
EC NB-IoT AT Commands Manual

Rev. EC NB-IoT_AT_Commands_Manual_V1.12

EIGENCOMM CONFIDENTIAL

Content

Content.....	2
1 Introduction.....	9
1.1 Definitions.....	9
1.2 AT command syntax	9
1.3 3GPP compliance.....	10
2 General Control Commands.....	11
2.1 Basic Commands (ITU-T V.250)	11
2.1.1 ATE Command echo.....	11
2.1.2 ATQ Result code suppression	11
2.2 3GPP Commands (27.007)	12
2.2.1 AT+CFUN set phone functionality.....	12
2.2.2 AT+CGSN request product serial number	13
2.2.3 AT+CGMR request manufacturer revision.....	15
2.2.4 AT+CMEE report mobile termination error.....	15
2.2.5 AT+COPS PLMN selection	17
2.2.6 AT+CREG network registration.....	19
2.2.7 AT+CEREG EPS network registration status.....	21
2.2.8 AT+CSQ get signal quality.....	24
2.2.9 AT+CESQ get extended signal quality.....	25
2.2.10 AT+CPSMS power saving mode setting.....	26
2.2.11 AT+CEDRXS eDRX setting	28
2.2.12 AT+CEDRXRDP eDRX read dynamic parameters	30
2.2.13 AT+CCIOPTOT clot optimization configuration	31
2.2.14 AT+CGCMOD PDP context modify.....	32
2.2.15 AT+CGATT PS attach or detach.....	33
2.2.16 AT+CGACT PDP context activate or deactivate	34
2.2.17 AT+CGDATA enter data state	35
2.2.18 AT+CGDCONT define a PDP context.....	36
2.2.19 AT+CGCONTRDP PDP context read dynamic parameters.....	41
2.2.20 AT+CGEQOS define EPS quality of service	43
2.2.21 AT+CGEQOSRDP EPS quality of service read dynamic parameters.....	44
2.2.22 AT+CGTFT traffic flow template.....	45
2.2.23 AT+CSODCP sending of originating data via the control plane	48
2.2.24 AT+CRTDCP reporting of terminating data via the control plane.....	50
2.2.25 AT+CGAPNRC APN rate control	51
2.2.26 AT+CGEREP Packet domain event reporting.....	52
2.2.27 +CGEV used to indicate EPS PDN connection and bearer resources operations status	54
2.2.28 AT+CGPADDR show PDP address(es).....	55
2.2.29 AT+CSCON signalling connection status	56
2.2.30 AT+CCLK return current date and time	57

2.2.31 AT+CIMI request international mobile subscriber identity	58
2.2.32 AT+CPIN enter PIN	59
2.2.33 AT+CLCK facility lock	60
2.2.34 AT+CPWD change password	61
2.2.35 AT+CSIM generic SIM access	62
2.2.36 AT+CRSM restricted SIM.....	63
2.2.37 AT+CCHO open logical channel	64
2.2.38 AT+CCHC close logical channel.....	65
2.2.39 AT+CGLA generic UICC logical channel access	66
2.2.40 AT+CTZU automatic time zone update.....	67
2.2.41 AT+CTZR time zone reporting	68
2.2.42 AT+CMOLR mobile originated location request.....	70
2.2.43 AT+CMTLR mobile terminated location request notification	73
2.2.44 AT+CMTLRA mobile terminated location request disclosure allowance	75
2.2.45 AT+CRCES reading coverage enhancement status	76
2.2.46 AT+CIPCA initial PDN context activation	77
2.2.47 AT+CGMI request manufacturer identification	79
2.2.48 AT+CGMM request model identification	79
2.2.49 AT+CPINR remaining PIN retries	80
2.2.50 AT+CGAUTH define PDP context authentication parameters	81
2.2.51 AT+IPR set UE baud rate	83
2.2.52 AT+CNMPSD No more PS data	84
2.2.53 AT+ICF set UE serial port character framing	84
2.2.54 AT+CNUM request subscriber number	85
2.2.55 AT+CEER extended error report.....	86
2.3 3GPP Commands (27.005)	87
2.3.1 AT+CMGS send message	87
2.3.2 AT+CSCA service center address	88
2.3.3 AT+CMGF message format.....	89
2.3.4 AT+CSMP set text mode parameters	90
2.3.5 +CMT new message received	91
3 Extended Commands	91
3.1 EC General Commands	91
3.1.1 AT+ECBAND	91
3.1.2 AT+ECLOGDBVER	92
3.1.3 AT+ECCFG.....	93
3.1.4 AT+ECPING	98
3.1.5 AT+ECIPERF.....	100
3.1.6 AT+ECFREQ	102
3.1.7 AT+ECRMFPLMN	104
3.1.8 AT+ECATTBEARER.....	105
3.1.9 AT+ECSNDDATA	107
3.1.10 +RECVNONIP	108
3.1.11 AT+ECPMUCFG	108

3.1.12 AT+ECSMSEND	109
3.1.13 AT+ECCGSN.....	110
3.1.14 AT+ECRFSTAT	111
3.1.15 AT+ECRST	111
3.1.16 AT+ECPSMR.....	112
3.1.17 AT+ECPLMNS.....	113
3.1.18 AT+ECCEQS.....	114
3.1.19 AT+ECSTATUS.....	115
3.1.20 AT+ECICCID	119
3.1.21 AT+ECBCINFO.....	120
3.1.22 AT+ECDNS.....	123
3.1.23 AT+ECDNSCFG	124
3.1.24 AT+ECEMMTIME	125
3.1.25 AT+ECPCFG	126
3.1.26 AT+ECSLEEP.....	128
3.1.27 AT+ECSIMSLEEP	129
3.1.28 AT+ECCGSNLOCK.....	130
3.1.29 AT+ECSAVEFAC.....	131
3.1.30 AT+DMCONFIG.....	131
3.1.31 AT+DMREGSTAT	132
3.1.31 AT+ECTASKINFO	133
3.1.32 AT+ECTASKHISTINFO	133
3.1.33 AT+ECSHOWMEM.....	133
3.1.34 AT+ECSYSTEST.....	134
3.1.35 AT+ECVOTECHK.....	135
3.1.36 AT+ECURC	136
3.1.37 AT+ECPTWEDRXS.....	138
3.1.38 +ECPIN	140
3.1.39 +ECPCFUN.....	141
3.1.40 +ECPADDR.....	141
3.1.41 AT+ECADC.....	142
3.1.42 AT+ECSWC.....	142
3.1.43 AT+ECSNTP.....	143
3.1.44 AT+ECIPR	144
3.1.45 AT+ECNPICFG.....	145
3.1.46 AT+ECPRODMODE	146
3.1.46 AT+ECLEDMODE	147
3.1.47 AT+ECPOWERCLASS.....	149
3.1.48 AT+ECPMUSTATUS.....	150
3.1.49 AT+ECSTATIS	151
3.1.50 AT+ECFSINFO	154
3.1.51 AT+ECFLASHMONITORINFO	155
3.1.52 AT+ECPURC	156
3.1.53 AT+ECFSFORMAT	158

3.1.54 AT+ECSIMPD.....	158
3.1.55 AT+ECABFOTACTL	159
3.1.56 AT+ECNBIOTRAI	159
3.1.57 AT+ECNBR14.....	160
3.1.58 AT+ECEVENTSTATIS	162
3.1.59 AT+NFWUPD	164
3.1.60 AT+ECUSATP	165
3.1.61 AT+ECPSSLPCFG	166
3.2 Socket Commands (solution A)	168
3.2.1 AT+SKTCREATE	168
3.2.2 AT+SKTCONNECT	169
3.2.3 AT+SKTBIND.....	169
3.2.4 AT+SKTSEND	170
3.2.5 AT+SKTSENDT	171
3.2.6 +SKTREC V	172
3.2.7 +SKTERR.....	173
3.2.8 AT+SKTSTATUS.....	173
3.2.9 AT+SKTDELETE	174
3.3 LwM2M Commands.....	174
3.3.1 AT+LWM2MCREATE	174
3.3.2 AT+LWM2MDELETE	175
3.3.3 AT+LWM2MADDOBJ.....	176
3.3.4 AT+LWM2MDELOBJ	177
3.3.5 +LWM2MREAD indicator of LwM2M server's reading command	178
3.3.6 +LWM2MWRITE indicator of LwM2M server's writing command	178
3.3.7 +LWM2MEXECUTE indicator of LwM2M server's execution command.....	179
3.3.8 +LWM2MOBSERVE indicator of LwM2M server's observation command	180
3.3.9 AT+LWM2MREADCONF	180
3.3.10 AT+LWM2MWRITECONF	182
3.3.11 AT+LWM2MEXECUTECONF	183
3.3.12 AT+LWM2MNOTIFY	183
3.3.13 AT+LWM2MUPDATE.....	184
3.3.14 Summary of <err> Codes	185
3.4 CoAP Commands.....	186
3.4.1 AT+COAPCREATE.....	186
3.4.2 AT+COAPDEL	186
3.4.3 AT+COAPADDRES	187
3.4.4 AT+COAPHEAD	187
3.4.5 AT+COAPOPTION	188
3.4.6 AT+COAPSEND	189
3.4.7 AT+COAPDATASTATUS	191
3.4.8 AT+COAPCFG	191
3.4.9 AT+COAPALISIGN	192
3.4.10 +COAPURC	193

3.5 MQTT Commands	194
3.5.1 AT+ECMTCFG	194
3.5.2 AT+ECMTOPEN	198
3.5.3 AT+ECMTCLOSE	199
3.5.4 AT+ECMTCONN	199
3.5.5 AT+ECMTDISC	200
3.5.6 AT+ECMTSUB	201
3.5.7 AT+ECMTUNS	202
3.5.8 AT+ECMTPUB	203
3.5.9 +ECMTSTAT	204
3.5.10 +ECMTRECV	204
3.6 Http Commands	205
3.6.1 AT+HTTPCREATE	205
3.6.2 AT+HTTPCON	206
3.6.3 AT+HTTPDESTROY	206
3.6.4 AT+HTTPSEND	207
3.6.5 +HTTPRESPH indicator of response header	208
3.6.6 +HTTPRESPC indicator of response content	209
3.6.7 +HTTPERR indicator of error message	209
3.7 OneNET Extension Commands	210
3.7.1 AT+MIPLCONFIG	210
3.7.2 AT+MIPLCREATE	211
3.7.3 AT+MIPLDELETE	211
3.7.4 AT+MIPLOPEN	212
3.7.5 AT+MIPLCLOSE	212
3.7.6 AT+MIPLADDOBJ	213
3.7.7 AT+MIPLDELOBJ	214
3.7.8 AT+MIPLNOTIFY	214
3.7.9 AT+MIPLREADRSP	216
3.7.10 AT+MIPLWRITERSP	218
3.7.11 AT+MIPLEXECUTERSP	219
3.7.12 AT+MIPLOBSERVERSP	220
3.7.13 AT+MIPLDISCOVERRSP	221
3.7.14 AT+MIPLPARAMETERRSP	222
3.7.15 AT+MIPLUPDATE	223
3.7.16 AT+MIPLVER	224
3.7.17 +MIPLREAD indicator of read request	224
3.7.18 +MIPLWRITE indicator of write request	225
3.7.19 +MIPLEXECUTE indicator of execute request	226
3.7.20 +MIPLOBSERVE indicator of observe request	227
3.7.20 +MIPLDISCOVER indicator of observe request	227
3.7.21 +MIPLPARAMETER indicator of observe parameter request	228
3.7.22 +MIPLEVENT indicator of event report	228
3.7.23 Summary of <err> Codes	229

3.8 China Telecom Extension Commands.....	230
3.8.1 AT+CTM2MVER.....	230
3.8.2 AT+CTM2MSETMOD.....	231
3.8.3 AT+CTM2MSETPM.....	232
3.8.4 AT+CTM2MREG.....	233
3.8.4 AT+CTM2MUPDATE.....	233
3.8.5 AT+CTM2MDEREG.....	234
3.8.6 AT+CTM2MSEND.....	234
3.8.7 AT+CTM2MCMDRSP.....	235
3.8.8 AT+CTM2MRMODE.....	236
3.8.9 AT+CTM2MREAD.....	237
3.8.10 +CTM2MRECV.....	237
3.8.11 +CTM2M.....	237
3.8.12 +CTM2MCMD.....	238
3.8.13 +CTM2M ERROR.....	239
3.9 Socket Command(solution B).....	240
3.9.1 AT+ECSOCR.....	240
3.9.2 AT+ECSOST.....	241
3.9.3 AT+ECSOSTT.....	243
3.9.4 AT+ECSOSTF.....	245
3.9.5 AT+ECSOSTFT.....	247
3.9.6 AT+ECQSOS.....	248
3.9.7 AT+ECSORF.....	249
3.9.8 AT+ECSOCO.....	250
3.9.9 AT+ECSOSD.....	251
3.9.10 AT+ECSOSDT.....	253
3.9.11 AT+ECSOCL.....	254
3.9.12 AT+ECSONMI.....	255
3.9.13 AT+ECSONMIE.....	256
3.9.14 +ECSOCLI.....	258
3.9.15 +ECSOSTR.....	259
3.9.16 +ECSOCO.....	259
3.9.17 Error code for socket command(solution B).....	260
3.10 Socket Command(TCP Server Mode).....	260
3.10.1 AT+ECSRVSOCRTCP.....	260
3.10.2 AT+ECSRVSOCCLTCPLISTEN.....	262
3.10.3 AT+ECSRVSOCCLTCPCLIENT.....	262
3.10.4 AT+ECSRVSOTCPSENDCLT.....	263
3.10.5 AT+ECSRVSOTCPSENDCLTT.....	264
3.10.6 AT+ECSRVSOTCPLIENSTATUS.....	265
3.10.7 +ECACCEPTTCPCLIENTSOCKET.....	266
3.10.8 +ECSRVSOTCPCLTRCV.....	267
3.10.9 +ECSRVSOCCLIENTCPCLOSE.....	268
4 Error Values.....	269

EIGENCOMM CONFIDENTIAL

1 Introduction

1.1 Definitions

For the purposes of the present document, the following syntactical definitions apply

<CR> Carriage return character.

<LF> Linefeed character.

<...> Name enclosed in angle brackets is a syntactical element. Brackets themselves do not appear in the command line.

[...] Optional subparameter of a command or an optional part of TA information response is enclosed in square brackets. Brackets themselves do not appear in the command line. When subparameter is not given in parameter type commands, new value equals to its previous value. In action type commands, action should be done on the basis of the recommended default setting of the subparameter.

NO_SAVE The parameter of the current AT command will be lost, if module is rebooted or current AT command doesn't have parameter

AUTO_SAVE The parameter of the current AT command will be kept in NVRAM automatically and take in effect immediately, and it won't be lost if module is reboot

AUTO_SAVE_REBOOT The parameter of the current AT command will be kept in NVRAM automatically and take in effect after reboot, and it won't be lost if module is reboot

- '-' means this AT command doesn't care the parameter saving mode

1.2 AT command syntax

1.2.1 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

Eigencomm use AT+EC<cmd> to implement self-extended command.

1.2.2 Command line

See figure 1 for general structure of a command line. Standardized basic commands are found only in ITU-T Recommendation V.250 [14]. 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 subparameters. Parameter type commands also have a read command (trailing ?) to check the current values of subparameters. Action type commands do not store the values of any of their possible subparameters, and therefore do not have a read command.

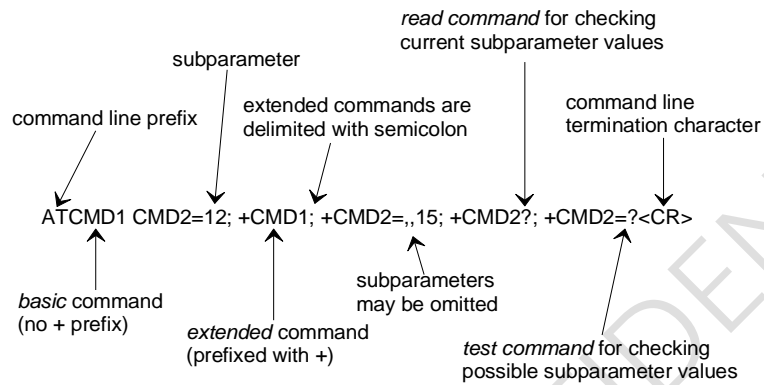


Figure 1: Basic structure of a command line

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

If subparameter 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 command was not processed due to an error related to MT operation.

1.3 3GPP compliance

Basic commands are compiled with ITU-T V.250(07/2003)

3GPP commands are compiled with the 3GPP TS 27.007 V16.0.0 (2019-03) and 3GPP TS 27.005 V15.0.0 (2018-06).

2 General Control Commands

2.1 Basic Commands (ITU-T V.250)

2.1.1 ATE Command echo

The setting of this parameter determines whether or not the DCE echoes characters received from the DTE during command state and online command state.

ATE

Set Command ATE<value>	Response OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	5s
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
```

2.1.2 ATQ Result code suppression

The setting of this parameter determines whether or not 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.

ATQ

Set Command ATQ<value>	Response OK If there is any error, response:
---------------------------	--

	+CME ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<value>	Integer type
0	DCE transmits unsolicited result codes
1	Unsolicited result codes are suppressed and not transmitted. Note: a) If set to "1", all unsolicited result codes are all suppressed, including: PING/IPERF/LWM2M unsolicited result codes; b) If set to "1", only suppress the unsolicited result codes; And AT response/result codes are not suppressed;
The default value is 0	

Example

```
ATQ0
OK
```

2.2 3GPP Commands (27.007)

2.2.1 AT+CFUN set phone functionality

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.

Read command returns the current setting of <fun>.

Test command returns values supported by the MT as compound values.

AT+CFUN

Set Command AT+CFUN=<fun> [, <rst>]	Response OK If there is any error, response: +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	25s
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 shall always be defaulted when <rst> is not given.
	1	Reset the MT before setting it to <fun> power level. (not supported and will be ignored)

Example

```

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

AT+CFUN?
+CFUN: 1
OK

AT+CFUN=1
OK

```

2.2.2 AT+CGSN request product serial number

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

Test command returns values supported as a compound value.

AT+CGSN

Set Command AT+CGSN=<snt>	Response When <snt>=0 and command successful: +CGSN: <sn> OK When <snt>=1 and command successful: +CGSN: <imei> OK
-------------------------------------	---

	<p>When <sn>=2 and command successful: +CGSN: <imeisv> OK</p> <p>When <sn>=3 and command successful: +CGSN: <svn> OK</p> <p>If there is any error, response: +CME ERROR: <err></p>
<p>Execution Command AT+CGSN</p>	<p>Response <sn> OK</p> <p>If there is any error, response: ERROR or +CME ERROR: <err></p>
<p>Test Command AT+CGSN=?</p>	<p>Response +CGSN: (list of supported <sn>s) OK</p>
<p>Maximum Response Time</p>	<p>5s</p>
<p>Parameter Saving Mode</p>	<p>NO_SAVE</p>

Parameter

<sn>	Integer type; indicating the serial number type that has been requested
0	Return <sn>
1	Returns the IMEI(International Mobile station Equipment Identity)
2	Returns the IMEISV(International Mobile station Equipment Identity and Software Version number)
3	Returns the SVN(Software Version Number)
<sn>	One or more lines of information text determined by the MT manufacturer (not support now)
<imei>	String type; in decimal format indicating the IMEI
<imeisv>	String type; in decimal format indicating the IMEISV
<svn>	String type; in decimal format indicating the current SVN which is a part of IMEISV

Example

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

AT+CGSN=2
+CGSN: "788596633100001"
OK
```

```

AT+CGSN=3
+CGSN: "01"

OK

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

OK

```

2.2.3 AT+CGMR request manufacturer revision

The execution command returns the manufacturer revision. Now it returns the firmware revision and build time.

AT+CGMR

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

Example

```

AT+CGMR
+CGMR:
-- Board: EC616_EVK --
-- SDK Version: EC616_SW_V001.000.xxx --
-- EVB Version: EC616_HW_V1.0 --
-- Compiled: Jul 23 2019 20:50:16 --

OK

AT+CGMR=?
OK

```

2.2.4 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.

AT+CMEE

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	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<mode>	Integer type	
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
```


2.2.5 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 done automatically by the MT or is forced by this command to operator <oper> (it shall be given in format <format>) to a certain access technology, indicated in <AcT>. If the selected operator is not available, no other operator shall be selected (except <mode>=4). If the selected access technology is not available, then the same operator shall be selected in other access technology. The selected operator name format shall also apply to further read commands (AT+COPS?). <mode>=2 forces an attempt to deregister from the network. The selected mode affects all further network registration (e.g. after <mode>=2, MT shall be unregistered until <mode>=0 or 1 is selected). This command should 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 should then be an empty field. The list of operators shall be 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 shall be delimited from the operator list by two commas.

AT+COPS

Set Command AT+COPS=<mode> [, <format> [, <oper> [, <AcT>]]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+COPS?	Response +COPS: <mode> [, <format> , <oper>] [, <AcT>]] OK If there is any error, response: +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 any error, response: +CME ERROR: <err>
Maximum Response Time	305s
Parameter Saving Mode	AUTO_SAVE Note: Set <mode> to 0,1,4 will save to NVM

Parameter

<mode>	Integer type
0	Automatic(<oper> field is ignored)
1	Manual(<oper> field shall be present, and <AcT> is optional) Note: <format> set to 2 is only supported in this case;
2	Deregister from network
3	Set only <format> (for read command AT+COPS?), do not attempt to register/deregister (<oper> and <AcT> fields are ignored);
4	Manual/automatic (<oper> field shall be present); if manual selection fails, automatic mode (<mode>=0) is entered; Note: <format> set to 2 is only supported in this case;
<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	Available
2	Current
3	Forbidden
<AcT>	Integer type; access technology selected
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
```

2.2.6 AT+CREG network registration

Set command controls the presentation of an unsolicited result code +CREG: <stat> when <n>=1 and there is a change in the MT's circuit mode network registration status 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.

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.

AT+CREG

Set Command AT+CREG=[<n>]	Response OK If there is any error, response: +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	5s
Parameter Saving Mode	AUTO_SAVE Note: Set of <n> will save to NVM, and the default value is 0.

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>]
<stat>	3	Enable network registration, location information and cause value information unsolicited result code +CREG: <stat>[, [<lac>], [<ci>], [<AcT>] [, <cause_type>, <reject_cause>]]
	Integer type	
	0	not registered, MT is not currently searching a new operator to register to
	1	registered, home network (not applicable) Note: As no CS service for NB, this is not applicable for NB.
<cause_type>	2	not registered, but MT is currently searching a new operator to register to
	3	registration denied
	4	unknown (e.g. out of GERAN/UTRAN/E-UTRAN coverage)
	5	registered, roaming (not applicable) Note: As no CS service for NB, this is not applicable for NB.
	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)
	8	attached for emergency bearer services only (not applicable) Note: As no emergency bearer for NB, this is not applicable for NB.
	9	registered for "CSFB not preferred", home network (not applicable) Note: As no CS service for NB, this is not applicable for NB.
	10	registered for "CSFB not preferred", roaming (not applicable) Note: As no CS service for NB, this is not applicable for NB.
	<tac>	String type two byte tracking area code
<ci>	String type Four byte cell ID in hexadecimal format	
<AcT>	Integer type; access technology of the serving cell 9 E-UTRAN(NB-S1 mode)	
<cause_type>	Integer type; indicates the type of <reject_cause> 0 Indicates that <reject_cause> contains an MM cause value, see 3GPP TS 24.008 [8] Annex G. 1 Indicates that <reject_cause> 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>.

Example

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

2.2.7 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 will provide the UE with additional information for the active time value and the extended periodic TAU value if there is a change of the network cell in E-UTRAN. The value <n>=5 further enhances the 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.

Test command returns values supported as a compound value.

AT+CEREG

Set Command AT+CEREG=<n>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CEREG?	When <n>=0, 1, 2 or 3 and command successful: +CEREG: <n>, <stat>[, [<tac>], [<ci>], [<AcT>

	<pre>[,<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>],[<P eriodic_TAU>]]]] OK If there is any error, response: +CME ERROR: <err></pre>
Test Command AT+CEREG=?	Response +CEREG: (list of supported <n>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE Note: Set of <n> will save to NVM, and the default value is 0.

Parameter

<n>	Integer type;
0	disable network registration unsolicited result code
1	enable network registration unsolicited result code +CEREG: <stat>
2	enable network registration and location information unsolicited result code +CEREG: <stat>[,<tac>],[<ci>],[<AcT>]]
3	enable network registration, location information and EMM cause value information unsolicited result code +CEREG: <stat>[,<tac>],[<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>][,,[,<Active-Time>],[<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; indicates the EPS registration status
0	Not registered, MT is not currently searching an operator to register to
1	Registered, home network
2	Not registered, but MT is currently trying to attach or searching an operator to register to
3	Registration denied

	4	Unknown (e.g. out of E-UTRAN coverage)
	5	Registered, roaming
	6	Registered for "SMS only", home network (not applicable)
	7	Registered for "SMS only", roaming (not applicable)
	8	Attached for emergency bearer services only (not applicable)
	9	Registered for "CSFB not preferred", home network (not applicable)
	10	Registered for "CSFB not preferred", roaming (not applicable)
<ta		String type; two bytes tracking area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)
<ci>		String type; four bytes E-UTRAN cell ID in hexadecimal format
<AcT>		Integer type; indicates the access technology of the serving cell
	0	GSM (not applicable)
	1	GSM Compact (not applicable)
	2	UTRAN (not applicable)
	3	GSM w/EGPRS (not applicable)
	4	UTRAN w/HSDPA (not applicable)
	5	UTRAN w/HSUPA (not applicable)
	6	UTRAN w/HSDPA and HSUPA (not applicable)
	7	E-UTRAN (not applicable)
	8	EC-GSM-IoT (A/Gb mode) (not applicable)
	9	E-UTRAN (NB-S1 mode)
<cause_type		Integer type; indicates the type of <reject_cause>
>	0	Indicates that <reject_cause> contains an EMM cause value
	1	Indicates that <reject_cause> contains a manufacturer-specific cause value
<reject_cau		Integer type; contains the cause of the failed registration. The value is of type as
se>		defined by <cause_type>.
<Active_Tim		String type; one byte in an 8-bit format. Requested Active Time value (T3324) to be
e>		allocated to the UE. (e.g. "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
<Periodic_T		String type; one byte in an 8-bit format. Requested extended periodic TAU value
AU>		(T3412) to be allocated to the UE in E-UTRAN. (e.g. "01000111" equals 70 hours). Bits 5 to 1 represent the binary coded timer value Bits 6 to 8 define the timer value unit

Example

```
AT+CEREG=5
```

```
OK
```

```
AT+CEREG?
```

```
+CEREG: 5,1,"5b49","0190271a",9
```

```
OK
```

```
AT+CEREG=?
```

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

```
OK
```

2.2.8 AT+CSQ get signal quality

The execution command returns received signal quality `<rssi>` and channel bit error rate `<ber>` from the MT. Please refer to Chapter 4 for possible `<err>` values.

The test command returns values supported as compound values.

AT+CSQ

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

Parameter

<code><rssi></code>	Integer type
0	-113dBm or less
1	-111dBm
2...30	-109...-53 dBm
31	-51 dBm or greater
99	not known or not detectable
<code><ber></code>	Integer type
0...7	as RXQUAL values in the table in 3GPP TS 45.008 [20] subclause 8.2.4
99	not known or not detectable

Example

```
AT+CSQ
```

```
+CSQ: 27,0
```

```
OK
```


2.2.9 AT+CESQ get extended signal quality

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

The test command returns values supported as compound values.

AT+CESQ

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

Parameter

<code><rxlev></code>	Integer type; not supported by NB-IoT 99 not known or not detectable
<code><ber></code>	Integer type; not supported by NB-IoT 99 not known or not detectable
<code><rscp></code>	Integer type; not supported by NB-IoT 255 not known or not detectable
<code><ecno></code>	Integer type; not supported by NB-IoT 255 not known or not detectable
<code><rsrq></code>	Integer type 0 rsrq<-19.5dB 1 -19.5dB<=rsrq<-19dB 2 -19dB<=rsrq<-18.5dB : 32 -4dB<=rsrq<-3.5dB 33 -3.5dB<=-3 dB 34 -3 dB <=rsrq 255 not known or not detectable
<code><rsrp></code>	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
```

2.2.10 AT+CPSMS power saving mode setting

The set command controls the setting of the UE's power saving mode (PSM) parameters. The command controls whether the UE wants to apply PSM or not. Please refer to the 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 will be disabled and data for all parameters in the command +CPSMS will be 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.

AT+CPSMS

Set Command AT+CPSMS=<mode>[,<Requested_Periodic-RAU>[,<Requested_GPRS-READY-timer>[,<Requested_Periodic-TAU>[,<Requested_Active-Time>]]]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CPSMS?	Response +CPSMS: <mode>,[<Requested_Periodic-RAU>],[<Requested_GPRS-READY-timer>],[<Requested_Periodic-

	TAU>], [<Requested_Active-Time>]
Test Command AT+CPSMS=?	Response +CPSMS: (list of supported <mode>s), (list of supported <Requested_Periodic-RAU>s, list of supported <Requested_GPRS-READY-timer>s), (list of supported <Requested_Periodic-TAU>s, list of supported <Requested_Active-Time>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<mode>	Integer type; indicates to disable or enable the use of PSM in the UE
0	Disable the use of PSM
1	Enable the use of PSM
2	Disable the use of PSM and discard all parameters for PSM
<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 represent the binary coded timer value Bits 6 to 8 define the timer value unit as follows: Bits 8 7 6 0 0 0 Value is incremented in multiples of 10 minutes 0 0 1 Value is incremented in multiples of 1 hour 0 1 0 Value is incremented in multiples of 10 hours 0 1 1 Value is incremented in multiples of 2 seconds 1 0 0 Value is incremented in multiples of 30 seconds 1 0 1 Value is incremented in multiples of 1 minute 1 1 0 Value is incremented in multiples of 320 hours 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. (e.g. "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 decihours

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: (0-2),,,"00000000"-11111111"), ("00000000"-11111111")
```

```
OK
```

2.2.11 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 will be disabled and data for all parameters in `AT+CEDRXS` command will be removed.

The set command also controls the presentation of an unsolicited result code `+CEDRXP`:

`<AcTtype>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]`

when `<n>=2` and there is a change in the eDRX parameters provided by the network.

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

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

AT+CEDRXS

Set Command <code>AT+CEDRXS=<mode>,<AcT-type>[,<Requested_eDRX_value>]</code>	Response OK If there is any error, response: <code>+CME ERROR: <err></code>
Read Command <code>AT+CEDRXS?</code>	Response <code>+CEDRXS: <AcT-</code>

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

Parameter

<mode>	Integer type; indicates to disable or enable the use of eDRX in the UE. This parameter is applicable to all specified types of access technology, i.e. the most recent setting of <mode> will take effect for all specified values of <AcT-type>.								
	<table border="1"> <tr> <td>0</td> <td>Disable the use of eDRX</td> </tr> <tr> <td>1</td> <td>Enable the use of eDRX</td> </tr> <tr> <td>2</td> <td>Enable the use of eDRX and enable the unsolicited result code: +CEDRXP: <AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]</td> </tr> <tr> <td>3</td> <td>Disable the use of eDRX and discard all parameters for eDRX.</td> </tr> </table>	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: <AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]	3	Disable the use of eDRX and discard all parameters for eDRX.
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: <AcT-type>[,<Requested_eDRX_value>[,<NW-provided_eDRX_value>[,<Paging_time_window>]]]								
3	Disable the use of eDRX and discard all parameters for eDRX.								
<AcT-type>	Integer type; indicates the type of access technology. AT+CEDRXS? is used to specify the relationship between the type of access technology and the requested eDRX value.								
	<table border="1"> <tr> <td>0</td> <td>Access technology is not using eDRX. This parameter value is only used in the unsolicited result code.</td> </tr> <tr> <td>5</td> <td>E-UTRAN (NB-S1 mode)</td> </tr> </table>	0	Access technology is not using eDRX. This parameter value is only used in the unsolicited result code.	5	E-UTRAN (NB-S1 mode)				
0	Access technology is not using eDRX. This parameter value is only used in the unsolicited result code.								
5	E-UTRAN (NB-S1 mode)								
<Requested_eDRX_value>	String type; half a byte in a 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element. (e.g."0010" equals 20.48 seconds)								
<NW-provided_eDRX_value >	String type; half a byte in a 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element. (e.g."0010" equals 20.48 seconds)								
<Paging_time_window >	string type; half a byte in a 4 bit format. The paging time window refers to bit 8 to 5 of octet 3 of the Extended DRX parameters information element. (e.g."0000" equals 2.56 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

```

2.2.12 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.

AT+CEDRXRDP

Execution Command AT+CEDRXRDP	Response +CEDRXRDP: <AcT-type>[, <Requested_eDRX_value>[, <NW-provided_eDRX_value>[, <Paging_time_window>]]] OK
Test Command AT+CEDRXRDP=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<AcT-type>	Integer type; indicates the type of access technology.
	0 Access technology is not using eDRX
	5 E-UTRAN (NB-S1 mode)
<Requested_eDRX_value>	String type; half a byte in a 4-bit format. (e.g."0010" equals 20.48 seconds)
<NW-provided_eDRX_value>	String type; half a byte in a 4-bit format. (e.g."0010" equals 20.48 seconds)
<Paging_time_window>	String type; half a byte in a 4-bit format. (e.g."0000" equals 2.56 seconds)

Example

```

AT+CEDRXRDP
+CEDRXRDP: 5,"0010","1101","0100"
OK

AT+CEDRXRDP=?
OK

```

2.2.13 AT+CCIOTOPT clot optimization configuration

The set command controls which Clot 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 Clot 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 Clot EPS optimizations by the network.

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

The test command returns values supported as compound values.

AT+CCIOTOPT

Set Command AT+CCIOTOPT=<n>[,<support_UE_opt>[,<preferred_ue_opt>]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CCIOTOPT?	Response +CCIOTOPT: <n>,<support_UE_opt>,<preferred_UE_opt> OK
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)
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<n>	Integer type; indicates the type of access technology.
0	Disable reporting
1	Enable reporting
3	Disable reporting and reset the parameters for Clot EPS optimization to the default values.
<support_UE_opt>	Integer type; indicates the UE's support for Clot EPS optimizations
1	Support for control plane Clot EPS optimization
2	Support for user plane Clot EPS optimization (not support

	now)
	3 Support for both control plane CloT EPS optimization and user plane CloT EPS optimizations
<preferred_UE_opt>	Integer type; indicates the UE's preference for CloT EPS optimizations
	0 No preference
	1 Preference for control plane CloT EPS optimization
	2 Preference for user plane CloT EPS optimization
<supported_Network_opt>	Integer type; indicates the Network support for CloT EPS optimizations
	0 No support
	1 Support for control plane CloT EPS optimization
	2 Support for user plane CloT EPS optimization
	3 Support for both control plane CloT EPS optimization and user plane CloT EPS optimization

Example

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

OK

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

OK

2.2.14 AT+CGCMOD PDP context modify

The execution 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.

AT+CGCMOD

Set Command AT+CGCMOD=<cid> (Note1)	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CGCMOD=?	Response +CGCMOD: (list of <cid>s associated with active contexts)

	OK
Maximum Response Time	70s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type; specifies a particular PDP context definition. <cid> values of 0-10 are supported.
-------	---

Note1:

1) Not support to specify several <cid>s, just not support: AT+CGCMOD=<cid>,<cid>[,...]

Example

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

OK
```

2.2.15 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 will be automatically deactivated when the attachment state changes to detached.

The read command returns the current Packet Domain service state.

The test command is used for requesting information on the supported Packet Domain service states.

AT+CGATT

Set Command AT+CGATT=<state>	Response OK If there is any error, response: +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	70s
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

AT+CGATT?
+CGATT: 1

OK
```

2.2.16 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, then the MT responds with a `+CME ERROR`.

For EPS, the activation request for an EPS bearer resource will be answered by the network by either an EPS dedicated bearer activation or EPS bearer modification request. The request must be accepted by the 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.

AT+CGACT

Set Command AT+CGACT=<state>,<cid> (Note1)	Response OK If there is any error, response: <code>+CME ERROR: <err></code>
Read Command AT+CGACT?	Response [+CGACT: <cid>,<state>]

	[<CR><LF>+CGACT:<cid>,<state>. [...] OK
Test Command AT+CGACT=?	Response +CGACT: (list of supported <state>s) OK
Maximum Response Time	70s
Parameter Saving Mode	NO_SAVE

Parameter

<state>	Integer type; indicates the activation state of PDP context activation.
0	Deactivated
1	Activated
<cid>	Integer type; specifies a particular PDP context definition. Only one <cid> can be activated or deactivated at the same time. <cid> values of 0-10 are supported. <cid> cid that defined in +CGDCONT/+CGDSCONT

Note1

- 1) <cid> must be specified, just not support to activate/deactivate all defined/activated bearers.
- 2) Not support to specify several <cid>s, just not support:
AT+CGACT=<state>,<cid>,<cid>[,...].

Example

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

OK

AT+CGACT?
+CGACT: 0,1

OK
```

2.2.17 AT+CGDATA enter data state

The execution 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 activations. <cid> should be specified (see the +CGDCONT) in order to provide the information needed for the context activation request.

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

AT+CGDATA

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

Parameter

<L2P>	String type; indicates the layer 2 protocol to be used between the TE and MT.
M-PT	Eigencomm specified protocol – PDP Type, such as IP/IPV6/IPV4V6/Non-IP
<cid>	Integer type; specifies a particular PDP context definition. <cid> values of 0-10 are supported.

Note:

- 1) This AT command is not fully followed the 3GPP 27.007, execution command just trigger MT to activate a PDP context, just same as: +CGACT=1,<cid>.
- 2) If PDP activation success, MT issues the result code: OK, not: CONNECT, as not support V.250 online data state now.

Example

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

2.2.18 AT+CGDCONT define a PDP context

The set command specifies 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, e.g. if the UE needs to transfer an APN. The number of PDP contexts that may be in a defined state at the same time is given by the range returned by the test command.

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

	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) [<CR><LF>+CGDCONT: (range 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 <PCSCF_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) [...]]
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type; specifies a particular PDP context definition. The parameter is local to the TE-MT interface and is used in other PDP context-related commands. The range of permitted values is returned by the test form of the command. <cid> values of 0-10 are supported.								
<PDP_type>	String type; specifies the type of packet data protocol. The default value is manufacturer specific. <table border="1"> <tr> <td>IP</td> <td>Internet Protocol</td> </tr> <tr> <td>IPV6</td> <td>Internet Protocol, version 6</td> </tr> <tr> <td>IPV4V6</td> <td>Virtual <PDP_type> introduced to handle dual IP stack UE capability</td> </tr> <tr> <td>Non-IP</td> <td>Transfer of Non-IP data to external packet data network</td> </tr> </table>	IP	Internet Protocol	IPV6	Internet Protocol, version 6	IPV4V6	Virtual <PDP_type> introduced to handle dual IP stack UE capability	Non-IP	Transfer of Non-IP data to external packet data network
IP	Internet Protocol								
IPV6	Internet Protocol, version 6								
IPV4V6	Virtual <PDP_type> introduced to handle dual IP stack UE capability								
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 max length is 99 characters;								
<PDP_addr>	String type; identifies the MT in the address space applicable to the PDP. (ignored with the set command)								
<d_comp>	Integer type; don't need for NB-IOT								
<h_comp>	Integer type; don't need for NB-IOT								

<IPv4AddrAlloc>	Integer type; control how the MT/TA requests to get the IPv4 address information
0	IPv4 address allocation through NAS signalling
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 (Not support)
2	PDP context is for new PDP context establishment
3	PDP context is for handover from a non-3GPP access network (Not support)
<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 +CGDCONT
1	Preference of P-CSCF address discovery through NAS signaling (Not support)
2	Preference of P-CSCF address discovery through DHCP (Not support)
<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
0	UE indicates that the PDP context is not for IM CN subsystem-related signaling only
1	UE indicates that the PDP context is for IM CN subsystem-related signaling only (Not support)
<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 (Not support)
<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 signalling
<Local_Addr_Ind>		Integer type; indicates to the network whether or not 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 (Not support)
<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 signalling
< 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 (Not support)

Example

```
AT+CGDCONT=1,"ipv6","apn1"
```

```
OK
```

```
AT+CGDCONT?
```

```
+CGDCONT: 0,"IP","snbiot.mnc006.mcc460.gprs","10.212.154.7"
```

```
+CGDCONT: 1,"IPV6","apn1"
```

```
OK
```

```
AT+CGDCONT=?
```

```
+CGDCONT: (0-10),"IP",,,,,,(0),(0,2),(0),(0),(0,1),(0),(0,1),(0),(0),(0)
```

```
+CGDCONT: (0-10),"IPV6",,,,,,(0),(0,2),(0),(0),(0,1),(0),(0),(0),(0),(0)
```

```
+CGDCONT: (0-10),"IPV4V6",,,,,,(0),(0,2),(0),(0),(0,1),(0),(0,1),(0),(0),(0)
```

```
+CGDCONT: (0-10),"Non-IP",,,,,,(0),(0,2),(0),(0),(0,1),(0),(0),(0),(0,1),(0)
```

```
OK
```

//note, if a bearer is activated with ipv4v6 addresses, two PDP context information will be returned as the response of reading command, such as:

```
AT+CGDCONT?
```

```
+CGDCONT: 0,"IPV4V6","ctnb.MNC011.MCC460.GPRS","11.50.125.207"
```

```
+CGDCONT:
```


0,"IPV4V6","ctnb.MNC011.MCC460.GPRS","36.14.8.120.0.80.3.53.0.2.0.1.83.170.233.153"

OK

2.2.19 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 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.

AT+CGCONTRDP

Set Command	Response
AT+CGCONTRDP[=<cid>]	<pre> [+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_valu e>]]]]]]]]]]]]]]]]]]]] [<CR><LF>+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 </pre> <p>If there is any error, response: +CME ERROR: <err></p>
Test Command AT+CGCONTRDP=?	<p>Response</p> <pre> +CGCONTRDP: (list of <cid>s associated with active contexts) OK </pre>

Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type; specifies a particular non secondary PDP context definition. <cid> values of 0-10 are supported.
<bearer_id>	Integer type; identifies the bearer.
<apn>	String type; a logical name that was used to select the GGSN or the external packet data network.
<local_addr_and_subnet_mask>	String type; shows the IP address and subnet mask of the MT. The string is given as dot-separated numeric (0-255)
<DNS_prim_addr>	String type; the IP address of the primary DNS server.
<DNS_sec_addr>	String type; the IP address of the secondary DNS server.
<P_CSCF_prim_addr>	String type; shows the IP address of the primary P-CSCF server. (Not displayed)
<P_CSCF_sec_addr>	String type; shows the IP address of the secondary P-CSCF server. (Not displayed)
<IM_CN_Signalling_Flag>	Integer type; shows whether the PDP context is for IM CN subsystem-related signalling only or not. (Not displayed)
<LIPA_indication>	Integer type; indicates that the PDP context provides connectivity using a LIPA PDN connection. (Not displayed)
<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 via a WLAN or not. (Not displayed)
<Local_Addr_Ind>	integer type; indicates whether or not the MS and the network support local IP address in TFTs. (Not displayed)
<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=0
+CGCONTRDP:
0,5,"CMNbiot.mnc004.mcc460.gprs","100.115.240.198.255.255.255.0","211.136.20.203","211.136.17.107"
OK
```

2.2.20 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 3GPP TS 24.301 [83] and 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.

AT+CGEQOS

Set Command AT+CGEQOS=<cid>[,<QCI>[,<DL_GBR>,<UL_GBR>[,<DL_MBR>,<UL_MBR>]]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CGEQOS?	Response [+CGEQOS: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>]] [<CR><LF> +CGEQOS:<cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>][...]] OK
Test Command AT+CGEQOS=?	Response +CGEQOS: (range of supported <cid>s), (list of supported <QCI>s)
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type; specifies a particular EPS Traffic Flows definition in EPS and PDP context definition. <cid> values of 0-10 are supported.
<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 non-guaranteed bit rate Traffic Flows
79	Value for non-guaranteed bit rate Traffic Flows
<DL_GBR>	Integer type; indicates DL GBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI

<UL_GBR>	Integer type; indicates UL GBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI
<DL_MBR>	Integer type; indicates DL MBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI
<UL_MBR>	Integer type; indicates UL MBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI

Example

```
AT+CGEQOS=0,9
OK

AT+CGEQOS=0,4,64,64,64,64
OK
```

2.2.21 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 secondary or non-secondary active PDP contexts.

AT+CGEQOSRDP

Set Command AT+CGEQOSRDP [=<cid>]	Response +CGEQOSRDP: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>][<DL_A MBR>,<UL_AMBR>] [<CR><LF>+CGEQOSRDP: <cid>,<QCI>,[<DL_GBR>,<UL_GBR>],[<DL_MBR>,<UL_MBR>][<DL_A MBR>,<UL_AMBR>] [...]] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CGEQOSRDP=?	Response +CGEQOSRDP: (list of <cid>s associated with active contexts)

	OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type; specifies a particular PDP context definition (see the +CGDCONT commands). <cid> values of 0-10 are supported.
<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 non-guaranteed bit rate Traffic Flows
79	Value for non-guaranteed bit rate Traffic Flows
[128-254]	Value range for Operator-specific QCIs
<DL_GBR>	Integer type; indicates DL GBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI
<UL_GBR>	Integer type; indicates UL GBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI
<DL_MBR>	Integer type; indicates DL MBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI
<UL_MBR>	Integer type; indicates UL MBR in case of GBR QCI. The value is in kbit/s. This parameter is omitted for a non-GBR QCI
<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: 0,9
```

```
OK
```

2.2.22 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 Packet Filters in the TFT for context number <cid> to become undefined. At any time there may exist only one PDP context with no associated TFT amongst all PDP contexts associated to one PDP address. At an attempt to delete a TFT, which would violate this rule, an +CME ERROR response is returned.

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

The test command returns values supported as compound values. If the MT supports several PDP types, the parameter value ranges for each PDP type are returned on a separate line. TFTs shall be used for PDP-type IP only.

AT+CGTFT

<p>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>]]]]]]]]]]</p>	<p>Response OK If there is any error, response: +CME ERROR: <err></p>
<p>Read Command AT+CGTFT?</p>	<p>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>] [<CR><LF>+CGTFT: <cid>,<packet filter identifier>,<evaluation precedence index>,<remote address and subnet mask>,<protocol number (ipv4) / next</p>

	<p>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> [...]]</p> <p>OK</p> <p>If there is any error, response: +CME ERROR: <err></p>
<p>Test Command AT+CGTFT=?</p>	<p>Response</p> <p>+CGTFT: <PDP_type>,(list of supported <packet filter identifier>s),(list of supported <evaluation precedence index>s),(list of supported <remote address and subnet mask>s),(list of supported <protocol number (ipv4) / next header (ipv6)>s),(list of supported <local port range>s),(list of supported <remote port range>s),(list of supported <ipsec security parameter index (spi)>s),(list of supported <type of service (tos) (ipv4) and mask / traffic class (ipv6) and mask>s),(list of supported <flow label (ipv6)>s),(list of supported <direction>s)</p> <p>[<CR><LF>+CGTFT: <PDP_type>,(list of supported <packet filter identifier>s),(list of supported <evaluation precedence index>s),(list of supported <remote address and subnet mask>s),(list of supported <protocol number (ipv4) / next header (ipv6)>s),(list of supported <local port range>s),(list of supported <remote port range>s),(list of supported <ipsec security parameter index (spi)>s),(list of supported <type of service (tos) (ipv4) and mask/ traffic class (ipv6) and mask>s),(list of supported <flow label (ipv6)>s),(list of supported <direction>s)</p> <p>[...]</p> <p>OK</p>
<p>Maximum Response Time</p>	<p>5s</p>
<p>Parameter Saving Mode</p>	<p>NO_SAVE</p>

Parameter

AT+CSODCP

Set Command AT+CSODCP=<cid>,<cpdata_length>,<cpdata>[,<RAI>[,<type_of_user_data>]]	Response OK If there is any error, response: +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)
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type; specifies a particular PDP context definition. <cid> values of 0-10 are supported. Note: If <cid> set to 0, just use current default bearer to send this originating data.
<cpdata_length>	Integer type. Indicates the number of octets of the <cpdata> information element. The max length is 950.
<cpdata>	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 will be completed with the transmission of the ESM DATA TRANSPORT message.
	2 The MT expects that exchange of data will be completed with the receipt of an ESM DATA 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
```

2.2.24 AT+CRTDCP reporting of terminating data via the control plane

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

AT+CRTDCP

Set Command AT+CRTDCP=[<reporting>]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CRTDCP?	Response +CMTDCP: <reporting> OK
Test Command AT+CRTDCP=?	Response +CMTDCP: (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	5s
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 +CMTDCP.
<cid>	Integer type. A numeric parameter which specifies a particular PDP context or EPS bearer context definition. The <cid> parameter is local to the TE-MT interface and identifies the PDP or EPS bearer contexts which have been setup via AT command (see the +CGDCONT commands). <cid> values of 0-10 are supported.
<cpdata_length>	Integer type. Indicates the number of octets of the <cpdata> information element. When there is no data to transmit, the value shall be set to zero.
<cpdata>	string of octets. Contains the user data container contents (refer 3GPP TS 24.301 [83] subclause 9.9.4.24). When there is no data to transmit, the <cpdata> shall be an empty string (""). And support "HEX" character format type.

Example

```
AT+CRTDCP=0
OK

AT+CRTDCP?
+CRTDCP: 0

OK

AT+CRTDCP=1
OK

AT+CRTDCP?
+CRTDCP: 1

OK
```

2.2.25 AT+CGAPNRC APN rate control

This execution command returns the APN rate control parameters (see 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.

AT+CGAPNRC

Set Command AT+CGAPNRC [=<cid>]	Response +CGAPNRC: <cid>[,<Additional_exception_reports>[,<Uplink_time_unit>[,<Maximum_uplink_rate>]]] [<CR><LF>+CGAPNRC: <cid>[,<Additional_exception_reports>[,<Uplink_time_unit>[,<Maximum_uplink_rate>]]] [...]] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CGAPNRC=?	Response +CGAPNRC: (list of <cid>s associated with active contexts) OK
Maximum Response Time	5s

Parameter Saving Mode	NO_SAVE
-----------------------	---------

Parameter

<cid>	Integer type; specifies a particular PDP context definition (see the +CGDCONT commands). <cid> values of 0–10 are supported.
<Additional_exception_reports>	Integer type; indicates whether or not 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 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 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 3GPP TS 24.008 [8] subclause 10.5.6.3.2

Example

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

OK
```

2.2.26 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.

AT+CGEREP

Set Command AT+CGEREP=<mode>[,<bfr>]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CGEREP?	Response +CGEREP: <mode>,<bfr> OK
Test Command AT+CGEREP=?	Response +CGEREP: (list of supported <mode>s),(list of supported <bfr>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_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. Note: default value
<bfr>	1 discard unsolicited result codes when MT-TE link is reserved (e.g. in on-line data mode); otherwise forward them directly to the TE
	integer type;
	0 MT buffer of unsolicited result codes defined within this command is cleared when <mode> 1 ; Only it now

Example

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

```
OK
```

```
AT+CGEREP?
```

```
+CGEREP: 1,0
```

```
OK
```

```
AT+CGEREP=?
```

```
+CGEREP: (0,1), (0)
```

```
OK
```

2.2.27 +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

+CGEV

```
+CGEV: <xxx>
```

Parameter

+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>	The format is found in command +CGDCONT. <cid> values of 0-10 are supported.
<pdnReason>	0 IPv4 only allowed 1 IPv6 only allowed 2 Single address bearer only allowed 3 Single address bearer only allowed and active second bearer failed 4 No reason
<bearerType>	0 NULL 1 default 2 Dedicated (Not application)
<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
<event_type >	integer type; indicates whether this is an informational event or whether the TE

has to acknowledge it.(not support now)

Example

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

2.2.28 AT+CGPADDR show PDP address(es)

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

AT+CGPADDR

Set Command AT+CGPADDR [=<cid>]	Response +CGPADDR: <cid>[,<PDP_addr_1>[,<PDP_addr_2>]] [<CR><LF>+CGPADDR:<cid>[,<PDP_addr_1>[,<PDP_addr_2>]]] [...] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CGPADDR=?	Response +CGPADDR: (list of defined <cid>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<cid>	Integer type; specifies a particular PDP context definition (see the +CGDCONT commands). <cid> values of 0-10 are supported.
<PDP_addr_1> <PDP_addr_2>	<PDP_addr_1> and <PDP_addr_2>: each is a string type that identifies the MT in the address space applicable to the PDP. Both <PDP_addr_1> and <PDP_addr_2> are omitted if none is available. Both <PDP_addr_1> and <PDP_addr_2> are included when both IPv4 and IPv6 addresses are assigned, with <PDP_addr_1> containing the IPv4 address and <PDP_addr_2> containing the IPv6 address. The string is given as dot-separated numeric (0-255) parameter of the form: a1.a2.a3.a4 for IPv4 and a1.a2.a3.a4.a5.a6.a7.a8.a9.a10.a11.a12.a13.a14.a15.a16 for IPv6.

Example

```
AT+CGPADDR
+CGPADDR: 0, "100.120.44.90"

OK
```

2.2.29 AT+CSCON signalling connection status

The set command controls the presentation of an unsolicited result code +CSCON. If $\langle n \rangle = 1$, +CSCON: $\langle mode \rangle$ 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 $\langle mode \rangle$ which shows whether the MT is currently in idle mode or connected mode.

Test command returns supported values as a compound value.

AT+CSCON

Set Command AT+CSCON= $\langle n \rangle$	Response OK If there is any error, response: +CME ERROR: $\langle err \rangle$
Read Command AT+CSCON?	Response +CSCON: $\langle n \rangle$, $\langle mode \rangle$ OK
Test Command AT+CSCON=?	Response +CSCON: (list of supported $\langle n \rangle$ s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

$\langle n \rangle$	Integer type
0	Disable unsolicited result code Note: default value
1	Enable unsolicited result code +CSCON: $\langle mode \rangle$
$\langle mode \rangle$	Integer type; indicates the signalling 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
```

2.2.30 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.

AT+CCLK

Set Command AT+CCLK=<time>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CCLK?	Response +CCLK: <time> OK
Test Command AT+CCLK=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<time>

String type

String type. The format is "yy/MM/dd,hh:mm:ss±zz", where

characters indicate year (four digits), month, day, hour, minute, second and time zone (indicates the difference, expressed in quarters of an hour, between the local time and GMT; and range is -96 ~ +96). For instance, 6th of May 2014, 22:10:00 GMT+2 hours equals "2014/05/06,22:10:00+08"

Note: the year should be after 2000 years, otherwise there will be asserted

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
```

2.2.31 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.

AT+CIMI

Set Command AT+CIMI	Response <IMSI> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CIMI=?	Response OK
Maximum Response Time	5s
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

```

2.2.32 AT+CPIN enter PIN

Set command sends to the MT a password which is necessary before it can be operated (SIM PIN, SIM PUK, etc.). If the PIN is to be entered twice, the TA shall automatically repeat 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.

AT+CPIN

Set Command AT+CPIN=<pin>[, <newpin>]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CPIN?	Response +CPIN: <code> OK
Test Command AT+CPIN=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<pin>, <newpin>	String type
<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

2.2.33 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) should be returned only if service is not active for any <class>. This command should be abortable when network facilities are set or interrogated. Test command returns facility values supported as a compound value.

AT+CLCK

Set Command AT+CLCK=<fac>,<mode>[,<passwd>]	Response OK If there is any error, response: +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 any error, response: +CME ERROR: <err>
Maximum Response Time	5s
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; shall be 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
```

2.2.34 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.

AT+CPWD

Set Command AT+CPWD=<fac>,<oldpwd>,<newpwd>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CPWD=?	Response +CPWD: list of supported (<fac>,<pwdlength>)s OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	5s
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>,<newpwd>	String type <oldpwd> shall be the same as password specified for the facility from the MT user interface or with command Change Password +CPWD and <newpwd> is the new password, maximum length of password can be determined with <pwdlength>
<pwdlength>	Integer type; maximum length of the password for the facility

Example

```
AT+CPWD=?
```

```
+CPWD: ("SC",8)
```

```
OK
```

2.2.35 AT+CSIM generic SIM access

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

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

AT+CSIM

Set Command AT+CSIM=<length>,<command>	Response +CSIM: <length>,<response> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CSIM=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<length>	Integer type length of the characters that are sent to TE in <code><command></code> or <code><response></code> (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 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 3GPP TS 51.011 [28] (hexadecimal character format)

Example

```
AT+CSIM=?
OK

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

OK
```

2.2.36 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 should be aware of the precedence of the GSM/UMTS application commands to the TE commands.

AT+CRSM

Set Command AT+CRSM=<command>[,<fileid>[,<P1>,<P2>,<P3>[,<data>[,<pathid>]]]]	Response +CRSM: <sw1>,<sw2>[,<response>] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CRSM=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<command>	Integer type; command passed on by the MT to the SIM; refer 3GPP TS 51.011 [28] 176 READ BINARY 178 READ RECORD 192 GET RESPONSE 214 UPDATE BINARY 220 UPDATE RECORD 242 STATUS All other values are reserved
<fileid>	Integer type; this is the identifier of a 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 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 3GPP TS 51.011 [28]
<data>	String type; information which shall be 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] (e.g. "7F205F70" in SIM and UICC case). The <pathid> shall only be used in the mode "select by path from MF" as defined in ETSI TS 102 221 [60]
<sw1>, <sw2>	Integer type; information from the SIM about the execution of the actual command. These parameters are delivered to the TE in both cases, on successful or failed execution of the command
<response>	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 datafield. This information includes the type of file and its size (refer 3GPP TS 51.011 [28]). After READ BINARY, READ RECORD or RETRIEVE DATA command the requested data will be returned. <response> is not returned after a successful UPDATE BINARY, UPDATE RECORD or SET DATA command.

Example

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

OK
```

2.2.37 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 will open a new logical channel; select the application identified by the <dfname> received with this command and return a session Id as the response. The ME shall restrict the communication between the TE and the UICC to this logical channel.

This <sessionid> is to be used when sending commands with Restricted UICC Logical Channel access +CRLA or Generic UICC Logical Channel access +CGLA commands.

AT+CCHO

Set Command AT+CCHO=<dfname>	Response <sessionid> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CCHO=?	Response OK

Maximum Response Time	5s
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 in order to target a specific application on the smart card (e.g. (U)SIM, WIM, ISIM) using logical channels mechanism.

Example

```
AT+CCHO="A00000004374506173732E496F54"
1
OK
```

2.2.38 AT+CCHC close logical channel

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

AT+CCHC

Set Command AT+CCHC=<sessionid>	Response +CCHC OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CCHC=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

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

Example

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

2.2.39 AT+CGLA generic UICC logical channel access

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

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

Although Generic UICC Logical Channel Access command +CGLA allows TE to take control over the UICC-MT interface, there are some functions of the UICC-MT interface that logically do not need to be accessed from outside the TA/MT. Moreover, for security reason the GSM network authentication should not be handled outside the TA/MT. Therefore, it shall not be allowed to execute a Run GSM Algorithm command or an Authenticate command in GSM context from the TE using +CGLA at all-time whether the +CGLA is locked or unlocked. This shall not forbid the TE to send Authenticate commands in other security contexts (e.g. EAP security context).

For example, the TA/MT shall forbid the transfer of the Authenticate command to a USIM application when parameters P2 = 0 (GSM security context). See 3GPP TS 31.102 for USIM authenticate command definition.

AT+CGLA

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

Parameter

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

<response>	String type
	response to the command passed on by the UICC to the MT in the format as described in 3GPP TS 31.101 (hexadecimal character format)

Example

```
AT+CGLA=1,38,"81F10000E0051010A11223344556677889900"
+CGLA: 42,
"009868119122009036664601FA483D3E000000253D165EF45278745B0F4365C32CBE23CEEBB331839000"
OK
```

2.2.40 AT+CTZU automatic time zone update

Set command enables and disables automatic time zone update via 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.

AT+CTZU

Set Command AT+CTZU=<onoff>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CTZU?	Response +CTZU: <onoff> OK
Test Command AT+CTZU=?	Response +CTZU: (lists of supported <onoff>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<onoff>	Integer type
0	Disable automatic time zone update via NITZ.
1	Enable automatic time zone update via NITZ. Note: default value

Example

```
AT+CTZU=1
```

```
OK
```

```
AT+CTZU?
```

```
+CTZU: 1
```

```
OK
```

```
AT+CTZU=?
```

```
+CTZU: (0-1)
```

```
OK
```

2.2.41 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.

AT+CTZR

Set Command AT+CTZR=<reporting>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CTZR?	Response +CTZR: <reporting> OK
Test Command AT+CTZR=?	Response +CTZR: (lists of supported <reporting>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<reporting>	Integer type
0	disable time zone change event reporting.

Note: 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

representing the sum of the local time zone (difference between the local time and GMT expressed in quarters of an hour) plus daylight saving time. The format is "±zz", expressed as a fixed width, two digit integer with the range -48 ... +56. To maintain a fixed width, numbers in the range -9 ... +9 are expressed with a leading zero, e.g. "-09", "+00" and "+09"

<dst>

Integer type, indicating 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 will be 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 will be 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+CMOLR=?	+CMOLR: (list of supported <enable>s), (list of supported <method>s), (list of supported <horacc-set>s), (list of supported <hor-acc>s), (list of supported <ver-req>s), (list of supported <veracc-set>s), (list of supported <ver-acc>s), (list of supported <vel-req>s), (list of supported <repmode>s), (list of supported <timeout>s), (list of supported <interval>s), (list of supported <shape-req>s), (list of supported <plane>s), (list of supported <NMEA-req>s), (list of supported <third-party-address>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<enable>	Integer type
	0 Disables reporting and positioning
	1 Enables reporting of NMEA strings by unsolicited result code +CMOLRN: <NMEA-string>. Lack of data at each timeout is indicated by an unsolicited result code +CMOLRE.
	2 Enables reporting of GAD shapes by unsolicited result code +CMOLRG: <location_parameters>. Lack of data at each timeout is indicated by an unsolicited result code +CMOLRE.
<method>	3 Enables reporting of NMEA strings and GAD shapes by unsolicited result codes +CMOLRG: <location_parameters> and +CMOLRN: <NMEA-string>. Lack of data at each timeout is indicated by an unsolicited result code +CMOLRE.
	0 Unassisted GPS. Autonomous GPS only, no use of assistance data.
	1 Assisted GPS.
	2 Assisted GANSS.
	3 Assisted GPS and GANSS.
	4 Basic self location (the network determines the position technology).
	5 Transfer to third party. This method makes the parameters <shape-

		rep> and <NMEA-rep> irrelevant (any values are accepted and disregarded). The third party address is given in the parameter <thirdparty-address>.
	6	Retrieval from third party. This method is to get the position estimate of the third party. The third party address is given in the parameter <third-party-address>.
<hor-acc-set>	0	Horizontal accuracy not set/specified.
	1	Horizontal accuracy set in parameter <hor-acc>.
<hor-acc>		Integer type. Requested accuracy as horizontal uncertainty exponent. The value range is 0-127.
<ver-req>	0	Vertical coordinate (altitude) is not requested, 2D location fix is acceptable. The parameters <ver-accset> and <ver-acc> do not apply.
	1	Vertical coordinate (altitude) is requested, 3D location fix is required.
<ver-acc-set>	0	Vertical accuracy not set/specified.
	1	Vertical accuracy set/specified in parameter <ver-acc>.
<ver-acc>		Integer type. Requested accuracy as vertical uncertainty exponent The value range is 0-127. The default value is implementation specific.
<vel-req>	0	Velocity not requested.
	1	Horizontal velocity requested.
	2	Horizontal velocity and vertical velocity requested.
	3	Horizontal velocity with uncertainty requested.
	4	Horizontal velocity with uncertainty and vertical velocity with uncertainty requested.
<rep-mode>		Integer type. Reporting mode.
	0	Single report, the timeout for the MO-LR response request is specified by <timeout>.
	1	Periodic reporting, the timeout for each MO-LR response request is specified by <timeout> and the interval between each MO-LR is specified by <interval>.
<timeout>		Integer type. Indicates how long the MS will wait for a response after a MO-LR. The value range is in seconds from 1 to 65535.
<interval>		Integer type. The parameter is applicable to periodic reporting only. Determine the interval between periodic MO-LRs. The value range is in seconds from 1 to 65535, and must be greater than or equal to <timeout>.
<shape-req>		Integer type. This parameter is a sum of integers each representing a certain GAD shape that will be accepted in the unsolicited result code <location_parameters>. Note that only one GAD shape is present per unsolicited result code.
	1	Ellipsoid point.
	2	Ellipsoid point with uncertainty circle.
	4	Ellipsoid point with uncertainty ellipse.
	8	Polygon.

	16	Ellipsoid point with altitude.
	32	Ellipsoid point with altitude and uncertainty ellipsoid.
	64	Ellipsoid arc.
<plane>		Integer type. The parameter specifies whether the control plane or SUPL will be used for MO-LR.
	0	Control plane.
	1	Secure user plane (SUPL).
<NMEA-rep>		String type. The supported NMEA strings are specified as a comma separated values inside one string. If the parameter is omitted or an empty string is given, no restrictions apply and all NMEA strings are supported.
<third-party-address>		String type. The parameter is applicable to reporting to third party only, and specifies the address to the third party. This parameter has to be specified when <method> value is set to 5 or 6.
<location_parameters>		String type in UTF-8. This parameter provides XML-formatted strings of GADshape positioning data as defined in table 8.50-2. This parameter shall not be subject to conventional character conversion as per +CSCS. The XML according to the DTD in table 8.50-2 may be provided in one or multiple unsolicited result codes.

Example

```
AT+CMOLR=0,4,1,0,1,1,4,1,0,2,1,64
```

```
OK
```

```
AT+CMOLR?
```

```
+CMOLR: 0,4,0,,0,,,0,0,10,15,1,0,,
```

```
OK
```

```
AT+CMOLR=?
```

```
+CMOLR: (0-3),(0-6),(0-1),(0-127),(0-1),(0-1),(0-127),(0-4),(0-1),(1-65535),(1-65535),(1,2,4,8,16,32,64),(0-1)
```

```
OK
```

2.2.43 AT+CMTLR mobile terminated location request notification

Set command enables Mobile Terminated Location Request (MT-LR) notifications to the TE. The parameter <subscribe> enables or disables notification by an unsolicited result code. It is possible to enable notification of MT-LR performed over the control plane or over SUPL or both. Relevant location request parameters are provided in the unsolicited result code +CMTLR: <handle-id>,<notification-

type>,<locationtype>,<client-external-id>,<client-name>],[<plane>].

Read command returns the current value of <subscribe>.

Test command returns the supported values as a compound value.

AT+CMTLR

Set Command AT+CMTLR=[<subscribe>]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CMTLR?	Response +CMTLR: <subscribe> OK
Test Command AT+CMTLR=?	Response +CMTLR: (lists of supported <subscribe> values) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<subscribe>	Integer type. Enables and disables the subscription for MT-LR notifications 0 Disables reporting and positioning. 1 Subscribe for notifications of MT-LR over control plane. 2 Subscribe for notifications of MT-LR over SUPL. 3 Subscribe for notifications of MT-LR over control plane and SUPL.
<handle-id>	Integer type. ID associated with each MT-LR used to distinguish specific request in case of multiple requests. The value range is 0-255.
<notification-type>	Integer type. Information about the user's privacy. 0 The subscription may stipulate that positioning the user by a third party is allowed and the network may choose to inform the user as a matter of courtesy. 1 Locating the user is permitted if the user ignores the notification. 2 Locating the user is forbidden if the user ignores the notification.
<location-type>	Integer type. Indicates what type of the location is requested. 0 The current location. 1 The current or last known location. 2 The initial location.
<client-external-id>	String type. Indicates the external client where the location information is sent to (if required).
<client-name>	String type. Contains the string identifying the external client requesting the user's location

<plane>	Integer type. The parameter specifies whether the MT-LR came over control plane or SUPL.
0	Control plane.
1	Secure user plane (SUPL).

Example

```

AT+CMTLR=1

OK

AT+CMTLR?
+CMTLR: 1

OK

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

OK

```

2.2.44 AT+CMTLRA mobile terminated location request disclosure allowance

Set command allows or disallows disclosure of the location to the TE as a result of MT-LR by the parameter <allow>.

Read command returns the current values.

Test command returns the supported values.

AT+CMTLRA

Set Command AT+CMTLRA=<allow>,<handle-id>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+CMTLRA?	Response +CMTLRA: <allow>,<handle-id> OK
Test Command AT+CMTLRA=?	Response +CMTLRA: (lists of supported <allow> values)

	OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<allow>	Integer type. Enables and disables the allowance for location disclosure.
0	Location disclosure allowed.
1	Location disclosure not allowed.
<handle-id>	Integer type. ID associated with each MT-LR used to distinguish specific request in case of multiple requests. The value range is 0-255.

Example

```
AT+CMTLRA=1,1
```

```
OK
```

```
AT+CMTLRA?
```

```
+CMTLRA: 1,1
```

```
OK
```

```
AT+CMTLRA=?
```

```
+CMTLRA: (0,1)
```

```
OK
```

2.2.45 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.

AT+CRCES

Execution Command	Response
AT+CRCES	+CRCES: <Act>,<CE_level>,<CC>

	OK
Test Command	Response
AT+CRCES=?	OK
Maximum Response Time	5s
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-GSM-IoT).
0	No Coverage Class 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=?

OK

```

2.2.46 AT+CIPCA initial PDN context activation

The set command controls whether an initial PDP context (see subclause 10.1.0) shall be 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.

AT+CIPCA

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

Parameter

<n>	Integer type. Activation of PDP context upon attach.	
	0	Do not activate(Not support)
	1	Always activate(Not support)
	2	Activate when not roaming(Not support)
<AttachWithoutPDN>	3	No change in current setting
	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
```

2.2.47 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 will consist of a single line containing the name of the manufacturer, but manufacturers may choose to provide more information if desired.

AT+CGMI

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

Example

```
AT+CGMI
```

```
+CGMI: "eigencomm"
```

```
OK
```

```
AT+CGMI=?
```

```
+CGMI: "manufacturer_ID"
```

```
OK
```

2.2.48 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 will consist of a single line containing the name of the product, but manufacturers may choose to provide more information if desired.

AT+CGMM

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

Example

```
AT+CGMM
+CGMM: "eigencomm","EC616"

OK

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

OK
```

2.2.49 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.

AT+CPINR

Set Command AT+CPINR=<code>	Response +CPINR: <code>,<retries>[,<default_retries>] OK If there is any error, response: +CME ERROR: <err>
Execution Command AT+CPINR	Response +CPINR: <code>,<retries>[,<default_retries>] +CPINR:

	<code><code>,<retries>[,<default_retries>]</code> OK If there is any error, response: +CME ERROR: <code><err></code>
Test Command	Response
AT+CPINR=?	OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

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

2.2.50 AT+CGAUTH define PDP context authentication parameters

Set command allows the TE to specify authentication parameters for a PDP context identified by the (local) context identification parameter `<cid>` used during the PDP context activation and the PDP context modification procedures. Since the `<cid>` is the same parameter that is used in the +CGDCONT and +CGDSCONT commands, +CGAUTH is effectively as an extension to these commands. Refer subclause 9.2 for possible `<err>` values.

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

The test command returns values supported as compound values

AT+CGAUTH

Set Command	Response
AT+CGAUTH= <code><cid></code> [, <code><auth_proto></code>],[<code><userid></code>],[<code><password></code>]]	OK If there is any error, response:

	+CME ERROR: <err>
Read Command AT+CGAUTH?	Response [+CGAUTH: <cid>,<auth_proto>,<userid>,<password>] [<CR><LF>+CGAUTH: <cid>,<auth_proto>,<userid>,<password>[...]] OK
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)
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<auth_proto>	Integer type
	0 None. Used to indicate that no authentication protocol is used for this P
	1 PAP
The default value is 0	
<userid>	string type
	userId string, the max length is 20 characters with 1 line end mark;
	The default value is ""(NULL)
<password>	string type
	password string, the max 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

```

2.2.51 AT+IPR set UE baud rate

Set command sets the UE baud rate to be used.

Read command returns the current baud rate.

Test command returns baud rates supported by the UE.

AT+IPR

Set Command AT+IPR=<rate>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+IPR?	Response +IPR:<rate> OK
Test Command AT+IPR=?	Response +IPR:(list of supported auto detectable <rate> values),(list of fixed-only <rate> values) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<rate>	Baud rate at which the UE will accept commands. Note: 1. If this value is 0 for set command, UE will enter auto baud rate detection mode and 'AT' or 'at' shall be sent for baud rate detection. 2. If this value is 0 for set command, the character format is forced to 8 data, none parity, 1 stop, +ICF=3(see AT+ICF section)
--------	--

Example

```
AT+IPR=115200
OK

AT+IPR?
+IPR:115200
OK

AT+IPR=?
```

```
+IPR: (600,1200,2400,4800,9600,19200,38400,57600,115200,230400), (0,300,600,1200,2400,4800,9600,19200,38400,57600,115200,230400,460800)
```

OK

2.2.52 AT+CNMPSD No more PS data

This command indicates that no application on the MT is expected to exchange data.

AT+CNMPSD

Execution Command AT+CNMPSD	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CNMPSD=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Note, this AT command can cause triggering of the user plane Release Assistance Indication. (NB R14 RAI feature)

Example

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

2.2.53 AT+ICF set UE serial port character framing

Set command sets the UE character framing to be used.

Read command returns the current character framing.

Test command returns character framing supported by the UE.

AT+ICF

Set Command AT+ICF=<format>[,<parity>]	Response OK If there is any error, response: +CME ERROR: <err>
---	---

Read Command AT+ICF?	Response +ICF:<format>,<parity> OK
Test Command AT+ICF=?	Response +ICF:(list of supported <format> values),(list of supported <parity> values) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<format>	Integer type. It determines the number of bits in the data bits, the presence of a parity bit, and the number of stop bits in the start-stop frame
1	8 Data; 2 Stop
2	8 Data; 1 Parity; 1 Stop
3	8 Data; 1 Stop
4	7 Data; 2 Stop
5	7 Data; 1 Parity; 1 Stop
6	7 Data; 1 Stop
<parity>	Integer type. It determines how the parity bit is generated and checked, if present
0	Odd
1	Even

Example

```
AT+ICF=2,1 // Set as 8 data, even parity, 1 stop
OK

AT+ICF?
+ICF:2,1
OK

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

2.2.54 AT+CNUM request subscriber number

Action command returns the MSISDNs (up to 4) related to the subscriber stored in the USIM.

AT+CNUM

Set Command AT+CNUM	Response [+CNUM: [<alpha>],<number>,<type>] [+CNUM: [<alpha>],<number>,<type>] [...] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+CNUM=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<alpha>	String type Optional alphanumeric string associated with <number>
<number>	String type Phone number of format specified by <type>
<type>	Type of address octet in integer format (refer 3GPP TS24.008 subclause 10.5.4.7), e.g. 129 Unknown type 145 International type (use “+” for international access code) 161 National type

2.2.55 AT+CEER extended error report

Execution command causes the TA to return one or more lines of information text <report>, determined by the MT manufacturer, which should offer the users of the TA an extended report of the reason for the last unsuccessful attach

AT+CEER

Execution Command AT+CEER	Response +CEER: <report> OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<report>	String type The total number of characters, including line terminators, in the information text shall not exceed 2041 characters.
----------	--

2.3 3GPP Commands (27.005)

2.3.1 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 should be abortable. Currently we only support TEXT mode, not support PDU mode.

For text mode:

- entered text (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.
- the TA shall send a four-character sequence <CR><LF><greater_than><space> (IRA 13, 10, 62, 32) after command line is terminated with <CR>; after that text can be entered from TE to ME/TA.
- the DCD signal shall be in ON state while text is entered.
- the echoing of entered characters back from the TA is controlled by V.25ter echo command E.
- the entered text should be formatted as follows:
 - if <dc> (set with +CSMP) indicates that 3GPP TS 23.038 [2] GSM 7 bit default alphabet is used and <fo> indicates that 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 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 shall be sent to the TE after every carriage return entered by the user);
 - if TE character set is "HEX": the entered text should consist of two IRA character long hexadecimal numbers which ME/TA converts into the GSM 7 bit default alphabet characters. (e.g. 17 (IRA 49 and 55) will be converted to character П (GSM 7 bit default alphabet 23)).
 - if <dc> indicates that 8-bit or UCS2 data coding scheme is used or <fo> indicates that 3GPP TS 23.040 [3] TP-User-Data-Header-Indication is set: the entered text should consist of two IRA character long hexadecimal numbers which ME/TA converts into 8-bit octet (e.g. two characters 2A (IRA 50 and 65) will be 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:

Currently we only support TEXT mode, not support PDU mode.

AT+CMGS

Set Command If text mode(AT+CMGF=1): AT+CMGS=<da>[,<toda>]<CR> Text is entered<ctrl-Z/ESC>	If sending successful: +CMGS: <mr> If there is any error, response: +CMS ERROR: <err>
Maximum Response Time	60s
Parameter Saving Mode	NO_SAVE

Parameter

<da>	String type; in text mode (AT+CMGF=1) Destination address
<toda>	integer type; Type of destination address
<mr>	3GPP TS 23.040 [3] TP-Message-Reference in integer format.

Example

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

2.3.2 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.

AT+CSCA

Set Command AT+CSCA=<sca>[,<tosca>]	Response OK If there is any error, response: +CMS ERROR: <err>
Read Command AT+CSCA?	Response +CSCA: <sca>,<tosca>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<sca>	String type; 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. 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
```

2.3.3 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). Currently we only support TEXT mode.

Test command returns supported modes as a compound value.

AT+CMGF

Set Command AT+CMGF=<mode>	Response OK If there is any error, response: +CMS ERROR: <err>
Read Command AT+CMGF?	Response +CMGF: <mode>
Test Command AT+CMGF=?	Response +CMGF: (list of supported <mode>s)
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<mode>	integer type;
	1 Text mode

Example

```
AT+CMGF=1
```

```
OK
```

```
AT+CMGF?
```

```
+CMGF: 1
```

```
OK
```

2.3.4 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 3GPP TS 23.040 [3], it shall be given as a hexadecimal coded string (refer e.g. <pdu>) with double quotes.

AT+CSMP

Set Command AT+CSMP=<fo>[, <vp>[, <pid>[, <dc>]]]]	Response OK If there is any error, response: +CMS ERROR: <err>
Read Command AT+CSMP?	Response +CSMP: <fo>,<vp>,<pid>,<dc>
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<fo>	integer type; First octet for sms submit pdu,status report required
<vp>	integer type; Validity period
<pid>	integer type; Protocol identifier
<dc>	integer type; Data coding scheme

Example

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

```
OK
```

```
AT+CSMP?
```

```
+CSMP: 33,167,0,0
```

OK

2.3.5 +CMT new message received

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

SMS-DELIVERs are routed directly to the TE using unsolicited result code. Currently only support TEXT mode SMS-DELIVER message display.

+CMT

+CMT: <oa>, <scts><CR><LF><data> (text mode enabled)

Parameter

<length>	Integer type; 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-Time-Stamp in time-string format 3GPP TS 23.040 [3] TP-Discharge-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. E.g. 6th of May 1994, 22:10:00 GMT+2 hours equals to "94/05/06,22:10:00+08"
<data>	String type; The content of deliver message in HEX string format

Example

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

3 Extended Commands

3.1 EC General Commands

3.1.1 AT+ECBAND

The 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.

AT+ECBAND

Set Command AT+ECBAND=<mode> [, <band1> [, <band2>...]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECBAND?	Response +ECBAND: <mode> , <band1> , <band2> , ... OK
Test Command AT+ECBAND=?	Response +ECBAND: (list of supported <mode>s) , (list of supported <band>s) OK
Maximum Response Time	25s
Parameter Saving Mode	AUTO_SAVE

Parameter

<mode>	Integer type 0 NB-IOT mode(current support NB-IoT only)
<band>	Integer type Band list in decimal number. <band> values of 1, 3, 5, 8 are supported. The default value is decided by RF Calibration table

Example

```
AT+ECBAND?  
+ECBAND: 0,5,8,1,3  
  
OK  
  
AT+ECBAND=?  
+ECBAND: (0) , (1,3,5,8)  
  
OK  
  
AT+ECBAND=0,5,8  
OK
```

3.1.2 AT+ECLOGDBVER

This read command returns current unilog database version information.

AT+ECLOGDBVER

Read Command AT+ECLOGDBVER?	Response +ECLOGDBVER: <LogDbVserion> OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

3.1.3 AT+ECCFG

The command set UE extended configuration.

The read command return current setting of each parameters.

The test command returns values supported as a compound value.

AT+ECCFG

Set Command AT+ECCFG=<param1>,<value1>[,<param2>,<value2>[,< param3>,<value3>,[...]]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECCFG?	Response +ECCFG: <param1>,<value1>,<param2>,<value2>...<paramN>,<valueN> OK
Test Command AT+ECCFG=?	Response +ECCFG: (list of supported <param>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

<param>	String type, name of configuration parameter. Note: the max number of configuration parameters is 8 for a single AT command.
"AutoApn"	Whether UE auto set the attached APN according to the inserted SIM card. Note: a) Supported values: (0, 1) b) Default value: 0, just use the APN which is set by AT+ECATTBEARER
"PsSoftReset"	Whether UE support protocol stack soft reset. Note:

	<ul style="list-style-type: none"> a) Supported values: (0, 1) b) Default value: 0.
"UsimPowerSave"	<p>Enable/disable USIM power save.</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (0, 1) b) Default value: 1.
"UsimSimulator"	<p>Enable/disable virtual SIM card for instrument test, such as CMW500.</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (0, 1) b) Default value: 0.
"Rohc"	<p>Whether UE support ROHC.</p> <p>Note:</p> <ul style="list-style-type: none"> c) Supported values: (0, 1) d) Default value: 1.
"Ipv6RsForTestSim"	<p>Whether UE trigger IPv6 NDP (RS) procedure to get IPv6 prefix address, when the SIM card inserted is a TEST SIM.</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (0, 1) b) Default value 0. c) IPv6 NDP (RS) procedure is triggered by default if the inserted SIM card is not for testing.
"Ipv6RsDelay"	<p>Delay in seconds then UE trigger IPv6 NDP (RS) procedure to get IPv6 prefix address.</p> <p>Note:</p> <ul style="list-style-type: none"> d) Supported values: (0-65535) e) Default value 15s. f) IPv6 NDP (RS) procedure is triggered by default if the inserted SIM card is not for testing.
"PowerCfun"	<p>Default CFUN state after UE power-on or reboot;</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (0, 1, 4) b) Default value: 1 c) If set to 0, UE remain CFUN0 state (neither turn on protocol/RF nor SIM) after power-on or reboot; And could turn on protocol/RF and SIM via AT+CFUN=1 d) If set to 1, UE auto turn on protocol, and connect the network after power-on or reboot; e) If set to 4, UE only turn on SIM, disable (turn off) protocol/RF, after power-on or reboot;

"PsPowerOnMaxDelay"	<p>Max delay in seconds after power on, UE will delay in a random value between 0 to max delay value.</p> <p>Note:</p> <p>a) Supported values: (0-65535)</p> <p>b) Default value: 0.</p>
"SupportSms"	<p>Whether UE support SMS.</p> <p>Note:</p> <p>a) Supported values: (0, 1)</p> <p>b) Default value: 1</p>
"TauForSms"	<p>Whether need to trigger TAU procedure, if UE support SMS capability, while NW not support.</p> <p>Note:</p> <p>a) Supported values: (0, 1)</p> <p>b) Default value: 0</p>
"PlmnSearchPowerLevel"	<p>Set the PLMN search level when UE OOS;</p> <p>Note:</p> <p>a) Supported values: (0, 1, 2, 3)</p> <p>0 - OOS PLMN search interval: 30 sec, 1 min, 2 min</p> <p>1 - OOS PLMN search interval: 5 min, 10 min, 15 min</p> <p>2 - OOS PLMN search interval: 10 min, 30 min, 1 hour</p> <p>3 - OOS PLMN search interval: 30 sec, then stop PLMN search, and let AT: AT+ECPLMNS to start PLMN search</p> <p>4 - Don't perform PLMN search when OOS, let user to decide next action (whether perform PLMN search or not)</p> <p>b) Default value: 1</p>
"EpcO"	<p>Whether UE need to use "EPCO" in "PDN CONNECTION REQUEST" carried in "ATTACH REQUEST", and "ESM INFORMATION RESPONSE"; If set to 0, just use "PCO".</p> <p>Note:</p> <p>a) Supported values: (0, 1)</p> <p>b) Default value: 1</p> <p>c) It must be restricted to execute in power off or air plane state</p>
"T3324MaxValueS"	<p>Set user control T3324 value in second.</p> <p>Note:</p> <p>a) Support values: (0-16777215)</p> <p>b) Default value: 16777215</p> <p>c) If T3324MaxValueS is smaller than 65535 and</p>

	<p>network configured T3324 is equal to or greater than T3324MaxValueS (or network not configured T3324),use the configured T3324MaxValueS</p> <p>d) If T3324MaxValueS is smaller than 65535 and network configured T3324 is smaller than T3324MaxValueS , use the network configured value</p> <p>e) If T3324MaxValueS is equal to or greater than 65535,just means disable this feature(use network configured value)</p>
"BarValueS"	<p>UE bar period due to SIB14,after timer expiry,UE can retry access to network.</p> <p>Note:</p> <p>c) Supported values: (1-600)</p> <p>d) Default value: 120.</p>
"MultiCarrier"	<p>Whether UE support multi-carrier feature.</p> <p>Note:</p> <p>a) Support values: (0, 1)</p> <p>b) Default value: 1</p> <p>c) It must be restricted to execute in power off or air plane state</p>
"MultiTone"	<p>Whether UE support multi-tone feature.</p> <p>Note:</p> <p>a) Supported values: (0, 1)</p> <p>b) Default value: 1</p> <p>c) It must be restricted to execute in power off or air plane state</p>
"SupportUpRai"	<p>Whether UE support L2 (MAC layer) RAI feature, which only valid whether set to R14 version.</p> <p>Note:</p> <p>a) Supported values: (0, 1)</p> <p>b) Default value: 0</p> <p>c) It must be restricted to execute in power off or air plane state</p>
"DataInactTimer"	<p>Set the value of "data inactivity timer" in seconds, if this timer is not configured by NW (in MAC-MainConfig-NB), just use this setting value.</p> <p>Note:</p> <p>a) Supported value: (0, 15-255)</p> <p>b) Default value: 60</p> <p>c) If set to 0, just means this timer is invalid, don't need to start.</p> <p>d) It must be restricted to execute in power off or</p>

	air plane state
"RelaxMonitorDeltaP"	<p>Set the value of "SearchDeltaP" in DB for Relex-Monitor feature. If this value is not configured by NW (in SIB-NB), just use this setting value.</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (0-15) b) Default value: 6 c) It must be restricted to execute in power off or air plane state
"DisableNCellMeas"	<p>Whether UE need to disable Ncell measurement</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (0, 1) b) Default value: 0 c) It must be restricted to execute in power off or air plane state
"NbCategory"	<p>Set the value of NB-IOT category.</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (1, 2) b) Default value: <ul style="list-style-type: none"> Set to 1 if "RelVersion" is set to 13. Set to 2 if "RelVersion" is set to 14. c) If "NbCategory" is not configured, use default value as described above. d) If "RelVersion" is 14 and "NbCategory" is 2, R14 features(twoHarq, multiCarrierNPRACH, multiCarrierPaging) are supported; if "RelVersion" is 14 and "NbCategory" is 1, twoHarq is not supported, other R14 features are supported. e) It must be restricted to execute in power off or air plane state
"RelVersion"	<p>Set the NB release version.</p> <p>Note:</p> <ul style="list-style-type: none"> a) Supported values: (13,14) b) Default value: 13 c) It must be restricted to execute in power off or air plane state
"DisableSib14Check"	<p>Whether UE support sib14 check during establish procedure.</p> <p>Note:</p> <ul style="list-style-type: none"> a) Support values: (0, 1) b) Default value: 0, means UE will check sib14 during establish procedure.

<value>	Integer type
	value of configuration

Example

```

AT+ECCCFG="Rohc",0
OK

AT+ECCCFG?
+ECCCFG: "AutoApn",0,
"PsSoftReset",0,"UsimPowerSave",1,"UsimSimulator",0,"Rohc",0,"Ipv6RsForTestSim",0,"Ipv6RsDelay",0,"PowerCfun",1,"PsPowerOnMaxDelay",0,"SupportSms",1,"TauForSms",0,"PlmnSearchPowerLevel",1,"Epco",1,"T3324MaxValueS",16777215,"BarValueS",0,"MultiCarrier",1,"MultiTone",1,"SupportUpRai",0,"DataInactTimer",60,"RelaxMonitorDeltaP",0,"DisableNCellMeas",0,"NbCategory",1,"RelVersion",13,"DisableSib14Check",1

OK

AT+ECCCFG=?
("AutoApn","PsSoftReset","UsimPowerSave","UsimSimulator","Rohc","Ipv6RsForTestSim","Ipv6RsDelay","PowerCfun","PsPowerOnMaxDelay","SupportSms","TauForSms","PlmnSearchPowerLevel","Epco","T3324MaxValueS","BarValueS","MultiCarrier","MultiTone","SupportUpRai","DataInactTimer","RelaxMonitorDeltaP","DisableNCellMeas","NbCategory","RelVersion","DisableSib14Check")

OK

```

3.1.4 AT+ECPING

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

The test command returns values supported as a compound value.

AT+ECPING

Set Command AT+ECPING=<ipaddr/Url>/<mode>[,<count>[,<size>[,<timeout>[,rai_enable]]]]	Response OK If there is any error, response: +SOCKET ERROR: <err>
Test Command	Response

AT+ECPING=?	+ECPING: (list of supported <ipaddr/Url/mode>s),(list of supported <count> s),(list of supported <size>s),(list of supported <timeout>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type 0 Stop ping
<ipaddr/Url>	String type, IP address or URL
<count>	Integer type, ping times 1-255 Note: default value: 4. 255 means all the time
<size>	Integer type, payload size 1-1500 Note: default value: 32 bytes
<timeout>	Integer type, UE ping reply timeout after ping request.(ms) Note: <timeout> values between 1 to 600000 are supported, and default value: 20000
<rai_enable>	Integer type, if the value is 1, UE will send the last ping request packet with RAI indication 0-1 Note: default value: 0

Note:

- When one PING reply received in <timeout>, an unsolicited result code: +ECPING: SUCC, dest: <dest_ip_addr>, RTT: <rtt_time>ms will sent to TE.
- If no PING reply received in <timeout>, an unsolicited result code: +ECPING: FAIL, dest: <dest_ip_addr>, time out: <timeout>ms will sent to TE.
- If this is an ERROR meet during PING procedure, an unsolicited result code: +ECPING: ERROR, cause: <cause> will sent to TE.
- When PING procedure is done, an unsolicited result code: +ECPING: DONE<CR><LF>+ECPING: dest: <dest_ip_addr>, <count> packets transmittted, <reply_count> received, <lost_percent>% packet loss<CR> rtt min/avg/max = <rtt_min> / <rtt_avg> / <rtt_max> ms will sent to TE.

Example

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

```
AT+ECPING="180.97.33.107",10,32,60000
OK
+ECPING: SUCC, dest: 180.97.33.107, RTT: 334 ms
+ECPING: SUCC, dest: 180.97.33.107, RTT: 179 ms
```

...

```
Stop ping:  
AT+ECPING=0  
OK
```

3.1.5 AT+ECIPERF

The command tests the TCP/IP's uplink and downlink IPERF performance.

The test command returns values supported as a compound value.

AT+ECIPERF

Set Command AT+ECIPERF=<action>[,<protocol>[,<port>[,<ipaddr>[,<tpt>[,payload_size[,<packet_number>[,<duration>[,<report_interval>]]]]]]]]	Response OK If there is any error, response: +SOCKET ERROR: <err>
Test Command AT+ECIPERF=?	Response +ECIPERF: (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	5s
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

Note: One type of IPERF UDP server, in this mode, UE will send one

	UDP packet to remote server to setup the UDP connection, then UE wait to receive the DL UDP packets, and start the DL UDP IPERF server.
	5 Stop IPERF server
<protocol>	Integer type
	0 UDP
	1 TCP
<port>	Integer type, UDP/TCP port number. Note: a) if <action> is 1 or 4, this <port> is the destination server port number. b) if <action> is 3, this <port> is the local IPERF server port number. c) Default value: 5001
<ipaddr>	String type, IP address. Note: a) if <action> is 1 or 4, this <ipaddr> is mandatory, and indicate the destination server address. b) If <action> is 3 and the test domain is ipv6 domain, this <ipaddr> is mandatory. it must be the UE local ipv6 address.
<tpt>	Integer type, throughput in bps. Note: default value: 20000, the max value is 1200000 bps
<payload_size>	Integer type, payload size of UL UDP/TCP IPERF packet. Used for client mode 36-1472 Note: default value 1350
<packet_number>	Integer type, packet number of UE send, when acted as a client mode. 1-65000 Note: default value 0, UE will send packets all the time
<report_interval >	Integer type, report internal of IPERF service result. UE send the following unsolicited result codes periodically in this interval (in seconds). a) If <action> is 1, the unsolicited result codes: +ECIPERF: Client SUCC, pkg sent bytes: <bytes>, UL through put: <tpt> bps b) If <action> is 3 or 4, the unsolicited result codes: +ECIPERF: Server SUCC, pkg recv bytes: <bytes>, DL through put: <tpt> bps Note: default value: 10
<duration>	Integer type, IPERF service duration in seconds Note: if not specified, IPERF will not stop, before meet an error, or received a terminate command.

Note:

- a) When IPERF client service is finished (terminated/timeout), UE send the unsolicited result codes: +ECIPERF: Client END, pkg sent total bytes: <bytes>, average UL through put: <tpt> bps

- b) When IPERF server service is finished (terminated/timeout), UE send the unsolicited result codes: +ECIPERF: Server END, pkg recv total bytes: <bytes>, average DL through put: <tpt> bps
- c) If happens an error which caused the IPERF service can't go on, UE send the unsolicited result codes: +ECIPERF: Client FAIL, <err>; Or +ECIPERF: Server FAIL, <err>

Example

```
AT+ECIPERF=1,0,5001,"180.167.122.150",10000

OK

+ECIPERF: Client SUCC, pkg sent bytes: 13720, UL through put: 10976 bps

+ECIPERF: Client SUCC, pkg sent bytes: 9604, UL through put: 7683 bps

+ECIPERF: Client SUCC, pkg sent bytes: 12348, UL through put: 9878 bps

+ECIPERF: Client SUCC, pkg sent bytes: 12348, UL through put: 9878 bps
AT+ECIPERF=0

OK

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

3.1.6 AT+ECFREQ

The command set prefer EARFCN list, lock or unlock cell.

Read command returns the current EARFCN setting.

The test command returns values supported as a compound value.

AT+ECFREQ

Set Command If cell unlock(mode = 0): AT+ECFREQ=<mode> If set prefer EARFCN list (mode = 1): AT+ECFREQ=<mode>[, <earfcn1>[, <earfcn2>...]] If cell lock (mode = 2): AT+ECFREQ=<mode>[, <earfcn>[, <phyCellId>]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECFREQ?	Response If neither set prefer EARFCN list nor lock

	EARFCN/cell : OK If set prefer EARFCN list: +ECFREQ: <1>,<arfcn1>,<arfcn2>,... If lock EARFCN or lock cell: +ECFREQ: <2>,<arfcn>,<phyCellId> OK If both set prefer EARFCN list and lock EARFCN/cell: +ECFREQ: <1>,<arfcn1>,<arfcn2>,... +ECFREQ: <2>,<arfcn>,<phyCellId> OK
Test Command AT+ECFREQ=?	Response +ECFREQ: (list of supported <mode>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

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

Note:

AT+ECFREQ must be restricted to execute in power off or air plane state.

Example

```
AT+ECFREQ=1,3738,3734
```

```
OK
```

```
AT+ECFREQ?
```

```
+ECFREQ: 1,3738,3734
```

```
OK
```

```
AT+ECFREQ=2,3738,143
```

```
OK
```

```

AT+ECFREQ?

+ECFREQ: 1,3738,3734

+ECFREQ: 2,3738,143

OK
AT+ECFREQ=0

OK
AT+ECFREQ?

+ECFREQ: 1,3738,3734

OK
AT+ECFREQ=3

OK
AT+ECFREQ?

OK

```

3.1.7 AT+ECRMFPLMN

Set command remove FPLMN in NVM or SIM.

The test command returns values supported as a compound value.

AT+ECRMFPLMN

Set Command AT+ECRMFPLMN=<mode>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECRMFPLMN=?	Response +ECRMFPLMN: (list of supported <mode>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

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

Example

```
AT+ECRMFPLMN=0
OK
```

3.1.8 AT+ECATTBEARER

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

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

The test command returns values supported as a compound value.

AT+ECATTBEARER

Set Command AT+ECATTBEARER=<PDP_type>[,<eitf>[,<apn>[,<IPv4AddrAlloc>[,<NSLPI>[,<IPv4_MTU_discovery>[,<NonIP_MTU_discovery>[,<auth_proto>[,<userId>[,<password>]]]]]]]]]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECATTBEARER?	Response +ECATTBEARER: <pdnType>,<eitf>,<apnStr>,<ipv4allocType>,<NSLPI>,<ipv4Mtu>,<nonIpMtu>,<auth_proto>,<userId>,<password> OK
Test Command AT+ECATTBEARER=?	Response +ECATTBEARER: (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), (list of supported <auth_proto>s), (userId string), (password string)
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

<PDP_type>	Integer type, PDP type
	1 IPv4
	2 IPv6
	3 IPv4v6
	5 NON IP
The default value is 3	
<eitf>	Integer type
	0 Security protected ESM information transfer not required
	1 Security protected ESM information transfer required
The default value is 1	
<apn>	string type
	Apn string, the max length is 99 characters;
The default value is ""(NULL)	
<IPv4AddrAlloc>	Integer type
	0 IPv4 address allocate through NAS signaling
	1 IPv4 address allocate through DHCP (not applicable)
The default value is 0	
<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.
	1 indicates that this PDP context is is to be activated with the value for the low priority indicator set to "MS is not configured for NAS signaling low priority".
	The default value is 0
<IPv4_MTU_discovery>	Integer type
	0 IPv4 MTU size discovery not influenced by +ECATTBEARER
	1 IPv4 MTU size discovery through NAS signaling
The default value is 1	
<NonIP_MTU_discovery>	Integer type
	0 Non-IP MTU size discovery not influenced by +ECATTBEARER
	1 Non-IP MTU size discovery through NAS signaling
The default value is 0	
<auth_proto>	Integer type
	0 None. Used to indicate that no authentication protocol is used for this P
	1 PAP
The default value is 0	
<userid>	string type
	userId string, the max length is 20 characters with 1 line end mark;
	The default value is ""(NULL)
<password>	string type

password string, the max length is 20 characters with 1 line end mark;
The default value is ""(NULL)

Example

3.1.9 AT+ECSENDDATA

The set command could send data via control plane or user plane

AT+ECSENDDATA

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

Parameter

<cid>	Integer type; specifies a particular PDP context definition. Note: If <cid> set to 0, just use current default bearer to send this originating data.						
<data_length>	Integer type. Indicates the number of octets of the <data> information element. The max length is 950.						
<data>	string of octets.						
<RAI>	Integer type. Indicates the value of the release assistance indication <table border="1"> <tr> <td>0</td> <td>No information available</td> </tr> <tr> <td>1</td> <td>The MT expects that exchange of data will be completed with the transmission of this UL packet.</td> </tr> <tr> <td>2</td> <td>The MT expects that exchange of data will be completed with the receipt of a DL packet.</td> </tr> </table>	0	No information available	1	The MT expects that exchange of data will be completed with the transmission of this UL packet.	2	The MT expects that exchange of data will be completed with the receipt of a DL packet.
0	No information available						
1	The MT expects that exchange of data will be completed with the transmission of this UL packet.						
2	The MT expects that exchange of data will be 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. <table border="1"> <tr> <td>0</td> <td>Regular data</td> </tr> </table>	0	Regular data				
0	Regular data						

Note:

Difference with AT+CSODCP, AT+CSODCP limit to transmit data over control plane to network, but this AT don't have such limitation.

Example

```
AT+ECSSENDDATA=5,2,"ABCD"
```

```
OK
```

3.1.10 +RECVNONIP

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

+RECVNONIP

```
+RECVNONIP: <cid>,<data_length>,<data>
```

Parameter

<cid>	Integer type; specifies a particular PDP context definition. <cid> values of 0-10 are supported.
<data_length>	Integer type. Indicates the number of octets of the <data> information element.
<data>	string of octets.

Example

3.1.11 AT+ECPMUCFG

The command set PMU mode.

Read command returns the current setup.

Test command returns values supported as a compound value.

AT+ECPMUCFG

Set Command AT+ECPMUCFG=<enable>[,<mode> >]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPMUCFG?	Response +ECPMUCFG: <enable>[,<mode>] OK Note:

	If PMU is disabled, <mode> will not return.
Test Command AT+ECPMUCFG=?	Response +ECPMUCFG: (range of supported <enable>s) , (list of supported <mode>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<ENABLE>	Integer type; specifies to enable PMU or not
0	Disable the PMU, <mode> will be ignored and only enter idle
1	Enable the PMU
	The default value is 1
<MODE>	Integer type; specifies to depth of sleep mode
0	Active
1	Idle
2	Sleep1
3	Sleep2
4	Hibernate
	The default value is 1

Example

```
AT+ECPMUCFG=1,4
OK

AT+ECPMUCFG=0
OK

AT+ECPMUCFG?
+ECPMUCFG: 0

AT+ECPMUCFG=?
+ECPMUCFG: (0-1), (0-4)
```

3.1.12 AT+ECMSSEND

The command is used to send one SMS. Currently we only support TEXT mode.

AT+ECMSSEND

Set Command AT+ECMSSEND=<mode>,<da>,[<tda>,<test_sms>]	Response OK If there is any error, response:
---	--

	+CME ERROR: <err>
Maximum Response Time	60s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type
	1 TEXT mode
<da>	String type; in text mode (AT+CMGF=1)
	Destination address for TXT mode
<toda>	integer type; Type of destination address
<test_sms>	String type
	Message's content for TXT mode

Example

```
AT+ECSMSSSEND=1,"1064899990000",,"hello"
+CMGS: 8
OK
AT+ECSMSSSEND=1,1064899990000,,"hello"
+CMGS: 8
OK
```

3.1.13 AT+ECCGSN

The command sets the imei and sn. Use AT+CGSN to read IMEI or SN

AT+ECCGSN

Set Command AT+ECCGSN=<type>,<sn/imei>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECCGSN=?	Response +ECCGSN: (list of supported <type>s) , (data) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<type>	String type
	Can be "IMEI" or "SN"

<sn/imei>	String type
	IMEI (15bytes character only)
	SN(31bytes maximum as visable character)
	The default IMEI is 866818039921444
	The default SN is ""(NULL)

Example

```
AT+ECCGSN="IMEI","788596633100008"
```

```
OK
```

```
AT+ECCGSN="SN","01"
```

```
OK
```

3.1.14 AT+ECRFSTAT

The command shows the status of RF calibration.
Set command will return the RF calibration staus.

Set Command AT+ECRFSTAT	Response +ECRFSTAT: calibrate done OK If RF is not calibrated, response: +ECRFSTAT: not calibrate
Test Command AT+ECRFSTAT=?	Response +ECRFSTAT OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Example

```
AT+ECRFSTAT?
```

```
+ECRFSTAT: calibrate done
```

```
OK
```

3.1.15 AT+ECRST

The command restart the chip.

AT+ECRST

Execution Command	Response
AT+ECRST	OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Example

```
AT+ECRST
OK
```

3.1.16 AT+ECPSMR

The command report the power saving mode status.

AT+ECPSMR

Set Command	Response
AT+ECPSMR=<n>	OK If there is any error, response: +CME ERROR: <err>
Read Command	Response
AT+ECPSMR?	+ECPSMR: <n>, <mode> OK
Test Command	Response
AT+ECPSMR=?	+ECPSMR: (range of supported <n>s) OK
Indicate	Response
	+ECPSMR: <mode>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<n>	Integer type 0: disable unsolicited result code 1: enable unsolicited result code +ECPSMR: <mode> The default value is 0
<mode>	Integer type 0: normal mode 1: power saving mode

Example

```
AT+ECPSMR=1
OK
```



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

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

3.1.17 AT+ECPLMNS

Set command is used to trigger a PLMN search while UE is out of service, if 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.

AT+ECPLMNS

Set Command AT+ECPLMNS	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPLMNS?	Response +ECPLMNS: <state>[,<oosTimeStep>] OK
Test Command AT+ECPLMNS=?	Response OK
Maximum Response Time	5s
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
<oosTimeStep>	Integer type. The rest of time (in seconds) of OOS PLMN search timer, only present when <state> is 3.

Example

```
AT+ECPLMNS
OK
```

```

AT+ECPLMNS?
+ECPLMNS: 3, 108
OK

```

3.1.18 AT+ECCESQS

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 +ECCESQ:

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.

AT+ECCESQS

Execution Command AT+ECCESQS=<report level>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECCESQS?	Response +ECCESQS: <report level> OK
Test Command AT+ECCESQS=?	Response +ECCESQS: (list of supported <report level>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<report level>	Integer type
	0 disable unsolicited report.
	1 report +CESQ: <rxlev>, <ber>, <rscp>, <ecno>, <rsrq>, <rsrp>
	2 report +ECCESQ: RSRP, <rsrp>, RSRQ, <rsrq>, SNR, <snr>
The default value is 0	
<rxlev>	Integer type; not supported by NB-IoT
	99 not known or not detectable
<ber>	Integer type; not supported by NB-IoT
	99 not known or not detectable

<rsrp>	Integer type; not supported by NB-IoT 255 not known or not detectable
<ecno>	Integer type; not supported by NB-IoT 255 not known or not detectable
<rsrp>	Integer type For +CESQ reporting, refer to sub clause 2.2.9 AT+CESQ For +ECCESQ reporting, the range is -156 dBm to -44 dBm
<rsrq>	Integer type For +CESQ reporting, refer to sub clause 2.2.9 AT+CESQ For +ECCESQ reporting, the range is -34 dB to 2.5 dB
<snr>	Integer type The range is -30 dB to 30 dB

Example

```

AT+ECCESQS=2
OK

AT+ECCESQS?
+ECCESQ: 2

OK

AT+ECCESQS=?
+ECCESQS: (0-2)

OK

```

3.1.19 AT+ECSTATUS

This action command returns some key parameter in UE side.

AT+ECSTATUS

Execution Command	Response
AT+ECSTATUS	+ECSTATUS: PHY, DlEarfcn:<dlEarfcn>, UlEarfcn:<ulEarfcn>, PCI:<pci>, Band:<band>, RSRP:<rsrp>, RSRQ:<rsrq>, SNR:<snr>, AnchorFreqOfst:<freqOfst>, NonAnchorFreqOfst:<freqOfst>, CeLevel:<ceLevel>, DlBler:<dlBler>, UlBler:<ulBler>, DataInactTimerS:<dataInactTimers>, RetxBSRTimerP:<retxBSRTimerP>, TAvalue:<taValue>, TxPower<txPower>, NBMode:<nbMode>

	<pre>+ECSTATUS: L2, SrbNum:<srbNum>, DrbNum:<drbNum> +ECSTATUS: RRC, State:<rrcState>, TAC:<tac>, CellId:<cellId> +ECSTATUS: EMM, EmmState:<emmState>, EmmMode:<emmMode>, PTWms:<ptwMs>, EDRXPeriodMs:<eDRXPeriodMs>, PsmExt3412TimerS:<psmExt3412TimerS>, T3324TimerS:<T3324TimerS>, T3346RemainTimeS:<T3346RemainTimeS> +ECSTATUS: PLMN, PlmnState:<plmnState>, PlmnType:<plmnType>, SelectPlmn:<selectPlmn> +ECSTATUS: ESM, ActBearerNum:<actBearerNum>, APN:<apn>, IPv4:<ipaddr> +ECSTATUS: CCM, Cfun:<cfun>, IMSI:<imsi> OK</pre>
Set Command AT+ECSTATUS=<module>	Response +ECSTATUS: <module>, <name>:<value>[, <name>:<value>[,...]] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSTATUS=?	Response +ECSTATUS: (list of supported <module>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<module>	Unquoted string Modules of protocol stack, supported value: PHY, L2, RRC, EMM, PLMN, ESM, CCM.
<dlEarfcn>	Integer type Downlink earfcn, value range is 0~262143
<ulEarfcn>	Integer type Uplink earfcn, value range is 0~262143
<pci>	Integer type Physical cell ID, value range is 0~503
<band>	Integer type

	Band, value range is 0~70
<rsrp>	Integer type Value in dBm, range is -156dBm ~ -44dBm
<rsrq>	Integer type Value in dB, range is -34dB ~ -2.5dB
<snr>	Integer type Value in dB, range is -30dB ~ 30dB
<AnchorFreqOfst>	Integer type Range is 0 ~ 21, stand for Frequency Offset {Invalid, -10,-9,-8,-7,-6,-5,-4,-3,-2,-1,-0.5,0,1,2,3,4,5,6,7,8,9}
<NonAnchorFreqOfst >	Integer type Range is 0 ~ 21, stand for Frequency Offset {Invalid, -10,-9,-8,-7,-6,-5,-4,-3,-2,-1,-0.5,0,1,2,3,4,5,6,7,8,9}
<ceLevel>	Integer type 0 CE level 0 1 CE level 1 2 CE level 2
<dlBler>	Integer type Downlink block error, value range is 0~10000
<ulBler>	Integer type Uplink block error, value range is 0~10000
<dataInactTimerS>	Integer type Data inactive timer in seconds, value range is 0~180
<retxBSRTimerP>	Integer type Timer for BSR reporting, value in number of PDCCH periods. Value pp4 corresponds to 4 PDCCH periods, pp16 corresponds to 16 PDCCH periods and so on.
<taValue>	Integer type Timing advance value. Value range is 0~1282. -1 is invalid value
<txPower>	Integer type Current TX power level in dBm. Value range is -45~23. -128 is invalid value
<NBMode>	String type Value range is "InBand Same PCI", "InBand Diff PCI", "Guard Band", "Stand alone", "Unknown"
<drbNum>	Integer type Value range is 0~2
<rrcState>	String type Value range is "DEACT", "OOS", "IDLE", "SUSPEND IDLE", "CONNECTED", "UNKONWN"

<tacl>	Integer type Value range is 0~65534
<cellId>	Integer type Value range is 0~268435455
<emmState>	String type Value range is "NULL", "DEREG", "REG INIT", "REG", "DEREG INIT", "TAU INIT", "SR INIT", "UNKNOWN"
<emmMode>	String type Value range is "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 Value range is "NO PLMN", "SEARCHING", "SELECTED", "UNKNOWN"
<plmnType>	String type Value range is "HPLMN", "EHPLMN", "VPLMN", "UPLMN", "OPLMN", "UNKNOWN"
<selectPlmn>	String 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 (string with double quotes)

Example

AT+ECSTATUS

```

+ECSTATUS: PHY, DlEarfcn:3738, UlEarfcn:21738, PCI:11, Band:8, RSRP:-91, RSRQ:-8, SNR:8,
AnchorFreqOfst:11, NonAnchorFreqOfst:0,CeLevel:0, DlBler:0/100, UlBler:0/100,
DataInactTimerS:0, RetxBSTimerP:0, TAvalue:-1, TxPower -128, NBMode:"Stand alone"

+ECSTATUS: L2, SrbNum:0, DrbNum:0

+ECSTATUS: RRC, State:"IDLE", TAC:23369, CellId:26224411

+ECSTATUS: EMM, EmmState:"REG", EmmMode:"IDLE", PTWms:5120, EDRXPeriodMs:40960,
PsmExT3412TimerS:0, T3324TimerS:300, T3346RemainTimeS:0

+ECSTATUS: PLMN, PlmnState:"SELECTED", PlmnType:"EHPLMN", SelectPlmn:"0x460,0xf000"

+ECSTATUS: ESM, ActBearerNum:1, APN:"cmnbiot.MNC004.MCC460.GPRS", IPv4:"100.83.34.10"

+ECSTATUS: CCM, Cfun:1, IMSI:"460043263600041"

OK

AT+ECSTATUS=PHY
+ECSTATUS: PHY, DlEarfcn:3738, UlEarfcn:21738, PCI:11, Band:8, RSRP:-91, RSRQ:-8, SNR:8,
AnchorFreqOfst:11, NonAnchorFreqOfst:0,CeLevel:0, DlBler:0/100, UlBler:0/100,
DataInactTimerS:0, RetxBSTimerP:0, TAvalue:-1, TxPower -128, NBMode:"Stand alone"

OK

AT+ECSTATUS=?
+ECSTATUS: (PHY,L2,RRC,EMM,PLMN,ESM,CCM)

OK

```

3.1.20 AT+ECICCID

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

AT+ECICCID

Execution Command	Response
AT+ECICCID	+ECICCID: <ICCID>
	OK
	If there is any error, response: +CME ERROR: <err>
Test Command	Response

AT+ECICCID=?	OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ICCID>	String type
	Integrated circuit card identification

Example

```
AT+ECICCID
+ECICCID: 89861119220009636664

OK
```

3.1.21 AT+ECBCINFO

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

AT+ECBCINFO

Execution Command AT+ECBCINFO[=<mode>[,<timeoutS>[,<save_for_later>[,<max_cell_number>[,<report_mode>]]]]]	Response +ECBCINFOSC: <earfcn>,<pci>,<rsrp>,<rsrq>,<mcc>,<mnc>,<cellid>,<tac> [<CR><LF>+ECBCINFONC: <earfcn>,<pci>,<rsrp>,<rsrq>[,<mcc>,<mnc>,<cellid>,<tac>] [...]] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECBCINFO=?	Response +ECBCINFO: (list of supported <mode>S), (range of supported <timeoutS>), (list of supported <save_for_later>S), (range of supported <max_cell_number>S), (list of supported <report_mode>S) OK
Maximum Response Time	305s
Parameter Saving Mode	NO_SAVE

Parameter

<mode>	Integer type
--------	--------------

	<p>0 Only search/measure neighbor cell signal quality: RSRP and RSRQ, don't acquire neighbor cell SIB-NB, don't return cell location information. In this mode, neighbor cell information in AT response is: +ECBCINFONC: <earfcn>, <pci>, <rsrp>, <rsrq></p> <p>Note: default value 0</p>
	<p>1 Search cell, including measure cell and acquire cell SIB1-NB to get cellID, PLMN, and tac info. In this mode, neighbor cell information in AT response is: +ECBCINFONC: <earfcn>, <pci>, <rsrp>, <rsrq>, <mcc>, <mnc>, <cellid>, <tac></p>
	<p>2 Read saved cell information, <save_for_later> set to 1 should be called before, otherwise, no cell information will be returned.</p>
<timeoutS>	<p>Integer type</p> <p>MAX measurement time in unit of seconds, cell information should be acquired/returned in this time period.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. Range: (4-300) 2. Default value: 8
<save_for_later>	<p>Integer type</p> <p>Whether need to save cell information, which could be returned in case of AT+ECBCINFO=2 later.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. As the cell information should be saved into flash when enter deep sleep mode: sleep2 and HIB, don't suggest to save it frequently. 2. List value: (0,1) 3. Default value: 0, not save.
<max_cell_number>	<p>Integer type</p> <p>Max number of cells to be measured or searched, including serving cell.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. The more cells needed, the longer measurement time it takes. 2. Range: (1-5) 3. Default value: 5
<report_mode>	<p>Integer type</p> <p>0 Synchronous mode. Report cell info in AT response. Note: Default value: 0</p> <p>1 Asynchronous mode. AT response immediately, measured cell information returned in URC: [<CR><LF>+ECBCINFOSC: <earfcn>, <pci>, <rsrp>, <rsrq>, <mcc>, <mnc>, <cellid>, <tac>[<CR><LF> [<CR><LF>+ECBCINFONC: <earfcn>, <pci>, <rsrp>, <rsrq>[, <mcc>, <mnc>, <cellid>, <tac>] [...]]</p>
<earfcn>	Integer type

	Indicate the EARFCN of the cell. Range 0 ~ 262143
<pci>	Integer type
	Indicate the physical cell ID. Range 0 ~ 503
<rsrp>	Integer type
	Indicate the measurement of RSRP value, in unit of dBm. Range -156 ~ -44
<rsrq>	Integer type
	Indicate the measurement of RSRQ value, in unit of dB. Range -34 ~ 25
<mcc>	String type
	Indicate the mobile country code
<mnc>	String type
	Indicate the mobile network code
<cellId>	String type
	Four byte E-UTRAN cell ID in hexadecimal format
<tac>	String type
	Two byte tracking area code in hexadecimal format

Note:

1. This AT command could be executed in cfun0/cfun4 state.
2. This AT command could also be executed, if SIM card is not inserted.
3. If the UE is not registered successfully (e.g. in cfun0/cfun4 state, or if no SIM card inserted), then the cell with the best signal quality (RSRP) will be regarded as the serving cell.
4. If the UE is registered, then the cell with the same registered PLMN (or EPLMN) will be returned.

Example

```
//test command
AT+ECBCINFO=?

+ECBCINFO: (0,1,2),(4-300),(0,1),(1-5),(0,1)

OK

//execute, default mode is 0
AT+ECBCINFO

+ECBCINFOESC: 3684,9,-87,-10,"460","00","0190271A","5B49"
+ECBCINFOESC: 3686,425,-86,-9
+ECBCINFOESC: 3688,172,-87,-12
+ECBCINFOESC: 3686,484,-89,-12

OK

//execute, mode set to 0
```

```
AT+ECBCINFO=0

+ECBCINFOESC: 3684,9,-84,-9,"460","00","0190271A","5B49"
+ECBCINFOFNC: 3688,172,-86,-11
+ECBCINFOFNC: 3686,425,-93,-16

OK

//execute, mode set to 1
AT+ECBCINFO=1,12,1,2

+ECBCINFOESC: 3684,9,-86,-9,"460","00","0190271A","5B49"
+ECBCINFOFNC: 3686,425,-88,-10,"460","00","0192781A","5B65"

OK

//execute, read last saved cell information
AT+ECBCINFO=2

+ECBCINFOESC: 3684,9,-86,-9,"460","00","0190271A","5B49"
+ECBCINFOFNC: 3686,425,-88,-10,"460","00","0192781A","5B65"

OK

//execute, asynchronous report mode
AT+ECBCINFO=1,8,0,5,1

OK

+ECBCINFOESC: 3684,9,-84,-9,"460","00","0190271A","5B49"

+ECBCINFOFNC: 3688,172,-86,-8,"460","00","0182201A","5B4A"

+ECBCINFOFNC: 3686,425,-88,-10,"460","00","0192781A","5B65"

+ECBCINFOFNC: 3684,124,-88,-15,"460","00","00D5205A","5B2C"

+ECBCINFOFNC: 3686,484,-92,-14,"460","00","0192891A","5B4A"
```

3.1.22 AT+ECDNS

This command to get the IP address for a specific URL. As a limitation now, only one IP address is

returned for a URL

AT+ECDNS

Execution Command AT+ECDNS=<url>	Response +ECDNS: <ipaddr> OK If there is any error, response: +SOCKET ERROR: <err>
Test Command AT+ECDNS=?	Response OK
Maximum Response Time	30s
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+ECDNS="www.baidu.com"  
+ECDNS: "39.156.66.14"  
  
OK
```

3.1.23 AT+ECDNSCFG

Set command set the default DNS server addresses configuration. It is the lowest priority for DNS server addresses setting. When the DNS server addresses is invalid which is assigned by core network or operator default setting. The DNS server addresses will be configured.

AT+ECDNSCFG

Execution Command AT+ECDNSCFG=<ipaddr1>[,<ipaddr2> [,<ipaddr3>[,<ipaddr4>]]]	Response OK If there is any error, response: +SOCKET ERROR: <err>
Read Command AT+ECDNSCFG?	Response +ECDNSCFG: <ipaddr1>[,<ipaddr2> [,<ipaddr3>[,<ipaddr4>]]] OK

Test Command AT+ECDNSCFG=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<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+ECDNSCFG?
+ECDNSCFG:
"114.114.114.114","114.114.115.115","240c:0000:0000:0000:0000:0000:6666", "240c:0000:
0000:0000:0000:0000:6644"

OK
AT+ECDNSCFG="114.114.114.12"
OK
```

3.1.24 AT+ECEMMTIME

This command report and get the Emm Time State, include T3346、T3448 and T3412/extend T3412.

AT+ECEMMTIME

Set Command AT+ECEMMTIME=<bitmap>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECEMMTIME?	Response +ECEMMTIME:0,<timeState>[,<remainTimeValue>] +ECEMMTIME:1,<timeState>[,<remainTimeValue>] +ECEMMTIME:2,<timeState>[,<remainTimeValue>] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECEMMTIME=?	Response +ECEMMTIME: (range of supported <bitmap>s) OK
Indicate	Response +ECEMMTIME:

<timeId>,<timeState>[,<timeValue>]

Parameter

<bitmap>	Integer type bit 0: enable/disable unsolicited result code T3346 bit 1: enable/disable unsolicited result code T3448 bit 2: enable/disable unsolicited result code T3412/ext T3412 The default value is 0
<timeId>	Integer type 0: Emm Timer: T3346 1: Emm Timer: T3448 2: Emm Timer: T3412/ext T3412
<timeState>	Integer type 0: Start 1: Stop 2: Expiry
<remainTimeValue>	Integer type Time remain value in seconds, Only include when <timeState> is 0.
<timeValue>	Integer type Time value in seconds, Only include when < timeState> is 0.

Example

```
AT+ECEMMTIME=7
OK

AT+ECEMMTIME?
+ECEMMTIME: 0,1
+ECEMMTIME: 1,1
+ECEMMTIME: 2,0,3240
OK

+ECEMMTIME: 2,0,3240
```

3.1.25 AT+ECPCFG

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

Read command returns the current plat config setting.

AT+ECPCF

G

Set	Response
-----	----------

Command AT+ECPCFG= <mode>,<value>	OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPCFG?	Response +ECPCFG: "faultAction":<value>,"dumpToATPort" :<value>,"startWDT":<value>,"logCtrl":<value>,"logLevel":<value>,"logBaudrate":<value>,"slpWaitTime":<value> OK
Test Command AT+ECPCFG=?	Response +ECPCFG: <option>,<setting> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	String type
	faultAction Set the hardfault 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 4 0: dump full exception info to flash and EPAT tool then trapped in endless loop 1: print necessary exception info then reset 2: dump full exception info to flash then reset 3: dump full exception info to flash and EPAT tool then reset 4: reset directly. Suggest set to this value when mass production stage.
	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. Suggest set to this value when mass production stage.

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: info 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

3.1.26 AT+ECSLEEP

This command is used for power consumption test. After executing this command, UE will enter related low power state. And UE could be wake up by wakeup PAD, after wake up, UE will reboot.

AT+ECSLEEP

Set Command AT+ECSLEEP=<state>	Response +ECSLEEP: <mode> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSLEEP=?	Response +ECSLEEP: <state> OK
Maximum Response Time	5s
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

3.1.27 AT+ECSIMSLEEP

The command set UE to allow SIM card sleep (power off SIM) or not (power on SIM) for AT+CSIM and AT+CRSM. It's only allowed when "UsimPowerSave" is 1 set by AT+ECCFG. If "UsimPowerSave" is 1, shall set SIM sleep not allowed (power on SIM) before use AT+CSIM or AT+CRSM, then set SIM sleep allowed (power off SIM) to save power after finish all commands of AT+CSIM or AT+CRSM.

The read command return current setting of each parameters.

The test command returns values supported as a compound value.

AT+ECSIMSLEEP

Set Command AT+ECSIMSLEEP=<mode>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECSIMSLEEP?	Response +ECSIMSLEEP: <mode> OK
Test Command AT+ECSIMSLEEP=?	Response +ECSIMSLEEP: (list of supported <mode>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

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

Note:

Default value: 1

3.1.28 AT+ECCGSNLOCK

This command is used to set lock flag for IMEI and SN. If locked, IMEI and SN could not be write via AT+ECCGSN command. The lock feature maybe required in production stage, to prevent customer's miss operation. If lock is set, it could not be cleared via AT command, the only way to clear it is via flash tool (erase related region).

The set command set lock for IMEI and SN.

The test command returns parameter supported as a compound value.

The read command returns current lock status.

AT+ECCGSNLOCK

Set Command AT+ECCGSNLOCK=<para>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECCGSNLOCK =?	Response +ECCGSN: <list of support para> OK
Read Command AT+ECCGSNLOCK?	Response +ECCGSN: <imeistatus,snstatus> OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	5s
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+ECCGSNLOCK=IMEI//lock imei
OK
```

```

AT+ECCGSNLOCK?
+ECCGSNLOCK: IMEI LOCKED,SN NOT LOCKED
OK

AT+ECCGSN=IMEI,XXXXXX
+ECCGSN:IMEI LOCKED
+CME ERROR: 50

```

3.1.29 AT+ECSAVEFAC

This command is used in production line, which saves related regions to default reliable region. e.g. after IMEI/SN was written or RF calibration was done. Default reliable regions is used to restore factory setting.

The set command saves related regions to default reliable region..

The test command returns mode supported as a compound value.

AT+ECSAVEFAC

Set Command AT+ECSAVEFAC=<mode>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSAVEFAC=?	Response +ECSIMSLEEP: <mode> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

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

3.1.30 AT+DMCONFIG

The command set the parameters need by registering to the Device Manager Platform of China Mobile.

The read command returns the relevant parameters that have been set.

AT+DMCONFIG

Set Command AT+DMCONFIG=<mode>,<lifetime>,<appkey>,<secret>,<platform>	Response OK If there is any error, response: +DM ERROR: <err>
Get Command AT+DMCONFIG?	Response +DMCONFIG: <mode>,<lifetime>,<appkey>,<secret>,<platform> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	Integer type 0 Not register to DM platform 1 Register to DM platform
<lifetime>	Integer type Lifetime, unit: minute
<appkey>	String type A sting of characters gived by China Mobile
<secret>	String type A sting of characters gived by China Mobile
<platform>	Integer type 0 Register to commercial platform 1 Register to test platform

3.1.31 AT+DMREGSTAT

The execution command get the DM client's register state.

AT+DMREGSTAT

Execution Command AT+DMREGSTAT	Response +DMREGSTAT: <state> OK If there is any error, response: +DM ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<state>	Integer type
0	Not register to DM platform, because not open the DM mode
1	Register to DM platform success
2	Register failed, register timeout or rejected by the server

3.1.31 AT+ECTASKINFO

The execution command returns all the task's name, id, status, priority and stack information. This command is only for debug purpose.

AT+ECTASKINFO

Execution Command AT+ECTASKINFO	Response +ECTASKINFO: <task information> OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

3.1.32 AT+ECTASKHISTINFO

The execution command shows most recent tasks' scheduling history. This command is only for debug purpose.

AT+ECTASKHISTINFO

Execution Command AT+ECTASKHISINFO	Response +ECTASKHISINFO: <task scheduling history> OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

3.1.33 AT+ECSHOWMEM

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

AT+ECSHOWMEM

Execution Command	Response
-------------------	----------

AT+ECSHOWMEM	+ECSHOWMEM: <curr_free_heap, min_free_heap> OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<curr_free_heap >	Integer type
	0—heap size current remained free memory size in heap
<min_free_heap >	Integer type
	0--heap size minimum heap memory size ever remaining in heap

3.1.34 AT+ECSYSTEST

This command is only for debug purpose.

The test command returns option supported as a compound value.

The set command triggers a test feature.

AT+ECSYSTEST

Test Command AT+ECSYSTEST=?	Response +ECSYSTEST: <option> OK
Set Command AT+ECSYSTEST=<option>	Response KO If there is any error, response: +CME ERROR: <err> When <test>="assert" or "fsassert" system assert
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

< option >	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 hardfault for test
-----------	------------------------------

3.1.35 AT+ECVOTECHK

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

This command is only for debug purpose.

AT+ ECVOTECHK

Set Command AT+ECVOTECHK	Response Sleep Vote Info: <vote info detail> OK
Maximum Response Time	5s
Parameter Saving Mode	NO SAVE

NOTE: vote info detail shows comprehensive information which will affect sleep process in EC616 SDK.

The detail info is separated into 5 parts.

Part1: user set sleep depth limitation.

Part2: EC internal sleep vote result.

Part3: application vote result.

Part4: user registered sleep depth callback.

Part5: driver vote result.

For more detail information, please refer to `slpman_ec616.h` in SDK and EC616 low power development manual.

Example

```
AT+ECVOTECHK
```

```
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_ec616.h`.)

```
EC SDK Vote for:              Hibn    (Note: SDK internal vote result) //part2
```

```
Detail: 0x0,0x0,0x0
```

```
Application Vote for:         Hibn    //part3
```

(Note: Normally, application use the vote API in `slpman_ec616.h` to control 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 show more detail of vote the vote result, including vote handle, name information, vote state and vote counter

Prohibit State can be "NULL", "Slp1", "Slp2", "Hibn"

Prohibit State: NULL → App do not prohibit to enter all sleep state

Prohibit State: Slp2 → App prohibit to enter sleep2, so the system can only sleep to sleep1)

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 info. The bitmap is corresponding to the enum type slpDrvVoteModule_t in slpman_ec616.h

The vote mask indicate the sleep flow do not care about 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, PMU module do not take ADC vote result into consideration. It will enter sleep as soon as UART clear the bit 0

For drivers, if the (bitmap & (~mask)) != 0 the system can not go to sleep1 or deeper state)

OK

3.1.36 AT+ECURC

The command close/open URC (unsolicited result code) report.

AT+ ECURC

Set Command AT+ECURC=<urcStr>,<value> >	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECURC?	Response +ECURC: "CREG":<value>,"CEREG":<value>,"CEDRXP":<value>," CCIOTOPTI":<value>,"CSCON":<value>,"CTZEU":<value >,"ECCESQ":<value>,"CGEV":<value>,"ECPSMR":<value >,"ECPTWEDRXP":<value>,"ECPIN":<value>,"ECPADDR": <value>,"ECPCFUN":<value>

	OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECURC=?	Response +ECURC: "ALL": (0-1), "CREG": (0-1), "CEREG": (0-1), "CEDRXP": (0-1), "CCIOTOPTI": (0-1), "CSCON": (0-1), "CTZEU": (0-1), "ECCESQ": (0-1), "CGEV": (0-1), "ECPSMR": (0-1), "ECPTWEDRXP": (0-1), "ECPIN": (0-1), "ECPADDR": (0-1), "ECPCFUN": (0-1) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE Note: Set of <value> will save to NVM, and the default value is 0.

Parameter

<urcStr>	String type;
"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
"CTZEU"	unsolicited result code +CTZEU
"ECCESQ"	unsolicited result code +ECCESQ
"CGEV"	unsolicited result code +CGEV
"ECPSMR"	unsolicited result code +ECPSMR
"ECPTWEDRXP"	unsolicited result code +ECPTWEDRXP
"ECPIN"	unsolicited result code +ECPIN
"ECPADDR"	unsolicited result code +ECPADDR
"ECPCFUN"	unsolicited result code +ECPCFUN
<value>	Integer type;
0	disable unsolicited result code report
1	enable unsolicited result code report

Example

```
AT+ECURC="ALL",1

OK

AT+ECURC?
+ECURC:
"CREG":1,"CEREG":1,"CEDRXP":0,"CCIOTOPTI":0,"CSCON":1,"CTZEU":1,"ECCESQ":1,"CGEV":1,"ECPSMR":1,"ECPTWEDRXP":1,"ECPIN":1,"ECPADDR":1,"ECPCFUN":1

OK
AT+ECURC =?

+ECURC: "ALL":(0-1),"CREG":(0-1),"CEREG":(0-1),"CEDRXP":(0-1),"CCIOTOPTI":(0-1),"CSCON":(0-1),"CTZEU":(0-1),"ECCESQ":(0-1),"CGEV":(0-1),"ECPSMR":(0-1),"ECPTWEDRXP":(0-1),"ECPIN":(0-1),"ECPADDR":(0-1),"ECPCFUN":(0-1)

OK
```

3.1.37 AT+ECPTWEDRXS

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 $\langle n \rangle = 2$ and there is a change of the paging time window and eDRX parameters provided by network:

```
+ECPTWEDRXP: <AcT-type>[,<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+ECPTWEDRXS=3`. In this form, paging time window and eDRX will be disabled and data for all parameters in `AT+ECPTWEDRXS` command will be removed.

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

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

AT+ECPTWEDRXS

Set Command	Response
-------------	----------

AT+ECPTWEDRXS=<mode>, <AcT-type>[, <Requested_Paging_time_window>[, <Requested_eDRX_value>]]	OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPTWEDRXS?	Response +ECPTWEDRXS: <AcT-type>, <Requested_Paging_time_window>, <Requested_eDRX_value>[, <NW_provided_eDRX_VALUE>[, <Paging_time_window>]] OK
Test Command AT+ECPTWEDRXS=?	Response +ECPTWEDRXS: (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)
Maximum Response Time	5s
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, i.e. the most recent setting of <mode> will take effect for all specified values of <AcT-type>. 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: +ECPTWEDRXP: <AcT-type>[, <Requested_Paging_time_window>[, <Requested_eDRX_value>[, <NW_provided_eDRX_value>[, <Paging_time_window>]]] 3 Disable the use of paging time window and eDRX and discard all parameters for pagint time window and eDRX.
<AcT-type>	Integer type; indicates the type of access technology. 0 Access technology is not using eDRX 5 NB IOT
<Requested_Paging_time_window >	String type; half a byte in a 4 bit format. The paging time window refers to bit 8 to 5 of octet 3 of the Extended DRX parameters information element. (e.g."0000" equals 2.56 seconds)
<Requested_eDRX_value>	String type; half a byte in a 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters

	information element. (e.g."0010" equals 20.48 seconds)
<NW-provided_eDRX_value >	String type; half a byte in a 4-bit format. The eDRX value refers to bit 4 to 1 of octet 3 of the Extended DRX parameters information element. (e.g."0010" equals 20.48 seconds)
<Paging_time_window >	String type; half a byte in a 4 bit format. The paging time window referes to bit 8 to 5 of octet 3 of the Extended DRX parameters information element. (e.g."0000" equals 2.56 seconds)

Example

```

AT+ECPTWEDRXS=1,5,"0011","0011"
OK

AT+ECPTWEDRXS?
+ECPTWEDRXS: 5,"0011","0011"

OK

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

OK

```

3.1.38 +ECPIN

This is an unsolicited result code used to indicate SIM PIN state. It's controlled by AT+ECURC.

+ECPIN

```
+ECPIN: <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.

Example

Example

3.1.41 AT+ECADC

This command is used to get thermal temperature and VBAT values sampled by ADC.

AT+ECADC

Test Command AT+ECADC=?	Response +ECADC: <option> OK
Set Command AT+ECADC=<option>	Response +ECADC: <option>,<value>[,<option>,<value>] OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<option>	String type
temp	Get current thermal temperature in unit of degree centigrade with 1 degree resolution.
vbat	Get current VBAT value in unit of mV.
all	Get current thermal temperature and VBAT value.
<value>	Integer type; Corresponding value of option

Example

```
AT+ECADC=all
+ECADC: TEMP,26,VBAT,3604
OK

AT+ECADC=temp
+ECADC: TEMP,26
OK

AT+ECADC=vbat
+ECADC: VBAT,3604
OK
```

3.1.42 AT+ECSWC

The command set UE to enable, disable, query and reset SIM write counter. The counter record the

execution of SIM write command, such as update binary and update record

The read command return current setting of <mode>.

AT+ECSWC

Set Command AT+ECSWC=<mode>	Response OK If there is any error, response: +CME ERROR: <err> When <mode>=2 and command successful: +ECSWC: <filename>,<counter> +ECSWC: <filename>,<counter> ... OK
Read Command AT+ECSWC?	Response +ECSWC: <mode> OK
Test Command AT+ECSWC=?	Response +ECSWC: (list of supported <mode>s) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	Integer type
0	Disable the SIM write counter
1	Enable the SIM write counter
2	Query the SIM write counter
3	Reset all SIM write counter as 0
<filename>	String type; SIM EF name, referred to TS31.102, "null" means no file writed.
<counter>	Integer type; if overflow occurs, restart from 0.

3.1.43 AT+ECSNTP

This command is used to synchronize the local time with the Universal Time Coordinated (UTC) via the SNTP server.

The test command returns the supported parameters

The set command sets server name and start to synchronize the local time with the Universal Time Coordinated (UTC). It will return immediately, the UTC content will be returned via URC.

AT+ECSNTP

Set Command	Response
-------------	----------

AT+ECSNTP=<server>[, <port>, <autosync>]	OK +ECSNTP:<time> If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSNTP=?	Response +ECSNTP:"IP ADDR/URL", (0-65535), (0,1) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<server>	String type Address of the NTP server. The format is a domain name or a dotted decimal IP address.
<port>	integer type The NTP server port number, default is 123
<autosync>	integer type Whether to automatically set synchronized time to local UTC. Default is 0. 0 Not set 1 Set
<time>	string type Time synchronized from NTP server The format is: "yy/mm/dd: hh/mm/ss, hour:minute:second"

Example

```
AT+ECSNTP=cn.pool.ntp.org
OK
+ECSNTP: 2020/08/01,08:53:48
```

3.1.44 AT+ECIPR

Set command sets the UE baud rate to be used.

Read command returns the current baud rate.

Test command returns baud rates supported by the UE.

AT+ECIPR

Set Command AT+ECIPR=<rate>	Response OK If there is any error, response:
---------------------------------------	---

	+CME ERROR: <err>
Read Command AT+ECIPR	Response +ECIPR:<rate> OK
Test Command AT+ECIPR=?	Response +ECIPR: (list of fixed-only <rate> values) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<rate>	Baud rate at which the UE will accept commands.
--------	---

Example

```

AT+ECIPR=115200
OK

AT+ECIPR?
+ECIPR:115200
OK

AT+ECIPR=?
+ECIPR: (300,600,1200,2400,4800,9600,19200,38400,57600,115200,230400,460800)

OK

```

3.1.45 AT+ECNPICFG

Set command is used to set NPI config

Read command returns the current NPI config setting.

AT+ECNPICFG

Set Command AT+ECNPICFG=<option>,<setting>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECNPICFG?	Response +ECNPICFG: "rfCaliDone":<value>,"rfNSTDone":<value>

	OK
Test Command AT+ECNPICFG=?	Response +ECNPICFG: <option>:<value>
	OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<option>	String type
	rfCaliDone Set RFCALI process status
	rfNSTDone Set RFNST process status
<setting>	Integer type
	For rfCaliDone , the values range is from 0 to 1 0: RFCALI process is not done 1: RFCALI process is done
	For rfNSTDone, the values range is from 0 to 1 0: RFNST process is not done 1: RFNST process is done

3.1.46 AT+ECPRODMODE

Set command is used to control entering NPI production mode.

Read command returns the current NPI production mode status.

AT+ECPRODMODE

Set Command AT+ECPRODMODE=<setting>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPRODMODE?	Response +ECPRODMODE: "prodModeEnable":<value1>,"prodModeEnter":<value2> OK
Test Command AT+ECPRODMODE=?	Response +ECPRODMODE: <status> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<setting>	String type
	prodModeEnable Enable production mode
	prodModeDisable Disable production mode
	prodModeEnter Enter production mode
	prodModeExit Exit production mode
<value1>	Integer type
	0: prodMode is disabled
	1: prodMode is enabled
<value2>	Integer type
	0: prodMode isn't entered
	1: prodMode is entered
<status>	reflects prodMode status

3.1.46 AT+ECLEDMODE

Set command is used to enable the Netlight function

Read command returns the whether the Netlight function is enable or not.

AT+ECLEDMODE

Set Command AT+ECLEDMODE=<state>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECLEDMODE?	Response +ECLEDMODE: <state> OK
Test Command AT+ECLEDMODE=?	Response +ECLEDMODE: (0,1) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<state>	Integer type
	0 Netlight function disable
	1 Netlight function enable

NOTE:

In EC NB-IoT SDK, Netlight is set to PAD 30 using PWM Instance 5. It can be changed by rewrite

NetLightInit function (in bsp_custom.c).

(1) Netlight function may cause extra power consumption

(2) All sleep is disabled, until NB enter Idle/PSM

(3) A pull-down resistor is needed to keep Pad low to avoid unwanted flicker

Netlight Indication Note	NB Status
HighLevel 64ms(LED ON)/ LowLevel 800ms(LED OFF)	Network searching
HighLevel 64ms(LED ON)/ LowLevel 2000ms(LED OFF)	Network connected in NB Connect state
Keep Low Level	NB Idle/PSM or other

EIGENCOMM CONFIDENTIAL

3.1.47 AT+ECPOWERCLASS

The command sets the band power class.

Read command returns the current band and related power class setting.

Test command returns the bands and related power class supported by the UE.

AT+ECPOWERCLASS

Set Command AT+ECPOWERCLASS=<band>,<power class>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPOWERCLASS?	Response +ECPOWERCLASS: <band1>,<power class> +ECPOWERCLASS: <band2>,<power class> ... OK
Test Command AT+ECPOWERCLASS=?	Response +ECPOWERCLASS: <band1>,(list of supported <power class>s) +ECPOWERCLASS: <band2>,(list of supported <power class>s) ... OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<band>	Integer type Band in decimal number.All bands supported by RF. 0 is to set power class of all bands.
<power class>	Integer type Power class in decimal number. 3 UE maximum output power is 23dBm 5 UE maximum output power is 20dBm 6 UE maximum output power is 14dBm

Note:

AT+ECPOWERCLASS must be restricted to execute under power off or air plane state.

Only when 'RelVersion' in AT+ECCFG is 14, parameter <power class> can be set to 6.

When set parameter <power class> to 6,parameter <band> must be set to 0.

Example

```
AT+ECPOWERCLASS?  
  
+ECPOWERCLASS: 1,3  
+ECPOWERCLASS: 3,3  
+ECPOWERCLASS: 5,3  
+ECPOWERCLASS: 8,3  
  
OK  
AT+ECPOWERCLASS=?  
  
+ECPOWERCLASS: 1,(3,5,6)  
+ECPOWERCLASS: 3,(3,5,6)  
+ECPOWERCLASS: 5,(3,5,6)  
+ECPOWERCLASS: 8,(3,5,6)  
  
OK  
AT+ECPOWERCLASS=1,5  
  
OK
```

3.1.48 AT+ECPMUSTATUS

The command obtains UE last PMU status.

AT+ECPMUSTATUS

Read Command AT+ECPMUSTATUS?	Response +ECPMUSTATUS: <slpstate>, <slptime> OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<slpstate>	String type
	Last PMU state UE entered:
Actv	Active state
Idle	Idle state
Slp1	Sleep1 state
Slp2	Sleep2 state
Hibn	Hibernate state

<slptime>

Integer type

Last sleep duration in resolution of 125ms, will be 0 if slpstate is Actv and Idle.

Example

```
AT+ECPMUSTATUS?
```

```
+ECPMUSTATUS: Actv, 0
```

```
OK
```

```
AT+ECPMUSTATUS?
```

```
+ECPMUSTATUS: Hibern, 3625
```

```
OK
```

3.1.49 AT+ECSTATIS

It is an action command. The execution command causes the MT to start/stop protocol statistic information reporting.

The test command returns a list of supported <interval> setting.

AT+ECSTATIS

Execution Command AT+ECSTATIS=<interval>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSTATIS=?	Response +ECSTATIS: list of supported <interval>s OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<interval>

Integer type

0

Stop statistic information report

(5-600)

UE report statistic information in this period, value is in unit of seconds.

Note 1:

When statistics reporting is turned on, the UE will disable to enter deep sleep state (Sleep2 & Hibernate)

Note 2:

[Statistical parameters description:](#)

Layer	Parameter	Description	Unit	Range
PHY DL	AvgRSRP	Average RSRP	dBm	
	AvgSnr	Average SNR	db	(-30,30)
	DIBler	DL block error rate	%	
	PhyDITpt	Physical layer DL average through put (contains HARQ failure data)	bps	
	AvgTBS	Average DL transport block size	byte	(0-317)
	Avgltbs	Average DL ITBS, index of table: 16.4.1.5.1-1 in 3GPP 36.213		(0-13)
	AvgNRep	Average DL repetition number		
	AvgSbfrmNum	Average sub-frame numbers per DL reception	ms	
	Harq2Ratio	HARQ2 occupancy ratio. HARQ ID 1 number/(HARQ ID 0 number + HARQ ID 1 number)	%	
PHY UL	UIBler	UL block error rate	%	
	PhyUITpt	Physical layer UL average through put (contains HARQ failure data)	bps	
	AvgTBS	Average UL transport block size	byte	(0-317)
	Avgltbs	Average UL ITBS, index of table: 16.5.1.2-2 in 3GPP 36.213		(0-13)
	AvgNRep	Average UL repetition number		
	AvgSbfrmNum	Average sub-frame numbers per UL transmission	ms	
	Harq2Ratio	HARQ2 occupancy ratio. HARQ ID 1 number/(HARQ ID 0 number + HARQ ID 1 number)	%	
	AvgScNum	Average UL sub-carrier number		(1-15)
MAC	MacUIBytes	Total amount of MAC layer data transmitted during the interval	byte	
	MacUIPadBytes	Total amount of MAC layer padding data transmitted during the interval	byte	
	MacDIBytes	Total amount of MAC layer data received during the interval	byte	
	MacDIPadBytes	Total amount of MAC layer padding data received during the interval	byte	
	MacUITpt	MAC layer UL average through put	bps	
	MacDITpt	MAC layer DL average through put	bps	
RLC	RlcUIPduBytes	Total amount of new RLC PDU data transmitted during the interval	byte	
	RlcUIRetxBytes	Total amount of re-transmitted RLC PDU data transmitted during the interval	byte	
	RlcDIPduBytes	Total amount of RLC PDU data received during the interval	byte	

	RlcUITpt	RLC layer UL average through put (not contained the re-transmission data)	bps	
	RlcDITpt	RLC layer DL average through put	bps	
PDCP	PdcpUIPduBytes	Total amount of PDCP layer data transmitted during the interval Note: here just means the data arrived in UE RLC layer, not means transmitted to network.	byte	
	PdcpDIPduBytes	Total amount of PDCP layer data received during the interval	byte	
	PdcpULDiscardBytes	Total amount of PDCP discarded data transmitted during the interval	byte	
	PdcpUITpt	PDCP layer UL average through put	bps	
	PdcpDITpt	PDCP layer DL average through put	bps	

Example

```
//start statistic information report
AT+ECSTATIS=10

OK

//UE report following URC in period of 10 seconds

+ECSTATIS: PHY DL, AvgRSRP: -75, AvgSnr: 9, DlBler: 0%, PhyDlTpt: 57 bps, AvgTBS: 9,
AvgItbs: 5, AvgNRep: 1, AvgSbfrmNum: 1, Harq2Ratio: 0%

+ECSTATIS: PHY UL, UlBler: 2%, PhyUlTpt: 48200 bps, AvgTBS: 125, AvgItbs: 11, AvgNRep: 1,
AvgSbfrmNum: 4, Harq2Ratio: 0%, AvgScNum: 11

+ECSTATIS: MAC, MacUlBytes:29250, MacUlPadBytes:0, MacDlBytes: 36, MacDlPadBytes: 20,
MacUlTpt: 46800 bps, MacDlTpt: 57 bps

+ECSTATIS: RLC, RlcUlPduBytes:29006, RlcUlRetxBytes:0, RlcDlPduBytes: 8, RlcUlTpt: 46409
bps, RlcDlTpt: 12 bps

+ECSTATIS: PDCP, PdcpUlPduBytes: 31424, PdcpDlPduBytes: 0, PdcpULDiscardBytes: 0,
PdcpUlTpt: 50278 bps, PdcpDlTpt: 0 bps

//stop
AT+ECSTATIS=0

OK

//test
```

```
AT+ECSTATIS=?
+ECSTATIS: (0,5-600)

OK
```

3.1.50 AT+ECFSINFO

The command lists file size and name, file system information including total and used space in unit of block.

AT+ ECFSINFO

Execution Command	Response
AT+ECFSINFO	<file system information> OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Example

```
AT+ECFSINFO

size      file name
58B       ccmconfig.nvm
302B      cemmconfigemminformation.nvm
72B       cemmconfigueinformation.nvm
205B      cemmesmconfig.nvm
240B      cemmplmnconfig.nvm
67B       cerrcconfig.nvm
936B      coap_nvnm
372B      dcxoFTBuff.nvm
452B      fileOpStatistic
296B      middlewareconfig.nvm
12B       npiconfig.nvm
21B       plat_config
2016B     slp_nvnm
232B      uiccctrlconfig.nvm
FS block: Totoal : 84  used: 10

OK
```

3.1.51 AT+ECFLASHMONITORINFO

The command is used to monitor flash information.

AT+ECFLASHMONITORINFO

Test Command AT+ECFLASHMONITORINFO=?	Response +ECFLASHMONITORINFO: (0,1,2,3) OK
Set Command AT+ECFLASHMONITORINFO=<option>	Response <monitor infomation> if option is 1,2,3 OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<option>	Integer type
	0 reset file system monitor counter
	1 show file write counter
	2 show file system block erase counter
	3 show pmu back up raw flash erase counter
	Note, macro FLASH_MONITOR_ENABLE shall be enabled for option 1 and 2

Example

```
AT+ECFLASHMONITORINFO=1
writeCount,writeBytesCount,file name
0,0B,ccmconfig.nvm
0,0B,cemmnconfigemminformation.nvm
0,0B,cemmnconfigueinformation.nvm
0,0B,cemmesmconfig.nvm
246,59040B,cemmplmnconfig.nvm
0,0B,cerrcconfig.nvm
0,0B,cisconfig.nvm
0,0B,coap_nvmm
0,0B,dcxoFTBuff.nvm
29972,4435856B,midwareconfig.nvm
2,8B,npiconfig.nvm
4,34B,plat_config
0,0B,slp_nvmm
5039,1169048B,uiccctrlconfig.nvm
OK

AT+ECFLASHMONITORINFO=2
37,35,359,364,339,325,330,221,240,259
```

```
265,246,238,230,223,230,239,234,365,252
360,252,378,261,271,276,262,267,261,244
243,239,242,243,239,231,237,278,359,253
266,283,326,245,334,234,261,239,0,279
280,398,299,385,274,242,367,235,227,237
357,260,260,276,277,247,240,238,234,0
0,359,0,409,404,400,312,278,256,248
239,226,231,246
OK
```

```
AT+ECFLASHMONITORINFO=3
PMU Flash Erase Count:
PlatPS Region1: 15 Phy Region1: 10
PlatPS Region2: 15 Phy Region2: 10
OK
```

```
AT+ECFLASHMONITORINFO=0
OK
```

```
AT+ECFLASHMONITORINFO=1
writeCount,writeBytesCount,file name
0,0B,ccmconfig.nvm
0,0B,cemmconfigeminformation.nvm
0,0B,cemmconfigueinformation.nvm
0,0B,cemmesmconfig.nvm
0,0B,cemmplmnconfig.nvm
0,0B,cerrcconfig.nvm
0,0B,cisconfig.nvm
0,0B,coap_nvm
0,0B,dcxoFTBuff.nvm
0,0B,middlewareconfig.nvm
0,0B,npiconfig.nvm
0,0B,plat_config
0,0B,slp_nvm
0,0B,uiccctrlconfig.nvm
OK
```

3.1.52 AT+ECPURC

The command enable/disable platform URC (unsolicited result code) report.

AT+ ECPURC

Set Command	Response
-------------	----------

AT+ECPURC=<urcStr>,<value>	OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPURC?	Response +ECPURC: "HIBNATE":<value> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECPURC=?	Response +ECPURC: "HIBNATE": (0-1) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE <i>Note: Set of <value> will save to NVM, and the default value is 0.</i>

Parameter

<urcStr>	String type;
	"HIBNATE" unsolicited result code +HIB
<value>	Integer type;
	0 disable unsolicited result code report
	1 enable unsolicited result code report

Example

```

AT+ECPURC="HIBNATE",1      //enable hibernate URC
OK

AT+ECPURC?
+ECPURC: "HIBNATE":1
OK

+HIB Enter
+HIB Exit

AT+ECPURC =?
+ECPURC: "HIBNATE": (0-1)
OK

```

3.1.53 AT+ECFSFORMAT

The execution command will erase file system region and platform configuration region

AT+ECFSFORMAT

Execution Command AT+ECFSFORMAT	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Note : after execute this command, user need to trigger a UE reset, then UE will write default parameter into the file system.

3.1.54 AT+ECSIMPD

The command set UE to enable or disable SIM presence detection when UE is not on deep sleep.

The read command return current setting of each parameters.

The test command returns values supported as a compound value.

AT+ECSIMPD

Set Command AT+ECSIMPD=<mode>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECSIMPD?	Response +ECSIMPD: <mode> OK
Test Command AT+ECSIMPD=?	Response +ECSIMPD: (list of supported <mode>s) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	Integer type
0	Disabled SIM presence detection
1	Enabled SIM presence detection

3.1.55 AT+ECABFOTACTL

The command set UE to enable or disable abup ota.

AT+ECABFOTACTL

Set Command AT+ECABFOTACTL=<mode>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECABFOTACTL?	Response +ECABFOTACTL: <mode> OK
Test Command AT+ECABFOTACTL=?	Response +ECABFOTACTL: (enable,disable) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	String type
enable	Enabled abup ota
disable	Disabled abup ota

3.1.56 AT+ECNBIOTRAI

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 will send 1 UL packet and no DL packet is expected, then the NB-IOT network will quickly release the current RRC connect.

AT+ECNBIOTRAI

Set Command AT+ECNBIOTRAI=<rai>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECNBIOTRAI=?	Response +ECNBIOTRAI: (0-1) OK

Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

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

Example

```
AT+ECNBIOTRAI=1
```

```
OK
```

```
AT+ECNBIOTRAI=?
```

```
+ECNBIOTRAI: (0-1)
```

```
OK
```

3.1.57 AT+ECNBR14

The execution command use to read/get UE and current network NB R14 feature capability.

AT+ECNBR14

Execution Command AT+ECNBR14	Response +ECNBR14: UeRelVersion,<UeRelVersion>,UeR14UpRai,<UeR14UpRai> <\r><\n> +ECNBR14: TwoHarq,<TwoHarq>,R14UpRai,<R14UpRai>, NonAnchorNPRACH,<NonAnchorNPRACH>, NonAnchorPaging,<NonAnchorPaging>, CpReest,<CpReest> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECNBR14=?	Response OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<UeRelVersion>	Integer type, UE release version, UE capability which could be set by AT:
----------------	---

	AT+ECCFG="RelVersion",13/14
	13 NB release version 13
	14 NB release version 14
<UeR14UpRai>	Integer type, UE R14 RAI capability, which could be set by AT: AT+ECCFG="SupportUpRai",0/1
	0 UE not support R14 RAI
	1 UE support R14 RAI
<TwoHarq>	Integer type, whether network enabled Two-HARQ feature in current RRC connection.
	0 Two-HARQ is not enabled
	1 Two-HARQ is enabled
<R14UpRai>	Integer type, whether network enabled R14 RAI in current RRC connection
	0 R14 RAI is not enabled
	1 R14 RAI is enabled
<NonAnchorNPRACH>	Integer type, whether NPRACH resource on non-anchor carrier is configured in current cell.
	0 NPRACH resource on non-anchor carrier is not configured
	1 NPRACH resource on non-anchor carrier is configured
<NonAnchorPaging>	Integer type, whether paging resource on non-anchor carrier is configured in current cell
	0 Paging resource on non-anchor carrier is not configured
	1 Paging resource on non-anchor carrier is configured
<CpReest>	Integer type, whether CP reestablishment procedure is supported in current cell.
	0 Not support CP reestablishment
	1 Support CP reestablishment

Note:

- 1> <TwoHarq> & <R14UpRai>, these two capabilities can only be set to TRUE (1) in RRC connected state.
- 2> <NonAnchorNPRACH> & <NonAnchorPaging>, these two cell/network capabilities are gotten/read from "SystemInformationBlockType22-NB", and this system information is not acquired when UE is set to R13, that is, if the UE is set to R13, these two capabilities are always return 0.

Example

```
AT+ECNBR14
+ECNBR14: UeRelVersion,14,UeR14UpRai,1
+ECNBR14: TwoHarq,1,R14UpRai,1,NonAnchorNPRACH,0,NonAnchorPaging,0,CpReest,0
```

OK

3.1.58 AT+ECEVENTSTATIS

The execution command use to set/get UE EMM and RRC specific event statistics.

AT+ECEVENTSTATIS

Execution Command AT+ECEVENTSTATIS =<mode>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECEVENTSTATIS=?	Response +ECEVENTSTATIS: list of supported <mode> OK
Read Command AT+ECEVENTSTATIS?	Response +ECEVENTSTATIS: <mode> +ECEVENTSTATIS: RRC, ConEstSucc:<num>, ConEstFail:<num> +ECEVENTSTATIS: EMM, AttachSucc:<num>, AttachFail:<num>, TAUSucc:<num>, TAUFail:<num>, SRSucc:<num>, SRFail:<num>, AuthFail:<num>, DetachNum:<num> +ECEVENTSTATIS: PLMN, OOS:<num> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	Integer type
0	Stop event statistic – default value
1	Start event statistic
2	Clear event statistic (Note: the mode should be executed in “UE Power Off state (AT+CFUN=0)”)

Note 1:

[Statistical parameters description:](#)

Layer	Parameter	Description	Range
EMM	AttachSucc	Total amount of Attach Success Procedure	0-65535
	AttachFail	Total amount of Attach Failure Procedure	0-65535

	TAUSucc	Total amount of Tracking Area Update Success Procedure	0-65535
	TAUFail	Total amount of Tracking Area Update Failure Procedure	0-65535
	SRSucc	Total amount of Service Request Success Procedure	0-65535
	SRFail	Total amount of Service Request Failure Procedure	0-65535
	AuthFail	Total amount of Authentication Reject Procedure	0-65535
	DetachNum	Total amount of MT-Detach and MO-Detach Procedure	0-65535
	OOS	Total amount of PLMN Out-of-service	0-65535
RRC	ConEstSucc	Total amount of RRC Connection Success Procedure	0-65535
	ConEstFail	Total amount of RRC Connection Failure Procedure	0-65535

Example

```

//start statistic mode
AT+ECEVENTSTATIS=1
OK

//Inquire EMM and RRC event statistics
AT+ECEVENTSTATIS?
+ECEVENTSTATIS: 1
+ECEVENTSTATIS: RRC, ConEstSucc:2, ConEstFail:0
+ECEVENTSTATIS: EMM, AttachSucc:1, AttachFail:0, TAUSucc:0, TAUFail:0, SRSucc:0, SRFail:0,
AuthFail:0, DetachNum:1
+ECEVENTSTATIS: PLMN, OOS:1

//Clear EMM and RRC event statistics
AT+CFUN=0
OK
AT+ECEVENTSTATIS=2
OK

//stop
AT+ECEVENTSTATIS=0
OK

//test
AT+ECEVENTSTATIS=?

```

+ECEVENTSTATIS: (0,1,2)

OK

3.1.59 AT+NFWUPD

The execution command will erase FOTA download region in internal flash, download the delta file, validate the delta file, query the name & the version of delta file, upgrade the delta file and inform the receiver the download of delta file has been completed.

AT+NFWUPD

Execution Command AT+NFWUPD=<cmd> [, <sn>, <nbytes>, <data>, <xor8sum>]	Response OK If there is any error, response: ERROR OR +CME ERROR: <err>
Test Command AT+NFWUPD=?	Response +NFWUPD: list of supported <cmd> OK If there is any error, response: ERROR OR +CME ERROR: <err>
Maximum Response Time	10s
Parameter Saving Mode	NO SAVE

Parameter

<cmd>	Integer type. 0 Erase the FOTA download region in internal FLASH. 1 Download the segments of delta file, which should be download in orders. And the parameters(<sn>, <nbytes>, <data>, <xor8sum>) is valid only on this condition. 2 Validate the delta file. 3 Query the name of delta file. 4 Query the version of delta file. 5 Upgrade the delta file. 6 Inform the receiver the download of delta file has been completed.
<sn>	Integer type. Segment No. of the delta file, which is started with 0 and increased by 1 in orders afterwards. Range: 0-65535.
<nbytes>	Integer type. Total bytes of delta file segment, which should be is multiple of 4

	bytes. Range: 4-256。
<data>	String type. The segment data in hex string format, the length of which is 2*<nbytes>.
<xor8sum>	String type. The result of XOR8 checksum in hex string format.

Example

```
AT+NFWUPD=1,0,200,75fabe4dd6d334916877e68a04c9fa89871afd5a965dca42a561afdb549
8ff186c8996d4ba31ee7d2e3e7aee35980ce80f269616e2dfe03ff75f53bf34776f73756e3037
ff2000d49b1d001b0000000046425a683131415926535920e20d98000017f6dffe90020a000c4
00004000100040000410080008114808000c80fa0005454d304c04308c11e81aa3c90c869a036
a6d468d076dd0655f348075e4cae4241b9848c67db91280dea539c3c5b143680fd1fe3faa7408
00322e0ba17724538509020e20d980000,1b
```

OK

```
AT+NFWUPD=?
+NFWUPD: (0-6)
```

OK

3.1.60 AT+ECUSATP

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

The read command return current setting of each parameters.

The test command returns values supported as a compound value.

AT+ECUSATP

Set Command AT+ECUSATP=<length>,<profile>	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECUSATP?	Response +ECUSATP: <length>,<profile> OK
Test Command AT+ECUSATP=?	Response OK
Maximum Response Time	5s
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 character format)

Example

3.1.61 AT+ECPSSLPCFG

The command set PS sleep2/HIB configuration.

The read command return current setting of each parameters.

The test command returns values supported as a compound value.

AT+ECPSSLPCFG

Set Command AT+ECPSSLPCFG=<minT3324>,<T1>[,<minT3412>,<T2>[,< minEDRXPeriod>,<T3>]]	Response OK If there is any error, response: +CME ERROR: <err>
Read Command AT+ECPSSLPCFG?	Response +ECPSSLPCFG: <minT3324>,<T1>,<minT3412>,<T2>,<minEDRXPeriod>,<T3> OK
Test Command AT+ECPSSLPCFG=?	Response +ECPSSLPCFG: (list of supported <param>s) OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE_REBOOT

Parameter

<param>	String type, name of configuration parameter.
"minT3324"	T3324 value in seconds. Note: a) Supported values: (0-65535)

		b) Default value: 0.
"minT3412"	T3412 value in seconds.	Note: a) Supported values: (0-4294967295) b) Default value: 0
"minEDRXPeriod"	Edrx value in seconds. When set to 65535,UE not allow enter sleep2/HIB in IDLE state	Note: a) Supported values: (0-65535) b) Default value: 0
<value>	Integer type	
	value of configuration	

NOTE:

1>"minT3324",<T1>

If NW assigned T3324<=T1,don't allow enter Sleep2/HIB when T3324 is running.

If NW not assign 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 not assign 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.

summary:

a)Condition allow enter Sleep2/HIB when T3324 is running : T3324>T1 && T3412>T2 && DRX/eDRX period>T3

b)Condition allow enter Sleep2/HIB when NW not assign T3324(PSM is disable) : DRX/eDRX period>T3

Example

```
AT+ECPSSLPCFG="minT3324",0,"minT3412",0,"minEDRXPeriod",0
```

OK

```
AT+ECPSSLPCFG?
```

```
+ECPSSLPCFG:"minT3324",0,"minT3412",0,"minEDRXPeriod",0
```

OK

```
AT+ECPSSLPCFG=?
```

```
+ECPSSLPCFG:(0-65535),(0-4294967295),(0-65535)
```

OK

3.2 Socket Commands (solution A)

3.2.1 AT+SKTCREATE

The command creates a socket on the UE and associates with specified protocol. UE supports up to 7 sockets(TCP or UDP) at the same time. And will return error if it is exceeded.

The test command returns values supported as a compound value.

AT+SKTCREATE

Set Command AT+SKTCREATE=<domain>,<type>,<protocol>	Response +SKTCREATE: <fd> OK If there is any error, response: +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	5s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type 0-9 Socket file description
<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=1,2,17  
+SKTCREATE: 1
```


OK

3.2.2 AT+SKTCONNECT

For TCP, the command connect socket with remote address and port.

For UDP, the command save remote address and port for send

AT+SKTCONNECT

Set Command AT+SKTCONNECT=<fd>, <addr>, <port>	Response OK If there is any error, response: +SOCKET ERROR: <err>
Test Command AT+SKTCONNECT=?	Response +SKTCONNECT: (list of supported <fd>s), (<addr>), (list of supported <port>s) OK
Maximum Response Time	35s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type 0-9 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

Example

3.2.3 AT+SKTBIND

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

AT+SKTBIND

Set Command AT+SKTBIND=<fd>, <addr>, <port>	Response OK If there is any error, response: +SOCKET ERROR: <err>
Test Command AT+SKTBIND=?	Response +SKTBIND: (list of supported <fd>s), (<addr>),

	(list of supported <port>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type 0-9 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

Example

3.2.4 AT+SKTSEND

Send a containing length bytes of data to remote port on remote addr.

AT+SKTSEND

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

Parameter

<fd>	Integer type 0-9 Socket file description returned by +SKTCREATE
<data len>	Integer type length of data in hex string format, the max length is 1400
<data>	Integer type Data in hex string
<rai info>	Integer type (option) 0-2 release assistance indication

	0	no rai info
	1	no further uplink or downlink data transmission subsequent to the uplink data transmission subsequent to the uplink data transmission is expected
	2	only a single downlink data transmission and no further uplink data transmission subsequent to the uplink data transmission is expected
	Note: default value: 0	
<except info>	Integer type (option)	
	0-1	expect data indication
	0	disable expect data indication
	1	enable expect data indication
	Note: default value: 0	

Example

Send data:23456

```
AT+SKTSEND=0,5,3233343536
```

```
OK
```

3.2.5 AT+SKTSENDT

Send a containing length bytes of data to remote port on remote address with transparent mode.

If the command contain the parameter of “data len”, It need input enough data with hex string format after “>” output. If the command did not contain the parameter of “data len”, the input of data must end with “CTRL+Z”.

AT+SKTSEND

Set Command AT+SKTSENDT=<fd>[,<data len>[,<rai info>[,<except info>]]]	Response OK If there is any error, response: +SOCKET ERROR: <err>
Test Command AT+SKTSENDT=?	Response +SKTSENDT: (list of supported <fd>s), (list of supported <data len>s), (list of supported <rai info>s), (list of supported <except info>s) OK
Maximum Response Time	60s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type
	0-9 Socket file description returned by +SKTCREATE
<data len>	Integer type
	length of data in hex string format, the max length is 1400
<data>	Integer type

	Data in hex string
<rai info>	Integer type (option)
	0-2 release assistance indication
	3 no rai info
	4 no further uplink or downlink data transmission subsequent to the uplink data transmission subsequent to the uplink data transmission is expected
	5 only a single downlink data transmission and no further uplink data transmission subsequent to the uplink data transmission is expected
	Note: default value: 0
<except info>	Integer type (option)
	0-2 expect data indication
	2 disable expect data indication
	3 enable expect data indication
	Note: default value: 0

Example

Send data:23456

```
AT+SKTSENDT=0,5
>3233343536
OK
```

```
AT+SKTSENDT=0
>3233343536
1A
OK
```

3.2.6 +SKTREC V

This is an unsolicited message to show data has been received on a socket.

+SKTREC V

```
+SKTREC V: <fd>, <len>, <data>
```

Parameter

<fd>	Integer type
	0-9 Socket file description that data from
<len>	String type
	Received data length(bytes)
<data>	String type
	Received data in hex string format

Example

3.2.7 +SKTERR

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

+SKTERR

+SKTERR: <fd>,<errno>

Parameter

<fd>	Integer type
	0-9 Socket file description that data from
<errno>	Integer type (Posix Errno defines)
	12 Out of memory error
	105 No buffer space available
	62 Timer expired
	113 No route to host
	115 Operation now in progress
	22 Invalid argument
	11 Operation would block
	107 Transport endpoint is not connected
	103 Software caused connection abort
104 Connection reset by peer	

Note: if there is some error with the socket which is connected, the socket will be closed automatic

Example

3.2.8 AT+SKTSTATUS

Get the status of a socket by file description.

AT+SKTSTATUS

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

Parameter

<fd>	Integer type
	0-9 Socket file description returned by +SKTCREATE
<status>	Integer type
	1 Not connected
	2 Connecting
	3 Connected

Example

3.2.9 AT+SKTDELETE

Delete a socket by file description.

AT+SKTDELETE

Set Command AT+SKTDELETE=<fd>	Response OK If there is any error, response: +SOCKET ERROR: <err>
Test Command AT+SKTDELETE=?	Response +SKTDELETE: (list of supported <fd>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<fd>	Integer type
	0-9 Socket file description returned by +SKTCREATE

Example

3.3 LwM2M Commands

3.3.1 AT+LWM2MCREATE

This command creates an instance of lwM2M client and register with lwM2M server. It need specify sever, port, ender point name, lifetime. Also can set psk id and psk if need DTLS.

AT+LWM2MCREATE

Set Command AT+LWM2MCREATE=<server>,<port>,<local port>,<endpoint>,<lifetime>[,<psk id>,<psk>]	Response +LWM2MCREATE: <clientId> OK If there is any error, response: +LWM2M ERROR: <err>
Test Command AT+LWM2MCREATE=?	Response +LWM2MCREATE: "<server>", (range of supported<port>), (range of supported<local port>), "<endpoint>", (range of supported<lifetime>), "<psk_id>", "<psk>" OK
Maximum Response Time	65s
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
<endpoint>	String type LwM2M client's endpoint name
<life time>	Integer type LwM2M client's life time
<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
```

3.3.2 AT+LWM2MDELETE

This command deletes a specified lwm2m client instance.

AT+LWM2MDELETE

Set Command AT+LWM2MDELETE=<clientId>	Response OK If there is any error, response: +LWM2M ERROR: <err>
Test Command AT+LWM2MDELETE=?	Response +LWM2MDELETE: (list of supported<clientId>) OK
Maximum Response Time	65s
Parameter Saving Mode	NO_SAVE

Parameter

<clientId>	Integer type Lwm2M client's id returned by +LWM2MCREATE
------------	---

Example

3.3.3 AT+LWM2MADDOBJ

This command adds a lwm2m object to a specified lwm2m client instance. Definitions of object, instance and resource. please refer to Lightweight Machine to Machine Technical Specification, ext-label Objects Produced by IPSO Alliance and oma-label Objects Produced by OMA.

<http://www.openmobilealliance.org/wp/OMNA/Lwm2M/Lwm2MRegistry.html>

AT+LWM2MADDOBJ

Set Command AT+LWM2MADDOBJ=<clientId>,<objectId>,<instanceId>,<resourceCount>,<resourceIds>	Response OK If there is any error, response: +LWM2M ERROR: <err>
Test Command AT+LWM2MADDOBJ=?	Response +LWM2MADDOBJ: (list of supported<clientId>), (range of supported<objectId>), (range of supported<instanceId>), (range of supported<resourceCount>), "<resourceIds>" OK
Maximum Response Time	65s
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>	String type Resources numbers separated by semicolons

Example

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

3.3.4 AT+LWM2MDELOBJ

This command deletes an object from a specified lwm2m client instance.

AT+LWM2MDELOBJ

Set Command AT+LWM2MDELOBJ=<clientId>,<objectId>	Response OK If there is any error, response: +LWM2M ERROR:<err>
Test Command AT+LWM2MDELOBJ=?	Response +LWM2MDELOBJ: (list of supported<clientId>), (range of supported<objectId>) OK
Maximum Response Time	65s
Parameter Saving Mode	NO_SAVE

Parameter

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

Example

3.3.5 +LWM2MREAD indicator of LwM2M server's reading command

This is an unsolicited message to represent lwm2m client receive the LwM2M server's reading command.

+LWM2MREAD

```
+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 want to read
<instanceId>	Integer type Instance id number that lwm2m server want to read
<resId>	Integer type Resource id number that lwm2m server want to read

Example

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

3.3.6 +LWM2MWRITE indicator of LwM2M server's writing command

This is an unsolicited message to represent lwm2m client receive the LwM2M server's writing command.

+LWM2MWRITE

```
+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 want to write
<instanceId>	Integer type Instance id number that lwm2m server want to write
<num>	Integer type The number of resources need to be write

<resId>	Integer type Resource id number that lwm2m server want 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 need to write to resource

Example

```
+LWM2MWRITE: 0,3306,111,5750,0,4,"54595045"
```

3.3.7 +LWM2MEXECUTE indicator of Lwm2m server's execution command

This is an unsolicited message to represent lwm2m client receive the Lwm2m server's execution command.

+LWM2MEXECUTE

```
+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 want to execute
<instanceId>	Integer type Instance id number that lwm2m server want to execute
<resId>	Integer type Resource id number that lwm2m server want to execute
<length>	Integer type value length
<valueStr>	String type value of execute command

Example

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

3.3.8 +LWM2MOBSERVE indicator of LwM2M server's observation command

This is an unsolicited message to represent lwm2m client receive the LwM2M server's observation command.

+LWM2MOBSERVE

```
+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 want to observe
<instanceId>	Integer type
	Instance id number that lwm2m server want to observe
<resId>	Integer type
	Resource id number that lwm2m server want 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
```

3.3.9 AT+LWM2MREADCONF

This command response lwm2m server's read command

AT+LWM2MREADCONF

Set Command	Response
AT+LWM2MREADCONF=<clientId>,<objectId>,<instanceId>,<resId>	OK

<instanceId>, <resId>, <valuetype>, <valuelen>, <value>	If there is any error, response: +LWM2M ERROR: <err>
Test Command AT+LWM2MREADCONF=?	Response +LWM2MREADCONF: (list of supported<clientId>), (range of supported<objectId>), (range of supported<instanceId>), (range of supported<resId>), (range of supported<valuetype>), (range of supported<valuelen>), "<value>" OK
Maximum Response Time	5s
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 value undefined
<valuelen>	Integer type Value length
<value>	String type Value, if type is opaque, value in hex string format

Example

Value type is string:

```
AT+LWM2MREADCONF=0,3306,0,5750,0,5,"hello"
```

Value type is opaque:

```
AT+LWM2MREADCONF=0,12001,0,4,1,5,"3432383330"
```

Value type is Integer:

```
AT+LWM2MREADCONF=0,3306,0,5851,2,3,"206"
```

Value type is float:

```
AT+LWM2MREADCONF=0,3303,0,5601,3,4,"3.14"
```

Value type is bool:

```
AT+LWM2MREADCONF=0,3306,0,5850,4,1,"1"
```

```
OK
```

3.3.10 AT+LWM2MWRITECONF

This command response LWM2M server's write command

AT+LWM2MWRITECONF

Set Command AT+LWM2MWRITECONF=<clientId>,<result>	Response OK If there is any error, response: +LWM2M ERROR: <err>
Test Command AT+LWM2MWRITECONF=?	Response +LWM2MWRITECONF: (list of supported<clientId>), (range of supported<result>) OK
Maximum Response Time	5s
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
```

3.3.11 AT+LWM2MEXECUTECONF

This command response lwM2M server's execute command

AT+LWM2MEXECUTECONF

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

Parameter

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

Example

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

3.3.12 AT+LWM2MNOTIFY

Notify lwM2M server a specified resource changed

AT+LWM2MNOTIFY

Set Command AT+LWM2MNOTIFY=<clientId>,<objectId>,<instanceId>,<resourceId>,<valuetype>,<valuelen>,<value>	Response OK If there is any error, response: +LWM2M ERROR: <err>
Test Command AT+LWM2MNOTIFY=?	Response +LWM2MNOTIFY: (list of supported<clientId>), (range of

	supported<objectId>), (range of supported<instanceId>), (range of supported<resourceId>), (range of supported<valuetype>), (range of supported<valuelen>), "<value>" OK
Maximum Response Time	10s
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
<resource>	Integer type Resource id number
<valuetype>	Integer type 0 string 1 opaque 2 Integer 3 float 4 bool Other value undefined
<valuelen>	Integer type Value length
<value>	String type Value, if type is opaque, value in hex string format

Example

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

3.3.13 AT+LWM2MUPDATE

This command updates the register information, with or not with object id's update

AT+LWM2MNOTIFY

Set Command	Response
AT+LWM2MUPDATE=<clientId>[, <withobj>]	OK

	If there is any error, response: +LWM2M ERROR: <err>
Test Command AT+LWM2MUPDATE=?	Response +LWM2MUPDATE: (list of supported<clientId>), (list of supported<withobj>) OK
Maximum Response Time	65s
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
```

3.3.14 Summary of <err> Codes

<err> Codes	Description
ERROR	Input wrong AT command, for example: contains chinese characters
PARAMETER ERROR	Input wrong parameter, for example: parameter out of range
CANNOT CREATE SEMPH	Can't create semaphore
CONFIG ERROR	Configuration lwm2m client error
NO FREE CLIENT	No free client left, now only support one client at one time
OPERATION NO SUPPORT	Not support operation, such as GET command
NO FIND CLIENT	Not find this client
ADD OBJECT FAILED	Fail to add object
NO FIND OBJECT	Not find this object id
DELETE OBJECT FAILED	Fail to delete the object
NETWORK NOT READY	Network not ready, can't use data service
INTERNAL ERROR	Internal error
REGISTER FAILED, BAD REQUEST	Bad request(register)
SERVER REGISTER REJECT	Reject by server when register
REGISTER FAILED, BAD	Bad parameter(register)

PARAMETER	
REGISTER TIMEOUT	Register timeout
SESSION INVALID	Session invalid
OBJECT ALREADY ADD	Object already add

3.4 CoAP Commands

3.4.1 AT+COAPCREATE

The command creates a CoAP client.

AT+COAPCREATE

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

Parameter

<localport> Integer type; client's port, values of 1-65535 are supported

3.4.2 AT+COAPDEL

The command deletes the CoAP client

AT+COAPDEL

Execution Command AT+COAPDEL	Response OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

3.4.3 AT+COAPADDRES

The command adds the CoAP resource

AT+COAPADDRES

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

Parameter

<length>	Integer type The CoAP client resources, range: 1-50
<resource>	String type The resource name

Note: this command is not supported now.

3.4.4 AT+COAPHEAD

The command adds the CoAP head

AT+COAPHEAD

Set Command AT+COAPHEAD=<mode>[, [<msgid>] [, <tkl>,<token>]]	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+COAPHEAD=?	Response +COAPHEAD: <mode>[, [<msgid>] [, <tkl>,<token>]] OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	10s
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. Range value: 0-65535
<tkl>	Integer type The token values length, only needed configure when the <mode> value is 2, 5. Range value: 1-8.
<token>	String type The token values, hexadecimal format string, only need configure when the <mode> value is 2, 5.

3.4.5 AT+COAPOPTION

The command adds the CoAP option

AT+COAPOPTION

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

Parameter

<opt cnt>	Integer type The option parameter count, range value: 1-12
<opt name>	String type, The option name, refer the RFC 7252 1 If-Match 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 value>	String type, The length of value string: 1-180. If the <opt_name> is 12 or 17, the <opt_value> should be the below value	
	"0", text-plain	
	"40", application/link-format	
	"41", application/xml	
	"42", application/octet-stream	
	"47", application/exi	
	"50", application/json	

3.4.6 AT+COAPSEND

The command send data to CoAP server.

AT+COAPSEND

Set Command AT+COAPSEND= <msgType>, <method>, <ipAddr>, <port>[, <length>, <data>]	Response OK If there is any error, response: +CME ERROR: <err>
Set Command AT+COAPSEND= <msgType>, <method/resrcode>, <ipAddr>, <port> Note: After ">" is responded, input the data to be sent. Tap "CTRL+Z" to send, and tap "ESC" to cancel the operation	Response OK If there is any error, response: +CME ERROR: <err>
Test Command	Response

AT+COAPSEND=?	+COAPSEND: <msgType>, <method/rsrcode> , <ipAddr>, <port> [, <length>, <data>] OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	10s
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
<rescode>	Integer type
	0 empty message
	201 2.01, created
	202 2.02, deleted
	203 2.03, valid
	204 2.04, changed
	205 2.05, content
	400 4.00, bad request
	401 4.01, unauthorized
	402 4.02, bad option
	403 4.03, forbidden
	404 4.04, not found
	405 4.05, method not allowed
	406 4.06, not acceptable
	412 4.12, precondition failed
	413 4.13, request entity to large
	415 4.15, unsupported content-format
	500 5.00, internal server error
	501 5.01, not implemented
	502 5.02, bad gateway
	503 5.03, service unavailable
	504 5.04, gateway timeout

	505 5.05, proxy not supported
<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, string type

3.4.7 AT+COAPDATASTATUS

The command gets the CoAP status

AT+COAPDATASTATUS

Read Command AT+COAPDATASTATUS?	Response +COAPDATASTATUS:<status> OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<status>	Integer type
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)

3.4.8 AT+COAPCFG

The command configs the CoAP client

AT+COAPCFG

Set Command AT+COAPCFG="Showra" [, <Showra>]	Response OK If there is any error, response:
--	---

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

Parameter

<Showra>	Integer type, Set whether or not to display the address of sender
0	do not display the IP address and port
1	display the IP address and port
<Showrspopt>	Integer type, Set whether or not to display the coap option of sender
0	do not display the coap option
1	display the coap option

3.4.9 AT+COAPALISIGN

The command gets the ali cloud sign

AT+COAPALISIGN

Set Command AT+COAPALISIGN="<devId>","<devName>" ,"<devSecret>","<productKey>"	Response +COAPALISIGN: "<sign>" OK If there is any error, response: +CME ERROR: <err>
Test Command AT+COAPALISIGN=?	Response +COAPALISIGN:"<dev_id>","<dev_name>"," "<dev_secret>","<product_key>" OK If there is any error, response: +CME ERROR: <err>
Maximum Response Time	10s
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 AliClode
<sign>	String type The calculated sign value

3.4.10 +COAPURC

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

+COAPURC

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

```
+COAPURC:"req", [<ip_addr>,<port>,<type>,<method>,<msgid>,<mode>[,<tkl>,<token>][,<opt_cnt>,<opt_name>,"<opt_value>"[,...]][,<length>,<data>]
```

Parameter

<ip_addr>	String type The CoAP server ip address, it will show when set AT+COAPCFG="Showra",1
<port>	Integer type The CoAP server port, it will show when set AT+COAPCFG="Showra",1
<type>	Integer type The CoAP Protocol of message type, range: 0-3, refer the 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>	String type The token value. Hexadecimal format
<opt_cnt>	Integer type The count of option, it will show when set AT+COAPCFG="Showrspopt",1
<opt_name>	Integer type The option name, it will show when set AT+COAPCFG="Showrspopt",1
<opt_value>	String type The option value, it will show when set AT+COAPCFG="Showrspopt",1
<length>	Integer type The data length. The max length is 512 bytes
<data>	String type Receive data from server

3.5 MQTT Commands

3.5.1 AT+ECMTCFG

The command creates a MQTT client.

AT+ECMTCFG

Set Command Configure the keep alive time AT+ECMTCFG="keepalive",<tcpconnectID>[,<keep-alive time>]	Response OK If <keep-alive time> is omitted, query the keep-alive time: +ECMTCFG: "keepalive",<keep-alive time> OK If there is any error, response: ERROR
Set Command Configure the session type AT+ECMTCFG="session",<tcpconnectID>[,<clean_session>]	Response OK If <clean_session> is omitted, query the session type: +ECMTCFG: "session",<clean_session> OK If there is any error, response:

<p>Set Command Configure the timeout of message delivery AT+ECMTCFG="timeout",<tcpconnectID>[,<pkt_timeout>[,<retry_times>][,<timeout_notice>]]</p>	<p>ERROR</p> <p>Response OK If <pkt_timeout>, <retry_times>, <timeout_notice> are omitted, query the timeout value of message delivery: +ECMTCFG: "timeout",<pkt_timeout>,<retry_times>,<timeout_notice> OK If there is any error, response: ERROR</p>
<p>Set Command Configure the Will information AT+ECMTCFG="will",<tcpconnectID>[,<will_fg>[,<will_qos>,<will_retain>,"<will_topic>","<will_msg>"]]</p>	<p>Response OK If <will_fg>, <will_qos>, <will_retain>, <will_topic> and <will_msg> are omitted, query the Will information: +ECMTCFG: "will",<will_fg>[,<will_qos>,<will_retain>,<will_topic>,<will_msg>] OK If there is any error, response: ERROR</p>
<p>Set Command Configure the MQTT protocol version to be used AT+ECMTCFG="version",<tcpconnectID>[,<version>]</p>	<p>Response OK If <version> is omitted, query the MQTT protocol version: +ECMTCFG: "version",<version> OK If there is any error, response: ERROR</p>
<p>Set Command Configure the device information for Ali cloud AT+ECMTCFG="aliauth",<tcpconnectID>[, "<product_key>","<device_name>","<device_secret>"]</p>	<p>Response OK If "<product_key>","<device_name>","<device_secret>" are omitted, query the device information: +ECMTCFG: "aliauth",<product_key>,<device_name>,<device_secret> OK</p>

	If there is any error, response: ERROR
Set Command Configure the cloud type and the format of sent data for the cloud AT+ECMTCFG="cloud",<tcpconnectID>,<cloud type>,<data type>	Response OK If there is any error, response: ERROR
Test Command AT+ECMTCFG=?	Response +ECMTCFG: +ECMTCFG: "keepalive", (0), (0-3600) +ECMTCFG: "session", (0), (0,1) +ECMTCFG: "timeout", (0), (1-60), (1-10), (0,1) +ECMTCFG: "will", (0), (0,1), (0-2), (0,1), "will_topic", "will_msg" +ECMTCFG: "version", (0), (3,4) +ECMTCFG: "aliauth", (0), "productkey", "devicename", "devicesecret" +ECMTCFG: "cloud", (0-255), (0-255) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE
Parameter	
<tcpconnectID>	String type MQTT socket identifier, the value is 0
<keep-alive time>	Integer type. The range is 0-3600. The default value is 120. Unit: second. It 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.
<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 about the client and treat the connection as "clean".
<pkt_timeout>	Integer type. Timeout of the packet delivery. The range is 1-60. The default value is 10.

	Unit: second.
<retry_times>	Integer type. (not supported) Retry times when packet delivery times out. The range is 0-10. The default value is 3.
<timeout_notice>	Integer type. (not supported) 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 When a client sends a PUBLISH message to a server, the server will not hold on to the message after it has been delivered to the current subscribers 1 When a client sends a PUBLISH message to a server, the server should hold on to the message after it has been delivered to the current subscribers
<will_topic>	Integer type. Will topic string, The maximum size is 255 bytes
<will_msg>	Integer 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. , The maximum size is 255 bytes
<version>	Integer type. Version of MQTT protocol, the default is MQTT v3.1.1 3 MQTT v3.1 4 MQTT v3.1.1
<product_key>	Integer type. Product key issued by Ali Cloud, The maximum size is 32 bytes
<device_name>	Integer type. Device name issued by Ali Cloud, The maximum size is 32 bytes
<device_secret>	Integer type. Device secret key issued by Ali Cloud, The maximum size is 64 bytes
<cloud_type >	Integer type 0 mosquitto 1 OneNet 2 alibaba 3-255 is reserved for customers
<data_type>	Integer type, the range is 0-255 For OneNet cloud 1 OneNet data type1 2 OneNet data type2

3 OneNet data type3

4 OneNet data type4

5 OneNet data type5

6 OneNet data type6

For Ali Cloud

1 Json data

2 string data

For other, data format insensitive

3.5.2 AT+ECMTOPE

The command is used to open a network for MQTT client.

AT+ECMTOPE

Set Command AT+ECMTOPE=<tcpconnectID>,"<host_name>", <port>	Response OK +ECMTOPE: <tcpconnectID>,<result> If there is any error, response: ERROR
Test Command AT+ECMTOPE=?	Response +ECMTOPE: (list of supported <tcpconnectID>s),"<host_name>",(list of supported <port>s) OK
Read Command AT+ECMTOPE?	Response [+ECMTOPE: <tcpconnectID>,"<host_name>",<po rt>] OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 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 The port of the server. The range is 1-65535
<result>	Integer type, Result of the command execution

-1	Failed to open network
0	Opened network successfully

3.5.3 AT+ECMTCLOSE

The command is used to close mqtt client.

AT+ECMTCLOSE

Set Command AT+ECMTCLOSE=<tcpconnectID>	Response OK +ECMTCLOSE: <tcpconnectID>,<result> If there is any error, response: ERROR
Test Command AT+ECMTCLOSE=?	Response +ECMTCLOSE: (list of supported <tcpconnectID>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 0
<result>	Integer type, Result of the command execution
	-1 Failed to close mqtt
	0 Close network successfully

3.5.4 AT+ECMTCONN

The command send MQTT connect packet.

AT+ECMTCONN

Set Command AT+ECMTCONN=<tcpconnectID>,"<clientID>" [,"<username>"[,"<password>"]]	Response OK +ECMTCONN: <tcpconnectID>,<result>[,<ret_code>] If there is any error, response: ERROR
Test Command	Response

AT+ECMTCONN=?	+ECMTCONN: (list of supported <tcpconnectID>s), "<clientID>" [, "<username>" [, "<password>"]] OK
Read Command AT+ECMTCONN?	Response [+ECMTCONN: <tcpconnectID>, <state>] OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 0
<clientID>	String type The client identifier, The maximum size is 48 bytes
<username>	String type User name of the client. It can be used for authentication, The maximum size is 48 bytes
<password>	String type Password corresponding to the user name of the client. It can be used for authentication, The maximum size is 96 bytes
<result>	Integer type 0 send data successfully and received ACK from server 1 send data successfully and received error ACK from server 2 send data fail
<ret_code>	Integer type 0 connect accepted 1 connect refused – incorrect protocol version 2 connect refused – invalid client identifier 3 connect refused – server unavailable 4 connect refused – bad user name or password 5 connect refused – not authorised 6 connect fail

3.5.5 AT+ECMTDISC

The command send MQTT disconnect packet.

AT+ECMTDISC

Set Command AT+ECMTDISC=<tcpconnectID>	Response OK
--	-----------------------

	+ECMTDISC: <tcpconnectID>,<result> If there is any error, response: ERROR
Test Command AT+ECMTDISC=?	Response +ECMTDISC: (list of supported <tcpconnectID>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 0
<result>	Integer type, Result of the command execution
	-1 Failed to disconnect network
	0 Disconnect network successfully

3.5.6 AT+ECMTSUB

The command send MQTT subscribe packet.

AT+ECMTSUB

Set Command AT+ECMTSUB=<tcpconnectID>,<msgID>,"<topic> <c>",<qos>	Response OK +ECMTSUB: <tcpconnectID>,<msgID>,<result>[, <value>] If there is any error, response: ERROR
Test Command AT+ECMTSUB=?	Response +ECMTSUB: (list of supported <tcpconnectID>s), (list of supported <msgID>s),"<topic>",<list of supported <qos>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 0
----------------	--

<msgID>	Integer type Message identifier of packet. The range is 1-65535
<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
<result>	Integer type 0 send data successfully and received ACK from server 1 send data successfully and received error ACK from server 2 send data fail
<value>	Integer type It is granted qos levels

3.5.7 AT+ECMTUNS

The command send MQTT unsubscribe packet.

AT+ECMTUNS

Set Command AT+ECMTUNS=<tcpconnectID>,<msgID>,"<topic> "	Response OK +ECMTUNS: <tcpconnectID>,<msgID>,<result> > If there is any error, response: ERROR
Test Command AT+ECMTUNS=?	Response +ECMTUNS: (list of supported <tcpconnectID>s), (list of supported <msgID>s), "<topic>" OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

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

<result>	Integer type
	0 send data successfully and received ACK from server
	1 send data successfully and received error ACK from server
	2 send data fail

3.5.8 AT+ECMTPUB

The command send MQTT publish packet .

AT+ECMTPUB

Set Command AT+ECMTPUB=<tcpconnectID>,<msgID>,<qos>,<retain>,"<topic>","<payload>"	Response OK +ECMTPUB: <tcpconnectID>,<msgID>,<result>[,<value>] If there is any error, response: ERROR
Test Command AT+ECMTPUB=?	Response +ECMTPUB: (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	5s
Parameter Saving Mode	NO_SAVE

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 0
<msgID>	Integer type Message identifier of packet. The range is 0-65535. It will be 0 only when <qos>=0
<qos>	Integer type Message QoS, can be 0,1 or 2
<retain>	Integer type 0 Server should not retain the message 1 Server should retain the message
<topic>	String type

	Topic that needs to be published. The maximum length is 255 bytes
<payload>	String type or Hex type Message that needs to be published. Maximum length is 1024 bytes. If in data mode, the maximum length is 1024 bytes
<result>	Integer type 0 send data successfully and received ACK from server 1 send data successfully and received error ACK from server 2 send data fail
<value>	Integer type It is granted qos levels

3.5.9 +ECMTSTAT

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

+ECMTSTAT

```
+ECMTSTAT: <tcpconnectID>,<err_code>
```

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 0
<err_code>	Integer type. Error code 1 Connection is closed or reset by peer

3.5.10 +ECMTRECV

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

+ECMTRECV

```
+ECMTRECV: <tcpconnectID>,<msgID>,<topic>,<data>
```

Parameter

<tcpconnectID>	Integer type MQTT socket identifier. The value is 0
<msgID>	String type The message identifier of packet
<topic>	string type The topic that received from MQTT server
<data>	String type Receive data from server

3.6 Http Commands

3.6.1 AT+HTTPCREATE

Set command creates a http or https client instance. Configure host, server certification, etc.

Test command returns values supported as a compound value.

Note: only one instance and http was fully verified. https and multiple instances will be test later.

AT+HTTPCREATE

Set Command AT+HTTPCREATE=<flag>,<host> [,<authuser>,<authpasswd>]	Response If there are more commands need to enter: +HTTP CMD: CONTIUE ENTER CMD If all commands has enter: +HTTPCREATE: <httpclientId> If there is any error, response: +HTTP ERROR: <err>
Test Command AT+HTTPCREATE=?	Response +HTTPCREATE: (list of supported< flag >s), “<host>” , “<authuser>” , “<authpasswd>” OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<flag>	Integer type
	1 not last part of command
	0 last part of command
<host>	string type
	http server's host name
<authuser>	String type
	Authentication user name
<authpasswd>	String type
	Authentication password
< httpclientId >	Integer type
	http Client Id ,0

Example

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

OK

3.6.2 AT+HTTPCON

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

Test command returns values supported as a compound value.

AT+HTTPCON

Set Command AT+HTTPCON=<httpClientId>	Response OK If there is any error, response: +HTTP ERROR: <err>
AT+HTTPCON=?	Response +HTTPCON: (list of supported< httpClientId >) OK
Maximum Response Time	40s
Parameter Saving Mode	NO_SAVE

Parameter

<httpClientId>	Integer type http client id returned by +HTTPCREATE
----------------	--

Example

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

3.6.3 AT+HTTPDESTROY

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

Test command returns values supported as a compound value.

AT+HTTPDESTROY

Set Command AT+HTTPDESTROY=<httpClientId>	Response OK If there is any error, response: +HTTP ERROR: <err>
AT+HTTPDESTROY=?	Response

	+HTTPDESTROY: (list of supported< httpclientId >) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<httpclientId>	Integer type http client id returned by +HTTPCREATE
----------------	--

Example

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

3.6.4 AT+HTTPSEND

Set command sends data to the http server.

Test command returns values supported as a compound value.

NOTE : only one send command could be processing before the related receiving is complete.

e.g. 2nd AT+HTTPSEND=xxx will return +HTTP ERROR: SEND FAILED.

AT+HTTPSEND

Set Command AT+HTTPSEND=<httpclientId>, <method>, <pathlen>, <path>[, <customheaderlen>, <customheader>, <contenttypelen>, <contentType>, <contentlen>, <content>]	Response OK If there is any error, response: +HTTP ERROR: <err>
AT+HTTPSEND=?	Response +HTTPSEND: (list of supported< httpclientId>), (list of supported< method>), (range of supported< pathlen>), "<path>" , (range of supported< customheaderlen>), "<customheader>" , (range of supported< contenttypelen>), "<contentType>" , (range of supported< contentlen>), "<content>" OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<httpClientId>	Integer type http client id returned by +HTTPCREATE
<method>	Integer type; http method 0 GET 1 POST 2 PUT 3 DELETE 4 HEAD
<pathlen>	Integer type Lenth of path,0-260
<path>	string type Path
<customheaderlen>	Integer type Length of custom header,0-255
<customheader>	string type Customheader in hex string
<contentTypelen>	Integer type Length of content type,0-64
<contentType>	string type Content type
<contentlen>	Integer type,0-1024 Length of content
<content>	String type User data need to send in hex string

Example

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

3.6.5 +HTTPRESPH indicator of response header

This is an unsolicited message to represent response header.

+HTTPRESPH

```
+HTTPRESPH: <clientId>,<responseCode>,<headerlen>,<header>
```

Parameter

<clientId>	Integer type http client id returned by +HTTPCREATE
<responseCode>	Integer type

	http response code
<headerlen>	Integer type Length of http response header
<header>	string type Header

3.6.6 +HTTPRESPC indicator of response content

This is an unsolicited message to represent response content.

+HTTPRESPC

```
+HTTPRESPC: <clientId>,<flag>,<contentlength>,<blockcontentlen>,<content>
```

Parameter

<clientId>	Integer type http client id returned by +HTTPCREATE
<flag>	Integer type; if has more data 0 No more data 1 Has more data
<contentlength>	Integer type Length of content
<blockcontentlen>	Integer type Current block length
<content>	string type content data string which is converted from content hex data, the length is 2*original hex data.

3.6.7 +HTTPERR indicator of error message

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

+HTTPERR

```
+HTTPERR: <clientId>,<errorcode>, [<rspcode>]
```

Parameter

<clientId>	Integer type http client id returned by +HTTPCREATE
<errorcode>	Integer type

2	URL parse error
3	could not resolve DNS
4	Protocol error
5	HTTP 404 error, NOT FOUND
6	HTTP 403 error, REFUSED
7	HTTP xxx error, with specific rspcode followed
8	Connection timeout
9	Connection error
10	Connection fatal error
11	Connection closed
13	Buffer overflow error

3.7 OneNET Extension Commands

3.7.1 AT+MIPLCONFIG

This command is used to enable/disable bootstrap mode and configure bootstrap server address or access server address.

Note: the command should be executed before the communication suite instance is created with AT+MIPLCREATE command.

AT+MIPLCONFIG

Set Command AT+MIPLCONFIG=<bsMode>,<ip>,<port>	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLCONFIG=?	Response +MIPLCONFIG: (list of supported< bsMode>s, "<ip>","<port>") OK
Maximum Response Time	5s
Parameter Saving Mode	AUTO_SAVE

Parameter

<bsMode>	Integer type 0 Disable bootstrap 1 Enable bootstrap The default value is 0
<ip>	When <bsMode>=0,<ip> represents access server IP When <bsMode>=1,<ip> represents bootstrap server IP
<port>	When <bsMode>=0,<port> represents access server port

When <bsMode>=1,<port> represents bootstrap server port

Example

```
AT+MIPLCONFIG=1,"183.230.40.39","5683"
```

3.7.2 AT+MIPLCREATE

This command creates an instance of communication to CMIoT OneNET platform.

AT+MIPLCREATE

Execution Command AT+MIPLCREATE	Response +MIPLCREATE:<ref> OK If there is any error, response: +CIS ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Example

3.7.3 AT+MIPLDELETE

This command deletes a specified OneNET communication instance.

AT+MIPLDELETE

Set Command AT+MIPLDELETE=<ref>	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLDELETE=?	Response +MIPLDELETE: (list of supported<ref>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
-------	--

Example

3.7.4 AT+MIPLOPEN

This command send login request to OneNET.

AT+MIPLOPEN

Set Command AT+MIPLOPEN=<ref>,<lifetime>[,<timeout>]	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLOPEN=?	Response +MIPLOPEN: (list of supported<ref>), (range of supported<lifetime>), range of supported<timeout>) OK
Maximum Response Time	10s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<lifetime>	Integer type lifetime value in second of the client, in this time client need send update msg to OneNET. range: 0xF-0x0FFFFFFF. unit: second
<timeout>	Integer type Timeout of registration. range: 30-65535. unit: second

Example

3.7.5 AT+MIPLCLOSE

This command send logout request to OneNET.

AT+MIPLCLOSE

Set Command AT+MIPLCLOSE=<ref>	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLCLOSE=?	Response +MIPLCLOSE: (list of supported<ref>) OK

Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
-------	--

Example

3.7.6 AT+MIPLADDOBJ

This command add an object from a specified OneNET communication instance.

AT+MIPLADDOBJ

Set Command AT+MIPLADDOBJ=<ref>,<objectid>,<instancecount>,<instancebitmap>,<attributecount>,<actioncount>	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLADDOBJ=?	Response +MIPLADDOBJ: (list of supported<ref>), (range of supported<objectid>), (range of supported<instancecount>), "<instanceBitmap>", (range of supported<attributecount >), (range of supported<actioncount >) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<objectid>	Integer type object id
<instancecount>	Integer type Number of instances, Maximum number is 100
<instancebitmap>	string type Instance bitmap, each bit represents an instance, 0 means not available, 1 means available

<attributecount>	Integer type
	Number of attributes
<actioncount>	Integer type
	Number of actions

Example

```
AT+MIPLADDOBJ=0,3306,1,"1",7,1
```

3.7.7 AT+MIPLDELOBJ

This command deletes an object from a specified OneNET communication instance.

AT+MIPLDELOBJ

Set Command AT+MIPLDELOBJ=<ref>,<objectid>	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLDELOBJ=?	Response +MIPLDELOBJ: (list of supported<ref>), (range of supported<objectid >) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<objectid>	Integer type object id

Example

3.7.8 AT+MIPLNOTIFY

This command notify OneNET that specified values changed.

AT+MIPLNOTIFY

Set Command AT+MIPLNOTIFY=<ref>,<msgid>,<objec	Response OK
--	-----------------------

tid>,<instanceid>,<resourceid>,<valuetype>,<len>,<value>,<index>,<flag>[,<ackid>[,<raiflag>]]	If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLNOTIFY=?	Response +MIPLNOTIFY: (list of supported<ref>), (range of supported<msgid>), (range of supported<objectid>), (range of supported<instanceid>), (range of supported<resourceid>), (range of supported<valuetype>), (range of supported<len>), "<value>", (list of supported<index>), (list of supported<flag>), (list of supported<ackid>), (list of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type	a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type	Msg id, msg id carried in +MIPLOBERVE
<objectid>	Integer type	object id
<instanceid>	Integer type	Instance id
<resourceid>	Integer type	Resource id
<valuetype>	Integer type	
	1	string
	2	opaque
	3	Integer
	4	float
	5	bool
<len>	Integer type	
	valuetype	len
	string	string len
	opaque	byte len
	integer	The number of bytes occupied by the integer. Optional 2, 4, 8
	float	The number of bytes occupied by the float. Optional 4, 8
	bool	1

<value>	string type	
	valuetype	value
	string	string
	opaque	Hex string
	interger	interger
	float	string
	bool	0:false; 1:true
<index>	Integer type	
	serial number, for one command may not be able to send the entire content. It's from N~0. 0 is last part of content.	
<flag>	Integer type; if there are multiple msg	
	1	First msg
	2	Middle msg
	0	Last msg
<ackid>	Integer type	
	0	This notify will be sent in Non-confirmable(NON) message
	1-65535	This notify will be sent in Confirmable(CON) message and the value will indicated by "+MIPLEVENT"
<raiflag>	Integer type	
	0 (PS_SOCKET_RAI_NO_INFO) disable RAI	
	1 (PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send	
	2 (PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received	

Example

```
AT+MIPLNOTIFY=0,msgid,3306,0,5750,1,5,"hello",0,0,116
+MIPLEVENT: 0,26,116
```

3.7.9 AT+MIPLREADRSP

After receive +MIPLREAD, This command return the read result to OneNET.

AT+MIPLREADRSP

Set Command AT+MIPLREADRSP=<ref>,<msgid>,<result>[, <objectid>,<instanceid>,<resourceid>, <valuetype>,<len>,<value>,<index>,<flag> >[,raiflag]]	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLREADRSP=?	Response +MIPLREADRSP: (list of supported<ref>), (range of supported<msgid>), (range of

	supported<objectid>), (range of supported<instanceid>), (range of supported<resourceid>), (range of supported<valuetype>), (range of supported<len>), "<value>", (list of supported<index>), (list of supported<flag>), (list of supported<ackid>), (list of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Msg id, msg id carried in +MIPLREAD
<result>	Integer type; Result of read command 1 205 content, read success 11 400 bad request 12 401 unauthorized 13 404 not found 14 405 method not allowed 15 406 not acceptable
<objectid>	Integer type object id
<instanceid>	Integer type Instance id
<resourceid>	Integer type Resource id
<valuetype>	Integer type 1 string 2 opaque 3 Integer 4 float 5 bool
<len>	Integer type Value length
<value>	string type value
<index>	Integer type serial number, for one command may not be able to send the entire content. It's

	from N~0. 0 is last part of content.
<Flag>	Integer type
	1 First msg
	2 Middle msg
	0 Last msg
<raiflag>	Integer type
	0 (PS_SOCKET_RAI_NO_INFO) disable RAI
	1 (PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send
	2 (PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received

Example

```
AT+MIPLREADRSP=0,msgid,1,3306,0,5750,1,5,"hello",0,0
```

3.7.10 AT+MIPLWRITERSP

After receive +MIPLWRITE, This command return the write result to OneNET.

AT+MIPLWRITERSP

Set Command AT+MIPLWRITERSP=<ref>,<msgid>,<result>[,raiflag]	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLWRITERSP=?	Response +MIPLWRITERSP: (list of supported<ref>), (range of supported<msgid>), (list of supported<result>), (list of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Msg id, msg id carried in +MIPLWRITE
<result>	Integer type; Result of read command
	2 204 changed, write success
	11 400 bad request
	12 401 unauthorized

	13	404 not found
	14	405 method not allowed
<raiflag>	Integer type	
	0 (PS_SOCKET_RAI_NO_INFO) disable RAI	
	1 (PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send	
	2 (PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received	

Example

```
AT+MIPLWRITERSP=0,msgid,2
```

3.7.11 AT+MIPLEXECUTERSP

After receive +MIPLEXECUTE, This command return the execute result to OneNET.

AT+MIPLEXECUTERSP

Set Command AT+MIPLEXECUTERSP=<ref>,<msgid>,<result>[,raiflag]	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLEXECUTERSP=?	Response +MIPLEXECUTERSP: (list of supported<ref>), (range of supported<msgid>), (list of supported<result>), (list of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Msg id, msg id carried in +MIPLEXECUTE
<result>	Integer type; Result of read command
	1 205 content
	2 204 changed, execute success
	11 400 bad request
	12 401 unauthorized
	13 404 not found
	14 405 method not allowed

	15	406 not acceptable
<raiflag>	Integer type	
	0 (PS_SOCKET_RAI_NO_INFO) disable RAI	
	1 (PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send	
	2 (PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received	

Example

```
AT+MIPLEXECUTERSP=0,msgid,2
```

3.7.12 AT+MIPLOBSERVERSP

After receive +MIPLOBSERVE, This command return the observe result to OneNET. It also can response the cancel observe.

AT+MIPLOBSERVERSP

Set Command AT+MIPLOBSERVERSP=<ref>,<msgid>,<result>	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLOBSERVERSP=?	Response +MIPLOBSERVERSP: (list of supported<ref>), (range of supported<msgid>), (list of supported<result>), (range of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Msg id, msg id carried in +MIPLOBSERVE
<result>	Integer type; Result of read command
	1 205 content
	2 204 changed, execute success
	11 400 bad request
	12 401 unauthorized
	13 404 not found
	14 405 method not allowed

	15	406 not acceptable
<raiflag>	Integer type	
	0 (PS_SOCKET_RAI_NO_INFO) disable RAI	
	1 (PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send	
	2 (PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received	

Example

```
AT+MIPLDISCOVER=0,msgid,1
```

3.7.13 AT+MIPLDISCOVERRSP

After receive +MIPLDISCOVER, This command return all resources of object id specified in +MIPLDISCOVER.

AT+MIPLDISCOVERRSP

Set Command AT+MIPLDISCOVERRSP=<ref>,<msgid>,<result>,<length>,<valuestring>[,<raiflag>]	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLDISCOVERRSP=?	Response +MIPLDISCOVERRSP: (list of supported<ref>), (range of supported<msgid>), (list of supported<result>), (range of supported<length>), \"<valuestring>\" , (list of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Msg id, msg id carried in +MIPLDISCOVER
<result>	Integer type; Result of read command
	1 205 content, discover success
	2 204 changed
	11 400 bad request
	12 401 unauthorized

	13	404 not found
	14	405 method not allowed
	15	406 not acceptable
<len>	Integer type	
	Value length, Maximum length is 255	
<valuestring>	string type	
	Object's resource id, multiple resources separated by “;”	
<raiflag>	Integer type	
	0 (PS_SOCKET_RAI_NO_INFO) disable RAI	
	1 (PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send	
	2 (PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received	

Example

```
AT+MIPLDISCOVERRSP=0,msgid,1,14,"5750;5850;5851"
```

3.7.14 AT+MIPLPARAMETERRSP

After receive +MIPLPARAMETER, This command return the execute result to OneNET.

AT+MIPLPARAMETERRSP

Set Command AT+MIPLPARAMETERRSP=<ref>,<msgid> ,<result>[,<raiflag>]	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLPARAMETERRSP=?	Response +MIPLPARAMETERRSP: (list of supported<ref>), (range of supported<msgid>), (list of supported<result>), (list of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Msg id, msg id carried in +MIPLPARAMETER
<result>	Integer type; Result of read command
	1 205 content
	2 204 changed, execute success

	11	400 bad request
	12	401 unauthorized
	13	404 not found
	14	405 method not allowed
	15	406 not acceptable
<raiflag>	Integer type	
	0 (PS_SOCKET_RAI_NO_INFO) disable RAI	
	1 (PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send	
	2 (PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received	

Example

```
AT+MIPLPARAMETERESP=0,msgid,2
```

3.7.15 AT+MIPLUPDATE

This command updates the register information,such as lifetime

AT+MIPLUPDATE

Set Command AT+MIPLUPDATE=<ref>,<lifetime>,<withobjectflag>[,<raiflag>]	Response OK If there is any error, response: +CIS ERROR: <err>
Test Command AT+MIPLUPDATE=?	Response +MIPLUPDATE: (list of supported<ref>), (range of supported<lifetime>), (list of supported<withobjectflag>), (list of supported<raiflag>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<lifetime>	Integer type lifetime value in second of the client
<withobjectflag>	Integer type; 1 Update object info at the same time

	0	Don't update object info
<raiflag>	Integer type	
	0	(PS_SOCKET_RAI_NO_INFO) disable RAI
	1	(PS_SOCKET_RAI_NO_UL_DL_FOLLOWED) enable RAI, release after send
	2	(PS_SOCKET_ONLY_DL_FOLLOWED) enable RAI, release after ACK has received

Example

```
AT+MIPLUPDATE=0,3600,0
```

3.7.16 AT+MIPLVER

This command return the version

AT+MIPLUPDATE

Read Command AT+MIPLVER?	Response +MIPLVER: <ver> OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<ver>	string type Version of OneNET protocol
-------	---

Example

```
AT+MIPLVER?
+MIPLVER: 2.2.0
OK
```

3.7.17 +MIPLREAD indicator of read request

This is an unsolicited message to represent remote reading command. OneNET request device to upload specified resource value.

+MIPLREAD

```
+MIPLREAD: <ref>,<msgid>,<objectid>,<instanceid>,<resourceid>
```

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Message id of this message
<objectid>	Integer type object id
<instanceid>	Integer type Instance id, -1 means read all the instance belong to the object
<resourceid>	Integer type Resource id, -1 means read all the resources belong to the instance

Example

3.7.18 +MIPLWRITE indicator of write request

This is an unsolicited message to represent remote writing command. OneNET request device to modify specified resource value.

+MIPLWRITE

```
+MIPLWRITE:
<ref>, <msgid>, <objectid>, <instanceid>, <resourceid>, <valuetype>, <len>,
<value>, <flag>, <index>
```

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Message id of this message
<objectid>	Integer type object id
<instanceid>	Integer type Instance id
<resourceid>	Integer type Resource id
<valuetype>	Integer type 1 string 2 opaque 3 Integer 4 float 5 bool
<len>	Integer type

	Value length
<value>	string type value
<Flag>	Integer type
	1 First msg
	2 Middle msg
	0 Last msg
<index>	Integer type
	serial number, for one command may not be able to send the entire content. It's from N~0. 0 is last part of content.

Example

3.7.19 +MIPLEXECUTE indicator of execute request

This is an unsolicited message to represent remote executing command. OneNET request device to execute some predefined operations on specified resource.

+MIPLEXECUTE

```
+MIPLEXECUTE:
<ref>,<msgid>,<objectid>,<instanceid>,<resourceid>,<len>,<arguments>
```

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Message id of this message
<objectid>	Integer type object id
<instanceid>	Integer type Instance id
<resourceid>	Integer type Resource id
<len>	Integer type Value length
<arguments>	string type Represent the operation

Example

3.7.20 +MIPLOBSERVE indicator of observe request

This is an unsolicited message to represent remote observe command. OneNET request device to upload the values of specified resource or all resources of specified instance when they change.

+MIPLOBSERVE

```
+MIPLOBSERVE: <ref>,<msgid>,<flag>,<objectid>,<instanceid>,<resourceid>
```

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Message id of this message
<Flag>	Integer type 1 Observe add 0 Observe cancel
<objectid>	Integer type object id,
<instanceid>	Integer type Instance id, -1 means observing all the instances and resources belong to the object
<resourceid>	Integer type Resource id, -1 means observing all the resources belong to the instance

Example

3.7.20 +MIPLDISCOVER indicator of observe request

This is an unsolicited message to represent remote discover command. OneNET request device to upload the resource id list of specified object id.

+MIPLDISCOVER

```
+MIPLDISCOVER: <ref>,<msgid>,<objectid>
```

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Message id of this message
<objectid>	Integer type

object id

Example

3.7.21 +MIPLPARAMETER indicator of observe parameter request

This is an unsolicited message to represent remote parameter command. with this message, OneNET can set interval and threshold value of specified resource.

+MIPLPARAMETER

```
+MIPLPARAMETER: <ref>,<msgid>,<objectid>,<instanceid>,<resourceid>,<len>,<parameter>
```

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<msgid>	Integer type Message id of this message
<objectid>	Integer type object id
<instanceid>	Integer type Instance id , -1 means observing all the instances and resources belong to the object
<resourceid>	Integer type Resource id, -1 means observing all the resources belong to the instance
<len>	Integer type Value length
<arguments>	string type Strategy parameter, like pmin=xxx;pmax=xxx;lt=xxx;gt=xx;st=xxx

Example

3.7.22 +MIPLEVENT indicator of event report

This is an unsolicited message to report status event.

+MIPLEVENT

```
+MIPLEVENT: <ref>,<evtid>[,<extend>,<ackid>]
```

Parameter

<ref>	Integer type a specified OneNET communication instance id returned by +MIPLCREATE
<eventid>	Integer type <ul style="list-style-type: none"> 1 Bootstrap start 2 Bootstrap success 3 Bootstrap failed 4 Connect success 5 Connect failed 6 Registration success 7 Registration failed 8 Registration timeout 9 Lifetime timeout 10 Status halt 11 Update success 12 Update failed 13 Update timeout 14 Update need 15 Un-registration done 20 Response failed 21 Response success 25 Notify failed 26 Notify success
<extend>	Integer type Extend parameter, if eventid is Response failed(20), extend is corresponding message id. If eventid is Update need(14), extend is lifetime remaining time(in seconds)
<ackid>	Integer type When the eventid is Notify success(26), the corresponding ackid issued when notify is carried.

Example

```
eventid is Update need, lifetime remaining 30s
+MIPLEVENT: 0,14,30
eventid is Response failed, message id is 1234
+MIPLEVENT: 0,20,1234
```

3.7.23 Summary of <err> Codes

<err> Codes	Description
100	unknown error
601	parameter error
602	status error

651	not support
652	sdk error
653	no instance
654	malloc fail
655	network not ready
656	configuration error

3.8 China Telecom Extension Commands

3.8.1 AT+CTM2MVER

This command sends request to get CTM2M version.

AT+CTM2MVER

Read Command AT+CTM2MVER?	Response +CTM2MVER: <lwm2m>,<ctm2m>,<ctmt>,<ctmv> OK If there is any error, response: +CTM2M ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<lwm2m>	String type Version of LWM2M
<ctm2m>	String type Version of CTM2M
<ctmt>	String type Chip type: EC616 etc.
<ctmv>	String type Software version of module

Example

```
AT+CTM2MVER?
+CTM2MVER: 1.0,1.1.0,EC616_EVK,V001.000.xxx
OK
```

3.8.2 AT+CTM2MSETMOD

This command sends command to set UE working mode.

AT+CTM2MSETMOD

Set Command AT+CTM2MSETMOD=<MOD_ID>,<MOD_DATA>	Response OK If there is any error, response: +CTM2M ERROR: <err>
Read Command AT+CTM2MSETMOD?	Response +CTM2MSETMOD: <idAuthMode>, <NAT type>,<onUQMode>, <autoHeartbeat>, <wakeupPolicy>,<protocolMode> OK
Test Command AT+CTM2MSETMOD=?	Response +CTM2MSETMOD: (list of supported <MOD_ID>s), (list of supported < MOD_DATA>s) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<MOD_ID>	Integer type 1---IDAuth_Mode 2--- NAT type 3--- ON_UQMode 4--- Auto_Heartbeat 5--- Wakeup_Notify 6--- Protocol_Mode
<MOD_DATA>	Integer type If MOD_ID=1: 1: default value, no authentication string (default) 2: SIMD authentication string from outside of module 3: SM9 authentication string from outside of module 4: SIMD authentication string from inside of module 5: SM9 authentication string from inside of module If MOD_ID=2: 0: enable NAT 1: disable NAT If MOD_ID=3: 1: UQ mode off (default)

- 2: UQ mode on
- If MOD_ID=4:
 - 1: auto heartbeat (default)
 - 2: no default auto heartbeat
- If MOD_ID=5:
 - 1: notify to MCU (default)
 - 2: not notify to MCU
- If MOD_ID=6:
 - 1: normal (default)
 - 2: enhance

Example

```
AT+CTM2MSETPM=1,1
```

3.8.3 AT+CTM2MSETPM

This command sends command to set the parameters to register to the China Telecom iot platform.

AT+CTM2MSETPM

Set Command AT+CTM2MSETPM =<Sever_IP>,<Port>,<Lifetime>	Response OK If there is any error, response: +CTM2M ERROR: <err>
Read Command AT+CTM2MSETPM?	Response +CTM2MSETPM: <Sever_IP>,<Port>,<Lifetime> OK
Test Command AT+CTM2MSETPM =?	Response +CTM2MSETPM: (list of supported <Sever_IP>), (range of supported<Port>),(range of supported<Lifetime>) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<Sever_IP>	String type IP address for LWM2M server.
<Port>	Integer type Port number for LWM2M server.
<Lifetime>	Integer type

Lifetime for LWM2M server and unit is second with minimum value 300

Example

```
AT+CTM2MSETPM=180.101.147.115,5683,86400
OK
AT+CTM2MSETPM?
+CTM2MSETPM: 180.101.147.115,5683,86400
OK
```

3.8.4 AT+CTM2MREG

This command sends command to register to the China Telecom iot platform.

AT+CTM2MREG

Execution Command AT+CTM2MREG	Response OK If there is any error, response: +CTM2M ERROR: <err>
Read Command AT+CTM2MREG?	Response +CTM2MREG:<status> OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Example

```
AT+CTM2MREG
OK
```

3.8.4 AT+CTM2MUPDATE

This command sends command to update binding mode to China Telecom iot platform.

AT+CTM2MUPDATE

Set Command AT+CTM2MUPDATE	Response OK If there is any error, response: +CTM2M ERROR: <err>
Maximum Response Time	40s
Parameter Saving Mode	NO_SAVE

Example

```
AT+CTM2MUPDATE
OK
+CTM2M: update,0,298
```

3.8.5 AT+CTM2MDEREG

This command sends command to exit China Telecom iot platform.

AT+CTM2MDEREG

Execution Command AT+CTM2MDEREG	Response +CTM2M: dereg,0 OK If there is any error, response: +CTM2M ERROR: <err>
Maximum Response Time	40s
Parameter Saving Mode	NO_SAVE

Example

```
AT+CTM2MDEREG
+CTM2M: dereg,0
OK
```

3.8.6 AT+CTM2MSEND

This command sends command to send business data to China Telecom iot platform.

AT+CTM2MSEND

Set Command AT+CTM2MSEND=<data>[,<mode>]	Response +CTM2MSEND: <msgID> OK If there is any error, response: +CTM2M ERROR: <err>
Test Command AT+CTM2MSEND=?	Response +CTM2MSEND: (list of supported <data>), (list of supported <mode>) OK
Maximum Response Time	40s
Parameter Saving Mode	NO_SAVE

Parameter

<data>	String type Hex string, Length should be less than 1024
--------	--

<mode>	Integer type
	0---CON mode
	1---NON mode
	2---NON with RAI flag
	3---CON with RAI flag
	Default value is 1

Example

```
AT+CTM2MSEND=0131323334
+CTM2MSEND: 554
OK
+CTM2M: send,31,554
```

3.8.7 AT+CTM2MCMDRSP

This command sends response to China Telecom iot platform.

AT+CTM2MCMDRSP

Set Command AT+CTM2MCMDRSP =<Msgid>,<Token>,<Rspcode>,<Uri_str >,<Observe>[,<Dataformat>,<Data>]	Response OK If there is any error, response: +CTM2M ERROR: <err>
Test Command AT+CTM2MCMDRSP=?	Response +CTM2MCMDRSP: (list of supported <Msgid>), (list of supported <Token>),(list of supported <Rspcode>), (list of supported <Uri_str>),(list of supported <Observe>), (list of supported <Dataformat>),(list of supported <Data>) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<Msgid>	Integer type It should be conform to the msgid of received message which needs to be replied.
<Token>	String type It should be conform to the token value of received message which needs to be replied.
<Rspcode>	Integer type Message response code. Something like 204, 205 etc.
<Uri_str >	String type

	/object id/instance id/resource id
<Observe>	Integer type 0--- set observe and no following parameters 1---cancel observe and no following parameters 8---just for response case 9---there are <Dataformat>,<data> parameters following
<Dataformat>	Integer type 1--- TLV format(application/vnd.oma.lwm2m+tlv) 2---opaque format(application/octet-stream) 7---TEXT format(text/plain) 8---JSON format(application/vnd.oma.lwm2m+json) 9---Core Link Param format(application/link-format)
<Data>	String type Length should be less than 1024

Example

```
AT+CTM2MCMRSP=51209,962AB03A,205,/3/0,0
OK
+CTM2M: send,31,51209
```

3.8.8 AT+CTM2MRMODE

This command set the display mode of the downlink data. When set to 1, the received downlink data will be displayed immediately, such as +CTM2MRECV: <data>. When set to 2, UE will cache data and display +CTM2MRECV, the data can be read by AT+CTM2MREAD. If there is cached data, UE will not enter Hibernate or Sleep2 mode. UE buffers up to 8 downlink data, after exceeding this limit, the oldest data will be discard first. 1 is default value, and restore to 1 when reset.

AT+CTM2MRMODE

Set Command AT+CTM2MCMRMODE=<mode>	Response OK If there is any error, response: +CTM2M ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	Integer type 0---Received the downlink data, buffer it and no indication. 1---Received the downlink data, immediately display +CTM2MRECV: <data>, 1 is default value 2---Received the downlink data, buffer it and display +CTM2MRECV
--------	--

3.8.9 AT+CTM2MREAD

This commands read the downlink data buffered by UE. It will return the first cached message every time and delete it from the cache. If there is no cached message, return OK directly.

AT+CTM2MREG

Execution Command AT+CTM2MREAD	Response <data> OK If there is any error, response: +CTM2M ERROR: <err>
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

3.8.10 +CTM2MRECV

This is an unsolicited message to represent received data message from China Telecom iot platform.

+CTM2MRECV

```
+CTM2MRECV: <data>
```

Parameter

<data>	String type
	Received data from platform

Example

```
+CTM2MRECV: 651A320502DA0034710B
```

3.8.11 +CTM2M

This is an unsolicited message to represent received async notification from China Telecom iot platform.

+CTM2M

```
+CTM2M: <operation>,<status code>[,<data1>,<data2>,<data3>]
```

Parameter

<operation>	String type
-------------	-------------

	Can be one type of below operations: reg/obsrv/update/dereg/send/lwstatus
<status code>	Integer type
	0---success
	1---timeout
	2---not send out packet
	9---receive platform RST packet and mean can't send UL to platform
	10---parameter error
	11---other errors
	13---authentication error
	14---UE not login
	22---iot protocol or lwm2m version mismatch
	24---lwm2m session invalid
	25---session load failure when quitting from sleep or after reboot
	26---Engine abnormal, need reboot by MCU
	28---TAU is due
	31---packet is already sent out
	32---object 19 not exist
<data1, 2, 3>	Integer type
	data1---msgID
	data2---N/A
	data3---N/A

Example

```
+CTM2M: lwstatus,29
```

3.8.12 +CTM2MCMD

This is a notification message from China Telecom iot platform to notify UE that some object operation is triggered and need to be replied.

+CTM2MCMD

```
+CTM2MCMD :
<msgid>,<cmdtype>,<token>,<uri_str>[,<observe>(,<dataformat>,<data>)]
```

Parameter

<msgid>	Integer type
	Message ID
<cmdtype>	Integer type
	0---Read
	1---Observe
	2---Write
	3---Write-Partial

	4---Write-Attribute 5---Discover 6---Execute 7---Create 8---Delete
<token>	Hex String type Async message response ID
<uri_str>	String type /objectid/instanceid/resourceid
<observe>	Integer type It is must when cmdtype=1 0---Set 1---Cancel
<dataformat>	Integer type 1---TLV 6---CoAP Param 7---text/plain 8---JSON
<data>	String type It is must when cmdtype=2、3、4、6、7 It is null when execute cmd has no arguments

Example

```
+CTM2M CMD: 3423,1,4AA33F97,/3/0,0
```

3.8.13 +CTM2M ERROR

This message indicates an error.

+CTM2M ERROR

```
+CTM2M ERROR:<errCode>
```

<errCode>	Description
8	Parameter not initalized
33	Already login
954	Not login
955	Login processing
957	Connect useless
958	No session
15	Object not observed

5	Psk error
969	Lifetime less than 300
972	Uri error
17	Data length not even
303	Operation not supported
1	Other error

3.9 Socket Command(solution B)

3.9.1 AT+ECSOCR

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

AT+ECSOCR

Set Command AT+ECSOCR=<type>,<protocol>[,<listen_port>[,<receive_control>[,<af_type>[,<ip_address>]]]]	Response <socket_id> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSOCR=?	Response +ECSOCR: (list of supported < type>), (list of supported < protocol>),(list of supported < listen_port>), (list of supported <receive_control>),(list of supported< af_type>), (list of supported < af_type>),(<ip_addr>) OK

Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<Type>	String type DGRAM UDP STREAM TCP
<Protocol>	Integer type Standard internet protocol definition. UDP is 17, and TCP is 6.
<listen_port>	Integer type A number in the range of 0-65535. This is the local port that will be included in sent messages and on which messages will be received. If it is 0 or omitted, the module will assign a random <listen_port> for this socket
<receive_control>	Integer type 0 the incoming messages will be ignored 1 the default value, the incoming messages will be received
<af_type>	String type AF_INET the default value. IPv4 AF_INET6 IPv6
<ip_address>	String type IP address. The IP address of the network assigned to UE.
<socket_id>	Integer type 0-9 A maximum of 7 sockets are supported, but other serviced may reduce this number.

Example

```
AT+ECSOCR="DGRAM",17,2233,1,"AF_INET"
1
OK
```

3.9.2 AT+ECSOST

Send a UDP datagram containing length bytes of data to <remote_port> on <remote_addr> . The command sends a UDP datagram containing length bytes of data to the specified host and port. It will return with the socket that it was sent on, and the number of bytes of data sent. If the amount of data is larger than the largest datagram that can be sent, return value of AT+ECSOST will indicate how much of the data was successfully sent.

AT+ECSOST

Set Command	Response
AT+ECSOST=<socket_id>,<remote_addr>,<remote_port>,<length>,<data>[,<sequence>[,<segment_id>,<segment_num>]]	<socket_id> ,<length>

	OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSOST=?	Response +ECSOST: (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	5s
Parameter Saving Mode	SAVE
Parameter	
<socket_id>	Integer type 0-9 Socket Id returned by AT+ECSOCR
<remote_addr>	String type Remote IP address
<remote_port>	Integer type 0-65535 This is the remote port on which messages will be received
<length>	Integer type 1-1400 Decimal length of data to be sent
<data>	String type data be sent in hex string format
<sequence>	Integer type

	1-255	If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported: +ECSOSTR: <socket_id>,<sequence>,<status>
<segment_id>	Integer type	
	1-4	One segment index of a segment message.
<segment_num>	Integer type	
	2-4	The total number which the messages will fragment.
<status>	Integer type	
	0	the sent status of datagram is fail
	1	the sent status of datagram is success

Example

```
AT+ECSOST=1,"180.167.122.150",5002,2,"ABAB"
1,2
OK
```

3.9.3 AT+ECSOSTT

Send a UDP datagram containing length bytes of data to <remote_port> on <remote_addr> with transparent mode.

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

If the command contain the parameter of "length", It need input enough data with hex string format after ">" output. If the command did not contain the parameter of "length", the input of data must end with "CTRL+Z".

AT+ECSOST

Set Command AT+ECSOSTT=<socket_id>,<remote_addr>,<remote_port>[,<length>[,<sequence>]]	Response <socket_id>,<length> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSOSTT=?	Response +ECSOSTT: (list of supported <socket_id>), (list of supported

	<remote_addr >), (list of supported< remote_port >), (list of supported < length>), (list of supported < sequence>) OK
Maximum Response Time	60s
Parameter Saving Mode	SAVE

Parameter	
<socket_id>	Integer type 0-9 Socket Id returned by AT+ECSOCR
<remote_addr>	String type Remote IP address
<remote_port>	Integer type 0-65535 This is the remote port on which messages will be received
<length>	Integer type 1-1400 Decimal length of data to be sent
<data>	String type data be sent in hex string format
<sequence>	Integer type 1-255 If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported: +ECSOSTR: <socket_id>,<sequence>,<status>
<status>	Integer type 2 the sent status of datagram is fail 3 the sent status of datagram is success

Example

```

AT+ECSOSTT=1,"180.167.122.150",5002,2
>ABAB
1,2
OK

AT+ECSOSTT=1,"180.167.122.150",5002
>ABAB
1A
1,2
OK

```

3.9.4 AT+ECSOSTF

Send a UDP datagram containing length bytes of data to <remote_port> on <remote_addr> and allows meta-data flags to be sent.

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

AT+ECSOSTF

<p>Set Command AT+ECSOSTF=<socket_id>,<remote_addr>,<remote_port>,<flag>,<length>,<data>[,<sequence>[,<segment_id>,<segment_num>]]</p>	<p>Response <socket_id>,<length> OK If there is any error, response: +CME ERROR: <err></p>
<p>Test Command AT+ECSOSTF=?</p>	<p>Response +ECSOSTF: (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 <</p>

	segment_num>) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<socket_id>	Integer type Socket Id returned by AT+ECSOCR
<remote_addr>	String type Remote IP address
<remote_port>	Integer type 0-65535 This is the remote port on which messages will be received
<flag>	Integer type Specifies the type of message transmission. Values of this argument are in hex format and are formed by logically OR'ing zero or more of the following flags: 0x100 Exception Message: Send messages with high priority 0x200 Release Indicator: indicate release after next message 0x400 Release Indicator: indicate release after next messages has been replied to If no flags are set, a value of 0 should be provided
<length>	Integer type 1-1400 Decimal length of data to be sent
<data>	String type data be sent in hex string format
<sequence>	Integer type 1-255 If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported: +ECSOSTR: <socket_id>,<sequence>,<status>
<segment_id>	Integer type 1-4 One segment index of a segment message.
<segment_num>	Integer type 2-4 The total number which the messages will fragment.
<status>	Integer type 0 the sent status of datagram is fail 1 the sent status of datagram is success

Example

```
AT+ECSOSTF=1,"180.167.122.150",5002,0x100,2,"ABAB"
```

```
1,2
```

```
OK
```

3.9.5 AT+ECSOSTFT

Send a UDP datagram containing length bytes of data to <remote_port> on <remote_addr> and allows meta-data flags to be sent with transparent mode.

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

If the command contain the parameter of "length", It need input enough data with hex string format after ">" output. If the command did not contain the parameter of "length", the input of data must end with "CTRL+Z".

AT+ECSOSTFT

Set Command AT+ECSOSTFT=<socket_id>,<remote_addr>,<remote_port>,<flag>[,<length>[,<sequence>]]	Response <socket_id>,<length> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSOSTFT=?	Response +ECSOSTFT: (list of supported <socket_id>), (list of supported <remote_addr>), (list of supported <remote_port>), (list of supported <flag>), (list of supported <length>),(list of supported <sequence>) OK
Maximum Response Time	60s
Parameter Saving Mode	SAVE
Parameter	
<socket_id>	Integer type Socket Id returned by AT+ECSOCR
<remote_addr>	String type Remote IP address

<remote_port>	Integer type 0-65535 This is the remote port on which messages will be received
<flag>	Integer type Specifies the type of message transmission. Values of this argument are in hex format and are formed by logically OR'ing zero or more of the following flags: 0x100 Exception Message: Send messages with high priority 0x200 Release Indicator: indicate release after next message 0x400 Release Indicator: indicate release after next messages has been replied to If no flags are set, a value of 0 should be provided
<length>	Integer type 1-1400 Decimal length of data to be sent
<data>	String type data be sent in hex string format
<sequence>	Integer type 1-255 If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported: +ECSOSTR: <socket_id>,<sequence>,<status>
<status>	Integer type 2 the sent status of datagram is fail 3 the sent status of datagram is success

Example

```
AT+ECSOSTFT=1,"180.167.122.150",5002,0x100,2
>ABAB
1,2
OK

AT+ECSOSTFT=1,"180.167.122.150",5002,0x100
>ABAB
1A
1,2
OK
```

3.9.6 AT+ECQSOS

The command queries the list of the pending upstream message.

AT+ECQSOS

Set Command	Response
AT+ECQSOS=<socket_id>[,<socket_id>[,<socket_id>[,<socket_id>[...]]]]	[+ECQSOS:<socket_id>,<sequence>]

	[...] OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECQSOS=?	Response +ECQSOS: (list of supported <socket_id>) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE
Parameter	
<socket_id>	Integer type 0-9 Socket Id returned by AT+ECSOCR
<sequence>	Integer type 1-255 If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported: +ECSOSTR: <socket_id>,<sequence>,<status>

Example

```
AT+ECQSOS=1
+ECQSOS:1,2
OK
```

3.9.7 AT+ECSORF

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

The command is use to receive data on a socket. When data arrives, a "+ECSONMI" response will be generated to indicate the socket the message was received on and also the amount of data. The AT+ECSORF command takes a length, which is the maximum amount of data that will be returned. If the requested length is larger than the actual size of the returned data, only the length of returned data is provided, and the remaining length is returned as 0. If the requested length is less than the amount of data returned, only the requested amount of data will be returned, plus an indication of the number of bytes remaining. Once a message has been fully read, a new "+ECSONMI" notification will be sent if there is another message to process.

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

AT+ECSORF

Set Command AT+ECSORF=<socket_id>,<req_length>	Response <socket_id>,<ip_addr>,<port>,<length>,<data>,<remaining_length>
---	---

	OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSORF=?	Response +ECSORF: (list of supported <socket_id>), (list of supported <req_length>) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<socket_id>	Integer type Socket Id returned by AT+ECSOCR
<ip_addr>	String type Remote IP address
<port>	Integer type 0-65535 This is the remote port on which messages will be sent from
<req_length>	Integer type 1-1357 Decimal length of data which want to read
<length>	Integer type 1-1358 Decimal length of data to be read
<data>	String type data be sent in hex string format
<remaining_length>	Integer type 0-1357 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+ECSORF=1,4
1,"180.167.122.150",5002,4,"ABABABAB",0
OK
```

3.9.8 AT+ECSOCO

The command connect a TCP server to the specified host and port. if the socket connection connect success, it will indicate the success result with the URC "+ECSOCO". If the socket connection connect fail, it will indicate the fail result with the URC "+ECSOCLI"

AT+ECSOCO

Set Command AT+ECSOCO=<socket_id>,<remote_addr>,<remote_port>	Response OK If there is any error, response:
---	---

	+CME ERROR: <err>
Test Command AT+ECSOCO=?	Response +ECSOCO: (list of supported <socket_id>), (list of supported <remote_addr>),(list of supported <remote_port >) OK
Maximum Response Time	30s
Parameter Saving Mode	SAVE

Parameter

<socket_id>	Integer type	
	0-9	Socket Id returned by AT+ECSOCR
<remote_addr>	String type	
		The remote IP address
<remote_port>	Integer type	
	0-65535	This is the remote port to be connected to

Example

```
AT+ECSOCO=1,"180.167.122.150",5002
OK
+ECSOCO: 1
```

3.9.9 AT+ECSOSD

The command sends a TCP datagram to the TCP server. It will return with the socket that it was sent on, and the number of bytes of data sent. If the amount of data is larger than the largest datagram that can be sent, then AT+ECSOSD return value will indicate how much the data was successfully sent.

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

AT+ECSOSD

Set Command AT+ECSOSD=<socket_id>,<length>,<data>[,<flag>[,<sequence>]]	Response <socket_id>,<length> OK If there is any error, response: +CME ERROR: <err>
Test Command	Response

AT+ECSOSD=?	+ECSOSTF: (list of supported <socket_id>), (list of supported <length>),(<data>),(list of supported <flag>) (list of supported <sequence>) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<socket_id>	Integer type 0-9 Socket Id returned by AT+ECSOCR
<length>	Integer type 1-1400 Decimal length of data to be sent
<data>	String type data be sent in hex string format
<flag>	Integer type Specifies the type of message transmission. Values of this argument are in hex format and are formed by logically OR'ing zero or more of the following flags: 0x100 Exception Message: Send messages with high priority 0x200 Release Indicator: indicate release after next message 0x400 Release Indicator: indicate release after next messages has been replied to If no flags are set, a value of 0 should be provided
<sequence>	Integer type 1-255 If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported: +ECSOSTR: <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+ECSOSD=1,2,"ABAB"
```

```
1,2
```

```
OK
```

3.9.10 AT+ECSOSDT

The command sends a TCP datagram to the TCP server with transparent mode. It will return with the socket that it was sent on, and the number of bytes of data sent. If the amount of data is larger than the largest datagram that can be sent, then AT+ECSOSDT return value will indicate how much the data was successfully sent.

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

If the command contain the parameter of “length”, It need input enough data with hex string format after “>” output. If the command did not contain the parameter of “length”, the input of data must end with “CTRL+Z”.

AT+ECSOSDT

Set Command AT+ECSOSDT=<socket_id>[,<length>[,<flag>[,<sequence>]]]	Response <socket_id>,<length> > OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSOSDT=?	Response +ECSOSDT: (list of supported <socket_id>), (list of supported <length>),(list of supported <flag>) (list of supported <sequence>) OK
Maximum Response Time	60s
Parameter Saving Mode	SAVE
Parameter	
<socket_id>	Integer type 0-9 Socket Id returned by AT+ECSOCR
<length>	Integer type 1-1400 Decimal length of data to be sent
<data>	String type data be sent in hex string format
<flag>	Integer type Specifies the type of message transmission. Values of this argument are in hex format and are formed by logically OR'ing zero or more of the following flags: 0x100 Exception Message: Send messages with high priority

	0x200	Release Indicator: indicate release after next message
	0x400	Release Indicator: indicate release after next messages has been replied to
	If no flags are set, a value of 0 should be provided	
<sequence>	Integer type	
	1-255	If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported: +ECSOSTR: <socket_id>, <sequence>, <status>
<status>	Integer type	
	2	the sent status of datagram is fail
	3	the sent status of datagram is success

Example

```
AT+ECSOSDT=1,2
```

```
>ABAB
```

```
1,2
```

```
OK
```

```
AT+ECSOSDT=1
```

```
>ABAB
```

```
1A
```

```
1,2
```

```
OK
```

3.9.11 AT+ECISOCL

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

AT+ECISOCL

Set Command AT+ECISOCL=<socket_id>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECISOCL=?	Response +ECSOSTF: (list of supported <socket_id>) OK
Maximum Response Time	5s

Parameter Saving Mode	SAVE
Parameter	
<socket_id>	Integer type
	0-9 Socket Id returned by AT+ECSOCL

Example

```
AT+ECSOCL=1
OK
```

3.9.12 AT+ECSONMI

The write command is used to set the unsolicited result code "+ECSONMI" to indicate arrived socket messages (the socket were not configure as private socket by AT+ECSONMIE command):

If <mode>=1, the UE will receive an unsolicited result code: "+ECSONMI: <socket>,<length>".

If <mode>=2, the UE will receive an unsolicited result code: "+ECSONMI: <socket>,<remote_ad dr>,<remote_port>,<length>,<data>".

If <mode>=3, the UE will receive an unsolicited result code: "+ECSONMI: <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

AT+ECSONMI

Set Command AT+ECSONMI=<mode>[,<max_public_dl_buffer>[,<max_public_dl_pkg_num>]]	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSONMI=?	Response +ECSONMI: (list of supported <mode>s), (list of supported <max_public_dl_buffer>),(list of supported <max_public_dl_pkg_num >) OK
Read Command AT+ECSONMI?	Response +ECSONMI:<mode>,<max_public_dl_buffer >,<max_public_dl_pkg_num> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<mode>	Integer type
	Control downlink data format.

	0	Disable indication messages unsolicited result code.
	1	Enable indication messages unsolicited result code: "+ECSONMI: <socket_id>,<length>"
	2	Enable indication messages unsolicited result code: "+ECSONMI: <socket_id>,<remote_addr>,<remote_port>,<length>,< data>
	3	Enable indication messages unsolicited result code: "+ECSONMI: <socket_id>,<length>,<data>"
<socket_id>	Integer type	
	0-9	Socket Id returned by AT+ECSOCR
<length>	Integer type	
	1-1358	Number of bytes of data in the first message
<remote_addr>	string type	
		The remote IP address which the message is sent from
<remote_port>	Integer type	
	0-65535	The remote port which the message if sent from
<data>	string type	
		Data received in hex string format. Maximum length of received data is 1358 bytes
<max_public_dl_buffer >	Integer type	
	1358-3072	The maximum downlink buffer total size for all sockets created by AT+ECSOCR which is not configure as private mode socket with the command "AT+ECSONMIE". The default value is 2048
<max_public_dl_pkg_num>	Integer type	
	8-16	The maximum downlink buffer message total number for all sockets created by AT+ECSOCR which is not configure as private mode socket with the command "AT+ECSONMIE". The default value is 8

Example

```
AT+ECSONMI=2,1500,9
```

```
OK
```

3.9.13 AT+ECSONMIE

The write command is used to set the unsolicited result code "+ECSONMI" to indicate arrived messages of a specified socket, the default mode is 3:

If <mode>=1, the UE will receive an unsolicited result code: "+ECSONMI: <socket>,<length>".

If <mode>=2, the UE will receive an unsolicited result code: "+ECSONMI: <socket>,<remote_addr>,<remote_port>,<length>,<data>".

If <mode>=3, the UE will receive an unsolicited result code: "+ECSONMI: <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

AT+ECSONMIE

Set Command AT+ECSONMIE=<socket_id>,<mode>[,<max_private_dl_buffer>[,<max_private_dl_pkg_num>]]	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSONMIE=?	Response +ECSONMIE: (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
Read Command AT+ECSONMIE?	Response [+ECSONMIE: <socket_id>,<mode>,<max_public_dl_buffer >,<max_public_dl_pkg_num>] [...] OK
Maximum Response Time	5s
Parameter Saving Mode	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: "+ECSONMI: <socket_id>,<length>" 2 Enable indication messages unsolicited result code for the specified socket: "+ECSONMI: <socket_id>,<remote_addr>,<remote_port>,<length>,<data>" 3 Enable indication messages unsolicited result code for the specified socket: "+ECSONMI:<socket_id>,<length>,<data>" 255 reset the mode setting which configure by the command "AT+ECSONMI" for the speceified socket
<socket_id>	Integer type 0-9 Socket Id returned by AT+ECSOCR

<length>	Integer type 1-1358	Number of bytes of data in the first message
<remote_addr>	string type	The remote IP address which the message is sent from
<remote_port>	Integer type 0-65535	The remote port which the message if sent from
<data>	string type	Data received in hex string format. Maximum length of received data is 1358 bytes
<max_private_dl_buffer>	Integer type 1358-2048	The maximum downlink buffer size for the specified socket. The default value is 1358
<max_private_dl_pkg_num>	Integer type 1-8	The maximum downlink buffer message number for the specified socket. The default value is 4

Example

```
AT+ECSONMIE=2,1500,6
OK
```

3.9.14 +ECSOCLI

This is an unsolicited message to notify that a socket has been closed. It returns the socket number and error code

+ECSOCLI

```
+ECSOCLI: <socket_id>,<errno>
```

Parameter

<socket_id>	Integer type 0-9	Socket Id returned by AT+ECSOCL
<errno>	Integer type (Posix Errno defines)	
	12	Out of memory error
	105	No buffer space available
	62	Timer expired
	113	No route to host
	115	Operation now in progress
	22	Invalid argument
	11	Operation would block
	107	Transport endpoint is not connected
	103	Software caused connection abort
	104	Connection reset by peer

Example

```
+ECSOCLI:1,104
```

3.9.15 +ECSOSTR

This is an unsolicited message to notify that one uplink datagram sent status with sequence
+ECSOSTR

```
+ECSOSTR: <socket_id>,<sequence>,<status>
```

Parameter

<socket_id>	Integer type	
	0-9	Socket Id returned by AT+ECSOCR
<sequence>	Integer type	
	1-255	If it is omitted, data sent will not be reported. If not omitted, when datagram is sent over RF or is discarded, then the result will be reported
<status>	Integer type	
	0	the sent status of datagram is fail
	1	the sent status of datagram is success

Example

```
+ECSOSTR:1,101,1
```

3.9.16 +ECSOCO

This is an unsolicited message to notify that the TCP connection has connected success
+ECSOCO

```
+ECSOSO: <socket_id>
```

Parameter

<socket_id>	Integer type	
	0-9	Socket Id returned by AT+ECSOCR

Example

```
AT+ECSOCO=1,"180.167.122.150",5002
```

```
OK
```

```
+ECSOCO:1
```

3.9.17 Error code for socket command(solution B)

Code of <err>	Description
402	Parameter invalid
403	Too much socket instance
404	Create socket error
405	operation not supported
406	Cannot find the socket
407	Socket Connect fail
408	Socket bind fail
409	Send data fail
410	The socket status is not connected
411	The socket status is already connected
412	The socket status is invalid
413	The socket connect timeout
414	The socket close fail
415	The socket happen fatal error
416	Can not allocate more memory
417	No more DL buffer resource
418	The socket is connecting
419	UL sequence is invalid
420	Send request fail
421	Unknown error

3.10 Socket Command(TCP Server Mode)

3.10.1 AT+ECSRVSOCRTCP

The command creates a TCP server socket on the UE and associates with specified protocol. If the TCP server socket create success, "+ECSRVSOCRTCP" unsolicited messages with "socket_id" will appear.

The TCP server socket can accept no more than three TCP clients, and the total number of TCP server and TCP client must be not bigger than nine.

If the remote TCP client connect the TCP server success, "+ECACCEPTTCPCLIENTSOCKET" unsolicited messages will appear with the TCP client information.

If the TCP server socket create repeatedly, It will notify that the TCP server socket has create with the URC "+ECSRVSOCRTCP".

AT+ECSRVSOCRTCP

Set Command	Response
AT+ECSRVSOCRTCP=<listen_port>[,<af_type>[,<ip_address>]	OK

]	+ECSRVSOCRTCP: <socket_id>,OK ,OK If there is any error, response: +CME ERROR: <err>
---	---

Test Command AT+ECSRVSOCRTCP=?	Response +ECSRVSOCRTCP: (list of supported <listen_port>), (list of supported< af_type>),(ip_addr>) OK
--	---

Maximum Response Time	5s
------------------------------	-----------

Parameter Saving Mode	SAVE
------------------------------	-------------

Parameter	
<listen_port>	Integer type A number in the range of 1-65535. This is the local port that will be included in sent messages and on which port will the TCP client connect.
<af_type>	String type AF_INET the default value. IPv4 AF_INET6 IPv6
<ip_address>	String type IP address. The IP address of the network assigned to UE.
<socket_id>	Integer type 0-9 A maximum of 5 sockets are supported, but other serviced may reduce this number.

Example

```
AT+ECSRVSOCRTCP=11111,"AF_INET"
OK
+ECSRVSOCRTCP: 1,OK

AT+ECSOCR="STREAM",6,22222,1,"AF_INET"
2
OK

AT+ECSOCO=2,"127.0.0.1",11111
OK
+ECSOCO: 2

+ECACCEPTTCPCLIENTSOCKET:1,3,127.0.0.1,22222
```

3.10.2 AT+ECSRVSOCCLTCPLISTEN

The command close a TCP server socket on the UE. When the TCP server socket close, the TCP clients which accept by the TCP server will be close auto.

AT+ ECSRVSOCCLTCPLISTEN

Set Command AT+ECSRVSOCCLTCPLISTEN=<socket_id>	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSRVSOCCLTCPLISTEN=?	Response +ECSRVSOCCLTCPLISTEN: (list of supported <socket_id> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE
Parameter	
<socket_id>	Integer type 0-9 A maximum of 7 sockets are supported, but other serviced may reduce this number.

Example

```
AT+ECSRVSOCRTCP=11111,"AF_INET"
OK
+ECSRVSOCRTCP: 1,OK

AT+ECSRVSOCCLTCPLISTEN=1
OK
```

3.10.3 AT+ECSRVSOCCLTCPCLIENT

The command close a TCP client connection which accept by TCP server socket.

If the command contain the parameter of "client_socket_id", it just close one TCP client connection.

If the command did not contain the parameter of "client_socket_id", it just close all TCP client connections which accept by the TCP server.

AT+ ECSRVSOCCLTCPCLIENT

Set Command AT+ECSRVSOCCLTCPCLIENT=<server_socket_id>[,<client_socke	Response OK
--	-----------------------

t_id>]	If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSRVSOCLETCCLIENT=?	Response +ECSRVSOCLETCCLIENT: (list of supported <server_socket_id>), (list of supported <client_socket_id>)) OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE
Parameter	
<server_socket_id> Integer type	
0-9	Socket Id returned by AT+ECSRVSOCRTCP
<client_socket_id> Integer type	
0-9	Socket Id returned by +ECACCEPTTCCLIENTSOCKET

Example

```
AT+ECSRVSOCLETCCLIENT=1,3
OK
```

3.10.4 AT+ECSRVSOTCPSENDCLT

Send a containing length bytes of data to the TCP client connection which accept by the TCP server socket.

AT+ECSRVSOTCPSENDCLT

Set Command AT+ECSRVSOTCPSENDCLT=<socket_id>,<data len>,<data>[,<rai info>[,<except info>]]	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSRVSOTCPSENDCLT=?	Response +ECSRVSOTCPSENDCLT: (list of supported <socket_id>s), (list of supported <data len>s), (<data>), (list of supported <rai info>s), (list of supported <expect info>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type
0-9	Socket Id returned by +ECACCEPTTCPCLIENTSOCKET
<data len>	Integer type
	length of data in hex string format, the max length is 1400
<data>	Integer type
	Data in hex string
<rai info>	Integer type (option)
0-2	release assistance indication
6	no rai info
7	no further uplink or downlink data transmission subsequent to the uplink data transmission subsequent to the uplink data transmission is expected
8	only a single downlink data transmission and no further uplink data transmission subsequent to the uplink data transmission is expected
	Note: default value: 0
<except info>	Integer type (option)
0-3	expect data indication
4	disable expect data indication
5	enable expect data indication
	Note: default value: 0

Example

Send data:23456

```
AT+ECSRVSOTCPSENDCLT=0,5,3233343536
```

```
OK
```

3.10.5 AT+ECSRVSOTCPSENDCLTT

Send a containing length bytes of data with transparent mode to the TCP client connection which accept by the TCP server socket.

If the command contain the parameter of "data_len", It need input enough data with hex string format after ">" output. If the command did not contain the parameter of "date_len", the input of data must end with "CTRL+Z".

AT+ECSRVSOTCPSENDCLT

Set Command AT+ECSRVSOTCPSENDCLT=<socket_id>[,<data len>[,<rai info>[,<except info>]]]	Response OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSRVSOTCPSENDCLT=?	Response +ECSRVSOTCPSENDCLT: (list of supported <socket_id>s), (list of supported <data

	len>s)(list of supported <rai info>s), (list of supported <expect info>s) OK
Maximum Response Time	5s
Parameter Saving Mode	NO_SAVE

Parameter

<socket_id>	Integer type 0-9 Socket Id returned by +ECACCEPTTCPCLIENTSOCKET
<data len>	Integer type length of data in hex string format, the max length is 1400
<data>	Integer type Data in hex string
<rai info>	Integer type (option) 0-2 release assistance indication 9 no rai info 10 no further uplink or downlink data transmission subsequent to the uplink data transmission subsequent to the uplink data transmission is expected 11 only a single downlink data transmission and no further uplink data transmission subsequent to the uplink data transmission is expected Note: default value: 0
<except info>	Integer type (option) 0-4 expect data indication 6 disable expect data indication 7 enable expect data indication Note: default value: 0

Example

Send data:23456

```
AT+ECSRVSOTCPSENDCLT=1,5
>3233343536
OK

AT+ECSRVSOTCPSENDCLT=1
>3233343536
1A
OK
```

3.10.6 AT+ECSRVSOTCPLISENSTATUS

The command query the TCP server socket status.

If the command contain the parameter of "socket_id", it will return the TCP socket status by the URC

“+ECSRVSOTCPLISTENSTATUS”.

AT+ ECSRVSOTCPLISTENSTATUS

Set Command AT+ECSRVSOTCPLISTENSTATUS=<socket_id>	Response +ECSRVSOTCPLISTENSTATUS : <socket_id>,<status> OK If there is any error, response: +CME ERROR: <err>
Test Command AT+ECSRVSOTCPLISTENSTATUS=?	Response +ECSRVSOTCPLISTENSTATUS : (list of supported <socket_id>) OK
Read Command AT+ECSRVSOTCPLISTENSTATUS?	Response +ECSRVSOTCPLISTENSTATUS : <socket_id>,<status> OK
Maximum Response Time	5s
Parameter Saving Mode	SAVE

Parameter

<socket_id>	Integer type 0-9 A maximum of 7 sockets are supported, but other serviced may reduce this number.
<status>	string Listening the TCP server socket is listening Not Listening the TCP server socket is not listening

Example

```
AT+ECSRVSOTCPLISTENSTATUS=1
+ECSRVSOTCPLISTENSTATUS: 1,Listening
OK
```

3.10.7 +ECACCEPPTCPCLIENTSOCKET

This is an unsolicited message to notify that one TCP client is accepted by TCP server

+ECACCEPPTCPCLIENTSOCKET

+ECACCEPPTCPCLIENTSOCKET: <server_socket_id>,<client_socket_id>,<client_ip_addr>,<client_port>	
---	--

Parameter

<server_socket_id>	Integer type 0-9 Socket Id returned by AT+ECSRVSOCRTCP
--------------------	--

<client_socket_id>	Integer type 0-9 Socket Id returned by +ECACCEPTTCPCLIENTSOCKET
<client_ip_addr>	String type The IP address of the TCP client
<client_port>	Integer type 1-65535 The port number of the TCP client

Example

```
AT+ECSRVSOCRTCP=11111,"AF_INET"
OK
+ECSRVSOCRTCP: 1,OK

AT+ECSOCR="STREAM",6,22222,1,"AF_INET"
2
OK

AT+ECSOCO=2,"127.0.0.1",11111
OK
+ECSOCO: 2

+ECACCEPTTCPCLIENTSOCKET:1,3,127.0.0.1,22222
```

3.10.8 +ECSRVSOTCPCLTRCV

This is an unsolicited message to notify that TCP server receive some packet from the TCP client

+ECSRVSOTCPCLTRCV

```
+ECSRVSOTCPCLTRCV: <client_socket_id>,<length>,<data>
```

Parameter

<client_socket_id>	Integer type 0-9 Socket Id returned by +ECACCEPTTCPCLIENTSOCKET
<length>	Integer type 1-1358 The length of receiving data
<data>	Hex string type The receiving data

Example

```
AT+ECSRVSOCRTCP=11111,"AF_INET"
OK
+ECSRVSOCRTCP: 1,OK

AT+ECSOCR="STREAM",6,22222,1,"AF_INET"
```

```

2
OK

AT+ECSOCO=2,"127.0.0.1",11111
OK
+ECSOCO: 2

+ECACCEPTTCPCLIENTSOCKET:1,3,127.0.0.1,22222

AT+ECSOSD=2,2,"1111"
2,2
OK

+ECSRVOTCPCLTRCV:3,2,"1111"

```

3.10.9 +ECSRVSOCCLIENTCPCLOSE

This is an unsolicited message to notify that TCP client is closed

+ECSRVSOCCLIENTCPCLOSE

+ECSRVSOCCLIENTCPCLOSE: <client_socket_id>,<errno>

Parameter

Parameter	Type	Description
<client_socket_id>	Integer type	Socket Id returned by AT+ECSOCR
	0-9	
<errno>	Integer type (Posix Errno defines)	
	12	Out of memory error
	105	No buffer space available
	62	Timer expired
	113	No route to host
	115	Operation now in progress
	22	Invalid argument
	11	Operation would block
	107	Transport endpoint is not connected
	103	Software caused connection abort
	104	Connection reset by peer

Example

```

AT+ECSRVSOCRTCP=11111,"AF_INET"
OK
+ECSRVSOCRTCP: 1,OK

AT+ECSOCR="STREAM",6,22222,1,"AF_INET"

```

```

2
OK

AT+ECSOCO=2,"127.0.0.1",11111
OK
+ECSOCO: 2

+ECACCEPTTCPCLIENTSOCKET:1,3,127.0.0.1,22222

AT+ECSOCL=2
OK

+ECSRVSOCCLIENTCPCLOSE:3,107

```

4 Error Values

If the AT command not implemented or format dose not match, it will output "ERROR". For general control commands compliant with the 3Gpp specifications. Please refer to 3GPP TS 27007 V14.5.0, sub-clause 9.2 for all possible <err> values. If an error occurs, it will output "+CME ERROR: <err>". Some common values are listed in the table below.

General Errors(27.007)

Code of <err>	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 support
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	Mission 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 packet filters
145	Syntactical errors in packet filters
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 non existent 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 via 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
309	PDN type and APN duplicate used

For general control commands compliant with 3GPP TS 27005. If an error occurs, it will output

“+CMS ERROR: <err>”. Some common values are listed in the table below.

General Errors(27.005)

Code of <err>	Description
300	ME failure
301	SMS service of ME reserved
302	operation not allowed
303	operation not supported
304	invalid PDU mode parameter
305	invalid text mode parameter
310	USIM not inserted
311	USIM PIN required
312	PH-(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 acknowledgement expected
500	unknown error

Annex A : Change history

Date	Subject/Comment	New version	Owner
2018-06-05	Initial	1.1	
2019-06-05	Change some AT's type	1.2	
2019-06-11	Add some AT and Chapter 4	1.3	
2019-08-30	Modify font format	1.4	
2019-12-10	Add some OneNET AT's description, change HTTPCREATE's description	1.5	
2020-01-14	Add AT commands: AT+CCHO, AT+CCHC, AT+CGLA,	1.6	xlhu

	AT+ECSIMSLLEP		
2020-02-19	Delete section: 3.1.38: Summary of <err> Codes	1.7	jcweng
2020-02-20	1. Add section: 3.1.38: +ECPIN 2. Modify AT+ECURC for single urc control 3. Add section: 2.2.49: AT+CPINR	1.7	xlhu
2020-02-20	1. Modify some coap command	1.7	yrhao
2020-02-24	1. Add section: 3.1.39: +ECPCFUN 2. Add section: 3.1.40: +ECPADDR 3. Modify AT+ECURC for +ECPCFUN/+ECPADDR	1.7	xlhu
2020-03-05	Add section 3.1.41 AT+ECADC	1.8	wl
2020-03-07	Update max response time for +COPS/+CGATT/ +ECBCINFO	1.8	xlhu
2020-03-11	Add section 3.1.42 AT+ECSWC	1.8	xlhu
2020-03-20	Modify section 3.1.6 AT+ECFREQ	1.8	flxing
2020-03-25	Add section 3.1.43 AT+ECSNTP	1.9	bchang
2020-04-01	1.Add section 2.2.50 AT+CGAUTH 2.Modify ECATTBEARER for PAP	1.9	jllwang
2020-04-02	1.Add section 2.2.51 AT+IPR 2.Add section 3.1.44 AT+ECIPR	1.10	wl
2020-04-03	1.Add section 3.1.45 AT+ECNPICFG	1.10	wl
2020-04-07	1. Modify section 3.1.3 AT+ECCFG	1.10	fwang
2020-04-09	Update AT+ECSWC	1.10	xlhu
2020-04-15	Modify AT+ECFREQ Parameter Saving Mode description	1.10	flxing
2020-04-16	Add AT+ECPCFG faultaction description	1.10	bchang
2020-04-16	Modify AT+ECATTBEARER parameter APN Length discription	1.10	jllwang
2020-04-21	1. Modify section 3.1.3 AT+ECCFG for configure DisableNCellMeas feature	1.10	fwang
2020-04-26	Modify section 3.1.11 AT+ECPMUCFG, change default value	1.10	bchang
2020-04-27	1. Fix typo in section 3.1.45 AT+ECNPICFG 2. Add section 3.1.46 AT+ECPRODMODE	1.11	wl
2020-04-30	1. Modify section 3.1.3 AT+ECCFG for configure NbCategory value	1.11	fwang
2020-04-30	Modify the max value from 950 to 1400 for AT socket sending data length	1.11	Xwang
2020-04-30	Disable PDU mode of SMS related command	1.11	xpliu
2020-05-06	Modify section 3.1.3 AT+ECCFG for: 1. Modify default value of RelaxMonitorDeltaP from 0 to 6 2. Modify NbCategory description	1.11	yyzhao
2020-05-11	Modify AT+CGDCONT parameter APN Length	1.11	jllwang

	discription		
2020-05-15	Modify 3.1.19 AT+ECSTATUS: Add taValue and txPower.	1.11	flxing
2020-05-15	Modify some OneNET AT, oceanConnect AT and LwM2M AT's maximum response time Add LwM2M AT's error codes	1.11	yjzhong
2020-05-19	Add section 3.1.47 AT+ECPOWERCLASS	1.11	flxing
2020-05-22	Modify AT+CPSMS/AT+CGEQOS test example	1.11	jlwang
2020-05-25	Add error code 309 - Pdn type and APN duplicate used	1.11	jlwang
2020-05-27	Update AT+ECCFG: 1. Add configuration parameters limit for single AT 2. Mod DataInactTimer support value: 0, (15-255)	1.11	xlhu
2020-05-29	Update AT+ECIPERF: 1. The Payload size description 2. TPT description	1.11	xwang
2020-06-01	Update 1. section 2.2.11 AT+CEDRXS eDRX setting parameter <AcT- type> 2. section 3.1.36 AT+ECURC read and test command	1.11	flxing
2020-06-03	Add section 2.2.52 AT+CNMPD	1.11	fwang
2020-06-05	Modify AT+CGDCONT/AT+CGCONTRDP test example	1.11	jlwang
2020-06-09	Update 2.2.13 AT+CCIOPT Add comments of parameter <supported_Network_opt>	1.11	flxing
2020-06-11	Add AT+DMREGSTAT	1.11	yjzhong
2020-06-15	Update ECSTATUS, add AnchorFreqOfst and NonAnchorFreqOfst	1.11	zifu
2020-06-17	Modify CGAUTH description and test example	1.11	jlwang
2020-06-18	Update AT+ECSTATUS with <module> display.	1.11	xlhu
2020-06-18	Add section 3.1.48 AT+ECPMUSTATUS	1.11	wl
2020-06-20	Modify AT+ECSNTP description: The default value of "autosync" change from 1 to 0	1.11	xwang
2020-06-22	Fix AT+ECDNSCFG only set ipv6 address fail issue	1.11	xwang
2020-06-23	Add "rai_enable" parameter for AT+ECPING command	1.11	xwang
2020-06-28	Modify AT+ECCFG parameter T3324MaxValueS comment	1.11	flxing
2020-06-28	Add AT CMD: AT+ECSTATIS	1.11	jcweng
2020-07-03	1. Add section 3.1.50 AT+ECFSINFO 2. Add section 3.1.51 AT+ECFLASHMONITORINFO	1.12	wl
2020-07-08	As PDP CID for attached bearer (initial PDP) has changed from 5 to 0, correct some AT description	1.12	jcweng
2020-07-10	Add section 3.1.52 AT+ECPURC	1.12	wl
2020-07-13	Add oceanconnect command error code	1.12	yjzhong
2020-07-16	Modify AT socket fd/id value description.	1.12	xwang
2020-07-17	1. Add section 2.2.53 AT+ICF	1.12	wl

	2. Add a note into 2.2.51 AT+IPR for the case that if AT+IPR=0 is set, character framing is forced to be 8 data, none parity, 1 stop		
2020-07-17	Add section 3.1.53 AT+ECFSFORMAT	1.12	wl
2020-07-27	Modify AT socket connection MAX number from 5 to 7	1.12	xwang
2020-07-27	Update section 3.1.21 AT+ECBCINFO, add tac parameter	1.12	fwang
2020-07-27	Add +ECSOCO URC	1.12	xwang
2020-07-28	Modify cid max value to 10	1.12	jlwang
2020-08-12	Modify +CIPCA param1 value arrange;	1.12	jlwang
2020-08-13	Update section 3.1.3 AT+ECCFG, add Ipv6RsForTestSim configuration	1.12	fwang
2020-08-19	Add AT+ECSIMPD	1.12	xlhu
2020-08-20	Update +ECSNTP URC format	1.12	xwang
2020-08-26	Add 2.2.54 AT+CNUM	1.12	xlhu
2020-09-01	Extended AT command: AT+ECBCINFO	1.12	jcweng
2020-09-10	Add section 3.1.56 AT command: AT+ECNBIOTRAI	1.12	fwang
2020-09-17	Add section 3.1.57 AT command: AT+ECNBR14	1.12	jcweng
2020-09-18	Change AT+CTM2MSETPMOD parameter's value	1.12	yjzhong
2020-09-21	Add at socket send with transparent mode	1.12	xwang
2020-09-21	Add at socket with TCP server mode	1.12	xwang
2020-09-22	Add section 3.1.58 AT command: AT+ECEVENTSTATIS	1.12	eysun
2020-09-23	Add description for AT+ECSIMSLEEP=?/AT+ECSWC=?/AT+ECSIMPD=?	1.12	xlhu
2020-09-27	Modify HTTP command	1.12	bchang
2020-11-04	Add section 2.2.55 AT command: AT+CEER	1.12	fwang
2020-11-18	Update AT+ECSIMSLEEP description	1.12	xlhu
2020-12-17	Enhance ECIPERF description	1.12	xwang
2021-01-29	Add 3.1.60 AT+ECUSATP	1.12	xlhu
2021-02-20	Add China Telecom AT at+ctm2mrmode and at+ctm2mread	1.12	yjzhong
2021-02-26	Add AT+ECCFG parameter PsSoftReset and BarValueS comment	1.12	flxing
2021-02-26	Add 3.1.61 AT+ECPSSLPCFG	1.12	flxing
2021-03-09	Update 3.1.12 AT+ECSMSEND example	1.12	flxing
2021-04-02	Update AT+ECCFG parameter "PlmnSearchPowerLevel" comment	1.12	flxing