

Set command transmits to the MT the <command> it then sends as it is to the selected UICC. In the same manner the UICC <response> is 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 then takes care of processing UICC information within the frame specified by GSM/UMTS.

Although the Generic UICC Logical Channel Access command +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 must not be handled outside the TA/MT. Therefore, it is not allowed to execute a Run GSM Algorithm command or an Authenticate command in GSM context from the TE using +CGLA at all-time whether the +CGLA is locked or unlocked. This does not forbid the TE to send Authenticate commands in other security contexts (for example, EAP security context).
For example, the TA/MT forbids the transfer of the Authenticate command to a USIM application when parameters P2 = 0 (GSM security context). See Characteristics of The Universal Subscriber Identity Module (USIM) Application (3GPP TS 31.102) for USIM authenticate command definition.

Table 3.41: AT+CGLA

AT+CGLA	Response
Set command AT+CGLA=<sessionid>,<length>,<command>	Response +CGLA: <length>,<response> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGLA=?	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<sessionid>	Integer type
	A session ID to be used to target a specific application on the Smart Card (for example (U)SIM, WIM, ISIM) using logical channels mechanism.
<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>	String type
	A session ID to be used to target a specific application on the Smart Card (for example (U)SIM, WIM, ISIM) using logical channels mechanism.
<response>	String type
	Response to the command passed on by the UICC to the MT in the format as described in UICC-Terminal Interface; Physical and Logical Characteristics 3GPP TS 31.101 (hexadecimal character format)

Example

```
AT+CGLA=1,38,"81F100000E0051010A11223344556677889900"
+CGLA: 42,"009868119122009036664601FA483D3E000000253D165EF45278745B0F436
5C32CBE23CEEBB331839000"
OK
```

3.2.40. AT+CTZU automatic time zone update

Set command enables and disables automatic time zone update through NITZ. If setting fails in an MT error , `+CME ERROR: <err>` is returned.

Read command returns the current settings in the MT. Test command returns supported on- and off-values as a compound value.

Table 3.42: AT+CTZU

AT+CTZU	Response

Set command
`AT+CTZU=<onoff>`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Read command
AT+CTZU?
	Response
	`+CTZU: <onoff>`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Test command `AT+CTZU=?`	Response `+CTZU:` (list of supported `<onoff>`s) `OK`
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	Integer type	
<onoff>	0	Disable automatic time zone update through NITZ.
	1	Enable automatic time zone update through NITZ. Default value.

Example

```
AT+CTZU=1
OK

AT+CTZU?
+CTZU: 1
OK

AT+CTZU=?
+CTZU: (0-1)
OK
```

3.2.41. AT+CTZR time zone reporting

This Set command controls the time zone change event reporting. If reporting is enabled, the MT returns the unsolicited result code `+CTZV: <tz>`, `+CTZE: <tz>,<dst>,[<time>]`, or `+CTZEU: <tz>,<dst>,[<utime>]` whenever the time zone is changed. The MT also provides the time zone upon network registration if provided by the network. If setting fails in an MT error, `+CME ERROR: <err>` is returned.

Read command returns the current reporting settings in the MT.

Test command returns supported `<reporting>` values as a compound value.

Table 3.43: AT+CTZR

AT+CTZR	Response

Set command
`AT+CTZR=<reporting>`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Read command
AT+CTZR?
	Response
	`+CTZR: <reporting>`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Test command `AT+CTZR=?`	Response `+CTZR:` (list of supported `<reporting>`s) `OK`
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<reporting>	Integer type	
	0	Disable time zone change event reporting. Default value.
	1	Enable time zone change event reporting by unsolicited result code `+CTZV: <tz>`
	2	Enable extended time zone and local time reporting by unsolicited result code `+CTZE: <tz>,<dst>,[<time>]`
	3	Enable extended time zone and universal time reporting by unsolicited result code `+CTZEU: <tz>,<dst>,[<utime>]`
<tz>	String type	
	Represent 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 to +56. To maintain a fixed width, numbers in the range -9 to +9 are expressed with a leading zero, for example, "-09", "+00" and "+09".	
<dst>	Integer type	
	Indicate 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	
	Value representing 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 MT from information provided by the network at the time of delivering time zone information and is present in the unsolicited result code for extended time zone and local time reporting if the universal time is provided by the network.	
<utime>	String type	
	Value representing 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 is present in the unsolicited result code for extended time zone and universal time reporting if provided by the network.	

Example

```
AT+CTZR=3
OK

AT+CTZR?
+CTZR: 3
OK

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

3.2.42. AT+CRCES reading coverage enhancement status

This command returns the coverage enhancement status of the MT. The terminal can consider the coverage enhancement status prior to deciding to transmit data. Depending on the coverage enhancement status, the terminal can refrain from transmitting data. The coverage enhancement status is only provided by the MT, if the access technology of the serving cell is E-UTRAN, EC-GSM-IoT, or E-UTRAN (NB-S1 mode). If the access technology of the serving cell is different, <Act>=0 is indicated.

Table 3.44: AT+CTZR

AT+CTZR	Response
Execution command AT+CRCES	Response +CRCES: <AcT>,<CE_level>,<CC> OK
Test command AT+CRCES=?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<AcT>	Integer type	
	Access technology of the serving cell.	
	0	Location disclosure allowed
	1	E-UTRAN
	2	EC-GSM-IoT (A/Gb mode)
	3	E-UTRAN (NB-S1 mode)
<CE_level>	Integer type	
	Coverage Enhancement (CE) level of the MT in the serving cell. Applicable only if <Act>=1 (E-UTRAN) or <Act>=3 (E-UTRAN (NB-S1 mode)).	
	0	No Coverage Enhancement in the serving cell
	1	Coverage Enhancement level 0
	2	Coverage Enhancement level 1
	3	Coverage Enhancement level 2
	4	Coverage Enhancement level 3
<cc>	Integer type	
	Coverage Class (CC) of the MT in the serving cell. Applicable only if <Act>=2 (EC- SMIoT).	
	0	No Coverage Enhancement in the serving cell
	1	Coverage Class 1
	2	Coverage Class 2
	3	Coverage Class 3
	4	Coverage Class 4
	5	Coverage Class 5

Example

```
AT+CRCES
+CRCES: 3,1,0
OK

AT+CRCES
+CRCES: 3,1,0
OK
```

3.2.43. AT+CIPCA initial PDN context activation

The Set command controls whether an initial PDP context is established automatically following an attach procedure when the UE is attached to GERAN or UTRAN RATs and whether the UE is attached to E-UTRAN with or without a PDN connection.

The Read command returns the current setting of the command.

The Test command returns values supported as a compound value.

Table 3.45: AT+CIPCA

AT+CIPCA	Response
Set command AT+CIPCA=[<n>[,<AttachWithoutPDN>]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CIPCA?	Response +CIPCA: <n>[,<AttachWithoutPDN>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CTZR=?	Response +CIPCA: (list of supported <n>s), (list of supported <AttachWithoutPDN>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<n>	Integer type	
	Activation of PDP context upon attach.	
	0	Do not activate
	1	Always activate
	2	Activate when not roaming
	3	No change in current setting
<AttachWithoutPDN>	Integer type	
	EPS Attach with or without PDN connection.	
	0	EPS Attach with PDN connection
	1	EPS Attach without PDN connection

Example

```
AT+CIPCA=3,0
OK

AT+CIPCA?
+CIPCA: 3,0
OK

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

3.2.44. AT+CGMI request manufacturer identification

Execution command causes the TA to return one or more lines of information text <manufacturer>, determined by the MT manufacturer, which is intended to permit the user of the TA to identify the manufacturer of the MT to which it is connected to. Typically, the text consists of a single line containing the name of the manufacturer, but manufacturers may choose to provide more information if required.

Table 3.46: AT+CGMI

AT+CGMI	Response
Execution command AT+CGMI	Response +CGMI: <manufacturer> OK
Test command AT+CGMI=?	Response +CGMI: <manufacturer_ID> OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Example

```
AT+CGMI
+CGMI: "qualcomm"
OK

AT+CGMI=?
+CGMI: "manufacturer_ID"
OK
```

3.2.45. AT+CGMM request model identification

Execution command causes the TA to return one or more lines of information text <model>, determined by the MT manufacturer, which is intended to permit the user of the TA to identify the specific model of the MT to which it is connected to. Typically, the text consists of a single line containing the name of the product, but manufacturers may choose to provide more information, if required.

Table 3.47: AT+CGMM

AT+CGMM	Response
Execution command AT+CGMM	Response +CGMM: <model> OK
Test command AT+CGMM=?	Response +CGMM: <model> OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Example

```
AT+CGMM
+CGMM: "qualcomm","QCX212"
OK

AT+CGMM=?
+CGMM:<list of supported technologies>,<model>
OK
```

3.2.46. AT+CPINR remaining PIN retries

Execution command and Set command cause the MT to return the number of remaining PIN retries for the MT passwords with intermediate result code `+CPINR: <code>,<retries>[,<default_retries>]` for standard PINs.

Table 3.48: AT+CPINR

AT+CPINR	Response

Set command
`AT+CPINR=<code>`
	Response
	`+CPINR: <code>,<retries>[,<default_retries>]`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Execution command
AT+CPINR
	Response
	`+CPINR: <code>,<retries>[,<default_retries>]`
	`+CPINR: <code>,<retries>[,<default_retries>]`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Test command `AT+CPINR=?`	Response `OK`
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

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

Example

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

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

3.2.47. AT+CGAUTH define PDP context authentication parameters

The set command allows the terminal equipment (TE) to specify authentication parameters for a packet data protocol (PDP) context identified by the (local) context identification parameter <cid>. The <cid> parameter is used during the PDP context activation and the PDP context modification procedures. As the <cid> is the

same parameter that is used in +CGDCONT and +CGDSCONT commands, the +CGAUTH command is effectively an extension to these commands.

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

The test command returns the values supported as compound values.

Table 3.49: AT+CGAUTH

AT+CGAUTH	Response
Set command AT+CGAUTH=<cid>[,<auth_proto>[,<userid>[,<password>]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CGAUTH?	Response [+CGAUTH: <cid>,<auth_proto>,<userid>,<password>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CGAUTH=?	Response +CGAUTH: (range of supported <cid>s), (list of supported <auth_proto>s), (range of supported <userid>s), (range of supported <password>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<auth_proto>	Integer type	
	0	Indicates that no authentication protocol is used for this PDP. Default value.
	1	PAP
<userid>	String type	
	Indicates the userid string. The maximum length is 20 characters with 1 line end mark. The default value is “”(NULL).	
<password>	String type	
	Indicates the password string. The maximum length is 20 characters with 1 line end mark. The default value is “”(NULL).	

Example

```
AT+CGAUTH=?
+CGAUTH: (0-10),(0-1),(20),(20)
OK

AT+CGAUTH?
+CGAUTH: 0,0,””,””
OK
```

```
AT+CGDCONT=1,IP
OK

AT+CGAUTH=1,1,"userid","psw"
OK
```

3.3. 3GPP COMMANDS (27.005)

3.3.1. AT+CMGS send message

Execution command sends message from a TE to the network (SMS-SUBMIT). Message reference value <mr> is returned to the TE on successful message delivery. Value can be used to identify message upon unsolicited delivery status report result code. If sending fails in a network or an ME error, final result code +CMS ERROR: <err> is returned. This command must be abortable.

For text mode:

- The Entered text Technical Realization of the Short Message Service (SMS) (3GPP TS 23.040) [3] (TP-Data-Unit) is sent to address <da> and all current settings (refer Set Text Mode Parameters +CSMP and Service Centre Address +CSCA) are used to construct the actual PDU in ME/TA
- TA sends a four-character sequence <CR><LF><greater_than><space> (IRA 13, 10, 62, 32) after command line is terminated with <CR>. After that text can be entered from TE to ME/TA.
- DCD signal is in ON state while text is entered.
- Echoing of entered characters back from the TA is controlled by V.25 echo command E.
- The entered text must be formatted as follows:
    - If <dcs> (set with +CSMP) indicates that Alphabets and Language-Specific Information (3GPP TS 23.038) [2] GSM 7-bit default alphabet is used and <fo> indicates that Technical Realization of the Short Message Service (SMS) (3GPP TS 23.040) [3] TP-User-Data-Header-Indication is not set.
    - If TE character set other than "HEX" (refer command Select TE Character Set +CSCS in AT Command Set for User Equipment (UE) (3GPP TS 27.007) [9]): ME/TA converts the entered text into the GSM 7-bit default alphabet according to rules of Annex A. Backspace can be used to delete last character and carriage returns can be used (previously mentioned four character sequence is sent to the TE after every carriage return entered by the user).
    - If TE character set is "HEX": the entered text must consist of two IRA character long hexadecimal numbers which ME/TA converts into the GSM 7-bit default alphabet characters (for example, 17 (IRA 49 and 55) is converted to character Π (GSM 7-bit default alphabet 23)).
    - If <dcs> indicates that 8-bit or UCS2 data coding scheme is used or <fo> indicates that Technical Realization of the Short Message Service (SMS) (3GPP TS 23.040) [3] TP-User-Data-Header-Indication is set: the entered text must consist of two IRA character long hexadecimal numbers which ME/TA converts into 8-bit octet (for example, two characters 2A (IRA 50 and 65) is converted to an octet with integer value 42).
- Sending can be cancelled by giving <ESC> character (IRA 27).
- <ctrl-z> (IRA 26) must be used to indicate the ending of the message body.

For PDU mode:

- <length> must indicate the number of octets coded in the TP layer data unit to be given (that is, SMSC address octets are excluded).
- TA sends 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 ME/TA.

- DCD signal is in ON state while PDU is given.
- The echoing of given characters back from the TA is controlled by V.25ter echo command E.
- PDU is hexadecimal format (similarly as specified for <pdu>) and given in one line. ME/TA 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 command Service Centre Address +CSCA is used. In this case the SMSC Type-of-Address octet is not present in the PDU, that is, TPDU starts right after SMSC length octet.
- Sending can be cancelled by giving <ESC> character (IRA 27).
- <ctrl-z> (IRA 26) must be used to indicate the ending of PDU.

Table 3.50: AT+CMGS

AT+CGAUTH	Response
Set command If text mode (AT+CMGF=1): AT+CMGS=<da>[,<toda>]<CR> Text is entered <ctrl-z/ESC> If PDU mode(AT+CMGF=0): **AT+CMGS=<length><CR>** PDU is given <ctrl-z/ESC>	Response +CMGS: <mr> If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	60 s
Parameter Saving Mode	NO_SAVE

Parameter

<da>	String type
	In text mode (AT+CMGF=1): Destination address.
<toda>	Integer type
	Type of destination address.
<length>	Integer type
	In PDU mode (AT+CMGF=0): the length of the actual TP data unit in octets (that is, the RP layer SMSC address octets are not counted in the length). The range is 7-220.
<mr>	Integer type
	Technical Realization of the Short Message Service (SMS) (3GPP TS 23.040) [3] TP-Message-Reference.

Example

```
AT+CMGF=1
OK

AT+CMGS="1064899990000"
>TEST
CTRL+Z(1a(hex))
+CMGS: 1
OK
```

3.3.2. AT+CSCA service center address

Set 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.

Table 3.51: AT+CSCA

AT+CSCA	Response
Set command AT+CSCA=<sca>[,<tosca>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CSCA?	Response +CSCA: <sca>,<tosca> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<sca>	String type
	Point-to-Point (PP) Short Message Service (SMS) Support on Mobile Radio Interface (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.
<tosca>	Integer type
	Point-to-Point (PP) Short Message Service (SMS) Support on Mobile Radio Interface (3GPP TS 24.011) [6] RP SC address Type-of-Address octet in integer format (when first character of <da> is + (IRA 43), default value is 145, otherwise default value is 129).

Example

```
AT+CSCA=”8613800200569”
OK

AT+CSCA?
+CSCA: “8613800200569”,129
OK
```

3.3.3. AT+CMGF message format

Set command tells the TA which input and output format of messages to use. <mode> indicates the format of messages used with send, list, read and write commands and unsolicited result codes resulting from received messages. Mode can be either PDU mode (entire TP data units used) or text mode (headers and body of the messages given as separate parameters).

Test command returns supported modes as a compound value.

Table 3.52: AT+CMGF

AT+CMGF	Response
Set command AT+CMGF=<mode>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CMGF?	Response +CMGF: <mode> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+CMGF=?	Response +CMGF: (list of supported <mode>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	Integer type	
<mode>	0	PDU mode
	1	Text mode

Example

```
AT+CMGF=1
OK

AT+CMGF?
+CMGF: 1
OK
```

3.3.4. AT+CSMP set text mode parameters

Set command is used to select values for additional parameters needed when SM is sent to the network or placed in a storage when text format message mode is selected. It is possible to set the validity period starting from when the SM is received by the SMSC (<vp> is in range 0... 255) or define the absolute time of the validity period termination (<vp> is a string). The format of <vp> is given by <fo>. If TA supports the EVPF, see Technical Realization of the Short Message Service (SMS) (3GPP TS 23.040) [3], it is given as a hexadecimal coded string (refer for example, <pdu>) with double quotes.

Table 3.53: AT+CSMP

AT+CSMP	Response
Set command AT+CSMP=<fo>[,<vp>[,<pid>[,<dcs>]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CSMP?	Response +CSMP: <fo>,<vp>,<pid>,<dcs> OK

Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<fo>	Integer type
	First octet for SMS submits PDU, status report required.
<vp>	Integer type
	Validity period.
<pid>	Integer type
	Protocol identifier.
<dcs>	Integer type
	Data coding scheme.

Example

```
AT+CSMP=33,167,0,0
OK

AT+CSMP?
+CSMP: 33,167,0,0
OK
```

3.3.5. +CMT new message received

SMS-DELIVERs are routed directly to the TE using unsolicited result code.

Table 3.54: +CMT

Message	Context
+CMT: <length><CR><LF><pdu>	PDU mode enabled
+CMT: <oa>,<scts><CR><LF><data>	Text mode enabled

Parameter

<length>	Integer type
	Length of PDU (PDU mode enabled)
	Deliver Message's source address (text mode enabled)
<oa>	String type
	Deliver Message's source address (text mode enabled)
<scts>	String type
	TP-Service-Centre-Timestamp in time-string format. Technical Realization of the Short Message Service (SMS). (3GPP TS 23.040) [3] TPDischarge-Time in time-string format: "yy/MM/dd,hh:mm:ss±zz", where characters indicate year (two last digits), month, day, hour, minutes, seconds, and time zone. Example: 6th of May 1994, 22:10:00 GMT+2 hours equal to "94/05/06,22:10:00+08".

Example

```
+CMT: ”106499990000”,”19.05.16 16:27:55 GMT:+8” hello
```

4. EXTENDED COMMANDS

4.1. QTI GENERAL COMMANDS

4.1.1. AT$QCBAND

This command sets the network mode and bands to be used.

Read command returns the current network mode and band list.

Test command returns network mode and bands supported by the UE.

Table 4.1: AT$QCBAND

AT$QCBAND	Response
Set command AT$QCBAND=<mode>[,<band1>[,<band2>…]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCBAND?	Response $QCBAND: <mode>,<band1>,<band2>… OK
Test command AT$QCBAND=?	Response $QCBAND: (list of supported <mode>s), (list of supported <band>s) OK
Maximum Response Time	25 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<mode>	Integer type	
	0	NB-IOT mode
<band>	Integer type	
	Band list in decimal number.	

Example

```
AT$QCBAND?
$QCBAND: 0,5,8,1,3
OK

AT$QCBAND=?
$QCBAND: (0),(1,3,5,8)
OK

AT$QCBAND=0,5,8
OK
```

4.1.2. AT$QCLOGDBVER

This Read command returns current unilog database version information.

Table 4.2: AT$QCLOGDBVER

AT$QCLOGDBVER	Response
Read command AT$QCLOGDBVER?	Response $QCLOGDBVER: <LogDbVserion> OK
Maximum Response Time	25 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<LogDbVserion>	Integer type
	Current unilog database version.

Example

```
AT$QCLOGDBVER?
$QCLOGDBVER: 0x0a1b2c3d
OK
```

4.1.3. AT$QCCFG

This command sets the UE extended configuration.

The Read command return current setting of each parameter.

The Test command returns values supported as a compound value.

Table 4.3: AT$QCCFG

AT$QCLOGDBVER	Response
Set command AT$QCCFG=<param1>,<value1>[,<param2>,<value2>[,<param3>,<value3>,[…]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCCFG?	Response $QCCFG: <param1>,<value1>,<param2>,<value2>…<paramN>,<valueN> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCCFG=?	Response $QCCFG: (list of supported <param>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

<param>	String type	
	Name of configuration parameter.	
	"AutoApn"	Whether UE auto set the attached APN according to the inserted SIM card. <value>=(0,1)
	"USIMPowerSave"	Enable/disable USIM power save. <value>=(0,1)
	"UsimSimulator"	Enable/disable virtual SIM card for instrument test, such as CMW500. <value>=(0,1)
	"Rohc"	Whether UE support ROHC. <value>=(0,1)
	"Ipv6RsForTestSim"	Whether UE trigger IPv6 NDP (RS) procedure to get IPv6 prefix address when the SIM card inserted is a TEST SIM. <value>=(0,1)
	"PowerCfun"	Default CFUN state after UE power-on or reboot. <value>=(0,1,4)
	"psPowerOnMaxDelay"	Max delay in seconds after power on, UE delays in a random value between 0 to max delay value. <value>=(0-65535)
	"SupportSms"	Whether UE support SMS. <value>=(0,1)
	"TauForSms"	Whether need to trigger TAU procedure, if UE support SMS capability, while NW does not support. <value>=(0,1)
	"PlmnSearchPowerLevel"	Set the PLMN search level when UE OOS. <value>=(0,1,2,3)
	"Epco"	Whether UE need to use "EPCO" in "PDN CONNECTION REQUEST" carried in "ATTACH REQUEST", and "ESM INFORMATION RESPONSE". If set to 0, use "PCO". <value>=(0,1)
	"T3324MaxValueS"	Set user control T3324 value in second. <value>=(0-0xFFFFFF)
	"MultiCarrier"	Whether UE support multi-carrier feature. <value>=(0,1)
	"MultiTone"	Whether UE support multi-tone feature. <value>=(0,1)
	"SupportUpRai"	Whether UE support L2 (MAC layer) RAI feature, which only valid whether set to R14 version. <value>=(0,1)
	"DataInactTimer"	Set the value of "data inactivity timer" in seconds, if this timer is not configured by NW (in MAC-MainConfig-NB), use this setting value. <value>=(0,40-254)
	"RelaxMonitorDeltaP"	Set the value of "SearchDeltaP" in DB for Relax-Monitor feature. If this value is not configured by NW (in SIB-NB), use this setting value. <value>=(0-15)
	"RelVersion"	Set the NB release version. <value>=(13,14)
	"DataInactMon_r14"	Whether UE support DataInactMon_r14. <value>=(0,1)
	"InterferenceRandomisation_r14"	Whether UE support InterferenceRandomisation_r14. <value>=(0,1)
	"SupportTwoHarq"	Whether UE support SupportTwoHarq. <value>=(0,1)

	"SupportMultiCarrierNPRACH"	Whether UE support SupportMultiCarrierNPRACH. <value>=(0,1)
	"SupportMultiCarrierPaging"	Whether UE support SupportMultiCarrierPaging. <value>=(0,1)
	"Cp-Reestablishment"	Whether UE support Cp-Reestablishment. <value>=(0,1)
<value>	Integer type	
	Value of configuration.	

Example

```
AT$QCCFG="Rohc",0
OK

AT$QCCFG?
$QCCFG: "AutoApn",0,"USIMPowerSave",1,"UsimSimulator",0,"Rohc",1,
"Ipv6RsForTestSim",0,"PowerCfun",1,"psPowerOnMaxDelay",0,"SupportSms",1,
"TauForSms",0,"PlmnSearchPowerLevel",1,"Epco",1,"T3324MaxValueS",
16777215,"MultiCarrier",1,"MultiTone",1,"SupportUpRai",0,
"DataInactTimer",60,"RelaxMonitorDeltaP",0,"RelVersion",13,
"CellSearchGuardTimer",1200,"DataInactMon_r14",0,
"InterferenceRandomisation_r14",0,"SupportTwoHarq",1,"SupportMultiCarrie
rNPRACH",0,"SupportMultiCarrierPaging",0,"CpReestablishment",0.
OK

AT$QCCFG=? ("AutoApn","USIMPowerSave","UsimSimulator","Rohc",
"Ipv6RsForTestSim","PowerCfun","psPowerOnMaxDelay","SupportSms",
"TauForSms","PlmnSearchPowerLevel","Epco","T3324MaxValueS",
"MultiCarrier","MultiTone","SupportUpRai","DataInactTimer",
"RelaxMonitorDeltaP","RelVersion",CellSearchGuardTimer,
"DataInactMon_r14","InterferenceRandomisation_r14","SupportTwoHarq",
"SupportMultiCarrierNPRACH","SupportMultiCarrierPaging",
"Cp-Reestablishment")
OK
```

4.1.4. AT$QCPING

This command sends an ICMP packet to the specified host address. AT$QCPING initiates the sending of a PING packet with payload size: <size> to the specified address. This either causes a packet to be returned if the remote system is connected and responding to PING packets or no response is received. If none of the response packet received within the timeout period <timeout>. It continues to send PING packet until the <count> number of times.

The Test command returns values supported as a compound value.

Table 4.4: AT$QCPING

AT$QCPING	Response
Set command AT$QCPING=[<ipaddr/Url>/<mode> [,<count>[,<size>[,<timeout>]]]]	Response OK If there is an error, the response is as follows: +SOCKET ERROR: <err>

Test command AT$QCPING=?	Response $QCCFG: (list of supported <param>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type	
	0	Stop ping
<ipaddr/Url>	String type	
	IP address or URL	
<count>	Integer type	
	Number of ping packages to send. Default is 4	
<size>	Integer type	
	Payload size. Default is 32.	
<timeout>	Integer type	
	UE ping reply timeout after ping request (in ms). Default is 20000.	

When one PING reply is received in <timeout>, an unsolicited result code: $QCPING: SUCC, dest: <dest_ip_addr>, RTT: <rtt_time> ms is sent to TE.

If no PING reply is received in <timeout>, an unsolicited result code: $QCPING: FAIL, dest: <dest_ip_addr>, time out: <timeout> ms is sent to TE.

If this is an ERROR meet during PING procedure, an unsolicited result code: $QCPING: ERROR, cause: <cause> is sent to TE.

When the PING procedure is performed, an unsolicited result code: $QCPING: DONE<CR><LF>$QCPING: dest: <dest_ip_addr>, <count> packets transmitted, <reply_count> received, <lost_percent>% packet loss<CR> rtt min/avg/max = <rtt_min> / <rtt_avg> / <rtt_max> ms is sent to TE.

Example

Ping 180.97.33.107 10 times with 32 bytes payload, timeout is 60 seconds:

```
AT$QCPING="180.97.33.107",10,32,60000
OK

$QCPING: SUCC, dest: 180.97.33.107, RTT: 334 ms
$QCPING: SUCC, dest: 180.97.33.107, RTT: 179 ms
…
```

Stop ping:

```
AT$QCPING=0
OK
```

4.1.5. AT$QCIPERF

This command tests the uplink and downlink IPERF performance of TCP/IP.

The Test command returns values supported as a compound value.

Table 4.5: AT$QCIPERF

AT$QCIPERF	Response
Set command AT$QCIPERF=<action>[,<protocol>[,<port>[,<ipaddr>[,<tpt>[,payload_size[,<packet_number>[,<duration>[,<report_interval>]]]]]]]	Response OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT$QCIPERF=?	Response $QCIPERF: (list of supported <action>s), (list of supported <protocol>s), (list of supported <port>s), (list of supported <tpt>s), (list of supported <payload_size>s), (list of supported <pkg_num>s), (list of supported <duration>s), (list of supported <report_interval>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<action>	Integer type	
	IPERF command	
	0	Terminate all IPERF services
	1	Start IPERF client
	2	Stop IPERF client
	3	Start IPERF server
	4	Start IPERF UDP NAT server
	5	Stop IPERF server
<protocol>	Integer type	
	0	UDP
	1	TCP
<port>	Integer type	
	UDP/TCP port number.	
<ipaddr>	String type	
	IP address.	
<tpt>	String type	
	Throughput in bps. Default value is 20000	
<payload_size>	Integer type	
	Payload size of UL UDP/TCP IPERF packet. Used for client mode.	

<packet_number>	Integer type
	Packet number of UE send when acted as a client mode.
<report_interval>	Integer type
	Report internal of IPERF service result. UE send the following unsolicited result codes periodically in this interval (in seconds). Default value is 10.
<duration>	Integer type
	IPERF service duration in seconds.

When the IPERF client service is finished (terminated/timeout), the UE sends the unsolicited result codes: `$QCIPERF: Client END, pkg sent total bytes: <bytes>, average UL through put: <tpt> bps`

When the IPERF server service is finished (terminated/timeout), the UE sends the unsolicited result codes: `$QCIPERF: Server END, pkg recv total bytes: <bytes>, average DL through put: <tpt> bps`

If an error occurs, causing the IPERF service to discontinue, the UE sends the unsolicited result codes: `$QCIPERF: Client FAIL, <err>; or $QCIPERF: Server FAIL, <err>`

Example

```
AT$QCIPERF=1,0,5001,"180.167.122.150",10000
OK

$QCIPERF: Client SUCC, pkg sent bytes: 13720, UL through put: 10976 bps
$QCIPERF: Client SUCC, pkg sent bytes: 9604, UL through put: 7683 bps
$QCIPERF: Client SUCC, pkg sent bytes: 12348, UL through put: 9878 bps
$QCIPERF: Client SUCC, pkg sent bytes: 12348, UL through put: 9878 bps

AT$QCIPERF=0
OK

$QCIPERF: Client END, pkg sent total bytes: 52136, average UL through
put: 9268 bps
```

4.1.6. AT$QCFREQ

This command sets prefer EARFCN list, lock or unlock cell.

The Read command returns the current EARFCN setting.

The Test command returns values supported as a compound value.

Table 4.6: AT$QCFREQ

AT$QCFREQ	Response
Set command If cell unlock or remove prefer EARFCN (mode = 0): AT$QCFREQ=<mode> If set prefer EARFCN list (mode = 1): AT$QCFREQ=<mode>[,<earfcn1>[,<earfcn2>…]] If cell lock (mode = 2): AT$QCFREQ=<mode>,<earfcn>[,<phyCellId>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCIPERF?	Response If mode = 0: $QCFREQ: <mode> If mode = 1: $QCFREQ: <mode>,<arfcn1>,<arfcn2>,… If mode = 2: $QCFREQ: <mode>,<arfcn>,<phyCellId> OK
Test command AT$QCFREQ=?	Response $QCFREQ: (list of supported <mode>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

<mode>	Integer type	
	0	Cancel cell lock and clear prefer EARFCN
	1	Set prefer EARFCN list
	2	EARFCN lock, or cell lock
<earfcn>	Integer type	
	E-UTRA Absolute Radio Frequency Channel Number	
<phyCellId>	Integer type	
	Physical cell ID	

The EARFCN of UE currently camped on cell will be added to the header of prefer EARFCN list.

Example

```
AT$QCFREQ=2,3734,145
OK

AT$QCREQ?
$QCFREQ: 2,3734,145
OK

AT$QCFREQ=0
OK
```

4.1.7. AT$QCRMFPLMN

Set command removes FPLMN in NVM or SIM.

The Test command returns values supported as a compound value.

Table 4.7: AT$QCRMFPLMN

AT$QCRMFPLMN	Response
Set command AT$QCRMFPLMN=<mode>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCRMFPLMN=?	Response $QCRMFPLMN: (list of supported <mode>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	Integer type	
<mode>	0	Remove FPLMN in NVM file and in SIM card
	1	Remove FPLMN in NVM file
	2	Remove FPLMN in SIM card

Example
```
AT$QCRMFPLMN=0
OK
```

4.1.8. AT$QCATTBEARER

The Set command is used to configure the PDN information request to establish during the attach process, if attach with PDN is required.

The Read command is used to obtain the configuration of the PDN information request to establish during the attach process.

The Test command returns values supported as a compound value.

Table 4.8: AT$QCATTBEARER

AT$QCATTBEARER	Response
Set command AT$QCATTBEARER=<PDP_type>[,<eitf>[,<apn>[,<IPv4AddrAlloc>[,<NSLPI>[,<IPv4_MTU_discovery>[,<NonIP_MTU_discovery>]]]]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCATTBEARER?	Response $QCATTBEARER: <pdnType>,<eitf>,<apnStr>,<ipv4allocType>,<NSLPI>,<ipv4Mtu>,<nonIpMtu> OK
Test command AT$QCATTBEARER=?	Response $QCFREQ: (list of supported <PDP_type>s), (list of supported <eitf>s), (list of supported <IPv4AddrAlloc>s), (list of supported <NSLPI>s), (list of supported <IPv4_MTU_discovery>s), (list of supported <NonIP_MTU_discovery>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

<PDP_type>	Integer type	
	PDP type	
	1	Cancel cell lock and clear prefer EARFCN
	2	IPv4
	3	IPv6 (Default)
	4	IPv4v6
	5	NON-IP
<eitf>	Integer type	
	0	Security protected ESM information transfer not required
	1	Security protected ESM information transfer required (Default)
<apn>	String type	
	Apn string, the max length is 63 characters. Default is “” (NULL).	
<IPv4AddrAlloc>	Integer type	
	0	IPv4 address allocate through NAS signaling (Default)
	1	IPv4 address allocate through DHCP
<NSLPI>	Integer type	
	0	Indicates that this PDP context is to be activated with the value for the low-priority indicator configured in the MT (Default)
	1	Indicates that this PDP context is to be activated with the value for the low-priority indicator set to "MS is not configured for NAS signaling low priority"
<IPv4_MTU_discovery>	Integer type	
	0	IPv4 MTU size discovery not influenced by $QCATTBEARER
	1	IPv4 MTU size discovery through NAS signaling (Default)

<NonIP_MTU_discovery>	Integer type	
	0	IPv4 MTU size discovery not influenced by $QCATTBEARER
	1	IPv4 MTU size discovery through NAS signaling (Default)

4.1.9. AT$QCSENDDATA

The Set command could send data through control plane or user plane.

Table 4.9: AT$QCSENDDATA

AT$QCSENDDATA	Response
Set command AT$QCSENDDATA=<cid>,<data_length>,<data>[,<RAI>[,<type_of_user_data>]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSENDDATA=?	Response $QCSENDDATA: (range of supported <cid>s), (maximum number of octets of user data indicated by <data_length>s), (list of supported <RAI>s), (list of supported <type_of_user_data>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<data_length>	Integer type	
	Indicates the number of octets of the <data> information element. The max length is 950	
<data>	String type	
	String of octets	
<RAI>	Integer type	
	Indicates the value of the release assistance indication	
	0	No information available
	1	The MT expects that exchange of data is completed with the transmission of this UL packet
	2	The MT expects that exchange of data is completed with the receipt of a DL packet
<type_of_user_data>	Integer type	
	Indicates whether the user data that is transmitted is regular or exceptional.	
	0	Regular data
	1	Exception data

Example

```
AT$QCSENDDATA=5,2,"ABCD"
OK
```

4.1.10. +RECVNONIP

This is an unsolicited code message used to indicate downlink NON-IP data.

Table 4.10: +RECVNONIP

Message	Parameters
+RECVNONIP:	<cid>,<data_length>,<data>

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<data_length>	Integer type	
	Indicates the number of octets of the <data> information element.	
<data>	String type	
	String of octets.	

4.1.11. AT$QCPMUCFG

This command sets PMU mode.

The Read command returns the current setup.

The Test command returns values supported as a compound value.

Table 4.11: AT$QCPMUCFG

AT$QCPMUCFG	Response
Set command AT$QCPMUCFG=<enable>[,<mode>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCPMUCFG?	Response $QCPMUCFG: <enable>[,<mode>] OK
Test command AT$QCPMUCFG=?	Response $QCFREQ: (range of supported <enable>s), (list of supported <mode>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<PDP_type>	Integer type	
	Specifies to enable PMU or not.	
	0	Disable the PMU (Default)
	1	Enable the PMU

<mode>	Integer type	
	Specifies to depth of sleep mode.	
	0	Active (Default)
	1	Idle
	2	Sleep1
	3	Sleep2
	4	Hibernate

Example

```
AT$QCPMUCFG=1,4
OK

AT$QCPMUCFG=0
OK
```

4.1.12. AT$QCSMSSEND

This command is used to send one SMS.

Table 4.12: AT$QCSMSSEND

AT$QCSMSSEND	Response
Set command AT$QCSMSSEND=<mode>,<pdu/da>, [<toda>,<test_sms>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	60 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type	
	0	PDU mode
	1	TXT mode
<pdu/da>	String type	
	In PDU mode: PDU	
	In TXT mode: destination	
<toda>	Integer type	
	Type of destination address	
<test_sms>	Integer type	
	Message's content for TXT mode	

Example

```
AT$QCSMSSEND=1,"1064899990000",,"hello"
OK

AT$QCSMSSEND=1,1064899990000,,"hello"
OK
```

4.1.13. AT$QCCGSN

This command sets the IMEI and SN. Use AT+CGSN to read IMEI or SN.

Table 4.13: AT$QCCGSN

AT$QCCGSN	Response
Set command AT$QCCGSN=<type>,<sn/imei>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCCGSN=?	Response $QCCGSN: (range of supported <type>s),(data) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<type>	String type
	"IMEI" or "SN"
<sn/imei>	String type
	IMEI (15 bytes character). Default is 866818039921444
	SN (31 bytes maximum as visible character). Default is ""(NULL)

Example

```
AT$QCCGSN="IMEI","788596633100008"
OK

AT$QCCGSN="SN","01"
OK
```

4.1.14. AT$QCRFSTAT

This command shows the status of RF calibration.

Table 4.14: AT$QCRFSTAT

AT$QCPMUCFG	Response
Set command AT$QCRFSTAT	Response $QCRFSTAT: calibrate done OK If RF is not calibrated, response: $QCRFSTAT: not calibrate
Test command AT$QCPMUCFG=?	Response $QCFREQ: <status> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<status>	Integer type
	Status of calibration

Example

```
AT$QCRFSTAT?
$QCRFSTAT: calibrate done
OK
```

4.1.15. AT$QCRST

This command restarts the chip.

Table 4.15: AT$QCRST

AT$QCRST	Response
Set command AT$QCRST	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Example

```
AT$QCRST
OK
```

4.1.16. AT$QCPSMR

This command reports the power-saving mode status.

Table 4.16: AT$QCPSMR

AT$QCPSMR	Response
Set command AT$QCPSMR=<n>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCPSMR?	Response $QCPSMR: <n>,<mode> OK
Test command AT$QCPSMR=?	Response $QCPSMR: (range of supported <n>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<n>	Integer type	
	0	Disable unsolicited result code $QCPSMR: <mode> (Default)
	1	Enable unsolicited result code $QCPSMR: <mode>
<mode>	Integer type	
	0	Normal mode
	1	Power-saving mode

Example

```
AT$QCPSMR=1
OK

AT$QCPSMR?
$QCPSMR: 1,0
OK

$QCPSMR: 1
```

4.1.17. AT$QCPLMNS

Execution command is used to trigger a PLMN search while the UE is out of service. If the UE is not out of service, +CME ERROR: <err> is returned.

Read command returns the current PLMN search state, and the reset of time of PLMN search timer.

Table 4.17: AT$QCPLMNS

AT$QCPLMNS	Response
Execution command AT$QCPLMNS	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCPLMNS?	Response $QCPLMNS: <state>[,<oosTimeStep>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCPLMNS=?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<state>	Integer type	
	0	Deactivated, no PLMN search is ongoing
	1	Searching, PLMN search is ongoing
	2	Selected, already selected a PLMN
	3	OOS, UE is out of service and has started a PLMN search timer

	Integer type
<oosTimeStep>	The rest of time (in seconds) of OOS PLMN search timer, only present when <state> is 3.

Example

```
AT$QCPLMNS
OK

AT$QCPLMNS?
$QCPLMNS: 3, 108
OK
```

4.1.18. AT$QCCESQS

The Set command controls the extended signal quality change event reporting. If reporting is enabled the MT returns the unsolicited result codes: +CESQ: <rxlev>,<ber>,<rscp>,<ecno>,<rsrq>,<rsrp>, or $QCCESQ: RSRP,<rsrp>,RSRQ,<rsrq>,SNR,<snr> whenever the extended signal quality is changed. If setting fails in an MT error, +CME ERROR: <err> is returned.

The Read command returns the current reporting settings in the MT.

The Test command returns values supported as compound values.

Table 4.18: AT$QCCESQS

AT$QCCESQS	Response
Set command AT$QCCESQS=<report level>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCCESQS?	Response $QCCESQS: <report level> OK If there is an error, the response is as follows: +CME ERROR: <err
Test command AT$QCCESQS=?	Response $QCCESQS: (list of supported <report level>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	Integer type	
<report level>	0	Disable unsolicited report. (Default)
	1	Report +CESQ:<rxlev>,<ber>,<rscp>,<ecno>,<rsrq>,<rsrp>
	2	Report $QCCESQ: RSRP,<rsrp>,RSRQ,<rsrq>,SNR,<snr>

Example

```
AT$QCCESQS=2
OK

AT$QCCESQS?
$QCCESQ: 2
OK

AT$QCCESQS=?
$QCCESQS: (0-2)
OK
```

4.1.19. AT$QCSTATUS

This execution command returns some key parameter on the UE side.

Table 4.19: AT$QCSTATUS

AT+CGMR	Response
Execution command AT$QCSTATUS	Response $QCSTATUS: PHY, DlEarfcn:<dlEarfcn>, UlEarfcn:<ulEarfcn>, PCI:<pci>, Band:<band>, RSRP:<rsrp>, RSRQ:<rsrq>, SNR:<snr>, CeLevel:<ceLevel>, DlBler:<dlBler>, UlBler:<ulBler>, DataInactTimerS:<dataInactTimers>, RetxBSRTimerP:<retxBSRTimerO>, NBMode:<nbMode> $QCSTATUS: L2, SrbNum:<srbNum>, DrbNum:<drbNum> $QCSTATUS: RRC, State:<rrcState>, TAC:<tac>, CellId:<cellId> $QCSTATUS: EMM, EmmState:<emmState>, EmmMode:<emmMode>, PTWMs:<ptwMs>, EDRXPeriodMs:<eDRXPeriodMs>, PsmExT3412TimerS:<psmExT3412TimerS>, T3324TimerS:<T3324TimerS>, T3346RemainTimeS:<T3346RemainTimeS> $QCSTATUS: PLMN, PlmnState:<plmnState>, PlmnType:<plmnType>, SelectPlmn:<selectPlmn> $QCSTATUS: ESM, ActBearerNum:<actBearerNum>, APN:<apn>, IPv4:<ipaddr> $QCSTATUS: CCM, Cfun:<cfun>, IMSI:<imsi> OK
Test command AT$QCSTATUS=?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<dlEarfcn>	Integer type	
	0 to 262143	Downlink earfcn
<ulEarfcn>	Integer type	
	0 to 262143	Uplink earfcn
<band>	Integer type	
	0 to 70	Band
<rsrp>	Integer type	
	-156dBm to -44dBm	Indicate the measurement of RSRP value
<rsrq>	Integer type	
	-34dB to -2.5dB	Indicate the measurement of RSRQ value
<snr>	Integer type	
	-30dB to 30dB	SNR value
<ceLevel>	Integer type	
	0	CE level 0
	1	CE level 1
	2	CE level 2
<dlBler>	Integer type	
	0 to 10000	Downlink block error
<ulBler>	Integer type	
	0 to 10000	Uplink block error
<dataInactTimerS>	Integer type	
	0 to 180	Data inactive timer in seconds
<retxBSRTimerP>	Integer type	
	0 to 180	Timer for BSR reporting, value in number of PDCCH periods.
<NBMode>	String type	
	Possible values are: "InBand Same PCI", "InBand Diff PCI", "Guard Band", "Stand alone"	
<srbNum>	Integer type	
	0 to 2	SRB value
<drbNum>	Integer type	
	0 to 2	DRB value
<rrcState>	String type	
	Possible values are: "DEACT", "OOS", "IDLE", "SUSPEND IDLE", "CONNECTED", "UNKNOWN"	
<tac>	Integer type	
	0 to 65534	Tracking area code
<cellId>	Integer type	
	0 to 268435455	Four byte E-UTRAN cell ID in hexadecimal format
<emmState>	String type	
	Possible values are: "NULL", "DEREG", "REG INIT", "REG DEINIT", "TAU INIT", "SR INIT", "UNKNOWN"	
<emmMode>	String type	
	Possible values are: "IDLE", "PSM", "CONNECTED", "UNKNOWN"	

<ptwMs>	Integer type	
	eDRX Paging Time Window in milliseconds	
<edrxPeriodMs>	Integer type	
	eDRX period in milliseconds	
<psmExT3412TimerS>	Integer type	
	Extended T3412 timer value in seconds	
<T3324TimerS>	Integer type	
	T3324 timer value in seconds	
<T3346RemainTimeS>	Integer type	
	If T3346 is running, set to the remaining time, else set to 0	
<plmnState>	String type	
	Possible values are: "NO PLMN", "SEARCHING", "SELECTED", "UNKNOWN"	
<plmnType>	String type	
	Possible values are: "HPLMN", "EHPLMN", "VPLMN", "UPLMN", "OPLMN", "UNKNOWN"	
<selectPlmn>	Integer type	
	Selected PLMN	
<actBearerNum>	Integer type	
	Activated bearer number	
<apn>	String type	
	Access point name	
<ipv4Addr ipv6Addr>	String type	
	Ipv4/Ipv6 address	
<fun>	Integer type	
	0	Minimum functionality
	1	Full functionality
	4	Turn off RF
<IMSI>	String type	
	International Mobile Subscriber Identity	

Example

```
AT$QCSTATUS

$QCSTATUS: PHY, DlEarfcn:3738, UlEarfcn:21738, PCI:11, Band:8, RSRP:-91,
RSRQ:-8, SNR:8, CeLevel:0, DlBler:0/100, UlBler:0/100,DataInactTimerS:0,
RetxBSRTimerP:0, NBMode:"Stand alone"

$QCSTATUS: L2, SrbNum:0, DrbNum:0

$QCSTATUS: RRC, State:"IDLE", TAC:23369, CellId:26224411

$QCSTATUS: EMM, EmmState:"REG", EmmMode:"IDLE", PTWMs:5120,
EDRXPeriodMs:40960,
PsmExT3412TimerS:0, T3324TimerS:300, T3346RemainTimeS:0

$QCSTATUS: PLMN, PlmnState:"SELECTED", PlmnType:"EHPLMN",
SelectPlmn:"0x460,0xf000"
```

```
$QCSTATUS: ESM, ActBearerNum:1, APN:"cmnbiot.MNC004.MCC460.GPRS",
IPv4:"100.83.34.10"

$QCSTATUS: CCM, Cfun:1, IMSI:"460043263600041"
OK
```

4.1.20. AT$QCICCID

Execution command causes the TA to return the ICCID of the UICC.

Table 4.20: AT$QCICCID

AT$QCICCID	Response
Execution command AT$QCICCID	Response $QCICCID: <ICCID> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCICCID=?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<ICCID>	String type
	Integrated circuit card identification

Example

```
AT$QCICCID
$QCICCID: 89861119220009636664
OK
```

4.1.21. AT$QCBCINFO

Execution command to return the basic serving cell information and neighbor cells information, mainly used for location service.

Table 4.21: AT$QCBCINFO

AT$QCBCINFO	Response
Execution command AT$QCICCID	Response $QCBCINFOSC: <earfcn>,<pci>,<rsrp>, <rsrq>,<mcc>,<mnc>,<cellid> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCICCID=?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<earfcn>	Integer type	
	0 to 262143	Indicate the EARFCN of the cell
<pci>	Integer type	
	0 to 503	Indicate the physical cell ID
<rsrp>	Integer type	
	-156dBm to -44dBm	Indicate the measurement of RSRP value
<rsrq>	Integer type	
	-34dB to -2.5dB	Indicate the measurement of RSRQ value
<mcc>	String type	
	Indicate the mobile country code	
<mnc>	String type	
	Indicate the mobile network code	
<cellId>	Integer type	
	0 to 268435455	Four-byte E-UTRAN cell ID in hexadecimal format

4.1.22. AT$QCDNS

This command is used to get the IP address for a specific URL.

Table 4.22: AT$QCDNS

AT$QCICCID	Response

Execution command
`AT$QCDNS=<url>`
	Response
	`$QCDNS: <ipaddr>`
	`OK`
	If there is an error, the response is as follows:
	`+SOCKET ERROR: <err>`

Test command `AT$QCDNS=?`	Response `OK`
Maximum Response Time	30 s
Parameter Saving Mode	NO_SAVE

Parameter

<url>	String type
	Domain name
<ipaddr>	String type
	If IPv4 type, output is dot-notation format, such as: "32.1.13.184" If IPv6 type, output is colon-notation format, such as: "2001:0DB8:0000:CD30:0000:0000:0000:0002"

Example

```
AT$QCDNS="www.google.com"
$QCDNS: "142.251.128.142"
OK
```

4.1.23. AT$QCDNSCFG

The Set command sets the default DNS addresses configuration. If the DNS address is not configured by the network, when activate a default bearer, use these DNS addresses.

The Read command returns the current setting of the default DNS addresses.

Table 4.23: AT$QCDNSCFG

AT$QCDNSCFG	Response
Set command AT$QCDNSCFG=<ipaddr1>[,<ipaddr2>[,<ipaddr3>[,<ipaddr4>]]]	Response OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Read command AT$QCDNSCFG?	Response $QCDNSCFG: <ipaddr1>[,<ipaddr2>[,<ipaddr3>[,<ipaddr4>]]] OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT$QCDNSCFG=?	Response $QCCESQS: (list of supported <report level>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	Integer type
<ipaddr>	If IPv4 type, output is dot-notation format, such as: "32.1.13.184" If IPv6 type, output is colon-notation format, such as: "2001:0DB8:0000:CD30:0000:0000:0000:0002"

Example

```
AT$QCDNSCFG?
$QCDNSCFG: "114.114.114.114","114.114.115.115","240c:0000:0000:0000:
0000:0000:0000:6666","240c:0000:0000:0000:0000:0000:0000:6644"
OK

AT$QCDNSCFG="114.114.114.12"
OK
```

4.1.24. AT$QCEMMTIME

This command reports and gets the Emm Time State, include T3346, T3448, and T3412/extend T3412.

Table 4.24: AT$QCEMMTIME

AT$QCEMMTIME	Response
Set command AT$QCEMMTIME=<bitmap>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCEMMTIME?	Response $QCEMMTIME:0,<timeState>[,<remainTimeValue>] $QCEMMTIME:1,<timeState>[,<remainTimeValue>] $QCEMMTIME:2,<timeState>[,<remainTimeValue>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCEMMTIME=?	Response $QCEMMTIME: (range of supported <bitmap>s) OK
Indicate	Response $QCEMMTIME: <timeId>,<timeState>[,<timeValue>]
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<bitmap>	Integer type	
	bit 0: enable/disable unsolicited result code T3346 (Default: 0) bit 1: enable/disable unsolicited result code T3448 bit 2: enable/disable unsolicited result code T3412/ext T3412	
<timeId>	Integer type	
	0	Emm timer: T3346
	1	Emm timer: T3448
	2	Emm timer: T412/ext T3412
<timeState>	Integer type	
	0	Start
	1	Stop
	2	Expiry
<remainTimeValue>	Integer type	
	Time remains value in seconds. Only include when <timeState> is 0.	
<timeValue>	Integer type	
	Time value in seconds. Only include when <timeState> is 0.	

Example

```
AT$QCEMMTIME=7
OK

AT$QCEMMTIME?
$QCEMMTIME: 0,1
$QCEMMTIME: 1,1
$QCEMMTIME: 2,0,3240
OK
```

4.1.25. AT$QCPCFG

Set command is used to set plat config. If the UE is not out of service, `+CME ERROR: <err>` is returned.

Read command returns the current plat config setting.

Table 4.25: AT$QCPCFG

AT$QCPCFG	Response

Set command
`AT$QCPCFG=<mode>,<value>`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Read command
`AT$QCPCFG?`
	Response
	`$QCPCFG: "faultAction":<value>,`
	`"dumpToATPort":<value>,"startWDT":<value>,`
	`"logCtrl":<value>,"logLevel":<value>,`
	`"logBaudrate":<value>,"slpWaitTime":<value>`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Test command
`AT$QCPCFG=?`
	Response
	`$QCPCFG: "faultAction":<value>,`
	`"dumpToATPort":<value>,"startWDT":<value>,`
	`"logCtrl":<value>,"logLevel":<value>,`
	`"logBaudrate":<value>,"slpWaitTime":<value>`
	`OK`

Maximum Response Time	5 s
Parameter Saving Mode	SAVE

Parameter

	String type
<mode>	"faultAction" Set the hard fault action mode
	"dumpToATPort" Set show assert dump in AT port or not
	"startWDT" Set watch dog mode
	"logCtrl" Set log control mode
	"logLevel" Set log print level
	"logBaudrate" Set log print baud rate
	"slpWaitTime" Set sleep wait time
<value>	Integer type
	For "faultAction", the values range is from 0 to 3 0: dump full exception information to flash and EPAT tool then trapped in endless loop 1: print necessary exception information then reset 2: dump full exception information to flash then reset 3: dump full exception information to flash and EPAT tool then reset
	For "dumpToATPort", the values range is from 0 to 1 0: not dump to AT port 1: dump to AT port
	For "startWDT", the values range is from 0 to 1 0: stop WDT

	1: start WDT
	For "logCtrl", the values range is from 0 to 2 0: unilog is disabled 1: only sw log is enabled 2: All log is enabled
	For "logLevel", the values range is from 0 to 5 0: debug log level 1: information log level 2: value log level 3: signal log level 4: warning log level 5: error log level
	For "logBaudrate", the values range is from 921600 to 6000000
	For "slpWaitTime", the values range is from 0 to 0xffff

4.1.26. AT$QCSLEEP

This command is used for power consumption test. After executing this command, the UE enters related Low Power state. Also, the UE can be woken up by wakeup PAD, after waking up the UE reboots.

Table 4.26: AT$QCSLEEP

AT$QCSLEEP	Response
Set command AT$QCSLEEP=<state>	Response $QCSLEEP: <mode> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSLEEP=?	Response $QCSLEEP: <state> OK
Maximum Response Time	5 s
Parameter Saving Mode	SAVE

Parameter

<state>	Integer type	
	0	HIB2
	1	HIB1
	2	SLEEP2
	3	SLEEP1
<mode>	String type	
	HIB2	Hibernate2 status
	HIB1	Hibernate1 status
	SLEEP2	Sleep2 status
	SLEEP1	Sleep1 status

Example

```
AT$QCSLEEP=?
$QCSLEEP: <state>
OK

AT$QCSLEEP=3
$QCSLEEP: SLEEP1
OK
```

4.1.27. AT$QCSIMSLEEP

This command sets the UE to allow SIM card sleep for power consumption.

The Read command returns current setting of each parameter.

The Test command returns values supported as a compound value.

Table 4.27: AT$QCSIMSLEEP

AT$QCSIMSLEEP	Response
Set command AT$QCSIMSLEEP=<mode>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCSIMSLEEP?	Response $QCSIMSLEEP: <state> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	SAVE

Parameter

	Integer type	
<mode>	0	Not allowed SIM sleep
	1	Allowed SIM sleep

4.1.28. AT$QCCGSNLOCK

This command is used to set lock flag for IMEI and SN. If locked, IMEI and SN cannot be written through the AT$QCCGSN command. The lock feature is required in the production stage to prevent any accidental operation by user. If lock is set, it cannot be cleared through the AT command; the only way to clear it is through flash tool (erase related region).

The Set command sets lock for IMEI and SN.

The Test command returns parameter supported as a compound value.

The Read command returns current lock status.

Table 4.28: AT$QCCGSNLOCK

AT$QCCGSNLOCK	Response
Set command AT$QCCGSNLOCK=<para>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCCGSNLOCK?	Response $QCCGSN: <imeistatus,snstatus> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCCGSNLOCK=?	Response $QCPCFG: "faultAction":<value>, "dumpToATPort":<value>,"startWDT":<value>, "logCtrl":<value>,"logLevel":<value>, "logBaudrate":<value>,"slpWaitTime":<value> OK
Maximum Response Time	5 s
Parameter Saving Mode	SAVE

Parameter

<para>	String type
	"IMEI"
	"SN"
<imeistatus>	String type
	"IMEI LOCKED"
	"IMEI NOT LOCKED"
<snstatus>	String type
	"SN LOCKED"
	"SN NOT LOCKED"

Example

```
AT$QCCGSNLOCK=IMEI
OK

AT$QCCGSNLOCK?
$QCCGSNLOCK: IMEI LOCKED,SN NOT LOCKED
OK

AT$QCCGSN=IMEI,XXXXXX
$QCCGSN:IMEI LOCKED
+CME ERROR: 50
```

4.1.29. AT$QCSAVEFAC

This command is used in the production line, which saves related regions to the default reliable region.

For example, after IMEI/SN was written or RF calibration was performed.

Default reliable regions are used to restore factory setting.
The Set command saves related regions to the default reliable region.

The Test command returns mode supported as a compound value

Table 4.29: AT$QCSAVEFAC

AT$QCSAVEFAC	Response
Set command AT$QCSAVEFAC=<mode>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSAVEFAC=?	Response $QCSAVEFAC: <mode> OK
Maximum Response Time	5 s
Parameter Saving Mode	SAVE

Parameter

	String type	
<mode>	"all"	All regions
	"rfregion"	Only RF Regions
	"other"	Regions except RF, currently IMEI/SN region

4.1.30. AT$QCTASKINFO

The Execution command returns the name, ID, status, priority and stack information of all the tasks. This command is only for debug purpose.

Table 4.30: AT$QCTASKINFO

AT$QCTASKINFO	Response
Execution command AT$QCTASKINFO	Response $QCTASKINFO: <task information> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

4.1.31. AT$QCTASKHISTINFO

The Execution command shows the most recent scheduling history of tasks. This command is only for debug purpose.

Table 4.31: AT$QCTASKHISTINFO

AT$QCTASKHISTINFO	Response
Execution command AT$QCTASKHISINFO	Response $QCTASKINFO: <task scheduling history> OK
Maximum Response Time	5 s

Parameter Saving Mode	NO_SAVE

4.1.32. AT$QCSHOWMEM

The Execution command shows the current heap memory usage. This command is only for debug purpose.

Table 4.32: AT$QCSHOWMEM

AT$QCSHOWMEM	Response
Execution command AT$QCSHOWMEM	Response $QCSHOWMEM: <curr_free_heap, min_free_heap> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<curr_free_heap>	Integer type
	Current remained free memory size in heap.
<min_free_heap>	Integer type
	Minimum heap memory size ever remaining in heap

4.1.33. AT$QCSYSTEST

This command is only for debug purpose.

The Test command returns option supported as a compound value.

The Set command triggers a test feature.

Table 4.33: AT$QCSYSTEST

AT$QCSAVEFAC	Response
Set command AT$QCSYSTEST=<option>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSYSTEST=?	Response $QCSYSTEST: <option> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	String type	
	"Handshake"	Perform handshake with UE
	"Assert"	Trigger a test assert
	"Testwdt"	Trigger watch dog test
	"Fsassert"	Trigger file system assert for test
	"Hardfault"	Trigger a hard fault for test

4.1.34. AT$QCVOTECHK

This command shows current vote state, which can help to analyze the reason of sleep failure.

This command is only for debug purpose.

Table 4.34: AT$QCVOTECHK

AT$QCVOTECHK	Response

Execution command
`AT$QCVOTECHK`
	Response
	`Sleep Vote Info:`
	`<vote info detail>`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

NOTE

Vote information detail shows comprehensive information that affects the sleep process in the HTNB32L SDK.

The detail information is separated into 5 parts.
Part1: user set sleep depth limitation.
Part2: QC internal sleep vote result.
Part3: application vote result.
Part4: user registered sleep depth callback.
Part5: driver vote result.
For more detail information, refer to slpman_qcx212.h in SDK and QCX212 low power development manual.

Example

```
AT$QCVOTECHK

Sleep Vote Info:
Deepest Sleep Mode: Hibn //part1
```

NOTE

Return the deepest Sleep mode allowed to enter. It can be set by using slpManSetPmuSleepMode, which can be found in slpman_qcx212.h.

```
QTI SDK Vote for: Hibn //part2
Detail: 0x0,0x0,0x0

Application Vote for: Hibn //part3
```

NOTE

Normally, applications use the vote API in slpman_qcx212.h to control the Sleep state. This item returns the application vote result.

```
Handle: 0 Name: CTIOT_NB Prohibit State: NULL Vote count: 0

Handle: 1 Name: ONENETSL Prohibit State: NULL Vote count: 0
```

NOTE

The sub item shows more details of the vote result, including vote handle, name information, vote state, and vote counter.

`Prohibit State` can be `NULL`, `Slp1`, `Slp2`, or `Hibn`.
`Prohibit State = NULL`: The application does not prohibit to enter all sleep state
`Prohibit State = Slp2`: The application prohibits to enter sleep2, so the system can only sleep to sleep

```
User defined Sleep Callback Vote for: Hibn //part4
```

NOTE

Vote result for UsrSlpDepth callback, which is registered by calling: slpManRegisterUsrSlpDepthCb().

```
Driver Vote bitMap: 0x0, with vote mask: 0x0 //part5
```

NOTE

Driver vote result and vote mask information. The bitmap is corresponding to the enum type slpDrvVoteModule_t in slpman_qcx212.h.

The vote mask indicates that the sleep flow does not depend on a specific driver vote result.
For example:
Driver Vote bitMap: 0x9, with vote mask: 0x8
UART and ADC do not allow to sleep, but as the vote mask is 0x8, the PMU module does not take the ADC vote result into consideration. It enters sleep as soon as UART clear the bit 0
For drivers, if the (bitmap & (~mask)) != 0 the system cannot go to sleep1 or deeper state)

```
OK
```

4.1.35. AT$QCURC

This command closes/opens URC (unsolicited result code) report.

Table 4.35: AT$QCURC

AT$QCURC	Response
Set command AT$QCURC=<urcStr>,<value>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCURC?	Response $QCURC: "CREG":<value>,"CEREG":<value>, "CEDRXP":<value>,"CCIOTOPTI":<value>, "CSCON":<value>,"QCCESQ":<value>, "CGEV":<value>,"QCPSMR":<value>, "PTWEDRX":<value>,"QCPTWEDRXP":<value>, "QCPIN":<value>,"QCPADDR":<value>, "QCPCFUN":<value> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCURC=?	Response $QCURC: "ALL":(0-1),"CREG":(0-1), "CEREG":(0-1),"CEDRXP":(0-1), "CCIOTOPTI":(0-1),"CSCON":(0-1), "QCCESQ":(0-1),"CGEV":(0-1), "QCPSMR":(0-1),"PTWEDRX":(0-1), "QCPTWEDRXP":(0-1),"QCPIN":(0-1), "QCPADDR":(0-1),"QCPCFUN":(0-1) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

	String type	
<mode>	"ALL"	All unsolicited result codes included as below
	"CREG"	Unsolicited result code +CREG
	"CEREG"	Unsolicited result code +CEREG
	"CEDRXP"	Unsolicited result code +CEDRXP
	"CCIOTOPTI"	Unsolicited result code +CCIOTOPTI
	"CSCON"	Unsolicited result code +CSCON
	"QCCESQ"	Unsolicited result code $QCCESQ
	"CGEV"	Unsolicited result code +CGEV
	"QCPSMR"	Unsolicited result code $QCPSMR
	"PTWEDRX"	Unsolicited result code +PTWEDRX
	"QCPTWEDRX"	Unsolicited result code $QCPTWEDRX
	"QCPIN"	Unsolicited result code $QCPIN
	"QCPADDR"	Unsolicited result code $QCPADDR
	"QCPCFUN"	Unsolicited result code $QCPCFUN
<mode>	Integer type	
	0	Disable unsolicited result code report
	1	Enable unsolicited result code report

Example

```
AT$QCURC="ALL",1
OK

AT$QCURC?
$QCURC:
"CREG":1,"CEREG":1,"CEDRXP":1,"CCIOTOPTI":1,"CSCON":1,"QCCESQ":1,
"CGEV":1,"QCPSMR":1,"PTWEDRX":1,"QCPTWEDRXP":1,"QCPIN":1
OK

AT$QCURC=?
$QCURC: "ALL":(0-1),"CREG":(0-1),"CEREG":(0-1),"CEDRXP":(0-1),
"CCIOTOPTI":(0-1),"CSCON":(0-1),"QCCESQ":(0-1),"CGEV":(0-1),
"QCPSMR":(0-1),"PTWEDRX":(0-1),"QCPTWEDRXP":(0-1),"QCPIN":(0-1)
OK
```

4.1.36. AT$QCPTWEDRXS

The Set command controls the setting of the UE's paging time window and eDRX parameters. It can be used to control whether the UE wants to apply paging time window and eDRX or not, as well as the requested eDRX value for NB-IoT.

The Set command also controls the presentation of the URC when <n>=2 and there is a change of the paging time window and eDRX parameters provided by network: `$QCPTWEDRXP: <AcTtype>[, <Requested_Paging_time_window>[,<Requested_eDRX_value>[, <NW_provided_eDRX_value>[,<Paging_time_window>]]]]`

A special form of the command can be given as `AT$QCPTWEDRXS=3`. In this form, paging time window and eDRX is disabled and data for all parameters in `AT$QCPTWEDRXS` command is 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.

Table 4.36: AT$QCPTWEDRXS

AT$QCPTWEDRXS	Response
Set command AT$QCPTWEDRXS=[<mode>[,<Ac-Type>[,<Requested_Paging_time_window>[,<Requested_eDRX_value>]]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCPTWEDRXS?	Response $QCPTWEDRXS: <AcTtype>,<Requested_Paging_time_window>,<Requested_eDRX_value>[,<NW_provided_eDRX_VALUE>[,<Paging_time_window>]] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCPTWEDRXS=?	Response $QCPTWEDRXS: (list of supported <mode>s),(list of supported <AcT-type>s),(list of supported <Requested_Paging_time_window>s),(list of supported <Requested_eDRX_value>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

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, that is, the most recent setting of <mode> takes effect for all specified values of <AcTtype>.	
	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 unsolicited result code.
	3	Disable the use of paging time window and eDRX and discard all parameters for paging time window and eDRX.
<AcT-type>	Integer type	
	Indicates the type of access technology.	
	5	NB IOT
<Requested_Paging_time_window>	String type	
	4-bit string. The paging time window refers to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (example "0000" equals 2.56 seconds).	
<Requested_eDRX	String type	

_value>	4-bit string. The eDRX value refers to bit 4 to 1 of octet 3 of the extended DRX parameters information element (example ”0010” equals 20.48 seconds).
<NW_provided_eDRX _value>	String type
	4-bit string. The eDRX value refers to bit 4 to 1 of octet 3 of the extended DRX parameters information element (example”0010” equals 20.48 seconds).
<Requested_Paging _time_window>	String type
	4-bit string. The paging time window refers to bit 8 to 5 of octet 3 of the Extended DRX parameters information element (example”0000” equals 2.56 seconds).

Example

```
AT$QCPTWEDRXS=1,5,”0011”,”0011”
OK

AT$QCPTWEDRXS?
$QCPTWEDRXS: 5,”0011”,”0011”
OK

AT$QCPTWEDRXS=?
$QCPTWEDRXS: (0,1,2,3), (5) ,(”0000”-”1111”),(”0000”-”1111”)
OK
```

4.1.37. $QCPIN

This is an unsolicited result code used to indicate SIM PIN state. It is controlled by AT$QCURC.

Table 4.37: $QCPIN

Message	Parameter
$QCPIN:	<code>

Parameter

<code>	String type	
	READY	MT is not pending for any password.
	SIM PIN	MT is waiting SIM PIN to be given.
	SIM PUK	MT is waiting SIM PUK to be given.
	SIM PUK BLOCKED	The SIM is locked.
	SIM NOT READY	The SIM is not ready.

4.1.38. $QCPCFUN

This is an unsolicited result code used to indicate the setting of <fun> from AT+CFUN when the MT powered on. It is controlled by AT$QCURC.

Table 4.38: $QCPCFUN

Message	Parameter
$QCPCFUN:	<fun>

Parameter

<fun>	String type	
	0	Minimum functionality

	1	Full functionality
	4	Turn off RF

4.1.39. $QCPADDR

This is an unsolicited result code used to print the PDP address. It is controlled by `AT$QCURC`.

Table 4.39: $QCPADDR

Message	Parameter
`$QCPADDR:`	`<cid>[,<PDP_addr_1>[,<PDP_addr_2>]]`

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<PDP_addr_1>	String type	
	Identifies the MT in the address space applicable to the PDP. Contains the IPv4 address. Omitted if not available.	
	The string is given as dot-separated numeric (0-255) parameter	
<PDP_addr_2>	String type	
	Identifies the MT in the address space applicable to the PDP. Contains the IPv6 address. Omitted if not available.	
	The string is given as dot-separated numeric (0-255) parameter	

4.1.40. AT$QCADC

The Set command returns current thermal temperature or battery voltage read from ADC.

The Test command shows the options available to read from ADC.

Table 4.40: AT$QCADC

AT$QCADC	Response

Set command
`AT$QCADC=<option>`
	Response
	`$QCADC: <option>,<value>[,<option>,<value>]`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Test command
`AT$QCADC=?`
	Response
	`$QCADC: <option>`
	`OK`

Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<option>	String type	
	"temp"	Get current thermal temperature in Celsius with 1 degree resolution.
	"vbat"	Get current VBAT value in mV.
	"all"	Get current thermal temperature and VBAT value.

<value>	String type
	Corresponding value of option

Example

```
AT$QCADC=all
$QCADC: TEMP,26,VBAT,3604
OK

AT$QCADC=temp
$QCADC: TEMP,26
OK

AT$QCADC=vbat
$QCADC: VBAT,3604
OK
```

4.1.41. AT$QCPDPCFGE

This command is used to update APN profile, which includes profile id, APN disable flag, APN class, APN bearer etc.

Table 4.41: AT$QCPDPCFGE

AT$QCPDPCFGE	Response
Set command AT$QCPDPCFGE=<profile_id>, <APN disable flag>,<timer_value>, <APN class>,<APN bearer>, <max_pdn_conn_per_block>, <max_pdn_conn_timer>, <pdn_req_wait_timer>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCPDPCFGE=?	Response $QCPDPCFGE=<profile_id>, <APN disable flag>,<timer_value>, <APN class>,<APN bearer>, <max_pdn_conn_per_block>, <max_pdn_conn_timer>, <pdn_req_wait_timer> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<profile_id>	Integer type	
	Sets PDN teardown time interval.	
	1-24	Supported values
<APN disable flag>	Integer type	
	Enable or disable APN disable flag.	

	0	Enable
	1	Disable
<timer_value>	Integer type	
	Indicates timer value. Parameter not used, must be 0	
	0	Supported value
<APN class>	Integer type	
	Indicates range of APN class.	
	0-5	Supported values
<APN bearer>	Integer type	
	Indicates APN Bearer values.	
	0-5	Supported values
<APN class>	Integer type	
	Indicates range of APN class.	
	1	GSM type
	2	UMTS type
	4	LTE type
	8	TDS type
	255	Supported values
<max_pdn_conn_per_block>	Integer type	
	Indicates maximum PDN connection per block.	
	0-1023	Supported values
<max_pdn_conn_timer>	Integer type	
	Indicates maximum PDN connection timer.	
	0-3600	Supported values
<pdn_req_wait_timer>	Integer type	
	Indicates PDN request wait timer	
	0-1023	Supported values

Example

```
AT$QCPDPCFGE=2,0,0,2,255,0,0,0
OK

AT$QCPDPCFGE=?
$QCPDPCFGE=<profile_id>,<APN disable flag>,<timer_value>,<APN class>,
<APN bearer>,<max_pdn_conn_per_block>,<max_pdn_conn_timer>,
<pdn_req_wait_timer>
OK
```

4.1.42. AT$QCESMCAUSE

Read command is used to return the current setting of ESM cause.

Table 4.42: AT$QCESMCAUSE

AT$QCESMCAUSE	Response
Read command AT$QCESMCAUSE?	Response $QCESMCAUSE:<rejCausePresent>,<causeType>,<rejCause> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<rejCausePresent>	Integer type	
	0	No reject cause present
	1	Reject cause present
<CauseType>	Integer type	
	Indicates the type of <reject_cause>	
	0	Indicates that <reject_cause> contains an ESM cause value
	1	Indicates that <reject_cause> contains a manufacturer-specific cause value
<rejCause>	Integer type	
	Contains the cause of the failed registration. The value is of type as defined by <cause_type>.	

Example

```
AT$QCESMCAUSE?
$QCESMCAUSE: 1,0,26
OK
```

4.1.43. AT$QCSTATE

The Read command returns the EMM states/sub-states.

Table 4.43: AT$QCSTATE

AT$QCSTATE	Response
Read command AT$QCSTATE?	Response $QCSTATE:<emmState> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

	String type
<emmState>	Possible values: “NULL”, ” DEREG”, ” REG INIT”, ” REG”, ” DEREG INIT”, ” TAU INIT”, ”SR INIT”, "DEG ATTEMPT TO ATTACH", "DEG PLMN SEARCH", "DEG NO IMSI", "DEG ATTACH NEEDED", "DEG NO CELL AVAILABLE", "REG ATTEMPTING TO UPDATE", "REG LIMITED SERVICE", "REG PLMN SEARCH", "REG UPDATE NEEDED", "REG NO CELL AVAILABLE", "REG ATTEMPTING TO UPDATE MM", "REG IMSI DETACH INTIATED", ”UNKNOWN”

Example

```
AT$QCSTATE?
QCSTATE:"REG"
OK
```

4.1.44. AT$QCLAPI

Set command set the LAPI. If the SIM card do not have EFNASCONFIG file, the module should follow the AT setting. If the SIM card has such a file, the module should follow the SIM card.

Read command returns current LAPI value.

Table 4.44: AT$QCLAPI

AT$QCLAPI	Response
Set command AT$QCLAPI=<Lapi>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCLAPI?	Response $QCLAPI:<Lapi> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

	Integer type	
<Lapi>	0	Disabled
	1	Enabled

Example

```
AT$QCLAPI = 0
OK
```

4.1.45. AT$QCDRX

Read command returns the DRX parameters.

Table 4.45: AT$QCDRX

AT$QCDRX	Response
Read command AT$QCDRX?	Response in idle mode $QCDRX: <drxcycle> OK Response in connected mode $QCDRX: <drxInactivityTimer>, <drxRetransmissionTimer>, <drxStartOffset>,<drxULRetransmissionTimer>, <longdrxCycle>,<onDurationTimer> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type	
	1	RRC IDLE
	2	RRC CONNECTED
<drxcycle>	Integer type	
	The value of DRX cycle in idle mode is in milliseconds.	
<drxInactivityTimer>	Integer type	
	Represents indexes in enums, which corresponds to seconds.	
	Index	Time in seconds
	0	1
	1	2
	2	3
	3	5
	4	7
	5	10
	6	15
	7	20
	8	40
	9	50
	10	60
	11	80
	12	100
	13	120
	14	150
	15	180
<longdrxcycle>	Integer type	

	Represents indexes in enums, which corresponds to sub-frames.	
	Index	Sub-frames
	0	256
	1	512
	2	1024
	3	1536
	4	2048
	5	3072
	6	4096
	7	4608
	8	6144
	9	7680
	10	8192
	11	9214
	12	spare4
	13	spare3
	14	spare2
	15	spare1
	16	1280_v1430
	17	2560_v1430
	18	5120_v1430
	19	10240_v1430
<drxStartOffset>	Integer type	
	The value is in number of sub-frames by step of (drx-cycle / 256).	
	0-255	Number of sub-frames
<drxOnDurationTimer>	Integer type	
	Represents indexes in enums, which corresponds to PDCCH periods.	
	Index	PDCCH Periods
	0	0
	1	1
	2	2
	3	4
	4	6
	5	8
	6	16
	7	32
	8	spare2
<drxRetransmissionTimer>	Integer type	
	Represents indexes in enums, which corresponds to PDCCH periods.	
	Index	PDCCH Periods
	0	0
	1	1
	2	2
	3	4
	4	6
	5	8

	6	16
	7	24
	8	33
	9	spare7
	10	spare6
	11	spare5
	12	spare4
	13	spare3
	14	spare2
	15	spare1
<drxULRetransmissionTimer>	Integer type	
	Represents indexes in enums, which corresponds to PDCCH periods.	
	Index	PDCCH periods
	0	256
	1	512
	2	1024
	3	1536
	4	2048
	5	3072
	6	4096
	7	4608
	8	6144
	9	7680
	10	8192
	11	9214
	12	spare4
	13	spare3
	14	spare2
	15	spare1
	16	1280_v1430

4.1.46. AT$QCPSSLPCFG

The set command PS sleep2/HIB configuration.

The read command returns current setting of each parameter.

The test command returns values supported as a compound value.

Table 4.46: AT$QCPSSLPCFG

AT$QCPSSLPCFG	Response
Set command AT$QCPSSLPCFG=<minT3324>,<T1>[,<minT3412>,<T2>[,<minEDRXPeriod>,<T3>]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCPSSLPCFG?	Response $QCPSSLPCFG: <minT3324>,<T1>,<minT3412>,<T2>,<minEDRXPeriod>,<T3> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCPSSLPCFG=?	Response $QCPTWEDRXS: (list of supported <param>s), OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

	String type	
<param>	"minT3324"	T3324 value in seconds. Range: 0-65535 Default: 0
	"minT3412"	T3412 value in seconds. Range: 0-65535 Default: 0
	"minEDRXPeriod"	eDRX value in seconds. When set to 65535, UE don't allow enter sleep2/HIB in IDLE state. Range: 0-65535 Default: 0

NOTE

1. "minT3324",<T1>
   If NW assigned T3324<=T1,don't allow enter Sleep2/HIB when T3324 is running.
   If NW is not assigned T3324(PSM is disable), allow enter Sleep2/HIB.
2. "minT3412",<T2>
   If NW assigned T3412/T3412ext<=T2, don't allow enter Sleep2/HIB when T3324 is running.
   If NW is not assigned T3324(PSM is disable), allow enter Sleep2/HIB.
3. "minEDRXPeriod",<T3>
   If NW assigned drx/edrx period<=T3, don't allow enter Sleep2/HIB when T3324 is running.
   If set T3 to 65535,don't allow enter Sleep2/HIB.
4. Summary:
   Condition allows enter Sleep2/HIB when T3324 is running : T3324>T1 && T3412>T2 && DRX/eDRX period>T3
   Condition allows enter Sleep2/HIB when NW not assign T3324(PSM is disable) : DRX/eDRX period>T3

Example

```
AT$QCPSSLPCFG=”minT3324”,0,”minT3412”,0,”minEDRXPeriod”,0
OK

AT$QCPSSLPCFG?
$QCPSSLPCFG:"minT3324",0,"minT3412",0,"minEDRXPeriod",0
OK

AT$QCPSSLPCFG=?
$QCPSSLPCFG:(0-65535),(0-4294967295),(0-65535)
OK
```

4.1.47. AT$QCNBIOTRAI

The command is used by the UE to request the NB-IOT network to quickly release the current RRC connect. The usage scenario are as follows:

- No information available
- UE sends one UL packet, and no DL packet is expected, then the NB-IOT network quickly releases the current RRC connect.

Table 4.47: AT$QCNBIOTRAI

AT$QCNBIOTRAI	Response
Set command AT$QCNBIOTRAI=<rai>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCNBIOTRAI=?	Response $QCNBIOTRAI(0-1) OK
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

	Integer type	
<rai>	0	No information available
	1	UE sends one UL packet and no DL packets expected

Example

```
AT$QCNBIOTRAI=1
OK

AT$QCNBIOTRAI=?
$QCNBIOTRAI: (0-1)
OK
```

4.1.48. AT$QCPLMNLOCK

Set command use to add/remove the PLMN in the list of PLMN in the table.

Read command display the current PLMN list from the table in NVM.

Table 4.48: AT$QCPLMNLOCK

AT$QCPLMNLOCK	Response
Set command AT$QCPLMNLOCK=<add/delete>,<mcc>,<mnc>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCPLMNLOCK?	Response $QCPLMNLOCK: <list_size> [PLMN list <PLMN1>,<PLMN2>…] OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<add/delete>	Integer type	
	0	Delete
	1	Add
<mcc>	String type	
	Indicate the mobile country code.	
<mnc>	String type	
	Indicate the mobile network code.	
<list_size>	String type	
	Indicate the mobile network code.	

Examples

```
AT$QCPLMNLOCK=1,405886
OK

AT$QCPLMNLOCK?
QCPLMNLOCK:<list 23>[PLMN list <405854>, <405855>, <405856>,<405872>,
<405857>, <405858>, <405859>, <405860>, <405861>,<405862>, <405873>,
<405863>, <405864>, <405874>, <405865>,<405866>, <405867>, <405868>,
<405869>, <405871>, <405870>,<405840>, <22201>, <000>, <000>, <000>,
<000>, <000>, <000>, <000>
OK
```

4.1.49. AT+CEINFO

The set command is used to enable/disable unsolicited report +CEINFO. (0=Disable, 1=Enable) Unsolicited report: +CEINFO: <Reporting>,<CE Enabled>,<UE State>,<Downlink Repetition Factor>,<Uplink Repetition Factor>,<RSRP<CINR>.

When +CEINFO=1 is received, unsolicited report +CEINFO shall be sent right away to indicate current status. After the initial report, unsolicited report shall be sent in following events:

- <CE Enabled> is changed
- <UE State> is changed
- <Downlink Repetition Factor> is changed
- <Uplink Repetition Factor> is changed

The read command returns the current CE Mode Information.

Table 4.49: AT+CEINFO

AT+CEINFO	Response
Set command AT+CEINFO=<value>	Response +CEINFO: <Reporting>,<CE Enabled>, <UE State>,<Downlink Repetition Factor>, <Uplink Repetition Factor>,<RSRP>,<CINR> OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+CEINFO?	Response +CEINFO: <Reporting>,<CE Enabled>, <UE State>,<Downlink Repetition Factor>, <Uplink Repetition Factor>,<RSRP>,<CINR> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

<Reporting>	Integer type	
	0	Disable
	1	Enable
<CE Enabled>	Integer type	
	0	Disable
	1	Enable
<UE State>	Char type	
	'I'	Idle
	'R'	RACH
	'C'	Connected
<Downlink Repetition Factor>	Integer type	
	0 to 255	Supported values
<Uplink Repetition Factor>	Integer type	
	0 to 255	Supported values

<RSRP>	Integer type	
	-156 to -44	Value in dBm
<CINR>	Integer type	
	-30 to 30	Value in dBm

4.1.50. AT$QCLAPISUPPORT

The set command is used to set the low access priority indicator (LAPI) value. If it is enabled, the module will not support LAPI even if EFNasconfig indicates LAPI support. If it is disabled, the module will work as per the SIM card.

The read command returns the current LAPI value.

Table 4.50: AT$QCLAPISUPPORT

AT$QCLAPISUPPORT	Response
Set command AT$QCLAPISUPPORT=<Lapi>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+QCLAPISUPPORT?	Response $QCLAPISUPPORT: <Lapi> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<Lapi>	Integer type	
	0	Disable
	1	Enable

Example

```
AT$QCLAPISUPPORT = 0
OK
```

4.1.51. AT$QCGCONTRDP

PDP context read dynamic parameters.

NOTE

This AT command is applicable for Verizon operator only.

The Set command returns the relevant information for an active non-secondary PDP context with the context identifier <cid>. If the mobile terminated (MT) has dual stack capabilities, at least one pair of lines with information is returned per <cid>. First one line with the IPv4 parameters followed by one line with the IPv6

parameters. If this MT with dual stack capabilities indicates more than two IP addresses of DNS servers, multiple of such pairs of lines are returned.

If the parameter <cid> is omitted, the relevant information for all active non secondary PDP contexts is returned.

The Test command returns a <cid> list associated with active non secondary contexts.

Table 4.51: AT$QCGCONTRDP

AT$QCGCONTRDP	Response
Set command AT$QCGCONTRDP[=<cid>]	Response [$QCGCONTRDP: <cid>,<bearer_id>,<apn> [,<local_addr_and_subnet_mask>[,<gw_addr> [,<DNS_prim_addr>[,<DNS_sec_addr> [,<PCSCF_prim_addr>[,<PCSCF_sec_addr> [,<IM_CN_Signalling_Flag>[,<LIPA_indication> [,<IPv4_MTU>[,<WLAN_Offload>[,<Local_Addr_Ind> [,<NonIP_MTU> [,<Serving_PLMN_rate_control_value>]]]]]]]]]]]]] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCGCONTRDP=?	Response $QCGCONTRDP: (a <cid> list associated with active contexts) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type	
	Specifies a particular PDP context definition.	
	1 … 11	Supported PDP context.
<bearer_id>	Integer type	
	Specifies a particular PDP context definition.	
<apn>	String type	
	A logical name that is used to select the GGSN or the external packet data network. The maximum configurable APN length is 99 bytes. If the value is null or omitted, then the subscription value will be requested.	
<local_addr_and_subnet_mask>	String type	
	The IP address and subnet mask of the MT.	
<gw_addr>	String type	
	The IP address of gateway.	
<DNS_prim_addr>	String type	
	The IP address of the primary DNS server.	
<DNS_sec_addr>	String type	
	The IP address of the primary DNS server.	
<P_CSCF_prim_addr>	String type	
	The IP address of the primary P-CSCF server.	

<P_CSCF_sec_addr>	String type
	The IP address of the secondary P-CSCF server.
<IM_CN_Signalling_Flag>	Integer type
	Shows whether the PDP context is for IM CN subsystem-related signaling only or not.
<LIPA_indication>	Integer type
	Indicates that the PDP context provides connectivity using a LIPA PDN connection.
<IPv4_MTU>	Integer type
	Shows the IPv4 MTU size in octets.
<WLAN_Offload>	Integer type
	Indicates whether traffic can be offloaded using the specified PDN connection through a WLAN or not.
<Local_Addr_Ind>	Integer type
	Indicates whether the MS and the network support local IP address in TFTs
<NonIP_MTU>	Integer type
	Shows the Non-IP MTU size in octets.
<Serving_PLMN_rate_control_value>	Integer type
	Indicates the maximum number of uplink messages the UE is allowed to send in a 6 minute interval.

4.1.52. AT$QCROAMINGDATA

Enable/disable data during roaming.

The Set command is used to enable/disable the data during roaming but not affect the network registering.

The Read command returns the current setting of <RoamData>.

Table 4.52: AT$QCROAMINGDATA

AT$QCLAPISUPPORT	Response
Set command AT$QCROAMINGDATA=<RoamData>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCROAMINGDATA?	Response $QCROAMINGDATA: <RoamData> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

	Integer type	
<RoamData>	0	Disable
	1	Enable

Example

```
AT$QCROAMINGDATA = 0
OK
```

4.1.53. AT$QCQOPS

The Set command is used to set the MCC and corresponding supported bands.

Read command returns the current settings of MCC and Band in NVM.

Table 4.53: AT$QCQOPS

AT$QCQOPS	Response
Set command AT$QCQOPS=<mcc>[,<band1>[,<band2>…]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCQOPS?	Response $QCQOPS: <mcc1>,<band1>,<band2>,… <mcc2>,<band1>,<band2>,… … OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<mcc>	Integer type	
	mcc in decimal number.	
	0 to 999	Supported values
<band>	Integer type	
	Band list in decimal numbers.	

Default values

Region	MCC	BANDS
USA	310	2, 5, 12, 13
EU	234	3, 20
Australia	505	3, 28
Japan	440	3, 8
China	460	3, 5, 8

Example

```
AT$QCQOPS?
$QCQOPS: 310,2,5,8,12,13
OK

AT$QCQOPS=310,2,5,12,13
OK
```

4.1.54. AT$QCDTRPM

Set command use to add/remove the PLMN in the list of PLMN in the table.

Read command display the current PLMN list from the table in NVM.

Table 4.54: AT$QCDTRPM

AT$QCDTRPM	Response
Set command AT$QCDTRPM=<add/delete>,<mcc>,<mnc>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCDTRPM?	Response $QCDTRPM: <list_size> [PLMN list <PLMN1>,<PLMN2>…] … OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	AUTO_SAVE

Parameter

<add/delete>	Integer type	
	0	Delete
	1	Add
<mcc>	String type	
	Indicate the mobile country code.	
<mnc>	String type	
	Indicate the mobile network code.	
<list_size>	String type	
	Indicate the mobile network code.	

Example

```
AT$QCDTRPM=1,90140
OK
```

4.1.55. AT+RPMPARAM

Set command used to set all RPM parameters.

Read command used to display all the RPM parameters.

Table 4.55: AT+RPMPARAM

AT+RPMPARAM	Response
Set command AT+RPMPARAM=[<rpmFlag>[,<N1>[,<T1>[,<T1_ext>[,<F1>[,<F2>[,<F3>[,<F4>]]]]]]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+RPMPARAM?	Response +RPMPARAM:<rpmFlag>,<N1>,<T1>,<T1_ext>,<F1>,<F2>,<F3>,<F4>,<isOmParamsValid>,<CBR1>,<CR1>,<CPDP1>,<CPDP2>,<CPDP3>,<CPDP4>,<LR1>,<LR2>,<LR3> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<rpmFlag>	Integer type	
	0	Disable
	1	Enable
<N1>	Integer type	
	Max no of SW resets per Hour allowed by RPM following permanent EMM reject.	
<T1>	Integer type	
	Average time before RPM resets modem following permanent EMM reject.	
<T1_ext>	Integer type	
	Average time before RPM resets modem following permanent EMM reject if T1 = 0xFF.	
<F1>	Integer type	
	Max number of PDN Connectivity Requests per Hour allowed by RPM following PDP Activation Ignore Scenario.	
<F2>	Integer type	
	Max number of PDN Connectivity Requests per Hour allowed by RPM following Permanent PDP Activation Reject.	
<F3>	Integer type	
	Max number of PDN Connectivity Requests per Hour allowed by RPM following Temporary PDP Activation Reject.	
<F4>	Integer type	
	Max number of PDN Connectivity Activation/ Deactivation Requests per Hour allowed by RPM	

<isOmParamsValid>	Integer type	
	If RPM params are present on SIM card. If not, all RPM OM functionality shall be disabled.	
	0	Disable
	1	Enable
<CBR1>	Integer type	
	Operational Management Counter related to N1.	
<CR1>	Integer type	
	Operational Management Counter related to T1	
<CPDP1>	Integer type	
	Operational Management Counters related to F1	
<CPDP2>	Integer type	
	Operational Management Counters related to F2	
<CPDP3>	Integer type	
	Operational Management Counters related to F3	
<CPDP4>	Integer type	
	Operational Management Counters related to F4	
<LR1>	Integer type	
	Leak rate(in Hours) for C-BR-1	
<LR2>	Integer type	
	Leak rate(in Hours) for C-R-1	
<LR3>	Integer type	
	Leak rate(in Hours) for CPDP-1 to C-PDP-4	

Example

```
AT+RPMPARAM = 1,1,,,60,,,
OK
```

4.1.56. AT+RPMVERSION

RPM version is read from sim card. Read command returns the current RPM version as 2.

Table 4.56: AT+RPMVERSION

AT+RPMVERSION	Response
Read command AT+RPMVERSION?	Response +RPMVERSION:<rpmversion> OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<rpmversion>	Integer type
	Stores RPM Version.

Example

```
AT+RPMVERSION?
+RPMVERSION: 2
OK
```

4.1.57. AT$QCUSATP

Set command transmits the <profile> to the MT to modify terminal profile of USAT.

The read command returns current setting of each parameter.

The test command returns values supported as a compound value.

Table 4.57: AT$QCUSATP

AT$QCUSATP	Response
Set command AT$QCUSATP=<length>,<profile>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+QCUSATP?	Response $QCUSATP: <length>,<profile> OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+QCUSATP?	Response OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<length>	Integer type
	Length of the characters that are sent to TE in <profile> (two times the actual length of the command or response)
<profile>	String type
	The profile describing the supported USAT facilities as specified for the terminal profile in 3GPP TS31.111 (hexadecimal format).

4.1.58. AT$QCSLTEST

Doppler Test:

Set command sets the speed estimation <Enable/Disable> and shows the valid speed, speed level, doppler, and SNR value in Result.

Attach Resume:

It is the empty signal we are sending to Lower Layer to resume the attach procedure when suspended by AT command.

Table 4.58: AT$QCSLTEST

AT$QCSLTEST	Response

Set command (Doppler Test)
`AT$QCSLTEST=14,<Disable/Enable>`
	Response
	`+QCDOPTEST: <SpeedValid>,<SpeedLvl>, <Doppler>,<SNR>`
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Set command (Attach Resume)
`AT$QCSLTEST=13`
	Response
	`OK`
	If there is an error, the response is as follows:
	`+CME ERROR: <err>`

Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<Disable/Enable>	Integer type	
	0	Disable
	1	Enable
<SpeedValid>	Integer type	
	0 to 165	Range
<SpeedLvl>	Integer type	
	0 to 155	Range when FC <= 1 GHz
	0 to 340	Range when FC > 1 GHz
<Doppler>	Integer type	
	-30 to 30	Range
<SNR>	Integer type	
	-20 to 40	Range in dB

Example

```
AT$QCSLTEST=14,1
+QCDOPTEST:0~165,0~155,-30~30,SNR
OK

AT$QCSLTEST=13
OK
```

4.2. SOCKETS COMMANDS (SOLUTION A)

4.2.1. AT+SKTCREATE

This command creates a socket on the UE and associates with the specified protocol. The UE supports up to five sockets (TCP or UDP) at the same time and returns an error if it is exceeded.

The Test command returns values supported as a compound value.

Table 4.59: AT+SKTCREATE

AT+SKTCREATE	Response
Set command AT+SKTCREATE=<domain>,<type>,<protocol>	Response +SKTCREATE: <fd> OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT+SKTCREATE=?	Response +SKTCREATE: (list of supported <domain>s), (list of supported <type>s), (list of supported <protocol>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE
<domain>	Integer type	
	1	IPV4
	2	IPV6
<type>	Integer type	
	1	TCP
	2	UDP
<protocol>	Integer type	
	Standard internet protocol definition.	
	6	IPPROTO_TCP
	17	IPPROTO_UDP

Example

```
AT+SKTCREATE=1,1,17
+SKTCREATE: 1
OK
```

4.2.2. AT+SKTCONNECT

For TCP, this command connects the socket with a remote address and port.

For UDP, this command saves remote address and port to send.

Table 4.60: AT+SKTCONNECT

AT+SKTCREATE	Response
Set command AT+SKTCONNECT=<fd>,<addr>,<port>	Response OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT+SKTCONNECT=?	Response +SKTCONNECT: (list of supported <fd>s), <addr>,(list of supported <port>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE
<addr>	String type	
	Remote address to connect or send to	
<port>	Integer type	
	Remote port to connect or send to	

4.2.3. AT+SKTBIND

This command binds socket with local address and port. If the address is default, it means any address.

Table 4.61: AT+SKTBIND

AT+SKTBIND	Response
Set command AT+SKTBIND=<fd>,<addr>,<port>	Response OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT+SKTBIND=?	Response +SKTCONNECT: (list of supported <fd>s), <addr>,(list of supported <port>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE
<addr>	String type	
	Address to bind. If address is defaults means any address.	
<port>	Integer type	
	Port to bind.	

4.2.4. AT+SKTSEND

Send a <data len> byte of data to the remote port on remote address.

Table 4.62: AT+SKTSEND

AT+SKTSEND	Response
Set command AT+SKTSEND=<fd>,<data len>,<data>[,<rai info>[,<except info>]]	Response OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT+SKTSEND=?	Response +SKTCONNECT: (list of supported <fd>s),(list of supported <data len>s),<data>,(list of supported <rai info>s),(list of supported <expect info>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE
<data len>	Integer type	
	Length of data in hexadecimal format, the max length is 512.	
<data>	Integer type	
	Data in hexadecimal format.	
<rai info>	Integer type	
	Release assistance indication.	
	0	No rai information. Default
	1	No further uplink or downlink data transmission after the uplink data transmission is expected
	2	Only a single downlink data transmission and no further uplink data transmission subsequent is expected
<except info>	Integer type	
	Expect data indication.	
	0	Disable expect data indication. Default
	1	Enable expect data indication.

Example

```
Data: 0123456789

AT+SKTSEND=0,A,30313233343536373839
OK
```

4.2.5. +SKTRECV

This is an unsolicited message to show that the socket has received data.

Table 4.63: +SKTRECV

Message	Parameter
+SKTRECV:	<fd>,<len>,<data>

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE
<len>	Integer type	
	Received data length (bytes)	
<data>	Integer type	
	Data in hexadecimal format.	

4.2.6. +SKTERR

This is an unsolicited message to show the error number when errors occur

Table 4.64: +SKTERR

Message	Parameter
+SKTERR:	<fd>,<errno>

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE
<errno>	Integer type	
	Received data length (bytes)	
	11	Operation would block
	12	Out of memory error
	22	Invalid argument
	62	Timer expired
	103	Software caused connection abort
	104	Connection reset by peer
	105	No buffer space available
	107	Transport endpoints are not connected
	113	No route to host
	115	Operation now in progress

4.2.7. AT+SKTSTATUS

Get the status of a socket by file description.

Table 4.65: AT+SKTSTATUS

AT+SKTSTATUS	Response
Set command AT+SKTSTATUS=<fd>	Response +SKTSTATUS: <status> OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT+SKTSTATUS=?	Response +SKTSTATUS: (list of supported <fd>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE
<status>	Integer type	
	1	Not connected
	2	Connecting
	3	Connected

4.2.8. AT+SKTDELETE

Delete a socket by file description.

Table 4.66: AT+SKTDELETE

AT+SKTSTATUS	Response
Set command AT+SKTDELETE=<fd>	Response +SKTDELETE: <status> OK If there is an error, the response is as follows: +SOCKET ERROR: <err>
Test command AT+SKTDELETE=?	Response +SKTDELETE: (list of supported <fd>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type	
	1 to 7	Socket file description returned by +SKTCREATE

4.3. LWM2M COMMANDS

4.3.1. AT+LWM2MCREATE

This command creates an instance of lwM2M client and registers with lwM2M server. You must specify <sever>, <port>, <enderpoint> name, and <lifetime>. If DTLS is needed, specify <psk id> and <psk>.

Table 4.67: AT+LWM2MCREATE

AT+LWM2MCREATE	Response
Set command AT+LWM2MCREATE=<server>,<port>,<local_port>,<enderpoint>,<lifetime>[,<psk_id>,<psk>]	Response +LWM2MCREATE: <clientId> OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MCREATE=?	Response +LWM2MCREATE: <server>, (list of supported <port>s), (list of supported <local_port>s), <enderpoint>,(list of supported <lifetime>s), <psk_id>,<psk> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type
	LwM2M client's id.
<server>	String type
	LwM2M server's URL or IP address.
<port>	Integer type
	LwM2M server's port number.
<local_port>	Integer type
	LwM2M client's local port.
<enderpoint>	String type
	LwM2M client's enderpoint name.
<lifetime>	Integer type
	LwM2M client's lifetime.
<psk_id>	String type
	LwM2M client's public identity.
<psk>	String type
	LwM2M client's pre shared key

Example

```
AT+LWM2MCREATE=”180.167.122.150”,5683,56830,”client0”,60
+LWM2MCREATE: 0
OK
```

4.3.2. AT+LWM2MDELETE

This command deletes a specified lwM2M client instance.

Table 4.68: AT+LWM2MDELETE

AT+LWM2MDELETE	Response
Set command AT+LWM2MDELETE=<clientId>	Response +LWM2MDELETE: <clientId> OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MDELETE=?	Response +LWM2MDELETE: (list of supported <clientId>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type
	LwM2M client's id returned by +LWM2MCREATE

4.3.3. AT+LWM2MADDOBJ

This command adds a an lwM2M object to a specified lwM2M client instance. For definitions of object, instance and resource. see Lightweight Machine to Machine Technical Specification, ext-label Objects Produced by IPSO Alliance and oma-label Objects Produced by OMA.

OMNA provides a unique Object or/and Resource Identifier: (Object ID) or (ResourceID):
http://www.openmobilealliance.org/wp/OMNA/LwM2M/LwM2MRegistry.html

Table 4.69: AT+LWM2MADDOBJ

AT+LWM2MADDOBJ	Response
Set command AT+LWM2MADDOBJ=<clientId>, <objectId>,<instanceId>, <resourceCount>,<resourceIds>	Response OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MADDOBJ=?	Response +LWM2MADDOBJ: (list of supported <clientId>s), (list of supported <objectId>s), (list of supported <instanceId>s), (list of supported <resourceCount>s), <resourceIds> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type
	LwM2M client's id returned by +LWM2MCREATE
<objectId>	Integer type
	Object id number
<instanceId>	Integer type
	Instance id number
<resourceCount>	Integer type
	Number of resources
<resourceIds>	Integer type
	Resources numbers separated by semicolons

Example

```
AT+LWM2MADDOBJ=0,3306,111,3,"5750;5850;5851"
OK
```

4.3.4. AT+LWM2MDELOBJ

This command deletes an object from a specified lwM2M client instance.

Table 4.70: AT+LWM2MDELOBJ

AT+LWM2MDELOBJ	Response
Set command AT+LWM2MDELOBJ=<clientId>,<objectId>	Response OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MDELOBJ=?	Response +LWM2MDELOBJ: (list of supported <clientId>s),(list of supported <objectId>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type
	LwM2M client's id returned by +LWM2MCREATE
<objectId>	Integer type
	Object id number

4.3.5. +LWM2MREAD

Indicator of LwM2M server's reading command.

This is an unsolicited message to represent that the lwM2M client has received the reading command of the LwM2M server.

Table 4.71: +LWM2MREAD

Message	Parameter
+LWM2MREAD:	<clientId>,<objectId>,<instanceId>,<resId>

Parameter

<clientId>	Integer type
	LwM2M client's id returned by +LWM2MCREATE
<objectId>	Integer type
	Object id number that lwM2M server wants to read
<instanceId>	Integer type
	Instance id number that lwM2M server wants to read
<resourceCount>	Integer type
	Resource id number that lwM2M server wants to read

Example

```
+LWM2MREAD: 0,3306,111,5750
```

4.3.6. +LWM2MWRITE

Indicator of LwM2M server's writing command

This is an unsolicited message to represent that the lwM2M client has received the writing command of the LwM2M server.

Table 4.72: +LWM2MWRITE

Message	Parameter
+LWM2MWRITE:	<clientId>,<objectId>,<instanceId>,<num>[,<resId>,<type>,<length>,<valueStr>]

Parameter

<clientId>	Integer type	
	LwM2M client's id returned by +LWM2MCREATE.	
<objectId>	Integer type	
	Object id number that lwM2M server wants to write.	
<instanceId>	Integer type	
	Instance id number that lwM2M server wants to write.	
<num>	Integer type	
	The number of resources need to be written.	
<resId>	Integer type	
	Resource id number that lwM2M server wants to write.	
<type>	String type	
	"S"	String type
	"O"	Opaque type
	"I"	Integer type
	"F"	Float type
<length>	Integer type	
	Value length in bytes.	

<valueStr>	String type
	Value needed to write to resource.

Example

```
+LWM2MWRITE: 0,3306,111,5750,O,4,”54595045”
```

4.3.7. +LWM2MEXECUTE

Indicator of LwM2M server's Execution command.

This is an unsolicited message to represent that the lwM2M client has received the execution command of the LwM2M server.

Table 4.73: +LWM2MEXECUTE

Message	Parameter
+LWM2MEXECUTE:	<clientId>,<objectId>,<instanceId>,<resId>,<length>,<valueStr>

Parameter

<clientId>	Integer type
	LwM2M client's id returned by +LWM2MCREATE.
<objectId>	Integer type
	Object id number that lwM2M server wants to execute.
<instanceId>	Integer type
	Instance id number that lwM2M server wants to execute.
<resId>	Integer type
	Resource id number that lwM2M server wants to execute.
<length>	Integer type
	Value length in bytes.
<valueStr>	String type
	Value of execute command.

Example

```
+LWM2MEXECUTE: 0,3303,0,5605,2,”ok”
```

4.3.8. +LWM2MOBSERVE

Indicator of LwM2M server's observation command.

This is an unsolicited message to represent that the lwM2M client has received the observation command of the LwM2M server.

Table 4.74: +LWM2MOBSERVE

Message	Parameter
+LWM2MOBSERVE:	<clientId>,<oper>,<objectId>,<instanceId>,<resId>

Parameter

<clientId>	Integer type	
	LwM2M client's id returned by +LWM2MCREATE.	
<oper>	Integer type	
	0	Observe object instance
	1	Cancel observe
<objectId>	Integer type	
	Object id number that lwM2M server wants to observe.	
<instanceId>	Integer type	
	Instance id number that lwM2M server wants to observe.	
<resId>	Integer type	
	Resource id number that lwM2M server wants to observe.	

Example

```
To observe 3306/111/5750:
+LWM2MOBSERVE: 0,0,3306,111,5750

Cancel observe 3306/111/5750:
+LWM2MOBSERVE: 0,1,3306,111,5750
```

4.3.9. AT+LWM2MREADCONF

This command responds to lwM2M server's Read command.

Table 4.75: AT+LWM2MREADCONF

AT+LWM2MREADCONF	Response
Set command AT+LWM2MREADCONF=<clientId>,<objectId>,<instanceId>,<resId>,<valuetype>,<valuelen>,<value>	Response OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MREADCONF=?	Response +LWM2MREADCONF: (list of supported <clientId>s), (list of supported <objectId>s), (list of supported <instanceId>s), (list of supported <resId>s), (list of supported <valuetype>s), (list of supported <valuelen>s),<value> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type	
	LwM2M client's id returned by +LWM2MCREATE.	
<objectId>	Integer type	
	Object id number.	
<instanceId>	Integer type	
	Instance id number.	
<resId>	Integer type	
	Resource id number.	
<valuetype>	Integer type	
	0	String
	1	Opaque
	2	Integer
	3	Float
	4	Bool
	Other	Undefined
<valuelen>	Integer type	
	Value length in bytes.	
<value>	String type	
	If <valuetype> is opaque, <value> must be hexadecimal string format.	

Example

```
Value type is string:
AT+LWM2MREADCONF=0,3306,0,5750,0,5,”hello”
OK

Value type is opaque:
AT+LWM2MREADCONF=0,12001,0,4,1,5,”3432383330”
OK

Value type is Integer:
AT+LWM2MREADCONF=0,3306,0,5851,2,3,”206”
OK

Value type is float:
AT+LWM2MREADCONF=0,3303,0,5601,3,4,”3.14”
OK

Value type is bool:
AT+LWM2MREADCONF=0,3306,0,5850,4,1,”1”
OK
```

4.3.10. AT+LWM2MWRITECONF

This command responds to lwM2M server's Write command

Table 4.76: AT+LWM2MWRITECONF

AT+LWM2MWRITECONF	Response
Set command AT+LWM2MWRITECONF=<clientId>,<result>	Response OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MWRITECONF=?	Response +LWM2MWRITECONF: (list of supported <clientId>s),(list of supported <objectId>s),(range of supported <result>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type	
	LwM2M client's id returned by +LWM2MCREATE.	
<result>	Integer type	
	Result of write command.	
	0x44	Write success
	0x8c	Time out
	0x84	Object not found
	Refer to RFC 7252	

Example

```
AT+LWM2MWRITECONF=0,68
OK
```

4.3.11. AT+LWM2MEXECUTECONF

This command responds to lwM2M server's Execute command

Table 4.77: AT+LWM2MEXECUTECONF

AT+LWM2MEXECUTECONF	Response
Set command AT+LWM2MEXECUTECONF=<clientId>,<result>	Response OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MEXECUTECONF=?	Response +LWM2MEXECUTECONF: (list of supported <clientId>s),(list of supported <objectId>s),(range of supported <result>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type	
	LwM2M client's id returned by +LWM2MCREATE.	
<result>	Integer type	
	Result of execute command.	
	0x44	Execute success
	0x8c	Time out
	0x84	Object not found
	Refer to RFC 7252	

Example

```
AT+LWM2MEXECUTECONF=0,68
OK
```

4.3.12. AT+LWM2MNOTIFY

Notifies the lwM2M server that a specified resource changed.

Table 4.78: AT+LWM2MNOTIFY

AT+LWM2MNOTIFY	Response
Set command AT+LWM2MNOTIFY=<clientId>,<objectId>,<instanceId>,<resId>,<valuetype>,<valuelen>,<value>	Response OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MNOTIFY=?	Response +LWM2MNOTIFY: (list of supported <clientId>s), (list of supported <objectId>s), (list of supported <instanceId>s), (list of supported <resId>s), (list of supported <valuetype>s), (list of supported <valuelen>s),<value> OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type
	LwM2M client's id returned by +LWM2MCREATE.
<objectId>	Integer type
	Object id number.
<instanceId>	Integer type
	Instance id number.
<resId>	Integer type
	Resource id number.

<valuetype>	Integer type	
	0	String
	1	Opaque
	2	Integer
	3	Float
	4	Bool
	Other	Undefined
<valuelen>	Integer type	
	Value length in bytes.	
<value>	String type	
	If <valuetype> is opaque, <value> must be hexadecimal string format.	

Example

```
AT+LWM2MNOTIFY=0,3303,0,5601,3,4,"3.14"
OK
```

4.3.13. AT+LWM2MUPDATE

This command updates the register information, with or without the object id's update.

Table 4.79: AT+LWM2MUPDATE

AT+LWM2MNOTIFY	Response
Set command AT+LWM2MUPDATE=<clientId>[,<withobj>]	Response OK If there is an error, the response is as follows: +LWM2M ERROR: <err>
Test command AT+LWM2MUPDATE=?	Response +LWM2MUPDATE: (list of supported <clientId>s), (list of supported <withobj>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type	
	LwM2M client's id returned by +LWM2MCREATE.	
<withobj>	Integer type	
	0	Not update the object information.
	1	Update the object information.

Example

```
AT+LWM2MUPDATE=0
OK
```

4.3.14. Summary of +LWM2M ERROR: <err> Codes

Table 4.80: +LWM2M ERROR: <err>

<err>	Description
ERROR	Wrong AT command input. For example, misspelled command.
PARAMETER ERROR	Wrong parameter input. For example, parameter out of range.
CANNOT CREATE SEMPH	Can't create semaphore.
CONFIG ERROR	LWM2M client configuration error.
NO FREE CLIENT	No free client left, now only support one client at a time.
OPERATION NO SUPPORT	Operation not supported, such as GET command.
NO FIND CLIENT	Can't find this client.
ADD OBJECT FAILED	Failed to add object.
NO FIND OBJECT	Can't find this object.
DELETE OBJECT FAILED	Failed to delete this object.
NETWORK NOT READY	Network not ready, can't use data service.

4.4. CoAP COMMANDS

4.4.1. AT+COAPCREATE

This command creates a CoAP client.

Table 4.81: AT+COAPCREATE

AT+COAPCREATE	Response
Set command AT+COAPCREATE=<local port>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COAPCREATE=?	Response +COAPCREATE:<1-65535> OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

<local port>	Integer type	
	1 to 65535	

4.4.2. AT+COAPDEL

This command deletes the CoAP client.

Table 4.82: AT+COAPDEL

AT+COAPCREATE	Response
Execution command AT+COAPDEL	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

4.4.3. AT+COAPADDRES

This command adds the CoAP resource.

Table 4.83: AT+COAPADDRES

AT+COAPADDRES	Response
Set command AT+COAPADDRES=<length>,<resource>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COAPADDRES=?	Response +COAPADDRES: <1-50>,"<resource>" OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

<length>	Integer type	
	1 to 50	The CoAP client resources.
<resource>	String type	
	The resource name.	

4.4.4. AT+COAPHEAD

This command adds the CoAP head.

Table 4.84: AT+COAPHEAD

AT+COAPHEAD	Response
Set command AT+COAPHEAD=<mode>[,[<msgid>][,<tkl>,<token>]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COAPHEAD=?	Response +COAPHEAD:<mode>[,[<msgid>][,<tkl>,<token>]] OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type	
	The CoAP Head and Token parameter.	
	1	Generate message id and token values randomly
	2	Generate message id, and configure the token values
	3	Only configure message id, not needed token values
	4	Configure message id, and generate the token values randomly
	5	Configure message and token values
<msgid>	Integer type	
	The message id only needed configure when the <mode> value is 3, 4, 5.	
	0 to 65535	Range
<tkl>	Integer type	
	The token values length, only needed configure when the <mode> value is 1, 2.	
	1 to 8	Range
<token>	String type	
	The token values, hexadecimal format string, only need configure when the <mode> value is 1, 2.	

4.4.5. AT+COAPOPTION

This command adds the CoAP option.

Table 4.85: AT+COAPOPTION

AT+COAPOPTION	Response
Set command AT+COAPOPTION= <opt cnt>,<opt name>,<opt value>[,...]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COAPOPTION=?	Response +COAPOPTION: <opt cnt>,<opt name>,"<opt value>"[,...] OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

<opt cnt>	Integer type	
	The option parameter count	
	1 to 12	Range
<opt name>	String type	
	Option name, refer the RFC 7252.	
	1	Option name, refer the RFC 7252.
	3	Uri-Host
	4	ETag
	5	If-None-Match
	6	Observe
	7	Uri-Port
	8	Location-Path
	11	Uri-Path
	12	Content-Format
	14	Max-Age
	15	Uri-Query
	17	Accept
	20	Location-Query
	23	Block2
	27	Block1
	28	SIZE
	35	Proxy-Uri
	39	Proxy-Scheme
	60	Size1
<opt name>	String type	
	The length of value string: 1-180. If the <opt_name> is 12 or 17, the <opt_value> must be the below value.	
	"0"	Text-plain
	"40"	Application/link-format
	"41"	Application/xml
	"42"	Application/octet-stream
	"47"	Application/ex
	"50"	Application/json

4.4.6. AT+COAPSEND

This command sends data to CoAP server.

Table 4.86: AT+COAPOPTION

AT+COAPHEAD	Response
Set command AT+COAPSEND= <msgType>,<method>, <ipAddr>,<port> Note: After ">" is responded, input the data to be sent. Press "CTRL+Z" to send or press "ESC" to cancel the operation	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COAPSEND=?	Response +COAPSEND: <msgType>,<method>, <ipAddr>,<port>[,<length>,<data>] OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

<msgType>	Integer type	
	0	CON, confirmable message(requires ACK/RST)
	1	NON, non-confirmable message(one-shot message)
	2	ACK, used to acknowledge confirmable message
	3	RST, reset, indicates error in received message
<method>	Integer type	
	1	GET
	2	POST
	3	PUT
	4	DELETE
<ipAddr>	String type	
	The CoAP Server IP address.	
<port>	Integer type	
	The CoAP Server Port.	
<length>	Integer type	
	The length of data to be sent, the max length is 512 Bytes.	
<data>	String type	
	The length of data to be sent in hexadecimal string.	

4.4.7. AT+COAPDATASTATUS

This command gets the CoAP status.

Table 4.87: AT+COAPDATASTATUS

AT+COAPDATASTATUS	Response
Test command AT+COAPDATASTATUS=?	Response +COAPDATASTATUS:<status> OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

	Integer type	
<status>	0	Have not sent
	1	Sent, waiting response of IoT platform (not supported)
	2	Sent failed (not supported)
	3	Timeout (not supported)
	4	Success
	5	Got reset message (not supported)

4.4.8. AT+COAPCFG

This command configures the CoAP client.

Table 4.88: AT+COAPCFG

AT+COAPCFG	Response
Set command AT+COAPCFG="Showra"[,<Showra>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Set command AT+COAPCFG="Showrspopt"[,<Showrspopt>]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT+COAPCFG?	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COAPCFG=?	Response OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

<Showra>	Integer type
	Set whether to display the address of sender.
<Showrspopt>	Integer type
	Set whether to display the CoAP option of sender.

4.4.9. AT+COAPALISIGN

This command gets the AliCloud sign.

Table 4.89: AT+COAPALISIGN

AT+COAPALISIGN	Response
Set command AT+COAPALISIGN=<devId>,<devName>,<devSecret>,<productKey>	Response +COAPALISIGN: "<sign>" OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT+COAPCFG=?	Response +COAPALISIGN: <devId>,<devName>,<devSecret>,<productKey> OK
Maximum Response Time	10 s
Parameter Saving Mode	NO_SAVE

Parameter

<devId>	String type
	Device ID issued by AliCloud.
<devName>	String type
	Device name issued by AliCloud.
<devSecret>	String type
	Device secret key issued by AliCloud.
<productKey>	String type
	Product key issued by AliCloud.
<sign>	String type
	The calculated sign value.

4.4.10. +COAPURC

This is an unsolicited message to indicate CoAP client receive data from CoAP server.

Table 4.90: +COAPURC

Message	Parameter
+COAPURC:	"rsp",[<ip_addr>,<port>,]<type>,<rspcode>,<msgid>[,<opt_cnt>,<opt_name>,"<opt_value>"[,...]][,<length>,<data>]

Parameter

<ip_addr>	String type	
	The CoAP server IP address, it shows when set AT+COAPCFG="Showra",1	
<port>	Integer type	
	The CoAP server port, it shows when set AT+COAPCFG="Showra",1	
<type>	Integer type	
	The CoAP Protocol of message type.	
	0 to 3	Range according to RFC 7252
<rspcode>	String type	
	The response code of CoAP Protocol. Refer to the RFC 7252.	

<method>	Integer type	
	The method of CoAP Protocol. Refer to the RFC 7252	
	1	GET
	2	POST
	3	PUT
	4	DELETE
<msgid>	Integer type	
	The CoAP message id.	
<mode>	Integer type	
	Indicates the existence of token, option, and data. Hexadecimal format.	
	Bit 0: The existence of token. Bit 1-6: The count of option. Bit 7: The existence of data.	
<tkl>	Integer type	
	The token value length.	
<token>	Integer type	
	The token value. Hexadecimal format.	
<opt_cnt>	Integer type	
	The count of option, it shows when set AT+COAPCFG="Showrspopt",1.	
<opt_name>	Integer type	
	The option name, it shows when set AT+COAPCFG="Showrspopt",1.	
<opt_value>	String type	
	The option value, it shows when set AT+COAPCFG="Showrspopt",1.	
<length>	Integer type	
	The data length. The max length is 512 bytes.	
<data>	String type	
	Received data from server.	

4.5. MQTT COMMANDS

4.5.1. AT$QCMTCFG

This command creates an MQTT client OR configures a network for the MQTT client.

Table 4.91: AT$QCMTCFG

AT$QCMTCFG	Response
Set command AT$QCMTCFG="mode",<value>,[…]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Set command AT$QCMTCFG="echomode", <tcpconnectID>,[<echo_mode>]	Response OK If <echo_mode> is omitted, query the data echo mode: $QCMTCFG: "echomode",<echo_mode> If there is an error, the response is as follows: +CME ERROR: <err>

Set command AT$QCMTCFG="dataformat",<tcpconnectID>,[<send_format>,[<recv_format>]]	Response OK If <send_format> and recv_format> are both fitted, query the format of sent/received data: $QCMTCFG: "dataformat",<send_format>,<recv_format> OK If there is an error, the response is as follows: +CME ERROR: <err>
Set command AT$QCMTCFG="keepalive",<tcpconnectID>[,<keep-alive time>]	Response OK If <keep-alive time> is omitted, query the keep-alive time: $QCMTCFG: "keepalive",<keep-alive time> OK If there is an error, the response is as follows: +CME ERROR: <err>
Set command AT$QCMTCFG="session",<tcpconnectID>[,<clean_session>]	Response OK If <clean_session> is omitted, query the session type: $QCMTCFG: "session",<clean_session> OK If there is an error, the response is as follows: +CME ERROR: <err>
Set command AT$QCMTCFG="timeout",<tcpconnectID>[,<pkt_timeout>[,<retry_times>][,<timeout_notice>]]	Response OK If <pkt_timeout>, retry_times>, <timeout_notice> are omitted, query the timeout value of message delivery: $QCMTCFG: "timeout",<pkt_timeout>,<retry_times>,<timeout_notice> OK If there is an error, the response is as follows: +CME ERROR: <err>
Set command AT$QCMTCFG="will",<tcpconnectID>[,<will_fg>[,<will_qos>,<will_retain>,"<will_topic>","<will_msg>"]]	Response OK If <echo_mode> is omitted, query the data echo mode: $QCMTCFG:"will",<will_fg>[,<will_qos>,<will_retain>,<will_topic>,<will_msg>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Set command AT$QCMTCFG="version",<tcpconnectID>[,<version>]	Response OK If <version> is omitted, query the MQTT protocol version: $QCMTCFG: "version",<version> OK If there is an error, the response is as follows: +CME ERROR: <err>

Set command AT$QCMTCFG="aliauth", <tcpconnectID> [,"<product_key>", "<device_name>", "<device_secret>"]	Response OK If "<product_key>", "<device_name>", "<device_secret>" are omitted, query the device information: $QCMTCFG: "aliauth",<product_key>, <device_name>,<device_secret> If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTCFG=?	Response $QCMTCFG: "echomode",(0),(0,1) $QCMTCFG: "dataformat",(0),(0,1), (0,1) $QCMTCFG: "keepalive",(0),(0-3600) $QCMTCFG: "session",(0),(0,1) $QCMTCFG: "timeout",(0),(1-60), (1-10),(0,1) $QCMTCFG: "will",(0),(0,1),(0-2), (0,1),"will_topic","will_msg" $QCMTCFG: "version",(0),(3,4) $QCMTCFG: "aliauth",(0),"productkey", "devicename","devicesecret" OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type	
	MQTT socket identifier, the value is 0.	
<echo_mode>	Integer type	
	Whether to echo the input data to UART in data mode.	
	0	Do not echo the input data to UART
	1	Echo the input data to UART
<send_format>	Integer type	
	The format of sent data	
	0	Text mode
	1	Hexadecimal mode
<recv_format>	Integer type	
	The format of received data	
	0	Text mode
	1	Hexadecimal mode
<keep-alive time>	Integer type	
	Defines the maximum time interval between messages received from a client. If the server does not receive a message from the client within 1.5 times of the keep-alive time period, it disconnects the client as if the client has sent a DISCONNECT message.	
	0 to 3600	Time range in seconds. Default: 120

<clean_session>	Integer type	
	Configure the session type.	
	0	The server must store the subscriptions of the client after it disconnects
	1	The server must discard any previously maintained information
<pkt_timeout>	Integer type	
	Timeout of the packet delivery.	
	0 to 60	Time range in seconds. Default is 10
<retry_times>	Integer type	
	Retry times when packet delivery times out.	
	0 to 10	Retry times. Default is 3
<timeout_notice>	Integer type	
	0	Not report timeout message when transmitting packet
	1	Report timeout message when transmitting packet
<will_fg>	Integer type	
	Configure the Will flag.	
	0	Ignore the Will flag configuration
	1	Require the Will flag configuration
<will_qos>	Integer type	
	Quality of service for message delivery.	
	0	At most once
	1	At least once
	2	Exactly once
<will_retain>	Integer type	
	The Will retain flag is only used on PUBLISH messages.	
	0	Server does not hold on to the message after it has been delivered to the current subscribers
	1	Server must hold on to the message after it has been delivered to the current subscribers
<will_topic>	String type	
	Will topic string	
<will_msg>	String type	
	The Will message defines the content of the message that is published to the will topic if the client is unexpectedly disconnected. It can be a zero-length message	
<version>	Integer type	
	Version of MQTT protocol.	
	3	MQTT v3.1
	4	MQTT v3.1.1. Default
<product_key>	Integer type	
	Product key issued by AliCloud.	
<device_name>	Integer type	
	Device name issued by AliCloud.	
<device_secret>	Integer type	
	Device secret key issued by AliCloud.	

4.5.2. AT$QCMTOPEN

This command is used to open a network for the MQTT client.

Table 4.92: AT$QCMTOPEN

AT$QCMTOPEN	Response
Set command AT$QCMTOPEN=<tcpconnectID>,"<host_name>",<port>	Response $QCMTOPEN: <tcpconnectID>,<result> OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCMTOPEN?	Response [$QCMTOPEN: <tcpconnectID>,"<host_name>",<port>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTOPEN=?	Response $QCMTOPEN: (list of supported <tcpconnectID>s),"<host_name>",(list of supported <port>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type	
	MQTT socket identifier. Set as 0.	
<host_name>	String type	
	The address of the server. It could be an IP address or a domain name. The maximum size is 100 bytes.	
<port>	Integer type	
	1 to 65535	The port of the server.
<result>	Integer type	
	-1	Failed to open network
	0	Opened network successfully

4.5.3. AT$QCMTCLOSE

This command is used to close a network for the MQTT client.

Table 4.93: AT$QCMTCLOSE

AT$QCMTCLOSE	Response
Set command AT$QCMTCLOSE=<tcpconnectID>	Response $QCMTCLOSE: <tcpconnectID>,<result> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTCLOSE=?	Response $QCMTCLOSE: (list of supported <tcpconnectID>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type
	MQTT socket identifier. Set as 0.

4.5.4. AT$QCMTCONN

This command is used to connect the MQTT client to a network.

Table 4.94: AT$QCMTCONN

AT$QCMTCONN	Response
Set command AT$QCMTCONN=<tcpconnectID>,"<clientID>"[,"<username>"[,"<password>"]]	Response $QCMTCONN: <tcpconnectID>,<result>[,<ret_code>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCMTCONN?	Response [$QCMTCONN: <tcpconnectID>,<state>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTCONN=?	Response $QCMTCONN: (list of supported <tcpconnectID>s),"<clientID>"[,"<username>"[,"<password>"]] OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type
	MQTT socket identifier. Set as 0.
<clientID>	String type
	The client identifier.
<username>	String type
	Username of the client. It can be used for authentication.

<password>	Integer type
	Password corresponding to the username of the client. It can be used for authentication.

4.5.5. AT$QCMTDISC

This command is used to disconnect the MQTT client from a network.

Table 4.95: AT$QCMTDISC

AT$QCMTDISC	Response
Set command AT$QCMTDISC=<tcpconnectID>	Response $QCMTDISC: <tcpconnectID>,<result> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTDISC=?	Response $QCMTDISC: (list of supported <tcpconnectID>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type
	MQTT socket identifier. Set as 0.

4.5.6. AT$QCMTSUB

This command sends MQTT subscribe packet.

Table 4.96: AT$QCMTSUB

AT$QCMTSUB	Response
Set command AT$QCMTSUB=<tcpconnectID>, <msgID>,"<topic>",<qos>	Response $QCMTSUB: <tcpconnectID>,<msgID>, <result>[,<value>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTSUB=?	Response $QCMTSUB: (list of supported <tcpconnectID>s), (list of supported <msgID>s),"<topic>", (list of supported <qos>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type	
	MQTT socket identifier. Set as 0.	
<msgID>	Integer type	
	1 to 65535	Message identifier of packet.
<topic>	String type	
	Topic that the client wants to subscribe to or unsubscribe from. The maximum length is 255 bytes.	
<qos>	Integer type	
	Message QoS. Can be 0, 1 or 2.	

4.5.7. AT$QCMTUNS

This command sends MQTT unsubscribe packet.

Table 4.97: AT$QCMTUNS

AT$QCMTUNS	Response
Set command AT$QCMTUNS=<tcpconnectID>,<msgID>,"<topic>"	Response $QCMTUNS: <tcpconnectID>,<msgID>,<result> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTUNS=?	Response $QCMTUNS: (list of supported <tcpconnectID>s),(list of supported <msgID>s),"<topic>" OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type	
	MQTT socket identifier. Set as 0.	
<msgID>	Integer type	
	1 to 65535	Message identifier of packet.
<topic>	String type	
	Topic that the client wants to subscribe to or unsubscribe from. The maximum length is 255 bytes.	

4.5.8. AT$QCMTPUB

This command sends MQTT publish packet.

Table 4.98: AT$QCMTPUB

AT$QCMTPUB	Response
Set command AT$QCMTPUB=<tcpconnectID>,<msgID>,<qos>,<retain>,"<topic>","<payload>"	Response $QCMTPUB: <tcpconnectID>,<msgID>,<result>[,<value>] OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCMTPUB=?	Response $QCMTPUB: (list of supported <tcpconnectID>s),(list of supported <msgID>s),(list of supported <qos>s),(list of supported <retain>s),"<topic>","<msg>" OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type	
	MQTT socket identifier. Set as 0.	
<msgID>	Integer type	
	1 to 65535	Message identifier of packet.
<qos>	Integer type	
	Message QoS. Can be 0, 1 or 2.	
<retain>	Integer type	
	0	Server must not retain the message
	1	Server must retain the message
<topic>	String type	
	Topic that the client wants to subscribe to or unsubscribe from. The maximum length is 255 bytes.	
<payload>	String type	
	Message that needs to be published. Maximum length is 700 bytes. If in data mode, the maximum length is 1024 bytes	

4.5.9. $QCMTSTAT

This is an unsolicited message to indicate that the MQTT client receives data from MQTT server.

Table 4.99: $QCMTSTAT

Message	Parameter
$QCMTSTAT:	<tcpconnectID>,<err_code>

Parameter

<tcpconnectID>	Integer type	
	MQTT socket identifier. Value is 0.	
<err_code>	Integer type	
	1	Connection is closed or reset by peer

4.5.10. $QCMTRECV

This is a message in response towards the data sent from client to server.

Table 4.100: $QCMTRECV

Message	Parameter
`$QCMTRECV:`	`<tcpconnectID>,<msgID>,<topic>,<data>`

Parameter

<tcpconnectID>	Integer type	
	MQTT socket identifier. Value is 0.	
<msgID>	Integer type	
	`1 to 65535`	Message identifier of packet.
<topic>	String type	
	The topic that received from MQTT server.	
<data>	String type	
	Receive data from server.	

4.6. HTTP COMMANDS

4.6.1. AT+HTTPCREATE

The Set command creates an http or https client instance, Configure host, server certification, and so on. It can create most 5 instances at one time.

The Test command returns values supported as a compound value.

Table 4.101: AT+HTTPCREATE

AT+HTTPCREATE	Response

Set command
`AT+HTTPCREATE=<host>[,<authuser>,<authpasswd>]`
	Response
	If there are more commands need to enter:
	`+HTTP CMD: CONTIUE ENTER CMD`
	If all command has entered:
	`+HTTPCREATE: <httpclientId>`
	If there is an error, the response is as follows:
	`+HTTP ERROR: <err>`

Test command
`AT+HTTPCREATE=?`
	Response
	`+HTTPCREATE: "<host>","<authuser>","<authpasswd>"`
	`OK`

Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<host>	String type	
	HTTP server's host name.	
<authuser>	String type	
	Authentication username.	
<authpasswd>	String type	
	Authentication password.	
<httpclientId>	Integer type	
	0 to 4	HTTP client ID.

Example

```
AT+HTTPCREATE="http://api.openweathermap.org:80"
+HTTPCREATE: 0
OK
```

4.6.2. AT+HTTPCON

The Set command creates a socket and connects with an http server and creates a task to receive data come from http server.

Test command returns values supported as a compound value.

Table 4.102: AT+HTTPCON

AT+HTTPCON	Response
Set command AT+HTTPCON=<httpclientId>	Response OK If there is an error, the response is as follows: +HTTP ERROR: <err>
Test command AT+HTTPCON=?	Response +HTTPCON: (list of supported <httpclientId>s) OK
Maximum Response Time	40 s
Parameter Saving Mode	NO_SAVE

Parameter

<httpclientId>	Integer type
	HTTP client id returned by +HTTPCREATE

Example

```
AT+HTTPCON=0
OK
```

4.6.3. AT+HTTPDESTROY

The Set command closes a socket, stops receive data from the http server, and frees the memory that was allocated by the client during creation.

Test command returns values supported as a compound value.

Table 4.103: AT+HTTPDESTROY

AT+HTTPDESTROY	Response
Set command AT+HTTPDESTROY=<httpclientId>	Response OK If there is an error, the response is as follows: +HTTP ERROR: <err>
Test command AT+HTTPDESTROY=?	Response +HTTPDESTROY: (list of supported <httpclientId>s) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<httpclientId>	Integer type
	HTTP client id returned by +HTTPCREATE

Example

```
AT+HTTPDESTROY=0
OK
```

4.6.4. AT+HTTPSEND

Set command sends data to the http server.

Test command returns values supported as a compound value.

NOTE

Only one send command can be processed before the related receiving is complete. For example, 2nd AT+HTTPSEND=xxx returns +HTTP ERROR: SEND FAILED.

Table 4.104: AT+HTTPSEND

AT+HTTPSEND	Response
Set command AT+HTTPSEND=<httpclientId>, <method>,<pathlen>,<path>, <customheaderlen>, <customheader>,<contentTypelen>, <contentType>,<contentlen>, <content>	Response OK If there is an error, the response is as follows: +HTTP ERROR: <err>
Test command AT+HTTPSEND=?	Response +HTTPSEND: (list of supported <httpclientId>s), (list of supported <method>s), (range of supported <pathlen>),"<path>", (range of supported <customheaderlen>), "<customheader>", (range of supported <contentTypelen>), "<contentType>", (range of supported <httpclientId>), "<content>" OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<httpclientId>	Integer type	
	HTTP client id returned by +HTTPCREATE.	
<method>	Integer type	
	0	GET
	1	POST
	2	PUT
	3	DELETE
	4	HEAD
<pathlen>	Integer type	
	0 to 260	Length of path.
<path>	String type	
	Path.	
<customheaderlen>	Integer type	
	0 to 255	Length of custom header.
<customheader>	String type	
	Customheader in hexadecimal string.	
<contentTypelen>	Integer type	
	0 to 64	Length of content type.
<contentType>	String type	
	Content type.	
<contentlen>	Integer type	
	0 to 1024	Length of content.

<content>	String type
	User data need to send in hexadecimal string.

Example

```
AT+HTTPSEND=0,0,89,
"/data2.5/weather?q=shanghai&appid=c592e14137c3471fa9627b44f6649db4&mode
=xml&units=metric"
OK
```

4.6.5. +HTTPRESPH

This is an unsolicited message to represent response header.

Table 4.105: +HTTPRESPH

Message	Parameter
+HTTPRESPH:	<httpclientId>,<responseCode>,<headerlen>,<header>

Parameter

<httpclientId>	Integer type
	HTTP client id returned by +HTTPCREATE.
<responseCode>	Integer type
	HTTP response code.
<headerlen>	Integer type
	HTTP response header length.
<header>	String type
	Header

4.6.6. +HTTPRESPC

Indicator of response header

This is an unsolicited message to represent response content.

Table 4.106: +HTTPRESPC

Message	Parameter
+HTTPRESPC:	<httpclientId>,<flag>,<contentlength>,<blockcontentlen>,<content>

Parameter

<httpclientId>	Integer type	
	HTTP client id returned by +HTTPCREATE.	
<flag>	Integer type	
	0	No more data
	1	Has more data
<contentlength>	Integer type	
	Length of content.	

<blockcontentlen>	Integer type
	Current block length
<content>	Integer type
	Content data string, which is converted from content hex data, the length is 2 times the original hex data.

4.6.7. +HTTPERR

This is an unsolicited message to represent an error message when an error occurs.

Table 4.107: +HTTPRESPC

Message	Parameter
+HTTPRESPC:	<httpclientId>,<errorcode>

Parameter

<httpclientId>	Integer type	
	HTTP client id returned by +HTTPCREATE.	
<errorcode>	Integer type	
	2	URL parse error
	4	Protocol error
	8	Connection timeout
	9	Connection error
	10	Connection fatal error
	11	Connection closed
	12	Need get more data
	13	Buffer overflow error
	14	Has more data

4.7. SOCKET COMMAND (SOLUTION B)

4.7.1. AT$QCSOCR

This command creates a socket on the UE and associates with specified protocol. If the port is set, receiving is enabled and "$QCSONMI" unsolicited messages appear for any message that is received on that port.

Table 4.108: AT$QCSOCR

AT$QCSOCR	Response
Set command AT$QCSOCR=<type>,<protocol>,<listen_port>[,<receive_control>[,<af_type>[,<ip_address>]]]	Response <socket_id> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSOCR=?	Response $QCSOCR: (list of supported <type>), (list of supported <protocol>), (list of supported <listen_port>), (list of supported <receive_control>), (list of supported <contentTypelen>), (list of supported <af_type>), "<ip_addr>" OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<type>	String type	
	"DGRAM"	UDP
	"STREAM"	TCP
<protocol>	Integer type	
	Standard internet protocol definition.	
	17	UDP
	6	TCP
<listen_port>	Integer type	
	This is the local port that is included in sent messages and on which messages are received. If it is 0 or omitted, the module assigns a random <listen_port> for this socket	
	0 to 65535	Range of supported values
<receive_control>	Integer type	
	0	The incoming messages are ignored
	1	The incoming messages are received. Default
<af_type>	String type	
	"AF_INET"	IPv4. Default
	"AF_INET6"	IPv6.
<ip_address>	String type	
	The IP address of the network assigned to UE.	

<socket_id>	Integer type
	Socket identification. A maximum of 5 sockets are supported, but other serviced may reduce this number.

Example

```
AT$QCSOCR="DGRAM",17,2233,1,"AF_INET"
1
OK
```

4.7.2. AT$QCSOST

This command sends a UDP datagram containing length bytes of data to <remote_port> on <remote_addr>.

This command sends a UDP datagram containing length bytes of data to the specified host and port. It returns 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, the return value of AT$QCSOST indicates how much of the data was successfully sent.

Table 4.109: AT$QCSOST

AT$QCSOST	Response
Set command AT$QCSOST=<socket_id>, <remote_addr>,<remote_port>, <length>,<data>[,<sequence> [,<segment_id>,<segment_num>]]	Response <socket_id>,<length> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSOCR=?	Response $QCSOCR: (list of supported <socket_id>), (list of supported <remote_addr>), (list of supported <remote_port>), (list of supported <length>),"<data>", (list of supported <sequence>), (list of supported <segment_id>), (list of supported <segment_num>), OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<remote_addr>	String type	
	Remote IP address.	
<remote_port>	Integer type	
	0 to 65535	Remote port on which messages are received.
<length>	Integer type	
	1 to 950	Decimal length of data to be sent.

<data>	String type	
	Data to be sent in hexadecimal string format.	
<sequence>	Integer type	
	1 to 255	If it is omitted, data sent is not reported. If not omitted, when datagram is sent over RF or is discarded, then the result is reported: $QCSOSTR: <socket_id>,<sequence>,<status>
<segment_id>	Integer type	
	1 to 4	One segment index of a segment message.
<segment_num>	Integer type	
	1 to 4	The total number which the messages fragment.
<status>	Integer type	
	0	The sent status of datagram is fail.
	1	The sent status of datagram is success.

Example

```
AT$QCSOST=1,”180.167.122.150”,5002,2,”ABAB”
1,2
OK
```

4.7.3. AT$QCSOSTF

This command sends a UDP datagram containing length bytes of data to <remote_port> on <remote_addr> and allows metadata flags to be sent.

This command sends a UDP datagram containing length bytes of data to the specified host:port. It returns 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, the return value of AT$QCSOSTF indicates how much of the data was successfully sent.

Table 4.110: AT$QCSOSTF

AT$QCSOSTF	Response
Set command AT$QCSOSTF=<socket_id>, <remote_addr>,<remote_port>, <flag>,<length>,<data>[,<sequence> [,<segment_id>,<segment_num>]]	Response <socket_id>,<length> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSOSTF=?	Response $QCSOCR: (list of supported <socket_id>), (list of supported <remote_addr>), (list of supported <remote_port>), (list of supported <flag>), (list of supported <length>),“<data>”, (list of supported <sequence>), (list of supported <segment_id>), (list of supported <segment_num>), OK
Maximum Response Time	5 s

Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<remote_addr>	String type	
	Remote IP address.	
<remote_port>	Integer type	
	0 to 65535	Remote port on which messages are received.
<flag>	Integer type	
	Specifies the type of message transmission. Values of this argument are in hexadecimal format and are formed by making logical OR with zero or more of the following flags. If no flags are set, enter value 0.	
	0x100	Exception Message: Send messages with high priority
	0x200	Release Indicator: indicate release after next message
	0x400	Release Indicator: indicate that release after next messages has been replied to
<length>	Integer type	
	1 to 950	Decimal length of data to be sent.
<data>	String type	
	Data to be sent in hexadecimal string format.	
<sequence>	Integer type	
	1 to 255	If it is omitted, data sent is not reported. If not omitted, when datagram is sent over RF or is discarded, then the result is reported: $QCSOSTR: <socket_id>,<sequence>,<status>
<segment_id>	Integer type	
	1 to 4	One segment index of a segment message.
<segment_num>	Integer type	
	1 to 4	The total number which the messages fragment.
<status>	Integer type	
	0	The sent status of datagram is fail.
	1	The sent status of datagram is success.

Example

```
AT$QCSOSTF=1,”180.167.122.150”,5002,0x100,2,”ABAB”
1,2
OK
```

4.7.4. AT$QCQSOS

This command queries the list of the pending upstream messages

Table 4.111: AT$QCQSOS

AT$QCSOSTF	Response
Set command AT$QCSOSTF=<socket_id>[,<socket_id>[,<socket_id>,[<…>]]]	Response $QCQSOS: <socket_id>,<sequence> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCQSOS=?	Response $QCQSOS: (list of supported <socket_id>) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<sequence>	Integer type	
	1 to 255	If it is omitted, data sent is not reported. If not omitted, when datagram is sent over RF or is discarded, then the result is reported: $QCSOSTR: <socket_id>,<sequence>,<status>

Example

```
AT$QCQSOS=1
$QCQSOS:1,2
OK
```

4.7.5. AT$QCSORF

This command can read up to <req_length> characters of data from <socket>, and the returned length is the actual number of characters returned.

This command is used to receive data on a socket. When data arrives, a "$QCSONMI" response is generated to indicate the socket the message was received on and the amount of data. The AT$QCSORF command takes a length, which is the maximum amount of data that is 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 is returned, plus an indication of the number of bytes remaining. Once a message has been fully read, a new "$QCSONMI" notification is 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 is discarded.

Table 4.112: AT$QCSORF

AT$QCSORF	Response
Set command AT$QCSORF=<socket_id>,<req_length>	Response <socket_id>,<ip_addr>,<port>, <length>,<data>,<remaining_length> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSORF=?	Response $QCSORF: (list of supported <socket_id>), (list of supported <req_length>) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<req_length>	Integer type	
	1 to 1357	Decimal length of data which wants to read.
<ip_addr>	String type	
	Remote IP address.	
<port>	Integer type	
	0 to 65535	Remote port on which messages are sent from.
<length>	Integer type	
	1 to 1358	Decimal length of data to be read.
<data>	String type	
	Data to be sent in hexadecimal string format.	
<remaining_length>	Integer type	
	1 to 1357	Amount of data left to read for this message as a decimal byte length. Remaining length is always 0. The remaining data is readable.

Example

```
AT$QCSORF=1,4
1,”180.167.122.150”,5002,4,”ABABABAB”,0
OK
```

4.7.6. AT$QCSOCO

This command connects a TCP server to the specified host and port

Table 4.113: AT$QCSOCO

AT$QCSOCO	Response
Set command AT$QCSOCO=<socket_id>,<remote_addr>,<remote_port>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSOCO=?	Response $QCSOCO: (list of supported <socket_id>), (list of supported <remote_addr>), (list of supported <remote_port>) OK
Maximum Response Time	30 s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<remote_addr>	String type	
	Remote IP address.	
<remote_port>	Integer type	
	0 to 65535	Remote port to connect.

Example

```
AT$QCSOCO=1,”180.167.122.150”,5002
OK
```

4.7.7. AT$QCSOSD

This command sends a TCP datagram to the TCP server. It returns 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$QCSOSD return value indicates how much the data was successfully sent.

If <sequence> is not omitted, when the datagram is asked for by the server or is discarded by UE, the result is reported.

Table 4.114: AT$QCSOSD

AT$QCSOSD	Response
Set command AT$QCSOSD=<socket_id>,<length>,<data>[,<flag>[,<sequence>]]	Response <socket_id>,<length> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSOSD=?	Response $QCSOSD: (list of supported <socket_id>),(list of supported <length>),"<data>",(list of supported <flag>),(list of supported <sequence>) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<length>	Integer type	
	1 to 950	Decimal length of data to be sent.
<data>	String type	
	Data to be sent in hexadecimal string format.	
<flag>	Integer type	
	Specifies the type of message transmission. Values of this argument are in hexadecimal format and are formed by making logical OR with zero or more of the following flags. If no flags are set, enter value 0.	
	0x100	Exception Message: Send messages with high priority
	0x200	Release Indicator: indicate release after next message
	0x400	Release Indicator: indicate that release after next messages has been replied to
<sequence>	Integer type	
	1 to 255	If it is omitted, data sent is not reported. If not omitted, when datagram is sent over RF or is discarded, then the result is reported: $QCSOSTR: <socket_id>,<sequence>,<status>
<status>	Integer type	
	0	The sent status of datagram is fail.
	1	The sent status of datagram is success.

Example

```
AT$QCSOSD=1,2,"ABAB"
1,2
OK
```

4.7.8. AT$QCSOCL

This command is used to close the specified socket. If there are pending messages to be read, they are dropped. No further unsolicited "$QCSONMI" notification is generated. If the socket has already been closed, or was never created, an error is returned.

Table 4.115: AT$QCSOCL

AT$QCSOCL	Response
Set command AT$QCSOCL=<socket_id>	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSOCL=?	Response $QCSOSD: (list of supported <socket_id>) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type
	Socket ID returned by AT$QCSOCR.

Example

```
AT$QCSOCL=1
OK
```

4.7.9. AT$QCSONMI

The Write command is used to set the unsolicited result code "$QCSONMI" to indicate arrived socket messages (the socket is not configured as private socket by AT$QCSONMIE command):

- If <mode>=1, the UE receives an unsolicited result code: "$QCSONMI: <socket>,<length>".
- If <mode>=2, the UE receives an unsolicited result code: "$QCSONMI: <socket>,<remote_addr>,<remote_port>,<length>,<data>".
- If <mode>=3, the UE receives an unsolicited result code: "$QCSONMI: <socket>,<length>,<data>".

The Read command returns the current setting of the command.,

The Write command is also used to set the public max downlink buffer size and the public max messages number.

Table 4.116: AT$QCSONMI

AT$QCSONMI	Response
Set command AT$QCSONMI=<mode> [,<max_public_dl_buffer> [,<max_public_dl_pkg_num>]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCSONMI?	Response $QCSONMI:<mode>, <max_public_dl_buffer>, <max_public_dl_pkg_num> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSONMI=?	Response $QCSONMI: (list of supported <mode>s), (list of supported <max_public_dl_buffer>), (list of supported <max_public_dl_pkg_num >) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type	
	Control downlink data format.	
	0	Disable indication messages unsolicited result code
	1	Enable indication messages unsolicited result code: "$QCSONMI: <socket_id>,<length>"
	2	Enable indication messages unsolicited result code: "$QCSONMI: <socket_id>,<remote_addr>,<remote_port>,<length>,<data>"
	3	Enable indication messages unsolicited result code: "$QCSONMI: <socket_id>,<length>,<data>"
<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<length>	Integer type	
	1 to 1358	Number of bytes of data in the first message.
<data>	String type	
	Data received in hexadecimal string format.	
<remote_addr>	String type	
	The remote IP address that the message is sent from.	
<remote_port>	Integer type	
	0 to 65535	The remote port which the message if sent from.
<max_public_dl_buffer>	Integer type	
	1358 to 3072	the maximum downlink buffer total size for all sockets created by AT$QCSOCR that is not configure as private mode socket with the command "AT$QCSONMIE". Default is 2048.

<max_public_dl_pkg_num>	Integer type	
	8 to 16	the maximum downlink buffer message total number for all sockets created by AT$QCSOCR which is not configure as private mode socket with the command "AT$QCSONMIE". Default is 8.

Example

```
AT$QCSONMI=2,1500,9
OK
```

4.7.10. AT$QCSONMIE

The Write command is used to set the unsolicited result code "$QCSONMI" to indicate arrived messages of a specified socket:

- If <mode>=1, the UE receives an unsolicited result code: "$QCSONMI: <socket>,<length>".
- If <mode>=2, the UE receives an unsolicited result code: "$QCSONMI: <socket>,<remote_addr>,<remote_port>,<length>,<data>".
- If <mode>=3, the UE receives an unsolicited result code: "$QCSONMI: <socket>,<length>,<data>".

The Read command returns the current setting of the command.

The Write command is also used to set the max downlink buffer size and the max messages number for the specified socket.

Table 4.117: AT$QCSONMI

AT$QCSONMI	Response
Set command AT$QCSONMIE=<socket_id>,<mode> [,<max_private_dl_buffer> [,<max_private_dl_pkg_num>]]	Response OK If there is an error, the response is as follows: +CME ERROR: <err>
Read command AT$QCSONMIE?	Response $QCSONMIE:<socket_id>,<mode>, <max_private_dl_buffer>, <max_private_dl_pkg_num> OK If there is an error, the response is as follows: +CME ERROR: <err>
Test command AT$QCSONMIE=?	Response $QCSONMIE: (list of supported <socket_id>s), (list of supported <mode>s), (list of supported <max_private_dl_buffer>), (list of supported <max_private_dl_pkg_num>) OK
Maximum Response Time	5 s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type	
	Control downlink data format.	
	0	Disable indication messages unsolicited result code for the specified socket.
	1	Enable indication messages unsolicited result code for the specified socket: "$QCSONMI: <socket_id>,<length>".
	2	Enable indication messages unsolicited result code for the specified socket: "$QCSONMI: <socket_id>,<remote_addr>,<remote_port>,<length>,<data>".
	3	Enable indication messages unsolicited result code for the specified socket: "$QCSONMI:<socket_id>,<length>,<data>".
	255	reset the mode setting which configures by the command "AT$QCSONMI" for the specified socket.
<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<length>	Integer type	
	1 to 1358	Number of bytes of data in the first message.
<data>	String type	
	Data received in hexadecimal string format.	
<remote_addr>	String type	
	The remote IP address that the message is sent from.	
<remote_port>	Integer type	
	0 to 65535	The remote port which the message if sent from.
<max_private_dl_buffer>	Integer type	
	1358 to 2048	The maximum downlink buffer size for the specified socket. Default is 1358.
<max_private_dl_pkg_num>	Integer type	
	1 to 8	The maximum downlink buffer message number for the specified socket. Default is 4.

Example

```
AT$QCSONMIE=2,1500,6
OK
```

4.7.11. $QCSOCLI

This is an unsolicited message to notify that a socket has been closed. It returns the socket number and error code.

Table 4.118: $QCSOCLI

Message	Parameter
$QCSOCLI:	<socket_id>,<errno>

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<errno>	Integer type	
	11	Operation would block.
	12	Out of memory error.
	22	Invalid argument
	62	Timer expired.
	103	Software caused connection abort.
	104	Connection reset by peer.
	105	No buffer space available.
	107	Transport endpoint is not connected.
	113	No route to host.
	115	Operation now in progress.

Example

```
$QCSOCLI:1,104
```

4.7.12. $QCSOSTR

This is an unsolicited message to notify that one uplink datagram has sent status with the sequence.

Table 4.119: $QCSOSTR

Message	Parameter
$QCSOSTR:	<socket_id>,<sequence>,<status>

Parameter

<socket_id>	Integer type	
	Socket ID returned by AT$QCSOCR.	
<sequence>	Integer type	
	1 to 255	If it is omitted, data sent is not reported. If not omitted, when datagram is sent over RF or is discarded, then the result is reported.
<sequence>	Integer type	
	0	The sent status of datagram is fail.
	1	The sent status of datagram is success.

Example

```
$QCSOSTR:1,101,1
```

4.7.13. Error code for socket command (solution B)

Table 4.120: +CME ERROR: <err>

<err> code	Description
1	Parameter invalid
2	Too much socket instance
3	Create socket error
4	Operation not supported
5	Cannot find the socket
6	Socket Connect fail
7	Socket bind fails
8	Send data fail
9	The socket status is not connected
10	The socket status is already connected
11	The socket status is invalid
12	The socket connect timeout
13	The socket close fails
14	The socket happens fatal error
15	Cannot allocate more memory
16	SIM PUK2 required
17	No more DL buffer resource
18	The socket is connecting
19	UL sequence is invalid
20	Unknown error

5. ERROR CODES

If the AT command is not implemented or the format dose not match, it provides an output "ERROR".

For general control commands compliant with the 3GPP specifications, see AT Command Set for User Equipment (UE) (3GPP TS 27.007) V14.5.0, sub-clause 9.2 for all possible <err> values. If an error occurs, it provides an output "+CME ERROR: <err>". Some common values are listed in the table below.

Table 5.1: General errors (27.007)

<err> code	Description
1	MT not connection
2	MT link reserved
3	Operation not allowed
4	Operation not supported
5	PH-SIM PIN required
6	PH-FSIM PIN required
7	PH-FSIM PUK required
10	SIM not inserted
11	SIM PIN required
12	SIM PUK required
13	SIM failure
14	SIM busy
15	SIM wrong
16	Incorrect password
17	SIM PIN2 required
18	SIM PUK2 required
20	Memory full
21	Invalid index
22	Not found
23	Memory failure
24	Text string too long
25	Invalid characters in text string
26	Dial string too long
27	Invalid characters in dial string
30	No network service
31	Network timeout
32	Network not allowed - emergency call only
40	Network personalization PIN required
41	Network personalization PUK required
42	Network subset personalization PIN required

43	Network subset personalization PUK required
44	Service provider personalization PIN required
45	Service provider personalization PUK required
46	Corporate personalization PIN required
47	Corporate personalization PUK required
48	Hidden key required
49	EAP method not supported
50	Incorrect Parameters
51	Command implemented but currently disabled
52	Command aborted by user
53	Not attached to network due to MT functionality restrictions
54	Modem not allowed - MT restricted to emergency calls only
55	Operation not allowed because of MT functionality restrictions
56	Fixed dial number only allowed - called number is not a fixed dial number
57	Temporarily out of service due to other MT usage
58	Language/alphabet not supported
59	Unexpected data value
60	System failure
61	Data missing
62	Call barred
63	Message waiting indication subscription failure
100	Unknown
103	Illegal MS
106	Illegal ME
107	GPRS services not allowed
108	GPRS services and non GPRS services not allowed
111	PLMN not allowed
112	Location area not allowed
113	Roaming not allowed in this location area
114	GPRS services not allowed in this plmn
115	No suitable cells in location area
122	Congestion
126	Insufficient resources
127	Missing or unknown APN
128	Unknown PDP address or PDP type
129	User authentication failed
130	Active reject by GGSN services gw or PDN gw
131	Active reject unspecified
132	Service option not supported

133	Requested service option not subscribed
134	Service option temporarily out of order
140	Feature not supported
141	Semantic errors in the TFT operation
142	Syntactical errors in the TFT operation
143	Unknown PDP context
144	Semantic errors in PF
145	Syntactical errors in PF
146	PDP context without TFT already activated
148	Unspecified GPRS error
149	PDP authentication failure
150	Invalid mobile class
171	Last PDN disconnection not allowed
172	Semantically incorrect message
173	Mandatory information element error
174	Information element nonexistent or not implemented
175	Conditional IE error
176	Protocol error unspecified
177	Operator determined barring
178	Max number of PDP contexts reached
179	Requested APN not supported in current rat and plmn combination
180	Request rejected bearer control mode violation
181	Unsupported oci value
182	User data transmission through control plane is congested
301	Internal error base
302	UE busy
303	Not power on
304	PDN not active
305	PDN not valid
306	PDN invalid type
307	PDN no parameter
308	UE fail

For general control commands compliant with Use of Data Terminal Equipment – Data Circuit Terminating Equipment (DTE - DCE) Interface for Short Message Service (SMS) and Cell Broadcast Service (CBS) (3GPP TS 27.005). If an error occurs, it provides an output "+CMS ERROR: <err>". Some common values are listed in the following table.

Table 5.2: General errors (27.005)

<err> code	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-(U)SIM 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 acknowledgment expected
500	Unknown error

LIST OF TABLES

REVISION HISTORY

Version	Date	Changes	Authors
01	06/07/2023	- Initial draft	GAA
02	10/07/2023	- Initial draft review	HBG

CONTACT

HT MICRON SEMICONDUTORES S.A.
Av. Unisinos, 1550 | 93022-750 | São Leopoldo | RS | Brasil
www.htmicron.com.br

DOCUMENT INFORMATION

Document Title:	iMCP HTNB32L-XXX AT Commands
Document Subtitle:	NB-IoT RF System-in-Package AT Commands Manual
Classification:	PUBLIC
Doc. Type:	USER DOCUMENTATION
Revision:	v.01
Date:	14/04/2023
Code:	ATC-HTNB32L-XXX

DISCLAIMER

This document is a property of HT Micron and cannot be reproduced without its consent.
HT Micron does not assume any responsibility for use what is described.
This document is subject to change without notice.