neoway

Get connected Get smart

N725

AT Commands Manual

Issue 1.2 Date 2023-07-11

Notice

This document is specifically for N725.

This document is intended for system engineers (SEs), development engineers, and test engineers.

THIS DOCUMENT PROVIDES SUPPORT FOR PRODUCT DESIGN BY THE USER. THE USER MUST DESIGN AND DEBUG THE PRODUCT ACCORDING TO THE SPECIFICATIONS AND PARAMETERS IN THIS DOCUMENT. NEOWAY ASSUMES NO RESPONSIBILITY FOR PERSONAL INJURY AND PROPERTY LOSS CAUSED BY USER'S IMPROPER OPERATION.

DUE TO PRODUCT VERSION UPDATES OR OTHER REASONS, THE CONTENT OF THIS DOCUMENT WILL BE UPDATED AS NECESSARY WITHOUT PRIOR NOTICE.

UNLESS OTHERWISE AGREED, ALL STATEMENTS, INFORMATION AND SUGGESTIONS IN THIS DOCUMENT CONSTITUTE NO WARRANTY WHETHER EXPRESS OR IMPLIED.

Neoway Technology Co., Ltd. provides customers with comprehensive technical support. For any inquiry, please contact your account manager directly or send an email to the following email address:

Sales@neoway.com

Support@neoway.com

Website: http://www.neoway.com

Contents

About This Document

Scope

This document is applicable to N725.

Audience

This document is intended for system engineers (SEs), development engineers, and test engineers.

Change History

Issue	Date	Change	Changed By
1.0	2022-12	• Initial draft	Song Haiqing
1.1	2023-01	• Added the Ecall commands	Dai Lintao
	2023-02	• Deleted some AT commands not supported by the module.	Song Haiqing
	2023-02	• Deleted the +NEOFTPFOTA command • Deleted SD card related command examples • Added the +PDPKEEPALIVE test command • Added the +DNSSERVER command example • Modified the +SSLTCPSEND command examples • Modified the +CERTADD command examples • Added a command example for AT+XISP=?	Li ChuYu XueJianyun Sun zhen
	2023-02	• Modified the AT+UDPTRANS examples • Modified the AT+IPSTATUS examples • Modified the AT+TRANSCLOSE examples • Modified the AT+TCPACK examples • Modified the AT+TCPTRANS examples • Modified the AT+UDPTRANS AT+TCPTRANS AT+TCPSETUP, AT+UDPSETUP • Added AT+UDPSEND parameter3 and parameter4, and modified the range of length; • Modified the AT+UDPSEND examples • Modified the AT+CTZR examples	Guo Yaobin

		• Added AT+TCPSEND parameter3 and parameter4, and modified the range of the parament length;	
	2023-02	• Modified the AT+TCPSETUP examples • Modified the AT+RECVMODE examples • Modified the AT$MYTYPE examples • Modified the AT$MYMODEM examples • Modified the AT+CMUX examples • Modified the AT+CGMM examples • Modified the AT+GMR Response • Modified the AT+CGSN format • Modified the AT+CPIN examples	SunZhen Zhao Guiqiang
1.2	2023-03	• Added the eCall commands	Xuebowen
	2023-04	• Modified the AT$MYGMR examples • Modified the AT+TCPSEND examples • Added +AT+SIGNAL examples • Added the +CGMR command	SunZhen XueJianyun
	2023-04	• Added +CME and +CMS error codes	Xuebowen
	2023-05	• Modified the AT+COPS format • Modified the AT$MYNETINFO examples • Modified the AT+UDPSETUP format,parameters and examples • Added +FTPLOGIN command • Added +FTPLOGOUT command • Added +FTPGET command • Added +FTPPUT command • Added +FTPSTATUS command • Added +NEOFTPFOTA command • Added +CIPGSMLOC command	Cuilisha ZhaoGuiqiang Xuejianyun YuHaijun Xuejianyun
	2023-05	• Modified the AT+UDPSETUP examples • Modified the AT+TCPTRANS examples • Modified the AT+UDPTRANS examples • Modified the AT+SSLTCPSETUP examples	ZengHuangming
	2023-05	• Modified the AT command about GPS	Yanhaifeng
	2023-05	• Added +UPDATETIME command	XueJianyun
	2023-05	• Modified the AT+UDPTRANS example • Modified the AT+TCPACK format • Modified the AT+IPSTATUS format • Deleted the AT+FSFAT parameter • Modified the AT+PDPKEEPALIVE format	lichuyu
	2023-05	• Modified the AT+NEOFTPFOTA example	YuHaijun
	2023-05	• Modified the AT+NWECALLCFG example	DaiLintao
	2023-05	• Modified the AT+TCPACK example	Guoyaobin

2023-05	• Modified the AT+IPSTATUS=<n> example	ZhaoGuiqiang
2023-06	• Added the AT+TCPKEEPALIVE command • Modified the AT+SSLTCPSEND=? format • Modified the AT+UDPREAD format • Added the AT+CERTDEL command response • Modified the AT+CREG/CEREG/CGREG format/examples • Modified the AT+TCPSETUP example • Modified the AT+TCPSEND example • Modified the AT+UDPSETUP example • Modified the AT+UDPSETUP example • Modified the AT+TCPSETUP example • Modified the AT+RECVMODE example	Lichuyu Cuilisha ZhaoGuiqiang SunZhen ZhaoGuiqiang
2023-06	• Modified the AT+NWGNSSMODE command • Modified the AT+NWGNSSAUX=<n>[,angle] command • Modified the AT+NWGNSSAUX=<n>[, angle] example • Modified the AT+TCPSETUP example • Modified the AT+TCPCLOSE response • Modified the AT+TCPTRANS response	Yanhaifeng Yanhaifeng Yanhaifeng Zhaoguiqiang Zhaoguiqiang Zhaoguiqiang
2023-06	• Modified the AT+UDPSETUP examples • Modified the AT+TCPSETUP examples • Modified the AT+TCPTRANS examples • Modified the AT+UDPTRANS examples • Modified the AT+SSLTCPSETUP examples • Modified the +MQTTSUB format • Modified the +CECALL Query format • Modified the +DNSSERVER response • Modified the example of AT+CGREG?	SunZhen DaiLintao Zhaoguiqiang Cuilisha
2023-06	• Modified the AT+NWGNSSMODE=? command information • Modified the AT+NWGNSSMODE? command information • Modified the +FTPLOGIN command format • Modified the +FTPGET command format • Modified the +FTPPUT command format • Modified the $MYNETAUTO command infomation and examples	Yanhaifeng XueJianyun
2023-06	• Modified the +TCPKEEPALIVE command examples • deleted some unsupported AT commands	Sunzhen Xuebowen

Conventions

Symbol	Indication
	This warning symbol means danger. You are in a situation that could cause fatal device damage or even bodily damage.
	Means reader be careful. In this situation, you might perform an action that could result in module or product damages.
	Means note or tips for readers to use the module

Related Documents

Neoway_N725_Datasheet

Neoway_N725_Product_Specifications

Neoway_N725_HW_User_Guide

Neoway_N725_EVK_User_Guide

1 General Introduction

1.1 Overview

Neoway modules can be driven via the serial interface using AT commands, which mainly comprise the the following two types of AT commands.

- International Standard AT Commands: in compliance with 3GPP TS27007, TS27005, and ITU-T Recommendation V.250.
- Neoway Extended AT Commands: to facilitate the implementation of a certain function, there are customized AT commands, such as internal protocol stack commands and other commands.

By utilizing AT commands, the MT can swiftly establish a wireless communication network, enabling diverse functionalities such as data transmission and remote control, thus accomplishing the interconnection of all things.

1.2 AT Interface

The AT interface is commonly used to send and receive AT commands when communicating with modems or modules. It can be accessed through the UART port or, in some cases, through the USB interface. Before debugging with AT commands, please refer to the module's EVK user guide for proper connection and interaction instructions.

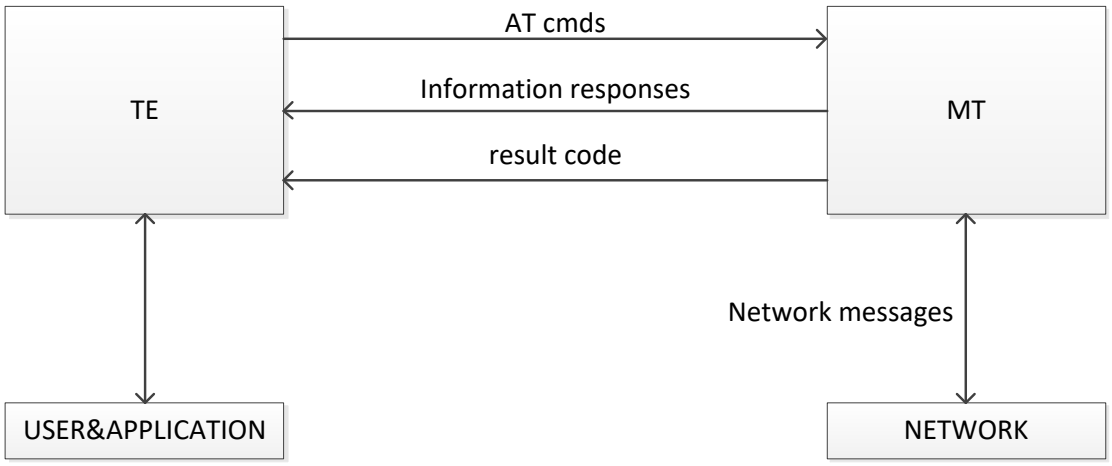
1.3 Definitions

In this document the following naming conventions are used:

- MT (Mobile Terminal): Neoway module
- TE (Terminal Equipment): Terminal that issues the command to the module.

AT commands, short for "Attention commands," are instructions that follow a specific syntax and are used to communicate with and control devices such as modems, cellular modules, and other telecommunication equipment. Below is the flow chart illustrating the response to AT commands.

Figure 1-1 AT command response flow chart



1.4 AT Command Syntax

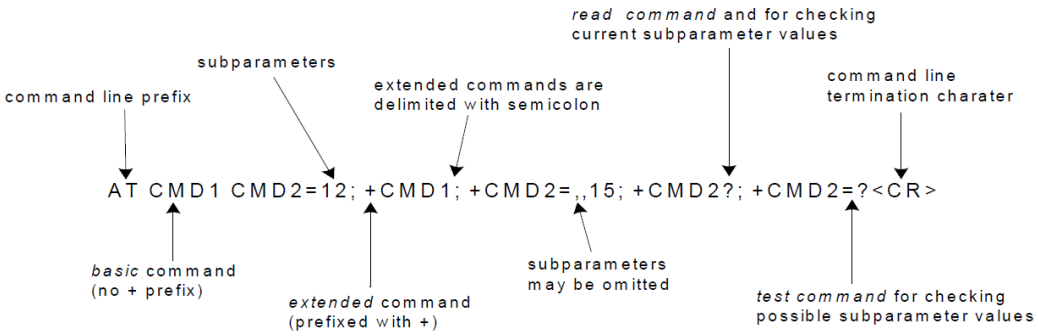
1.4.1 Symbols

- <CR>: The default termination character, which can be modified using the S3 command.
- <LF>: The default line feed character, which can be modified using the S4 command.
- <...>: the name in angle brackets is a parameter. The brackets themselves do not appear in the command line.
- [...]: the square brackets represent the optional parameters of a command or an optional part of the DCE information text response. Brackets themselves do not appear in the command line.
- Underscore: When a parameter value contains an underscore, it indicates that it is the default setting for that particular parameter.

1.4.2 Command Line

The AT commands are typically issued to the cellular modules using a command line with the following generic syntax:

Figure 1-2 AT command line



Prefix character

"**AT**" or "**at**" is a prefix that is set at the beginning of each command line. Neoway modules only recognize this type of AT commands.

Termination character

AT commands typically use **<CR>** as the default termination character, which is represented as "**0x0D**" in ASCII format.

Command parameter and response types

Numeric: Parameter data type, such as integer type, ranges from 0 to 100, with reference to the test command.

String: Parameter of string type.

- Enclosed in double quotation marks, consisting of byte stream without quotation marks or commas. Strings are by default enclosed in double quotation marks. If a string parameter is not enclosed, it will be specified separately in the command.
- If the character string content of an AT command includes special characters such as quotes, commas, slashes, etc., they need to be escaped using a backslash "\". The specific escape sequences are as follows:
  - \\: Escapes a backslash.

- \,: Escapes a comma. Note that the comma used to separate parameters does not need to be escaped.
- \": Escapes a double quotation mark. Double quotation marks used to indicate string parameters do not need to be escaped.

Command line length

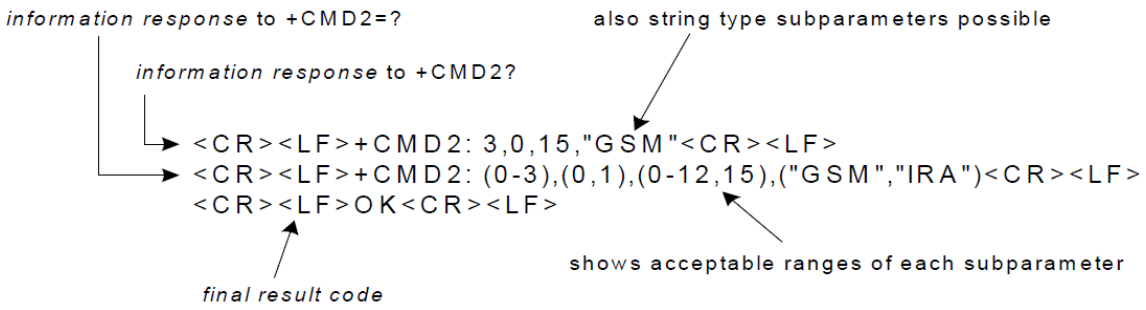
For the N725 module, the maximum length of a command line is 1536 bytes, including the command and all parameters.

Multiple commands can be placed on the same line using a semicolon (;). In this case, only the first command needs to have the "AT" prefix, while the subsequent commands do not require the "AT" prefix. If a command fails to execute, it does not affect the execution of the remaining commands.

AT command response

AT command responses can be divided into response information and result codes, as shown in the example below:

Figure 1-3 Command response example



- Response information syntax:

  Default format: <CR><LF>response<CR><LF>. The response can be a single line or multiple lines.

- Result code syntax:

  Default format: <CR><LF><result code><CR><LF>
  <CR><LF>OK<CR><LF> indicates that the command was executed successfully.
  <CR><LF>ERROR<CR><LF> indicates that the command execution failed.

The format of the response information and result codes returned by AT commands may vary depending on the settings of the ATV and AT+CMEE commands. For more details, please refer to the ATV and AT+CMEE commands, as well as Appendix 错误!未找到引用源。“错误!未找到引用源。”.

1.4.3 Unsolicited Result Code (URC)

URCs are string messages reported by the module without any request from the Terminal Equipment (TE). They are automatically sent by the module when specific events occur. Typical events that can trigger URCs include incoming calls (ringing), received short messages, network disconnections, and so on. For example:

```
When the AT+CRGE=1 command is enabled, if there is any change in the Mobile Terminal's (MT)
network registration status,
+CREG: <Stat> will be reported.
```

The URCs supported by the module can be found in Appendix 错误!未找到引用源。. Please refer to that section for more detailed information.

1.5 Supported Character Sets

The supported character sets in the AT command interface are as follows:

- GSM
- UCS2
- IRA

You can configure and query the character set using the AT+CSCS command (defined in 3GPP TS 27.007). The character set setting affects SMS sending and receiving, as well as the input and display of text fields in phonebook entries and SMS broadcast messages.

1.6 AT Command Type

Table 1-1 AT Command Type

Command type	Syntax	Function
Basic command	AT<CMD>[< VALUE >]	<CMD> is a single letter (A to Z), or it can be the "&" character followed by a single letter. <VALUE> is a decimal number, which can be one or multiple digits. Leading zeros in <VALUE> will be ignored.

Register commands	ATS<n>=[< VALUE >]	"<n>" represents the index of the S register, while "<VALUE>" represents the assigned parameter value.
Extended commands		
Set	AT+CMD=<VALUE><CR>	Store a value or values for later use
Execute	AT+CMD[=<VALUE>]<CR>	Invoke a function of the module.
Query	AT+CMD?<CR>	Determine the current value or values stored.
Test	AT+CMD=?<CR>	Determine the range of parameter values or parameter lengths that are supported

Please note that all AT command examples provided in this document do not include syntax indicators. However, they do comply with the syntax rules specified in this section.

1.7 Command Response Time-Out

Every command issued to the Neoway module returns a result response, if response codes are enabled (default). The time needed to process the given command and return the response varies, depending on the command type.

Commands that do not interact with the SIM or the network, and only involve internal setups or readings, have an immediate response, and the maximum response timeout is default to 300 ms. Commands that interact with the SIM/USIM, the network, or the peripherals could take many seconds to send a response, depending on SIM configuration (e.g., number of contacts stored in the phonebook, number of stored SMS), on the network the command may interact with (e.g., network quality, network congestion, and so on), or on the peripheral type.

In the table below are listed only the commands whose interaction with the SIM, the network, or the peripheral could lead to long response timings. For other commands not listed in the table below, the maximum response time is 300 ms.

No.	Command	Estimated maximum time to get response (Seconds)
1	AT+COPS	600
2	AT+CPIN	5
3	AT+CLCK	15
4	AT+CPWD	5
5	AT+CPMS	5

6	AT+CMGR	15
7	AT+CMGL	30
8	AT+CMGS	60
9	AT+CMGW	5
10	AT+CMSS	60
11	AT+CMGD	5
12	AT+CSCA	5
13	AT+XIIC	30
14	AT+TCPSETUP	60
15	AT+TCPSEND	30
16	AT+TCPCLOSE	10
17	AT+UDPSETUP	20
18	AT+UDPSEND	30
19	AT+CLOSELISTEN	1
20	AT+CLOSECLIENT	1
21	AT+TCPSENDS	30
22	AT+TCPTRANS	60
23	AT+TRANSCLOSE	10
24	AT+TCPSRVTRANS	1
41	ATD	60
43	ATA	30
44	ATH	30
46	AT+CLIP	5
47	ATS0	300
48	AT+CLCC	5
49	AT+SSLTCPSETUP	30
50	AT+SSLTCPCLOSE	10
51	AT+SSLTCPSEND	30
52	AT+CERTADD	30
79	AT+NWGNSSPWR	30
80	AT+NWGNSSNMEA	10
81	AT+NWGNSSAUX	30
82	AT+NWGPSSMODE	10

85	AT+SETSERVER	10
89	AT$QCRMCALL	60
97	AT+NEOFTPFOTA	Determined by upgrade package size and network environment.
98	AT+FSWF	Customized, ranging from 1 to 60s.
99	AT$MYSYSINFO	120
100	AT$MYNETINFO	120

1.8 Operating Mode of The AT Interface

The AT interface can operate in these modes:

- Command mode: in this mode, the module treats all received characters as AT commands and executes them, returning the corresponding execution results.
- Transparent data mode: after establishing a PPP connection or entering transparent data mode through AT commands (such as TCP, FTP, etc.), the module enters data transmission mode. In this mode, the module sends all received characters as data to be transmitted.

In transparent data mode, using the "+++" command (without carriage return or line feed) allows switching to command mode. In command mode, using the "ATO" command (without carriage return or line feed) switches to transparent data mode.

2 Boot LOG Instruction

If the module is powered on with a fixed baud rate, the UART port will provide a power-on response of "+PBREADY".

If the module is powered on in auto-baud rate mode, please wait for 10 seconds after power-up and send the command "AT" to the module. Once the module completes the AT initialization and responds with "OK", it will provide the power-on response of "+PBREADY".

For auto-baud rate mode, please refer to the AT+IPR command.

After power-up, the module will automatically enter command mode. In command mode, users can use AT commands to query, configure, and perform other operations on the module.

3 General Commands

3.1 ATI – Querying the Manufacturer Information

To query the manufacturer information, including manufacturer, model, and version

Format

Type	Command	Response
Execute	ATI<CR>	<CR><LF><manufacturer> <CR><LF><module_version> <CR><LF><soft_version><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<manufacturer>	module manufacturer
<module_version>	module model
<soft_version>	software version

Example

```
ATI
NEOWAY                                          Manufacturer
N725
V003                                            Module model
                                                Version

OK
```

3.2 AT+GMR – Querying the Software Version

To query the software version.

Format

Type	Command	Response
Execute	AT+GMR<CR>	<CR><LF><reversion><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<reversion> software version

Example

```
AT+GMR
N725_D0B0CM_BZ_V003                                    Query

OK
```

3.3 AT+CGMR – Querying the Software Version

To query the software version.

Format

Type	Command	Response
Execute	AT+CGMR<CR>	<CR><LF>+CGMR: <reversion><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<reversion> software version

Example

```
AT+CGMR
+CGMR: N725-A01-STD-BZ-002                             Query

OK
```

3.4 AT+CSQ – Querying Signal Quality

To query the receiving signal strength indication (RSSI) and bit error rate (BER) of the channel.

Format

Type	Command	Response
Execute	AT+CSQ<CR>	<CR><LF>+CSQ: <signal>,<ber><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<signal> The following table shows the relationship between the CSQ and the RSSI.

	signal	Rssi
0	<4 or 99	<-107dBm or unknown
1	<10	<-93dBm
2	<16	<-71 dBm
3	<22	<-69 dBm
4	<28	<-57 dBm
5	>=28	>=-57 dBm

<ber>

0...7	Refer to the value of RXQUAL in the table of GSM 05.08 8.2.4.
99	Not known or not detectable

Example

```
AT+CSQ
+CSQ: 19,2                                    Query the strength of the current signal.

OK
```

3.5 AT+CREG – Querying Network Registration Status

To query the network registration status of the module

Format

Type	Command	Response
Execute	AT+CREG=[<n>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CREG?<CR>	<CR><LF>+CREG: <n>,<stat>[,[<lac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CREG=?<CR>	<CR><LF>+CREG: (list of supported <n>s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<n> integer type;Specifies whether to enable unsolicited result codes for network registration.
0: disable network registration unsolicited result code
1: enable network registration unsolicited result code +CREG: <stat>
2: enable network registration and location information unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>]]
3: enable network registration, location information and cause value information unsolicited result code
+CREG: <stat>[,[<lac>],[<ci>],[<AcT>][,<cause_type>,<reject_cause>]]

<stat> integer type; circuit mode registration status
0: not registered, the module is not currently searching for an operator to register to
1: registered with a home network
2: not registered, but the module is currently trying to attach or searching for an operator to register to
3: registration denied
4. unknown code
5: registered, roaming
6: registered for "SMS only", home network (applicable only when <AcT> indicates E-UTRAN)
7: registered for "SMS only", roaming (applicable only when <AcT> indicates E-UTRAN)
8: attached for emergency bearer services only (see NOTE 2) (not applicable)
9: registered for "CSFB not preferred", home network (applicable only when <AcT> indicates E-UTRAN)
10: registered for "CSFB not preferred", roaming (applicable only when <AcT> indicates E-UTRAN)
11: Only emergency services are available

<lac> Two-byte location area code in hexadecimal format, string type

<ci> Four-byte cell ID in hexadecimal format, string type

<Act> The access technology of the serving cell, integer type
0: GSM
1: GSM compact
2: UTRAN
3: GSM w/EGPRS
4: UTRAN w/HSDPA
5: UTRAN w/HSUPA
6: UTRAN w/HSDPA and w/HSUPA
7: E-UTRAN
8: UTRAN w/HSPA+

<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=1                          Enable unsolicited codes of network registration.
OK
AT+CREG?                           Query the network registration status of the module.
+CREG: 0,1

OK
AT+CREG=?                          Query the value range of the network registration status
+CREG: (0-3)                       parameter.

OK
```

3.6 AT+CGREG – Querying GPRS Network Registration Status

Query the current state of GPRS network registration, reference 3GPP TS 27007.

Format

Type	Command	Response
Execute	AT+CGREG=[<n>]<CR>	<CR><LF>OK<CR><LF>

		Or <CR><LF>ERROR<CR><LF>
Query	AT+CGREG?	<CR><LF>+CGREG: <n>,<stat>[,[<lac>],[<ci>],[<AcT>],[<rac>][,<cause_type>,<reject_cause>]]<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CGREG=?	<CR><LF>+CGREG: (list of supported<n>s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<n>	integer type 0: disable network registration unsolicited result code (default settings). 1: enable network registration unsolicited result code. +CGREG: <stat> 2: enable network registration and location information unsolicited result code. +CGREG: <stat>[,<[lac>,]<[ci>],[<AcT>],[<rac>]] 3: enable network registration, location information and GMM cause value information unsolicited result code +CGREG: <stat>[,[<lac>],[<ci>],[<AcT>],[<rac>][,<cause_type>,<reject_cause>]]
<stat>	integer type; indicates the GPRS 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 GERAN/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 (see NOTE 2) (applicable only when <AcT> indicates 2,4,5,6) 9: registered for "CSFB not preferred", home network (not applicable) 10: registered for "CSFB not preferred", roaming (not applicable) 11: attached for emergency bearer services only
<lac>	string type; two byte location area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)
<ci>	string type; four byte GERAN/UTRAN cell ID in hexadecimal format.
<Act>	0: GSM 1: GSM compact 2: UTRAN 3: GSM w/EGPRS

	4: UTRAN w/HSDPA 5: UTRAN w/HSUPA 6: UTRAN w/HSDPA and HSUPA 7: E-UTRAN (not applicable)
<rac>	string type; one byte routing area code in hexadecimal format
<cause_type>	integer type; indicates the type of <reject_cause>. 0: Indicates that <reject_cause> contains a GMM 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+CGREG?                                   Query the network registration status of
+CGREG: 0,1                                 the module.
                                            Register GPRS Network.
OK
AT+CGREG=1                                  Enable network registration unsolicited
OK                                          code.
AT+CGREG=?                                  Query the value range of the network
+CGREG: (0-3)                               registration status parameter.

OK
```

3.7 AT+CEREG – Querying EPS Network Registration Status

To query the EPS network registration status of the module

Format

Type	Command	Response
Execute	AT+CEREG=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CEREG?<CR>	<CR><LF>+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>[,<cause_type>,<reject_cause>]]]<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CEREG=?<CR>	<CR><LF>+CEREG: (list of supported<n>s) <CR><LF>

<CR><LF>OK<CR><LF>

Parameter

<n>	Specifies whether to enable network registration unsolicited result code. 0: disable network registration unsolicited result code (default). 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>]]
<stat>	Network status 0: not registered, the module is not currently searching for a new operator to register 1: registered to the home network 2: not registered, but the module is currently trying to searching for a base station 3: registration denied 4. Unknown code 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 (See NOTE 2) 9: registered for "CSFB not preferred", home network (not applicable) 10: registered for "CSFB not preferred", roaming (not applicable) 11: attached for emergency bearer services only
<tac>	string type; two byte tracking area code in hexadecimal format (e.g. "00C3" equals 195 in decimal)
<ci>	four byte E-UTRAN cell ID in hexadecimal format, string type
<Act>	the access technology of the serving cell, integer type 0: GSM (not applicable) 1: GSM Compact (not applicable) 2: UTRAN (not applicable) 3: GSM w/EGPRS (see NOTE 3) (not applicable) 4: UTRAN w/HSDPA (see NOTE 4) (not applicable) 5: UTRAN w/HSUPA (see NOTE 4) (not applicable) 6: UTRAN w/HSDPA and HSUPA (see NOTE 4) (not applicable) 7: E-UTRAN
<cause_type>	integer type; indicates the type of <reject_cause>. 0: Indicates that <reject_cause> contains an EMM cause value, see 3GPP TS 24.301 [83] Annex A. 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+CEREG?                          Query the network registration status of the module.
+CEREG: 0,1

OK
AT+CEREG=1                         Enable network registration unsolicited code.
OK
AT+CEREG=?                         Query the value range of the network registration
+CEREG: (0-3)                      status parameter.

OK
```

3.8 AT+COPS – Selecting and Registering a Network

To select and register a network.

Format

Type	Command	Response
Execute	AT+COPS=[<mode>[,<format>[,<oper>>[,<AcT>]]]]<CR>	<CR><LF>OK<CR><LF>
Query	AT+COPS?<CR>	<CR><LF>+COPS: <mode>[,<format>,<oper>[,<AcT>]]<CR><LF><CR><LF>OK<CR><LF>
Test	AT+COPS=?<CR>	<CR><LF>+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)]<CR><LF><CR><LF>OK<CR><LF>

Parameter

<mode> To set automatic network selection or manual selection:
0: Automatic selection (ignore the parameter <per>)
1: Manual selection
2: Deregister from the network

	3: Set <format>only 4: Manual/automatic selection (if the manual selection fails, automatic mode starts)
<format>	0: Long alphanumeric <oper> (default value) 1: Short format alphanumeric <oper> 2: Numeric <oper>
<oper>	It is given in <format>. This field may be in 16-character long alphanumeric format, 8-characters short alphanumeric format, or 5-character numeric format (MCC/MNC).
<AcT>	indicates the radio access technology and its value can be 0, 1, and 2. 0: GSM 1: GSM compact 2: UTRAN 3: GSM w/EGPRS 4: UTRAN w/HSDPA 5: UTRAN w/HSUPA 6: UTRAN w/HSDPA and HSUPA 7: E-UTRAN 9: CDMA

Example

```
AT+COPS=0,0                          Automatic network selection is enabled. Long
OK                                   alphanumeric mode.
AT+COPS=0,2
OK                                   Set to digital mode
AT+COPS?
+COPS: 0,0,"CHINA MOBILE",7          China Mobile

OK
AT+COPS?
+COPS: 0,2,"46000",7                 If it is set to digital mode, get the number 46000

OK
AT+COPS?
+COPS: 0,0,"CHINA UNICOM",7          China Unicom

OK
AT+COPS?
+COPS: 0,2,"46001",7                 If it is set to digital mode, then get the number
                                     46001.
OK
AT+COPS?
+COPS: 0,0,"CHINA TELECOM",9         If it is set to digital mode, then get the number
                                     46003.
OK
AT+COPS?
+COPS: 0,2,46011,9                   If it is set to digital mode, then get the number
                                     46011.
OK
AT+COPS=?
+COPS: (2,"CHINA
MOBILE","CMCC","46000"),(1,"CHN-      Query the parameter ranges.
CT","CT","46011"),(1,"CHN-
```



```
UNICOM","UNICOM","46001"),,(0,1,
2,3,4),(0,1,2)

OK
AT+COPS=2                                    Deregister the network.
OK
```

3.9 AT+CIMI – Querying the IMSI

To query the international mobile subscriber identification (IMSI).

Format

Type	Command	Response
Execute	AT+CIMI<CR>	<CR><LF><IMSI><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<IMSI> The international mobile subscriber identification.
A character string of 15 digits and starts with 3-bits of MCC and 2-bits of MNC. It is used to authenticate the SIM card.

Example

```
AT+CIMI
460020188385503                              Obtain the IMSI number.
OK
AT+CIMI                                      Query the IMSI.
ERROR                                        No SIM card is installed.
```

3.10 AT+CGSN – Querying the IMEI

To query the International Mobile Equipment Identity (IMEI) of the module.

Format

Type	Command	Response
Execute	AT+CGSN<CR>	<CR><LF><IMEI><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<IMEI> International Mobile Equipment Identity, a character string of 15 digits.

Example

```
AT+CGSN
355897043139120

OK
```

On a 3GPP2 network, the return code is an 8-digit ESN.

3.11 AT+CCID – Obtaining the ICCID of the SIM Card

To obtain the integrated circuit card identifier (ICCID) of the SIM card

Format

Type	Command	Response
Execute	AT+CCID<CR>	<CR><LF>+CCID:<ICCID><CR><LF> <CR><LF>OK<CR><LF> **Or** <CR><LF>ERROR<CR><LF>

Parameter

<ICCID> SIM card ID, a string of 20 digits.

Example

```
AT+CCID                                  Read command
+CCID:89860002190810001367

OK
AT+CCID                                  Read command
ERROR                                    The SIM card is not inserted.
```

3.12 AT+CGMM – Querying the Module Model

To query the module model

Format

Type	Command	Response
Execute	AT+CGMM<CR>	<CR><LF><model><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<model> module model

Example

```
AT+CGMM                                  Query the product model.
N725

OK
```

3.13 AT+GMM – Querying the Module Model

To query the the module model.

Format

Type	Command	Response
Execute	AT+GMM<CR>	<CR><LF><model><CR><LF>

	<CR><LF>OK<CR><LF>

Parameter

N/A

Example

```
AT+GMM
N725                                                    Query the product model.

OK
```

3.14 AT+IPR – Setting Baud Rate

To set the baud rate of the module. The default baud rate is 0, indicating automatic baud rate detection.

The setting by this command is not saved after the module is powered down.

Format

Type	Command	Response
Execute	AT+IPR=<baud rate><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+IPR?<CR>	<CR><LF>+IPR: <baud rate> <CR><LF>OK<CR><LF>
Test	AT+IPR=?<CR>	<CR><LF>+IPR: (list of supported <baud rate>s) <CR><LF> <CR><LF>OK<CR><LF>

Parameter

<baud rate> The value can be 0,300,600,1200,2400,4800,9600,19200,38400,57600,115200,230400,460800, 921600.

Example

```
AT+IPR=115200                                   Set the baud rate to 115200 bps.
OK
AT+IPR?
+IPR: 115200                                    Query the current baud rate.

OK
AT+IPR=?
+IPR:
0,300,600,1200,2400,4800,9600,19
200,38400,57600,115200,230400,46                Query the available baud rate range.
0800,921600

OK
AT+IPR=100                                      Set the baud rate to 100.
ERROR                                           The value is not allowed
```

3.15 AT+CFUN – Setting Module Functionality

To select the level of functionality of the module by setting **<fun>**.

The settings are not be saved after the module is powered down.

Format

Type	Command	Response
Execute	AT+CFUN=[<fun>[,<rst>]]<CR>	<CR><LF>+CFUN: (list of supported <fun>s),(list of supported <rst>s) <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CFUN?<CR>	<CR><LF>+CFUN:<fun> <CR><LF>OK<CR><LF>
Test	AT+CFUN=?	<CR><LF>+CFUN: (list of supported <fun>s),(range of supported <rst>) <CR><LF>OK<CR><LF>

Parameter

<fun> power saving function mode
0: turn off radio and SIM power
1: Full functionality (default)
4: Turn off the TX and RX circuits (Flight mode)

<rst>	Specifies whether to restart the module 0: do not reset the module before setting it to <fun> power level 1: reset the module before setting it to <fun> power level

Example

```
AT+CFUN=1                                          Set full functionality.
OK
AT+CFUN?
+CFUN: 1                                           Query current function level.
OK                                                 Full functionality
AT+CFUN=?
+CFUN: (0-1,4) ,(0-1)                              Query available parameter value ranges.
OK
```

3.16 AT+CMUX – Activating Multiplexing Mode

To activate multiplexing mode.

Format

Type	Command	Response
Execute	AT+CMUX=<mode>[,<subset>[,<port_speed>[,<N1>[,<T1>[,<N2>[,<T2>[,<T3>[,<k>]]]]]]]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CMUX=?<CR>	<CR><LF>+CMUX: (list of supported <mode>values),(list of supported <subset> values),(value range of<port_speed>),(value range of<N1>),(value range of<T1>),(value range of<N2>),(value range of<T2>),(value range of<T3>),(value range of<k>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode>	The mode of MUX that is enabled, integer type 0: Basic option (default value) 1: Advanced option (not supported)
<subset>	Subset of frame format, integer type 0: UIH frames used only (default value) 1: UI frames used only (not supported currently)
<port_speed>	UART port rate, integer type

	1: 9600bit/s 2: 19200bit/s 3: 38400bit/s 4: 57600bit/s 5: 115200bit/s (default) 6: 230400bit/s
<N1>	Maximum frame size. Integer type, ranging from 1 to 1509. The default value is 31. For Advanced option, the default value is 64.
<T1>	Acknowledgement timer in unite of ten milliseconds, integer type, ranging from 1 to 255, where 10 is default (equal to 100 ms).
<N2>	Maximum number of re-transmissions, integer type, ranging from 0 to 5. The default value is 3.
<T2>	response timer for the multiplexer control channel in units of ten milliseconds, integer type. Ranging from 2 to 255. The default value is 30 (300 ms).
<T3>	Wake up response timer in seconds, integer type. Not supported.
<k>	window size, integer type. Not supported

Example

```
AT+CMUX=0                                          Basic option, the rest of the parameters
OK                                                 use the default values
AT+CMUX=2                                          The parameter value exceeds available
ERROR                                              range
AT+CMUX=0,0,,512,254,5,255
                                                   Basic option.
OK
AT+CMUX=1,0,,512,254,5,255                         Advanced option.
ERROR                                              Not supported
AT+CMUX=?
+CMUX: (0),(0),(1-6),(16-1509),(1-255),(0-
                                                   Query the available range of parameters.
100),(2-255),(1-255),(1-7)
OK
AT+CMUX?
                                                   Incorrect command syntax.
ERROR
```

3.17 AT+CCLK – Clock

To set and query the real-time clock.

The setting by this command is saved after the module is powered down.

The default clock is GMT+0.

Format

Type	Command	Response
Execute	AT+CCLK=<time><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CCLK?<CR>	<CR><LF>+CCLK: <time><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<time> Character string in format of "YY/MM/DD,hh:mm:ss[+TZ]".
TZ: Two digits, indicating the time lag between the local time and the GMT time. This information is optional because it can be displayed only when the network supports it.

Example

```
AT+CCLK="08/07/01,14:54:01"                    Set the real-time clock of the module.
OK
AT+CCLK?
+CCLK: "08/07/01,14:54:10"                     Query the setting of the real-time clock.
OK
AT+CCLK=14/07/02,10:48:50
ERROR                                          Incorrect command syntax.
```

3.18 AT+CPIN – Entering PIN Code

To query the PIN status and enter PIN code.

Format

Type	Command	Response
Execute	AT+CPIN=<pin>[,<newpin>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CPIN?<CR>	<CR><LF>+CPIN: <code><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<code>	READY: No password SIM PIN: Enter PIN code. SIM PUK: Enter PUK code. SIM PIN2: Enter PIN2 code. SIM PUK2: Enter PUK2 code.

Example

```
AT+CPIN?
+CPIN: READY                                    Query whether PIN code is required.
                                                No password is required.
OK
AT+CPIN?
+CPIN: SIM PIN                                  Query whether PIN code is required.
                                                Required
OK
AT+CPIN="1234"                                  Input correct PIN code.
OK
                                                Unlock
+PBREADY
AT+CPIN?
+CPIN: SIM PUK                                  PUK code is required if PIN code is input
                                                incorrectly for three times
OK
AT+CPIN="12345678","4321"                       Input PUK code, and new PIN code.
OK

                                                Unlock
+PBREADY
```

- To enter PIN code, lock current SIM card (running AT+CLCK="SC",1,"1234") and then restart the module.
- If PIN code is input incorrectly for three times, PUK is required to unlock.

3.19 AT+CLCK – Locking/Unlocking Module

To lock, unlock or interrogate module.

Format

Type	Command	Response
Execute	AT+CLCK=<fac>,<mode>[,<passwd>[,<class>]]<CR>	• When <mode> =2: <CR><LF>+CLCK: <status>[,<class1>

		[<CR><LF>+CLCK: <status>,<class2>[...]]<CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> • When <mode> is not equal to 2: <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CLCK=?<CR>	<CR><LF>+CLCK: (list of supported <fac> values) <CR><LF> <CR><LF>OK<CR><LF>

Parameter

<fac>	A pair of quotation marks are required for the value. "OI": Outgoing international calls "SC": SIM card "AI": all incoming calls "IR": incoming calls when roaming outside the homing place "AO": All outgoing calls "OX": All outgoing international calls except to the home country "AB": all calling services "AG": all outgoing call services "AC": all incoming call services "FD": SIM fixed dialing memory feature "PS": lock phone to the SIM card "PN": network authentication "PU": network subsystem authentication "PP": service provider authentication "PC": corporate authentication
<mode>	0: Unlock 1: Lock 2: Query the status
<status>	0: not active 1: active
<passwd>	Password or code, string type. A pair of quotation marks are required for the value.
<classx>	1: Voice service 2: Data service 4: Fax service 8: SMS 16: Synchronous data service 32: Asynchronous data service 64: Dedicated packet access

128: Dedicated PAD access

Example

```
AT+CLCK="SC",2
+CLCK: 0

OK
AT+CLCK=?
+CLCK: ("CS", "PS", "PF", "SC", "AO", "OI", "OX", "AI",
"IR", "NT", "NM", "NS", "NA", "AB", "AG", "AC", "FD",
"PN", "PU", "PP", "PC" )

OK
AT+CLCK="SC",1,"1234"
+CLCK: 1

OK
AT+CLCK="SC",0,"1234"
+CLCK: 0

OK
AT+CLCK="SC",1,"2222"
ERROR
```

Query the network information related to the module.

Lock the current SIM card. "1234" is the PIN code of current SIM card.

Unlock the current SIM card. "1234" is the PIN code of current SIM card.

The PIN code is incorrect.

3.20 AT+CPWD – Modifying Password

To modify the password of the lock function of the module.

Format

Type	Command	Response
Execute	AT+CPWD=<fac>,<oldpwd>,<newpwd><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CPWD=?<CR>	<CR><LF>+CPWD:(list of supported (<fac>,<pwdlength>)s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<fac>	A pair of quotation marks is required for the value. "CS":CNTRL (lock CoNTRoL surface (e.g. phone keyboard)) "PS":PH-SIM (lock PHone to SIM/UICC card installed in the currently selected card slot) (MT asks password when other than current SIM/UICC card inserted; MT may remember certain amount of previously used cards thus not requiring password when they are inserted) "PF":lock Phone to the very First inserted SIM/UICC card (also referred in the present document as PH-FSIM) (MT asks password when other than the first SIM/UICC card is inserted) "P2":SIM PIN2 "OI": outgoing international calls "AI": all incoming calls "IR": incoming calls when roaming outside the homing place "SC": SIM card "AO": all outgoing calls "OX": all outgoing international calls except to the home country "AB": all calling services "AG": all outgoing call services "AC": all incoming call services "FD": fixed dialing of the SIM card "PN": network authentication "PU": network subsystem authentication "PP": service provider authentication "PC": corporate authentication "NT":barr incoming calls from numbers Not stored to TA memory "NM":barr incoming calls from numbers Not stored to MT memory "NS":barr incoming calls from numbers Not stored to SIM/UICC memory "NA":barr incoming calls from numbers Not stored in Any memory
<oldpwd>	Old password or code, string type. A pair of quotation marks is required for the value.
<newpwd>	New password or code, string type. A pair of quotation marks is required for the value.

Example

```
AT+CPWD=?
+CPWD:
("CS",16),("PS",16),("PF",16),("SC",16),
("AO",16),("OI",16),("OX",16),("AI",16),
("IR",16),("AB",16),("AG",16),("AC",16),
("FD",16),("NT",16),("NM",16),("NS",16),
("NA",16),("PN",16),("PU",16),("PP",16),
("PC",16),("P2",16)

OK
```

Query the service range of the PIN password allowed by the module.

```
AT+CPWD="SC","1234","0000"
OK
```

Modify the PIN code of the current SIM card. "1234" is the old PIN code and "0000" is the new PIN code.

To modify the PIN code, lock the SIM card (running AT+CLCK="SC",1,"1234").

3.21 AT+CGDCONT – Defining PDP Context

To set the packet data protocol (PDP) parameter values for a PDP context identified by the (local) context identification parameter, <cid>.

Format

Type	Command	Response
Set	AT+CGDCONT=<cid>[,<PDP_type>[,<APN>[,<PDP_addr>[,<d_comp>[,<h_comp>]]]]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CGDCONT?<CR>	<CR><LF>+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CGDCONT=?<CR>	<CR><LF>+CGDCONT: [list of supported (<cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>)] <CR><LF>OK<CR><LF>

Parameter

<cid>	(PDP Context Identifier) a numeric parameter that specifies a particular PDP context definition. Minimum value = 1.
<PDP_type>	(Packet Data Protocol type) a string parameter. IP: Internet Protocol (IETF STD 5) PPP: Point to Point Protocol (IETF STD 51) IPV4V6: V4V6 dual-stack network protocol
<APN>	Access Point Name. A string parameter which is a logical name that is used to select the GGSN or the external packet data network.
<PDP_address>	a string parameter that identifies the terminal in the address space applicable to the PDP. TE will provide a value for this parameter after PDP starts if it is null or omitted. If TE fails to provide, the subscription value will be requested.
<d_comp>	numeric parameter that controls PDP data compression. Used only for SNDCP
<h_comp>	numeric parameter that controls PDP header compression. 0 - off (default if value is omitted)
<pd1>, …	zero to N string parameters whose meanings are specific to the <PDP_type>

<pdN>

Example

```
AT+CGDCONT=1,"IP","CMNET"                                  Set PDP type to IP and APN to CMNET.
OK
AT+CGDCONT?
+CGDCONT: 1,"IP","CMNET"                                   Query current PDP format.

OK
AT+CGDCONT=?
+CGDCONT: (0-15),"IP",,,(0-3),(0-
4),(0,1),(0,1),(0-2),(0,1)

+CGDCONT: (0-15),"IPV6",,,(0-3),(0-
4),(0,1),(0,1),(0-2),(0,1)                                 Query the available value range of PDP
                                                           format.
+CGDCONT: (0-15),"IPV4V6",,,(0-3),(0-
4),(0,1),(0,1),(0-2),(0,1)

+CGDCONT: (0-15),"PPP",,,(0-3),(0-
4),(0,1),(0,1),(0-2),(0,1)

OK
```

3.22 AT+XGAUTH – PDP Authentication

PDP authentication.

Before sending this command, send AT+CGDCONT.

To use internal protocol stack, add this command to the process.

<cid> is same as that in +CGDCONT.

Format

Type	Command	Response
Set	AT+XGAUTH=<cid>,<auth>[,<name>,<pwd>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+XGAUTH=?	<CR><LF>+XGAUTH: (list of supported <cid>),(value range of<auth>),(length of <name>),(length of <pwd>) <CR><LF>OK<CR><LF>

Parameter

<cid>	PDP context identifier
<auth>	authentication may be: 0: NONE 1: PAP (default) 2: CHAP 3: PAP or CHAP
<name>	user name
<pwd>	password

Include <name> and <pwd> into the command when <auth> is equal 0.

Example

```
AT+XGAUTH=1,1,”gsm”,”1234”                    Set the first PDP authentication.
OK
AT+XGAUTH=?                                   Query the available value range of
+XGAUTH: (1),(0-3),128,128                    parameters.
OK
```

3.23 AT+CGATT – Setting GPRS Attach and Detach

To set GPRS attach and detach.

The settings are be saved after the module is powered off.

By default, the module can automatically perform GPRS attach.

Ensure that the GPRS attach is set before the PPP connection is set up. It is recommended to add the AT+CGATT? command to the process to query the GPRS status. If the module returns 1, set up PPP connection directly; otherwise, set GPRS attach manually by executing the command AT+CGATT=1.

Format

Type	Command	Response
Set	AT+CGATT=<state><CR>	<CR><LF>GPRS DISCONNECTION<CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>OK<CR><LF>

		Or <CR><LF>ERROR<CR><LF>
Query	AT+CGATT?<CR>	<CR><LF>+CGATT: <state> <CR><LF>OK<CR><LF>
Test	AT+CGATT=?<CR>	<CR><LF>+CGATT: (value range of <state>) <CR><LF>OK<CR><LF>

Parameter

<state>	0: indicates detach 1: indicates attach

Example

```
AT+CGATT=1                          GPRS attach is set successfully.
OK
AT+CGATT=0                          GPRS detach is set successfully.
OK
AT+CGATT=0
GPRS DISCONNECTION                  Send this command after setting a PPP connection.

OK
AT+CGATT=0                          ERROR is returned because no SIM card is installed.
ERROR
AT+CGATT?
+CGATT: 0                           Query the GPRS status.

OK
AT+CGATT=?
+CGATT: (0-1)                       Query the valid parameter values for the command.

OK
```

3.24 ATE1/ATE0 – Enabling & Disabling the Terminal Display

To enable or disable the terminal display function of the AT commands. The settings by this command are not saved after the module is powered off.

The terminal display function is enabled by default.

If the command is sent after dialing up to connect the network, terminal display is disabled automatically.

ATE is equal to ATE1.

Format

Type	Command	Response
Execute	• ATE1<CR> • ATE0<CR>	<CR><LF>OK<CR><LF> <CR><LF>OK<CR><LF>

Parameter

N/A

Example

```
ATE1
OK                                          Turn on module AT command echo function
AT                                          Send AT, serial tools show "AT" and "OK".
OK
ATE0                                        Turn off the module AT command echo
OK                                          function

OK                                          Send AT, serial tools only show "OK"
```

3.25 ATD*99# – GPRS Dialing Command

To initialize a GPRS connection using external protocol stacks.

This command is applicable only to external protocol stacks.

Ensure that the module has registered the network and set APN before dialing any number.

Format

Type	Command	Response
Execute	ATD*99#<CR>	<CR><LF>CONNECT<CR><LF>

Parameter

N/A.

Example

```
ATD*99#
CONNECT
```

3.26 AT+ENPWRSAVE－Enabling or Disabling Sleep Mode

To enable or disable sleep mode.

The settings by this command are not saved after the module is powered off.

Sleep mode is triggered by inputting low level at DTR by default.

After this command is sent and low (or high) level is input at DTR, the module can enter sleep mode unless circuit of each part inside the module allows.

Format

Type	Command	Response
Execute	AT+ENPWRSAVE=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+ENPWRSAVE?<CR>	<CR><LF>+ENPWRSAVE: <n><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<n> 0: Forbid sleep mode. (default)
1: Allow sleep mode (Low level at DTR triggers sleep mode)
2: Allow sleep mode (High level at DTR triggers sleep mode)

Example

```
AT+ENPWRSAVE=1                          Enable the sleep mode.
OK
AT+ENPWRSAVE?
+ENPWRSAVE: 1                           Query current sleep mode status.

OK
```

3.27 AT+SIGNAL－Setting Blinking Status Signal Indicator

To set the different blinking status of the signal indicator.

The default status setting is 7.

If the status is set to 0 to 6, the indicator will be always on when a call or SIM message incoming in sleep mode.

Format

Type	Command	Response
Set	AT+SIGNAL=<value><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+SIGNAL?<CR>	<CR><LF>+SIGNAL: <value><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+SIGNAL=?<CR>	<CR><LF>+SIGNAL: (value range of<value>) <CR><LF>OK<CR><LF>

Parameter

<value> Integers, ranging from 0 to 7
0: Blink once every second in normal situation. Being off or on if any abnormality occurs.
1: Blink once every second after the module is connected to the GPRS network. Being off in any other situations.
2: Flash and blink. Flash every 250 ms for the GPRS data service and blink every second in other normal situations.
3: Be on after the GPRS network is connected and blink every second in other situations.
4: Be on after the GPRS network is connected and being off in other situations.
5: Be off if the SIM card cannot be detected after the module is started, blink every second if the SIM card is detected, and be on after the GPRS network is connected.
6: Four indicator states:

- If no SIM card is installed or the SIM card does not register network, the indicator blinks every one second and is on for 0.1 second.
- If the SIM card registered network, the indicator blinks every three second and is on for 0.1 second.
- If the GPRS network is connected, the indicator blinks every 250 ms and is on for 0.1 second.

- The indicator is always on during a call.

7: four indicator states

- Off: No SIM card, not registered
- On: registered network
- On for 0.2 second and off for 1.8 second: Obtained the IP address
- On for 1.8 second and off for 0.2 second: Connected to the server

<low_interval> 10 to 65535 ms

<high_interval> 10 to 65535 ms

Example

```
AT+SIGNAL?
+SIGNAL: 2                                  The current signal indicator status is 2.

OK
AT+SIGNAL=3
OK                                          Set current signal indicator status to 3.
AT+SIGNAL=11
ERROR                                       The parameter is set to an incorrect
                                            value.
AT+SIGNAL=?
+SIGNAL: (0-7)                              The available value of the signal
OK                                          indicator status ranges from 0 to 7.
```

3.28 AT+CUSD – Sending USSD Data

To send Unstructured Supplementary Service Data (USSD)

This command supports 3GPP only.

Format

Type	Command	Response
Execute	AT+CUSD=[<n>[,<str>[,<dcs>]]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CUSD?<CR>	<CR><LF>+CUSD: <m>[,<str>,<dcs>]<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CUSD=?<CR>	<CR><LF>+CUSD: (range of supported <m>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<n>	specifies whether to display return codes 0: do not display return codes 1: display return codes 2: Cancel the request (not available for read command)
<str>	USSD string. If this parameter is not set, the module does not access network.
<dcs>	3GPP TS 23.038 [25] Cell Broadcast Data Coding Scheme in integer format (default 0).
<m>	0: no further user action required 1: further user action required 2: USSD terminated by network 3: other local client has responded 4: operation not supported 5: network response times out

Example

```
AT+CUSD=1,"*100#",15                                                    The operator supports
OK                                                                      this data service.

+CUSD: 2,
"04110430043B0430043D0441003A003300350031002C0035003104400020",
72
AT+CUSD=?                                                               Query the value range
+CUSD: (0-2)                                                            of parameters.

OK
```

4 SMS Commands

4.1 AT+CSMS – Selecting SMS Services

To select an SMS service among SMS-MO, SMS-MT, and SMS-CB

This command is not supported on a CDMA network.

Format

Type	Command	Response
Set	AT+CSMS=<service><CR>	<CR><LF>+CSMS: <mt>,<mo>,<bm><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CSMS?<CR>	<CR><LF>+CSMS: <service>,<mt>,<mo>,<bm><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CSMS=?<CR>	<CR><LF>+CSMS: (value range of <service>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<service>	0: GSM03.40 and GSM03.41. SMS-related AT commands support GSM07.05 Phase 2. 1: GSM03.40 and GSM03.41. SMS-related AT commands support GSM07.05 Phase 2+.
<mt>,<mo>,<bm>	0: not support 1: support

The default settings of this command are 0, 1, 1, 1.

Example

```
AT+CSMS=1
```

```
+CSMS: 1,1,1                                    Set SMS service to 1.

OK
AT+CSMS?                                        Query the current parameter values.
+CSMS: 1, 1, 1, 1

OK
AT+CSMS=?                                       Query the value range of SMS service.
+CSMS: (0,1)

OK
```

4.2 AT+CPMS – Setting Preferred SMS Storage

To set preferred SMS storage.

The settings by this commands are saved after the module is powered off.

Format

Type	Command	Response
Set	AT+CPMS=<mem1><CR>	<CR><LF>+CPMS: <used1>, <total1>, <used2>, <total2>, <used3>, <total3><CR><LF> <CR><LF>OK<CR><LF>
Query	AT+CPMS?<CR>	<CR><LF>+CPMS: <mem1>, <used1>, <total1>, <mem2>, <used2>, <total2>, <mem3>, <used3>, <total3><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CPMS=?<CR>	<CR><LF>+CPMS: (list of supported <mem1>s), (list of supported <mem2>s),(list of supported <mem3>s) <CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mem1> Memory from which SMS messages are read and deleted, string type
"SM": SIM only
"ME": ME only
"MT": any of storages associated with ME (SIM first)
"SR": Status Report Storage

<used> Used quantity

<total> Total capacity of the storage

Example

```
AT+CPMS="SM"                                          Set the SMS storage to "SM",
+CPMS: 0, 50, 0, 50, 0, 50                            that is, store SMS messages in
                                                      SIM card.
OK
AT+CPMS?                                              Query the capacity of current
+CPMS: "SM", 0, 50, "SM", 0, 50,"SM", 0, 50           SMS storage.

OK
AT+CPMS=?                                             Query the available storages.
+CPMS: ("SM,ME"),("SM,ME"),("SM,ME")

OK
AT+CPMS="SM"                                          No SIM card is inserted.
+CMS ERROR: 314
```

4.3 AT+CMGF – Setting SMS Inputting Mode

To set the SMS inputting mode

Format

Type	Command	Response
Set	AT+CMGF=[<mode>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CMGF?<CR>	<CR><LF>+CMGF: <mode><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CMGF=?<CR>	<CR><LF>+CMGF: (range of supported <mode>s) <CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode> 0: PDU mode (default)
1: text mode

Example

```
AT+CMGF=1               Set the SMS to text mode.
OK
AT+CMGF?                Query the current mode of SMS message input.
+CMGF: 1

OK
AT+CMGF=?               Query the value range of SMS mode setting.
+CMGF: (0-1)

OK
```

4.4 AT+CSCS – Setting the TE Character Set

To set the format of the TE character set.

The default character set is "IRA".

Format

Type	Command	Response
Set	AT+CSCS=[<chset>]<CR>	<CR><LF>OK<CR><LF>
Query	AT+CSCS?<CR>	<CR><LF>+CSCS: <chset><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CSCS=?<CR>	<CR><LF>+CSCS: (list of supported <chset>s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<chset>

- "GSM": default GSM alphabet (GSM03.38.6.2.1)
- HEX": string consisting of 0x00 to 0xFF in hexadecimal format. E.g. 032FE6 is 3 8-bit characters, which are respectively 3, 47, and 230. Conversion is not required between these characters and the source MT character set. (Not supported)
- "IRA": international reference alphabet (ITU-T T.50)
  "8859-1": ISO 8859 Latin 1 character set (not supported)
- "UCS2": 16-bit universal multiple-octet coded character set (USO/IEC10646). The UCS2 character string is converted into a hexadecimal number (ranging from 0x0000 to 0xFFFF). UCS2 encoding is used only in some character string of the statement.

Example

```
AT+CSCS="GSM"                              Set GSM character set.
OK
AT+CSCS?                                   Query the format of current character set.
+CSCS: "IRA"

OK
AT+CSCS=?                                  Query the character set formats that the
+CSCS: ("IRA","UCS2","HEX","GSM")          module supports.

OK
```

4.5 AT+CNMI – Setting SMS Indication Mode

To set the mode how the module informs users of new SMS messages received from the network.

Format

Type	Command	Response
Set	AT+CNMI=[<mode>[,<mt>[,<bm>[,<ds> [,<bfr>]]]]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CNMI?<CR>	<CR><LF>+CNMI: [<mode>[,<mt>[,<bm>[,<ds>[,<bfr>]]]]]<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CNMI=?<CR>	<CR><LF>+CNMI: (value range of <mode>),(value range of <mt>),(list of supported <bm>s),(value range of <ds>s),(value range of <bfr>s) <CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode> Set the instruction mode after receiving SMS messages. The default value is 0.
0: SMS instruction codes can be saved in the buffer of the module. If the TA is full, the old codes can be saved in other place or replaced with new codes. (default)
1: when the module is online, it will discard saved SMS instruction codes and reject new codes. In other situations, the codes are displayed on the end device.
2: when the module is online, the SMS instruction codes are saved in the buffer of the module. After the connection is released, the SMS instruction codes are output through UART. In other situations, codes are directly displayed on the end device.

3: when the module is online, SMS indicator code and other data are transmitted together and the code will be displayed on the device.

<mt> Set the format of the new SMS instruction codes. The default value is 0.

0: SMS instruction codes will not be sent to the end device (default)

1: The format of the new SMS instruction codes is +CMTI: "MT" ,<index>. The SMS message is stored rather than directly displayed.

2: The format of the new SMS instruction codes is **+CMT :<oa>,<scts>,<tooa>,<lang>,<encod>,<priority>[,<cbn>],<length><CR><LF><data> (text mode)**. SMS messages are directly displayed rather than stored.

3: Use the report codes defined by **<mt>=2 t**o transmit SMS instruction codes to the end device. The SMS instruction codes in other modes are the same as that of **<mt>=1**.

<bm> Set the format of the new cell broadcast codes. The default value is **1**.

0: not send the instruction information of new cell broadcast. The cell broadcast will not be stored.

1: the cell broadcast instruction code is **+CBMI:"BC" ,<index>** and the cell broadcast is stored. (default)

2: the format of the new cell broadcast instruction codes is **+CBM:<oa>,[<alpha>,]<scts>[,<tooa>,<length>] <CR><LF><data>**(text mode). The cell broadcast will be directly displayed rather than stored.

<ds> report status of SMS message sending. The default value is 0.

0: no status report of SMS message sending

1: the format of the SMS sending status report is **+CDS :<fo>,<mr>,[<ra>],[<tora>],<scts>, <dt>,<st>**(text mode).

2: if a status report is stored, then the following unsolicited result code is sent: +CDSI:<mem>,<index>

<bfr> The default value is 0.

0: when **<mode>** is set to **1** or **2**, codes defined by this command and stored in TA will be sent to TE. The module will return **OK** before transmitting the codes.

1: when **<mode>** is set to **1** or **2**, the codes defined by this command and stored in TA will be cleared.

- The default settings of this command are **0, 0, 0, 0, 1**.
- The recommended setting is **+CNMI: 2,1,0,0,0** (new messages are stored on SIM card rather than displayed directly). **+CNMI: 2,2,0,0,0** (new messages are displayed rather than stored on SIM card) is not supported.
- SMS message types:

Class 0: Displayed not stored

Class 1: Stored in ME

Class 2: Stored in SIM

Class 3: Sent to TE

Example

```
AT+CNMI=1,1,0,0,0                          Set the SMS message indication mode.
OK
```

```
AT+CNMI=?                                   Query the value ranges of the parameters.
+CNMI: (0-2),(0-3),(0,2,3),(0-1),(0-1)

OK
AT+CNMI?                                    Query the current setting of the
+CNMI: 1,1,0,0,0                            parameters.

OK
```

4.6 AT+CMGR – Reading SMS Messages

To read SMS messages stored in current memory (use the **AT+CPMS** command to specify the current memory)
If the received message is unread, its status in the storage changes to received read after executing this command.

Format

Type	Command	Response
Execute	AT+CMGR=<index><CR>	Text mode (+CMGF=1) • SMS-DELIVER <CR><LF>+CMGR: <stat>,<oa>,[<alpha>],<scts>[,<tooa>,<fo>, <pid>,<dcs>,<sca>,<tosca>,<length>] <CR><LF><data><CR><LF> <CR><LF>OK<CR><LF> • SMS-SUBMIT: <CR><LF>+CMGR: <stat>,<da>,[<alpha>][,<toda>, <fo>,<pid>, <dcs>,[<vp>],<sca>,<tosca>,<length>] <CR><LF><data><CR><LF> <CR><LF>OK<CR><LF> • SMS-STATUS-REPORT: <CR><LF>+CMGR: <stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>, <st> • SMS-COMMAND: <CR><LF>+CMGR: <stat>,<fo>,<ct>[,<pid>,[<mn>], [<da>], [<toda>],<length> <CR><LF><data>] • CBM-STORAGE: <CR><LF>+CMGR: <stat>,<sn>,<mid>,<dcs>, <page>,<pages> <CR><LF><data>

	PDU mode (+CMGF=0) <CR><LF>+CMGR: <stat>,[<alpha>],<length> <CR><LF><pdu><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<index> location value **<index>** from preferred message storage **<mem1>** to the TE

<stat> Status of SMS messages in the storage

- TEXT mode
  "REC UNREAD": received unread
  "REC UNREAD": received read
  "STO UNSENT": stored unsent
  "STO SENT": stored sent
- PDU mode
  0: received unread
  1: received read
  2: stored unsent
  3: stored sent

<oa> String type, 3GPP TS 23.040 TP-Originating-Address Address-Value field. BCD numbers (or GSM 7-bit default alphabet characters) are converted to characters of the currently selected TE character set (refer to AT+CSCS command in 3GPP TS 27.007). The type of address is given by **<tooa>**.

<alpha> String type alphanumeric representation of **<da>** or **<oa>** corresponding to the entry found in MT phonebook. Implementation of this feature is manufacturer specified. The used character set should be the one selected with AT+CSCS command (see definition of this command in 3GPP TS 27.007).

<scts> String type, 3GPP TS 23.040 TP-Service-Centre-Time-Stamp (refer to **<dt>**).

<tooa> Integer type, 3GPP TS 24.011 TP-Originating-Address Type-of-Address octet (default refer to **<toda>**).

<fo> Depending on the command or result code: First octet of 3GPP TS 23.040 SMS-DELIVER, SMS-SUBMIT (default 17), SMS-STATUS-REPORT, or SMS-COMMAND in integer format. If a valid value has been entered once, the parameter can be omitted.

<pid> 3GPP TS 23.040 TP-Protocol-Identifier in integer format (default 0).

<dcs> Depending on the command or result code: 3GPP TS 23.038 SMS Data Coding Scheme (default 0), or Cell Broadcast Data Coding Scheme in integer format.

<sca> 3GPP TS 24.011 RP SC address Address-Value field in string format. BCD numbers (or GSM 7-bit default alphabet characters) are converted to characters of the currently selected TE character set (refer to AT+CSCS command in 3GPP TS 27.007). The type of address is given by **<tosca>**.

<tosca> Integer type, 3GPP TS 24.011 RP SC address Type-of-Address octet (default refer to **<toda>**).

<length> Integer type, indicating in the text mode (AT+CMGF=1) the length of the message body

	<data> in characters, or in PDU mode (AT+CMGF=0) the length of the actual TP data unit in octets (i.e. the RP layer SMSC address octets are not counted in the length).
<data>	If **<dcs>**indicates that GSM 03.38 default alphabet is used and **<fo>** indicates that GSM 03.40 TP-User-Data-Header-Indication is not set: • ME/TA converts GSM alphabet into current TE character set when TE character set is not configured to HEX by +CSCS. • ME/TA converts each 7-bit octet into hexadecimal numbers containing two IRA characters when TE character set is configured to HEX by +CSCS. If **<dcs>**indicates that 8-bit or UCS2 data coding scheme is used, or **<fo>** indicates that GSM 03.40 TP-User-Data-Header-Indication is set: ME/TA converts each 8-bit octet into hexadecimal numbers containing two IRA characters (e.g. octet with integer value 42 is presented to TE as two characters 2A (IRA 50 and 65)
<da>	String type, 3GPP TS 23.040 TP-Destination-Address Address-Value field. BCD numbers (or GSM 7-bit default alphabet characters) are converted to characters of the currently selected TE character set (refer to **AT+CSCS** command in 3GPP TS 27.007). The type of address is given by **<toda>**
<toda>	Integer type, 3GPP TS 24.011 TP-Destination-Address Type-of-Address octet
<vp>	GSM 03.40 TP-Validity-Period Its format determined by **<fo>** of SMS-SUBMIT: integer (167 by default) or string type (refer to **<dt>**)
<mr>	Integer type, GSM 03.40 TP-Message-Reference.
<ra>	String type, GSM 03.40 TP-Recipient-Address, refer to **AT+CSCS**.
<tora>	Integer type, GSM 04.11 TP-Recipient-Address Type-of-Address, refer to **<toda>**.
<dt>	String type, GSM 03.40 TP-Discharge-Time, in format of yy/MM/dd,hh:mm:ss±zz
<st>	Integer type, GSM 03.40 TP-Status
<ct>	Integer type, GSM 03.40 TP-Command-Type
<sn>	Integer type, GSM 03.41 CBM Serial Number.
<mid>	Integer type, GSM 03.41 CBM Message Identifier.
<page>	Integer type, GSM 03.41 CBM Page Parameter 4-7 bit
<pages>	Integer type, GSM 03.41 CBM Page Parameter 0-3 bit
<pdu>	In the case of SMS: 3GPP TS 24.011 SC address followed by 3GPP TS 23.040 TPDU in hexadecimal format: ME/TA converts each octet of TP data unit into two IRA character long hexadecimal number (e.g. octet with integer value 42 is presented to TE as two characters 2A (IRA 50 and 65)).

Example

```
AT+CMGR=1                                                    Read the message
+CMGR: "REC READ","66421",,"11/09/13,16: 37: 59+32"          indexed as 1.
050003140401E27778592EA7E7EBE9373C3C279BCF68F59AADC7FED62779BA596
D7EBAEB5B91EBD16A5D46C35F98406A744E311A95C32594DA75688B50EADACA6D
689150EADF1B2BC5E579AD575E5B5582D5EABD5624C36A3D56C375C0E1693CD68
35DB0D9783A15C91D2E06BDAA558AC1F60C52B937CADCD2B747AA9021BDEC627E
8E9441BD42655DEF446
```

```
OK
AT+CMGF=0                                                    Set PDU mode.
OK
AT+CSCS="UCS2"
OK

+CMTI: "SM",39
AT+CMGR=39                                                   Incoming SMM.
+CMGR: 0,,23                                                 Read the message.
0891683110501905F0240BA18177377949F500004130623125032304683 41A0D

OK
AT+CMGF=1                                                    Set the text mode.
OK
AT+CSCS="GSM"
OK

+CMTI: "SM",40                                               Incoming SMS
AT+CMGR=40                                                   message.
+CMGR: "REC UNREAD","18777397945",,"14/03/26,13: 57: 58+32"  Read the SMS
hello world                                                  message.

OK
```

4.7 AT+CMGL – SMS Message List

To read SMS messages of one type from the current memory specified by the **+CPMS** command

Format

Type	Command	Response
Execute	AT+CMGL[=<stat>]<CR>	Text mode (+CMGF=1) • SMS-SUBMITs or SMS-DELIVERs: <CR><LF>+CMGL: <index>,<stat>,<oa/da>,[<alpha>], [<scts>][,<tooa/toda>,<length>]<CR><LF> <data><CR><LF>[<CR><LF>+CMGL: <index>,<stat>,<da/oa>,[<al pha>],[<scts>][,<tooa/toda>,<length>]<CR><LF><data >[...]]<CR><LF> <CR><LF>OK<CR><LF> • SMS-STATUS-REPORTs: <CR><LF>+CMGL: <index>,<stat>,<fo>,<mr>,[<ra>], [<tora>],<scts>,<dt>,<st><CR><LF> <CR><LF>+CMGL: <index>,<stat>,<fo>,<mr>,[<ra>],

		[<tora>],<scts>,<dt>,<st>[...]] • SMS-COMMANDs: <CR><LF>+CMGL: <index>,<stat>,<fo>,<ct> [<CR><LF>+CMGL: <index>,<stat>,<fo>,<ct>[...]] • CBM storage: <CR><LF>+CMGL: <index>,<stat>,<sn>,<mid>, <page>,<pages><CR><LF><data> [<CR><LF>+CMGL: <index>,<stat>,<sn>,<mid>, <page>,<pages><CR><LF><data>[...]] PDU mode (+CMGF=0) <CR><LF>+CMGL: <index>,<stat>,[<alpha>],<length><CR><LF><pdu><CR><LF> [<CR><LF>+CMGL: <index>,<stat>,[<alpha>],<length><CR><LF><pdu><CR><LF> [...]] <CR><LF>OK<CR><LF>
Test	AT+CMGL=?<CR>	<CR><LF>+CMGL: (list of supported <stat>s) <CR><LF> <CR><LF>OK<CR><LF>

Parameter

<stat>	String type or numeric type When set **AT+CMGF=1**, "REC UNREAD": received unread "REC UNREAD": received read "STO UNSENT": stored unsent "STO SENT": stored sent "ALL": all SMS messages When set AT**+CMGF=0**, 0: received unread 1: received read 2: stored unsent 3: stored sent 4: all SMS messages
<index>	Location value **<index>** from preferred message storage **<mem1>** to the TE
<oa>	String type, 3GPP TS 23.040 TP-Originating-Address Address-Value field. BCD numbers (or GSM 7-bit default alphabet characters) are converted to characters of the currently selected TE character set (refer to AT+CSCS command in 3GPP TS 27.007). The type of address is given by **<tooa>**.
<da>	String type, 3GPP TS 23.040 TP-Destination-Address Address-Value field. BCD numbers (or GSM 7-bit default alphabet characters) are converted to characters of the currently selected TE character set (refer to AT+CSCS command in 3GPP TS 27.007). The type of address is given by **<toda>**

<alpha>	String type alphanumeric representation of **<da>** or **<oa>** corresponding to the entry found in MT phonebook. Implementation of this feature is manufacturer specified. The used character set should be the one selected with **AT+CSCS** command (see definition of this command in 3GPP TS 27.007).
<scts>	String type, 3GPP TS 23.040 TP-Service-Centre-Time-Stamp (refer to **<dt>**).
<tooa>	Integer type, 3GPP TS 24.011 TP-Originating-Address Type-of-Address octet (default refer to **<toda>**).
<toda>	Integer type, 3GPP TS 24.011 TP-Destination-Address Type-of-Address octet
<length>	Number of octets of the given TP-level data unit (octets that do not contain the service center address)
<data>	If **<dcs>**indicates that GSM 03.38 default alphabet is used and **<fo>** indicates that GSM 03.40 TP-User-Data-Header-Indication is not set: • ME/TA converts GSM alphabet into current TE character set when TE character set is not configured to HEX by +CSCS. • ME/TA converts each 7-bit octet into hexadecimal numbers containing two IRA characters when TE character set is configured to HEX by +CSCS. If **<dcs>**indicates that 8-bit or UCS2 data coding scheme is used, or **<fo>** indicates that GSM 03.40 TP-User-Data-Header-Indication is set: • ME/TA converts each 8-bit octet into hexadecimal numbers containing two IRA characters (e.g. octet with integer value 42 is presented to TE as two characters 2A (IRA 50 and 65)
<fo>	Depending on the command or result code: First octet of 3GPP TS 23.040 SMS-DELIVER, SMS-SUBMIT (default 17), SMS-STATUS-REPORT, or SMS-COMMAND in integer format. If a valid value has been entered once, the parameter can be omitted.
<mr>	Integer type, 3GPP TS 23.040 TP-Message-Reference
<ra>	String type, 3GPP TS 23.040 TP-Recipient-Address Address-Value field. BCD numbers (or GSM default alphabet characters) are converted to characters of the currently selected TE character set (refer to **AT+CSCS** command). The type of address is given by **<tora>**.
<tora>	Integer type, 3GPP TS 24.011 TP-Recipient-Address Type-of-Address octet (default refer to **<toda>**).
<scts>	String type, 3GPP TS 23.040 TP-Service-Centre-Time-Stamp (refer to **<dt>**).
<dt>	GSM 03.40 TP-Discharge-Time, in format of yy/MM/dd,hh:mm:ss±zz
<st>	Integer type, GSM 03.40 TP-Status.
<ct>	Integer type, GSM 03.40 TP-Command-Type
<sn>	Integer type, GSM 03.41 CBM Serial Number
<mid>	Integer type, GSM 03.41 CBM Message Identifier
<page>	Integer type, GSM 03.41 CBM Page Parameter 4-7 bit
<pages>	Integer type, GSM 03.41 CBM Page Parameter 0-3 bit
<pdu>	In the case of SMS: 3GPP TS 24.011 SC address followed by 3GPP TS 23.040 TPDU in hexadecimal format: ME/TA converts each octet of TP data unit into two IRA character long hexadecimal number (e.g. octet with integer value 42 is presented to TE as two characters 2A (IRA 50 and 65)).

Example

```
AT+CMGL="ALL"
+CMGL: 1,"REC READ","10010",,"14/06/23,14:42:27+32"
0500034F0302672C77ED4FE14E2D768452694F596D4191CF5305542B53E052A053056D4191CFFF0C8BF76CE8610
F533A52064F7F7528FF093002672C6B2167E58BE27ED3679C5B5857285EF665F6FF0C8BF74EE551FA8D264E3A51
C63002767B96468054901A624B673A84254E1A53850020007700610070002E00310030003000310030002E00630
06F006D

+CMGL: 2,"REC READ","10010",,"14/06/23,14:42:27+32"
0500034F03016E2999A863D0793AFF0C622A6B62003667080032003265E5FF0C60A85F536708595799105185 6D4
191CF5DF24F7F752800340033002E00360031004D0042FF0C52694F596D4191CF003200350036002E0033003900
4D0042FF08598260A88BA28D2D4E867EA256F4811662164E9196C0621660A6005400566D4191CF53E052A05305F
F0C5219

OK.
AT+CMGL=?                                          Query in text format (AT+CMGF=1).
+CMGL: "REC UNREAD", "REC READ", "STO UNSENT",
"STO SENT", "ALL"

OK

AT+CMGL=4
+CMGL: 0,1,,24
0891683108705505F0240D91684137471463F400002221
32908374230433DA8C06

OK
AT+CMGL=?                                          Query in PDU format (AT+CMGF=0).
+CMGL: (0-4)

OK

AT+CMGL=ALL                                        A pair of quotation marks ("") is
ERROR                                              required for the parameter.
AT+CMGF=1                                          The parameter should be set to 0.
OK
AT+CMGL=4
ERROR
AT+CMGF=0                                          The parameter should be set to 1.
OK
AT+CMGL="ALL"
ERROR
```

4.8 AT+CMGS – Sending SMS Messages

To send an SMS message from the module to the network

The network will return reference value **<mr>** to the module after the SMS message is sent successfully.

Format

Type	Command	Response
Execute	• AT+CMGS=<da>[,<toda>]<CR> text is entered<Ctrl+Z/ESC> (Text mode) • AT+CMGS=<length><CR> PDU is given<Ctrl+Z/ESC> (PDU mode)	• Text mode (+CMGF=1): <CR><LF>+CMGS: <mr>[,<scts>]<CR><LF> <CR><LF>OK<CR><LF> • PDU mode (+CMGF=0): <CR><LF>+CMGS: <mr>[,<ackpdu>]<CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<da>	The destination number to which the SMS message is sent in text mode
<toda>	Type of destination address.3GPP TS 24.011 TP-Destination-Address Type-of-Address octet in integer format.
<text>	SMS message content in text mode
<length>	the byte length of the SMS message content in PDU mode
<mr>	storage location
<CR>	end character
<Ctrl+Z>	indicates the end of the input message, ➔➔ in the example.
<ESC>	indicates giving up the input message
<scts>	Service center time stamp. 3GPP TS 23.040 TP-Service-Centre-Time-Stamp in time-string format (refer to <dt>).
<ackpdu>	3GPP 23.040 RP-User-Data element of RP-ACK PDU

Example

```
AT+CMGS="66358"<CR>                                  Text mode(+CMGF=1)
> This is the text➔                                  ➔is the symbol after pressing Ctrl+Z.
+CMGS: 171

OK
AT+CMGS="15889758493"<CR>                            AT+CMGF=1 might not be executed.
> This is the text➔

ERROR
AT+CMGS=33<CR>                                       PDU mode (+CMGF=0)
```

```
>0891683108705505F001000B815118784271F2000814
6DF157335E025B9D5B89533A59276D6A80545EFA
+CMGS: 119
OK
```

4.9 AT+CMGW – Writing SMS Messages

To write an SMS message into the memory

The location information **<index>** will be returned after the message is saved correctly.

If PDU messages is sent through a UART debugging tool, press the **Enter** button or enter <CR> in hexadecimal format.

Format

Type	Command	Response
Execute	• AT+CMGW[=<oa/da>[,<tooa/toda>[,<stat>]]]<CR>text is entered<Ctrl+Z/ESC>(text mode) • AT+CMGW=<length>[,<stat>]<CR>PDU is given<Ctrl+Z/ESC> (PDU mode)	<CR><LF>+CMGW:<index><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<da>	The destination number to which the SMS message is sent in text mode
<toda>	Type of destination address.3GPP TS 24.011 TP-Destination-Address Type-of-Address octet in integer format.
<stat>	Status of SMS messages in the storage
<text>	SMS message content in text mode
<length>	The byte length of the SMS message content in PDU mode
<index>	Location information
<CR>	End character
<Ctrl+Z>	Indicates the end of the input message
<ESC>	Indicates giving up the input message

Example

```
AT+CMGW="091137880"<CR>                                    Text mode (+CMGF=1)
>"This is the text"<Ctrl+Z>
+CMGW: 15
```

```
OK
AT+CMGW=091137880                                   A pair of quotation marks ("") is
ERROR                                               required for the number in text
                                                    mode.
AT+CMGW=31<CR>                                      PDU mode (+CMGF=0)
>0891683108705505F001000B813124248536F3000812004
00026002A535A53D153A653C1532052C7<Ctrl+Z>
+CMGW: 1

OK
```

4.10 AT+CMSS – Sending Messages from Storage

To send an SMS message specified by **<index>** in the memory (SMS-SUBMIT)

The network returns reference value **<mr>** to the end device after the SMS message is sent successfully.

Format

Type	Command	Response
Execute	AT+CMSS=<index>[,<da>[,<toda>]]<CR>	• Text mode (+CMGF=1)]: <CR><LF>+CMSS: <mr>[,<scts>]<CR><LF> <CR><LF>OK<CR><LF> • PDU mode (+CMGF=0): <CR><LF>+CMSS: <mr>[,<ackpdu>]<CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<index>	Message location
<da>	The destination number of the SMS messages
<toda>	Type of address
<mr>	Message reference number
<scts>	Service center time stamp
<ackpdu>	3GPP 23.040 RP-User-Data element of RP-ACK PDU

Example

```
AT+CMSS=2                                   Send the SMS messages stored in memory 2.
+CMSS: <mr>

OK
AT+CMSS=2                                   No SMS message is stored in memory 2 or the SMS
ERROR                                       message number in memory 2 is incorrect.
```

4.11 AT+CMGD – Deleting SMS Messages

To delete SMS messages from the current memory.

Format

Type	Command	Response
Execute	AT+CMGD=<index>[,<delflag>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CMGD=?<CR>	<CR><LF>+CMGD: (list of <index>),(value range of <delflag>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<index> The recording number of the stored SMS messages

<delflag> Integer

0: delete the SMS messages with the specified recording numbers.

1: delete all read SMS messages.

2: delete all read and sent SMS messages.

3: delete all read, sent, and unsent SMS messages.

4: delete all messages.

If **<delflag>** is set, ignore the parameter **<index>**.

Example

```
AT+CMGD=0,3                             Delete all read, sent, and unsent SMS messages.
```

```
OK
AT+CMGD=?                                Query the value ranges of parameters.
+CMGD: (0,1,2,3),(0-4))

OK
AT+CMGD=5
ERROR                                    The 5th message does not exist.
```

4.12 AT+CSCA – Setting SMS Center Number

To set the SMS center number.

This command is not supported on a CDMA network.

Format

Type	Command	Response
Set	AT+CSCA=<sca>[,<tosca>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CSCA?<CR>	<CR><LF>+CSCA: <sca>,<tosca><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<sca> SMS center number

<tosca> The format of the SMS center number.
129 indicates national number.
145 indicates international number.

Example

```
AT+CSCA="8613800755500",145              Set an international SMSC number.
OK
AT+CSCA=8613800755500,145                A pair of quotation marks ("") are not
OK                                       mandatory for SMSC number.
AT+CSCA?                                 Query the SMSC number.
+CSCA: "+8613800755500",145

OK
```

4.13 AT+CSMP – Setting Text Mode Parameters

To select required values for the additional parameters in the text mode, and set the validity period since the message is received from the SMSC, or the absolute time defining the end of the validity period.

Format

Type	Command	Response
Set	AT+CSMP=[<fo>[,<vp>[,<pid>[,<dcs>]]]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CSMP?<CR>	<CR><LF>+CSMP:<fo>,<vp>,<pid>,<dcs><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<fo> Determined by the command or the first 8 bits of the result code **GSM 03.40 SMS-DELIVER**; SMS-SUBMIT (default value: 17); or adopt the integer-type SMS-COMMAND (default value: 2)

<vp>

Value	Validity Period
0-143	(vp+1)*5mins, 12 hours at most
144-167	12hours +((vp–143)*30mins), 24 hours at most
168-196	(vp–166)*1day
197-255	(vp–192)*1week

<pid> Integer-type TP-protocol-ID (default value: 0)

<dcs> Encoding plan for integer-type cell broadcast data (default value: 0)

The default setting is **„0,0** on a 3GPP network.

Example

```
AT+CSMP=17,167,0,0                  Text mode parameters.
OK                                  No status report; the validity period of the
                                    information is 24 hours; Only messages in text format
                                    can be sent.
AT+CSMP?                            Query the current settings of the text mode.
+CSMP: 17,167,0,8
```

```
OK
```

4.14 AT+CSDH – Displaying the Parameters of the Text Mode

To set whether the detailed header information is displayed in the result code in text mode

This command is valid in text mode, which can be set by **AT+CMGF=1**.

Format

Type	Command	Response
Set	AT+CSDH=[<show>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CSDH?<CR>	<CR><LF>+CSDH: <show><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CSDH=?<CR>	<CR><LF>+CSDH: (value range of <show>) <CR><LF><CR><LF>OK<CR><LF>

Parameter

<show>: 0: not display (default value)
1: display

Example

```
AT+CSDH=0                                                    Set the header information to
OK                                                           not display
AT+CMGR=0                                                    Read the 0th message.
+CMGR:"RECREAD","13510895077",,"15/07/23,20:58:28+32"
Abc

OK
AT+CSDH=1                                                    Set the detailed header
OK                                                           information to display.
AT+CMGR=0
+CMGR:"RECREAD","13510895077",,"15/07/23,20:58:28+32",       Read the 0th message.
161,36,0,0,"+8613010888500",145,3
Abc
```

```
OK
AT+CSDH?                                        Query the current parameter
+CSDH: 0                                        setting of the command.

OK
AT+CSDH=?                                       Query the value range of
+CSDH: (0,1)                                    parameter in the command.

OK
```

4.15 AT+CSAS – Saving the Setting

To save the setting.

Only the parameter settings of AT+CSCA, AT+CSMP, and AT+CSCB can be saved by executing this command.

Format

Type	Command	Response
Execute	AT+CSAS[=<profile>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CSAS=?<CR>	<CR><LF>(list of supported <profile>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<profile> 0-255: save settings (manufacturer specific profile number where settings are to be stored)

Example

```
AT+CSAS=1                                       Save the setting.
OK
AT+CSAS=0                                       Save the setting.
OK

AT+CSAS=?                                       Query the valid parameter values for the
+CSAS: 0-255                                    command.
```

```
OK
```

5 TCP/UDP Client Commands

5.1 AT+NETAPN – Setting Network APN

To set the network APN.

Format

Type	Command	Response
Execute	AT+NETAPN="APN","USERNAME","PASSWORD"<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NETAPN?<CR>	<CR><LF>NETAPN: "APN","USERNAME","PASSWORD" <CR><LF>OK<CR><LF>

Parameter

APN	GPRS network access point
USERNAME	GPRS user name
PASSWORD	GPRS password

Example

```
AT+NETAPN="CMNET","",""          Set GPRS APN to CMNET and leave user account and password
OK                               blank.
AT+NETAPN=CMNET,,
ERROR                            A pair of quotation marks is required for each parameter.
AT+NETAPN?
+NETAPN:"","",""                 Query the current settings of APN parameter.
OK
```

5.2 AT+XIIC – Activating PDP Context

To activate PDP context.

Send **AT+CGDCONT** to set APN before executing this command.

Ensure that the module is registered on the network before the **AT+XIIC=1** command is executed.

Use **AT+GREG?** to check whether the module is registered on the network or not. If **+CREG: 0,1** or **+CREG: 0,5** is returned, the module is not registered on the network.

Format

Type	Command	Response
Execute	AT+XIIC=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+XIIC?<CR>	<CR><LF>+XIIC: 1,<ip><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<n>	0: Disconnect the PPP link 1: Activate the PPP link.
<ip>	IP address

Example

```
AT+XIIC=1                                          The module is required to set up a PPP link.
OK
AT+XIIC?                                           The PPP link is set up successfully and the
+XIIC:    1,10.107.216.162                         IP address is 10.107.216.162.
OK                                                 There are four spaces before 1.
AT+XIIC=1                                          Reconnect an established ppp link.return OK
OK
AT+XIIC?
+XIIC:    0,0.0.0.0                                The PPP link is set up successfully.
OK                                                 There are four spaces before 1.
AT+XIIC?
+XIIC:    1,10.6.162.140                           When PPP is activated in dual stack mode the
+XIIC:                                             IPV4 address is 10.6.162.140, and the IPV6
1,240e:bf:d209:b6e0:1:1:de27:e51a                  address is 240e:bf:d209:b6e0:1:1:de27:e51a
OK                                                 (there are 4 spaces before 1)
AT+XIIC=0
OK                                                 Disconnect a PPP link
```


5.3 AT+TCPSETUP – Setting Up TCP Connection

To set up a TCP Connection.

Use the **AT+XIIC=1** command to activate PDP context before sending this command.

In IPv6 single stack, ip_type is IPv6 by default. In IPv4 single stack and IPv4v6 double stack, ip_type is IPv4 by default.

Parameter [,<ip_type>] is temporarily not supported. <ip> can recognize ip_type automatically.

Format

Type	Command	Response
Execute	AT+TCPSETUP=<n>,<ip>,<port>[,<ip_type>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>+TCPSETUP: ERROR<CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	<CR><LF>+TCPSETUP: <n>,<result><CR><LF>	

Parameter

<n>	Socket ID, ranging from 0 to 5
<ip>	Destination IP address, in xx.xx.xx.xx or domain name format
<port>	Destination port ID in decimal ASCII code
<ip_type>	Type of the destination IP address. (Not supported temporarily) IPv4 IPv6
<result>	Result code OK FAIL ERROR1

Example

```
AT+TCPSETUP=0,220.199.66.56,6800                    Set up a connection to
OK                                                  220.199.66.56,6800 on socket 0.

+TCPSETUP: 0,OK                                     Successful
AT+TCPSETUP=0,neowayjsr.oicp.net,60010              Set up a connection to
OK                                                  neowayjsr.oicp.net, 60010 on
                                                    socket 0.
+TCPSETUP: 0,OK
```

```
                                                    Successful
AT+TCPSETUP=1,2408:84fb:213:bdc1:7d7b:b58b:2a92:d5  The link
64,25800,IPV6 (Not supported temporarily)           to
OK                                                  2408:84fb:213:bdc1:7d7b:b58b:2a92:
                                                    d564,25800 is successfully set up on
+TCPSETUP: 1,OK                                     socket 1.
+TCPCLOSE: 0,Link Closed                            The socket is closed.
AT+TCPSETUP=1,192.168.20.6,7000                     Fails to set up a connection to
OK                                                  192.168.20.6,7000 on socket 1. The
                                                    server is not started, the IP
+TCPSETUP: 1,FAIL                                   address is incorrect, or the SIM
                                                    card is out of credit.
AT+TCPSETUP=0,neowayjsr.oicp.net,60010              A TCP/UDP connection has been set
OK                                                  up on socket 0.

+TCPSETUP: 0,ERROR1
AT+TCPSETUP=6,192.168.20.6,7000                     Parameters are set incorrectly.
+TCPSETUP: ERROR
AT+TCPSETUP=0.58.60.184.213.10012                   Parameters are set incorrectly.
ERROR
AT+TCPSET=0,58.60.184.213,10012                     The AT command is not complete.
ERROR
```

5.4 AT+UDPCLOSE – Closing a UDP Connection

To close a UDP connection

Format

Type	Command	Response
Execute	AT+UDPCLOSE=<n><CR>	<CR><LF>+IPSTATUS: <n>,<CONNECT or DISCONNECT or CONNECTING or DISCONNECTING>[,<TCP or UDP>,<send-buffer-size>] Or <CR><LF>ERROR<CR><LF>
URC	+UDPCLOSE: <n>,<result>	

Parameter

<n>	Socket ID, ranging from 0 to 5
<CONNECT or DISCONNECT or CONNECTING or DISCONNECTING>	• CONNECT • DISCONNECT • CONNECTING • DISCONNECTING>
<TCP or UDP>	Connection type, TCP or UDP

<send-buffer-size>	Available send buffer size of the module, in

Example

```
AT+UDPCLOSE=1                    The TCP link on socket 1 is closed successfully.
+UDPCLOSE: 1,OK
AT+UDPCLOSE=6                    Socket number error
+UDPCLOSE: ERROR
```

5.5 AT+TCPSEND – Sending TCP Data

To send TCP data.

The module will return > after this command is sent. Send TCP data 50 ms to 100 ms later.

Ensure that the TCP connection is set up before sending TCP data.

The **AT+IPSTATUS** command is recommended to check the buffer size before sending data.

Format

Type	Command	Response
Execute	AT+TCPSEND=<n>,<length>[[,<content>][,mode]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	+TCPSEND: <n>[,<length>],<result>	

Parameter

<n>	Socket ID, ranging from 0 to 5. A TCP connection is established on the socket.
<length>	The length of data to be sent, in bytes; when <content><mode>is missing, and the value range is 1~4096 When<mode>is 0, the value range of<length>is 1~1024 When<mode>is 1, the value range of<length>is 1~512
<content>	Data sent in command mode does not support escape mode
<mode>	HEX send enable switch 0: Send in ASCII mode 1: TX in HEX mode

Example

```
    AT+TCPSEND=0,1                          1-byte data is successfully sent through socket
    >                                       0.
    OK

    +TCPSEND: 0,1
    AT+TCPSEND=0,1024                       Send 1024-byte data. E.g.:123…
    >
    +TCPSEND: ERROR                         Congestion.
    AT+TCPSEND=0,1024,,1
    >                                       Send 1024-byte HEX data in Buffer mode
    OK                                      successfully.

    +TCPSEND: 0,1024
    AT+TCPSEND=0,16,"\/!@#$%^&*()_+=-"
    OK                                      Send data in command mode successfully.

    +TCPSEND: 0,16
    AT+TCPSEND=0,3,”313233”,1
    OK                                      Send HEX data in command mode successfully.

    +TCPSEND: 0,3
    AT+TCPSEND=0,10                         After the data sending command is input and > is
    >                                       returned, no more data is entered in 30 seconds.
    +TCPSEND: 0,OPERATION EXPIRED           Then the expiration information is displayed.
    AT+TCPSEND=0,1                          One-byte data fails to be sent on socket 0
    +TCPSEND: SOCKET ID OPEN FAILED         because the link is not established.
    AT+TCPSEND=0,4097                       4097-byte data fails to be sent on socket 0
    +TCPSEND: DATA LENGTH ERROR             because data length exceeds the limit.
AT+TCPSEND=0,1,01,1                         Send HEX data in command mode successfully.
    OK

    +TCPSEND: 0,1
```

5.6 AT+RECVMODE – Setting Receive Mode

To set the receive mode of TCP and UDP data.

Do not send this command during communication because it clears the buffer.

This command also works for UDP data.

The settings are not saved after the module is powered off.

Format

Type	Command	Response
Set	AT+RECVMODE=<n>[,<mode>]<CR>	<CR><LF>OK<CR><LF>

		Or <CR><LF>ERROR<CR><LF>
Query	AT+RECVMODE?<CR>	<CR><LF>+RECVMODE: <n>,<mode> <CR><LF>OK<CR><LF>
Test	AT+RECVMODE=?<CR>	<CR><LF>+RECVMODE: (list of supported <n>s), (list of supported <mode>s) <CR><LF>OK<CR><LF>

Parameter

<n>	receive mode 0: buffer the TCP or UDP data received and MCU sends command to read the data 1: print the TCP or UDP data received to UART directly (default) 2: print the TCP or UDP data received to UART directly (output in data mode directly as well)
<mode>	0: report in ASCII mode (default) 1: report in HEX mode

Example

```
AT+RECVMODE=0                          Set data receiving mode.
OK
AT+RECVMODE=1,1                        HEX report in command mode
OK
AT+RECVMODE=?
+RECVMODE: (0-2),(0-1)                 Query value ranges of parameters in this command
OK
AT+RECVMODE?
+RECVMODE: 1,0
OK
```

5.7 +TCPRECV – Notifying TCP Data Output

To notify TCP data output.

When the module receives TCP data from the network, the UART prints the data automatically.

Format

Type	Command
URC	+TCPRECV:<n>,<length>,<data><CR>

Parameter

<n>	Socket ID, ranging from 0 to 5
<length>	Length of the data received
<data>	Data received Add **0x0d 0x0a** to the end of the data. Identify the end based on **<length>**.

Example

```
+TCPRECV:0,10,1234567890                    10-byte data is successfully received on
                                            socket 0. The data is 1234567890.
```

5.8 AT+TCPREAD – Reading TCP Data

To read TCP data from the buffer.

Execute **+RECVMODE** to select data receive mode.

Format

Type	Command	Response
Set	AT+TCPREAD=<n>,<length><CR>	<CR><LF>+TCPREAD: <n>,<length>,<data> <CR><LF>OK<CR><LF> **Or** <CR><LF>ERROR<CR><LF>

Parameter

<n>	Socket ID, ranging from 0 to 5.
<length>	Maximum length of data allowed to read, ranging from 1 to 2048, byte.
<data>	Data that is read.

Example

```
+TCPRECV: 0,10                              Socket 0 receives data.
AT+TCPREAD=0,100                            Read data.
+TCPREAD: 0,10,1234567890                   The data read is 1234567890.
OK
```

5.9 AT+TCPCLOSE – Closing a TCP Connection

To close a TCP connection

Formats

Type	Command	Response
Execute	AT+TCPCLOSE=<n><CR>	<CR><LF>+TCPCLOSE: <n>,OK<CR><LF> Or <CR><LF>+TCPCLOSE: ERROR<CR><LF>
URC	<CR><LF>+TCPCLOSE: <n>,<result><CR><LF>	

Parameter

<n>	Socket ID, ranging from 0 to 5
<result>	OK FAIL Link Closed

Example

```
AT+TCPCLOSE=1                    Close the TCP connection.
+TCPCLOSE: 1,OK                  The TCP connection on socket 1 is closed
                                 successfully.
AT+TCPCLOSE=2                    Socket number error
+TCPCLOSE: ERROR

+TCPCLOSE: 0,Link Closed          The server sends TCP connection closing command or
                                 the network encounters abnormality or weak signals.
```

5.10 AT+UDPSETUP – Setting up a UDP Connection

To set up a UDP connection

Use the **AT+XIIC=1** command to activate PDP context before executing this command.

In IPv6 single stack, ip_type is IPv6 by default. In IPv4 single stack and IPv4v6 double stack, ip_type is IPv4 by default.

Parameter [,<ip_type>] is temporarily not supported. <ip> can recognize ip_type automatically.

Format

Type	Command	Response
Execute	AT+UDPSETUP=<n>,<ip>,<port>[,<ip_type>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>+UDPSETUP: ERROR<CR><LF>
URC	<CR><LF>+UDPSETUP: <n>,<result><CR><LF>	

Parameter

<n>	Socket ID, ranging from 0 to 5
<ip>	Destination IP address, in xx.xx.xx.xx format or domain name format (www.*XXXX*.com)
<port>	Destination port ID in decimal ASCII code
<ip_type>: Type of the destination IP address.	IPv4 IPv6 (Not supported temporarily)
<result>	OK FAIL ERROR1

Example

```
AT+UDPSETUP=1,220.199.66.56,7000                          The connection to 220.199.66.560.7000
OK                                                        is successfully set up on socket 1.

+UDPSETUP: 1,OK

AT+UDPSETUP=0,neowayjsr.oicp.net,60010                    Set up a connection to
OK                                                        neowayjsr.oicp.net,60010 on socket 0

+UDPSETUP: 0,OK                                           Successful

AT+UDPSETUP=1,2408:84fb:213:bdc1:7d7b:b58b:2a92:d564,25
801,IPV6 (Not supported temporarily)                      The link to
OK                                                        2408:84fb:213:bdc1:7d7b:b58b:2a92:d564
                                                          ,25801 is successfully set up on
+UDPSETUP: 1,OK                                           socket 1.

AT+UDPSETUP=0,58.60.184.213,11008                         A TCP/UDP connection is already set up
OK                                                        on socket 0.

+UDPSETUP: 0,ERROR1

AT+UDPSETUP=1,192.168.20.6,7000                           Fail to set up the connection to
```

```
OK                                                    192.168.20.6,7000 on socket 1.

+UDPSETUP: 1,FAIL
AT+UDPSETUP=6,192.168.20.6,6800                       Socket ID is set incorrectly.
+UDPSETUP: ERROR
AT+UDPSETUP=0.58.60.184.213.10012                     Punctuation mark is used incorrectly.
ERROR
AT+UDPSET=0,58.60.184.213,10012                       The AT command is not complete.
ERROR
```

5.11 AT+UDPSEND – Sending UDP Data

To send UDP data.

The module will return > after this command is sent. Send UDP data 50 ms to 100 ms later.

Ensure that the UDP link is set up before sending UDP data.

To decrease the packet loss rate, do not send more than 1472 each time.

Backslash is used for data link escape. For how to send quotation marks or backlash in character string, see the example.

The **mode** parameter can be omitted. ASCII data sent supports escape mode by default.

When ASCII data is sent in command mode, length of the content of third parameter must be less than 1024 (including escape characters).

To send data containing more than 15 commas, use buffer mode.

The settings are not saved.

Format

Type	Command	Response
Execute	AT+UDPSEND=<n>,<length>[[,<content>][,mode]]<CR>	<CR><LF>> <CR><LF>OK<CR><LF> <CR><LF>+UDPSEND: <n>,<length> Or <CR><LF>+UDPSEND: <n>,OPERATION EXPIRED<CR><LF> Or

	<CR><LF>+UDPSEND: DATA LENGTH ERROR<CR><LF>

Parameter

<n>	Socket ID, ranging from 0 to 5. A UDP connection is established on the socket.
<length>	The length of data to be sent, in bytes; when <content><mode>is missing, and the value range is 1~4096 When<mode>is 0, the value range of<length>is 1~1024 When<mode>is 1, the value range of<length>is 1~512
<content>	Data sent in command mode does not support escape mode
<mode>	HEX send enable switch 0: Send in ASCII mode 1: TX in HEX mode

Example

```
AT+UDPSEND=0,2                                  Send 2-byte data on socket 0. Then send the
>                                               characters to be sent 50 ms to 100 ms after
OK                                              the module returns >.
                                                The data is sent successfully.

+UDPSEND: 0,2

AT+UDPSEND =0,512,,1                            Send 512-byte HEX data in buffer mode.
>
OK

                                                Data is sent successfully.

+UDPSEND: 0,512

AT+UDPSEND=0,16,"\/!@#$%^&*()_+=-"              Command mode
OK


+UDPSEND: 0,16

AT+UDPSEND=0,3,”313233”,1                       Command mode, HEX format
OK


+UDPSEND: 0,3

AT+UDPSEND=0,4097                               4097-byte data fails to be sent on socket 0
+UDPSEND: DATA LENGTH ERROR                     because data length exceeds the limit.

AT+UDPSEND=0,10                                 After the data sending command is input
>                                               and > is returned, no more data is entered
+UDPSEND: 0,OPERATION EXPIRED                   in 30 seconds. Then the expiration
                                                information is displayed.
```


5.12 +UDPRECV – Notifying UDP Data Ouput

To notify UDP data output.

When the module receives UDP data from the network, the UART prints the data automatically.

Format

Type	Command
URC	+UDPRECV: <n>,<length>[,<data>]<CR>

Parameter

<n>	Socket ID, ranging from 0 to 5
<length>	Length of the data received
<data>	Data received Identify the end based on **<length>**.

Example

```
+UDPRECV: 0,10,1234567890                    10-byte data is successfully received on
                                             socket 0. The data is 1234567890.
```

5.13 AT+UDPREAD – Reading UDP Data

To read UDP data.

You can use the AT+RECVMODE command to select the data receive mode.

Format

Type	Command	Response
Execute	AT+UDPREAD=<n>,<length><CR>	<CR><LF>+UDPREAD: <n>, <length>, <data><CR><LF> <CR><LF>OK<CR><LF> **Or** <CR><LF>ERROR<CR><LF>

Parameter

<n>	Socket ID, ranging from 0 to 5
<length>	Maximum length of data allowed to read
<data>	Data that is read, 1 to 2048 bytes

Example

```
+UDPRECV: 0                              Socket 0 receives data.
AT+UDPREAD=0,100                         Read data.
+UDPREAD: 0,10,1234567890                The data read is 1234567890.
OK
```

5.14 AT+UDPCLOSE – Closing a UDP Connection

To close a UDP connection

Format

Type	Command	Response
Execute	AT+UDPCLOSE=<n><CR>	<CR><LF>+UDPCLOSE: <n>,OK<CR><LF> Or <CR><LF>+UDPCLOSE:ERROR<CR><LF>
URC	+UDPCLOSE: <n>,<result>	

Parameter

<n>	socket ID, ranging from 0 to 5
<result>	OK FAIL

Example

```
AT+UDPCLOSE=1                    The TCP link on socket 1 is closed successfully.
+UDPCLOSE: 1,OK

AT+UDPCLOSE=6                    Socket number error
+UDPCLOSE: ERROR
```

5.15 AT+IPSTATUS – Querying TCP/UDP Socket Status

To query the TCP/UDP socket status.

Format

Type	Command	Response
Execute	AT+IPSTATUS=<n><CR>	<CR><LF>+IPSTATUS: <n>,<CONNECT or DISCONNECT>[,<TCP or UDP>,<send-buffer-size>] <CR><LF>OK<CR><LF> Or <CR><LF>+IPSTATUS:1,DISCONNECT<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	Socket ID, ranging from 0 to 5
<CONNECT or DISCONNECT>	Socket status, CONNECT or DISCONNECT, or CONNECTING or DISCONNECTING
<TCP or UDP>	Socket type, TCP or UDP
<send-buffer-size>	The size of the available send buffer on the module, in decimal ASCII mode, unit: byte. <send-buffer-size> is not supported on UDP connection.

Example

```
AT+IPSTATUS=0                                   A TCP connection has been set up on socket 0 and
+IPSTATUS: 0,CONNECT,TCP,10240                  the buffer size is 4096 bytes.

AT+IPSTATUS=0                                   A UDP connection has been set up on socket 0.
+IPSTATUS: 0,CONNECT,UDP,0

AT+IPSTATUS=1                                   No TCP or UDP connection is set up on socket 1.
+IPSTATUS: 1,DISCONNECT

AT+IPSTATUS=6                                   The socket number in the command is incorrect.
ERROR
```


5.16 AT+TCPACK – Querying Status of Data Sent by TCP Socket

To query the size of data successfully sent by the TCP socket and the size of the data successfully received.

Format

Type	Command	Response
Execute	AT+TCPACK=<n><CR>	<CR><LF>+TCPACK: <n>,<data_sent>,<acked_recv><CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+TCPACK: <n>,DISCONNECT<CR><LF> Or <CR><LF>+TCPACK: NO TCP LINK<CR><LF>

Parameter

<n>	socket ID, ranging from 0 to 5
<data_sent >	Size of data successfully sent through this socket, unsigned 64-bit integer in decimal ASCII. The unit is byte
<acked_recv>	Size of data acknowledged by the receiver, unsigned 64-bit integer in decimal ASCII. The unit is byte

20-byte data has been transmitted from socket 0 and the receiver acknowledged 20-byte data.

Example

```
AT+TCPACK=0                          20-byte data has been transmitted from socket 0 and
+TCPACK: 0,20,20                     the receiver acknowledged 20-byte data.

AT+TCPACK=0                          128-byte data has been transmitted from socket 0
+TCPACK: 0,128,120                   and the receiver acknowledged 120-byte data.

AT+TCPACK=1                          No connection is set up on socket 1.
+TCPACK: 1,DISCONNECT

AT+TCPACK=2                          A UDP connection is set up on socket 2.
```

```
+TCPACK: NO TCP LINK

AT+TCPACK=6                                  The socket number in the command is incorrect.

ERROR
```

5.17 AT+DNSSERVER－Setting the DNS Server

To set primary and secondary DNS servers.

In general, you do not have to set DNS server, which will be issued by base station during PPP negotiation.

Format

Type	Command	Response
Set	AT+DNSSERVER=<n>,<dns-ip><CR>	<CR><LF>+DNSSERVER: OK<CR><LF> Or <CR><LF>+DNSSERVER: ERROR<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+DNSSERVER?<CR>	<CR><LF>+DNSSERVER: dns1:<dns-ip1>;dns2:<dns-ip2><CR><LF>

Parameter

<n> DNS server number, ranging from 1 to 2.
<dns-ip> IP address of DNS server.

Example

```
AT+DNSSERVER?
                                             Query DNS server.(default IPV4)
+DNSSERVER: dns1:0.0.0.0;dns2:0.0.0.0

AT+DNSSERVER=1,114.114.114.114


+DNSSERVER: OK
AT+DNSSERVER=2,202.96.134.133
                                             Set DNS server.(IPV4)
+DNSSERVER: OK
AT+DNSSERVER?


+DNSSERVER:
```

```
dns1:114.114.114.114;dns2:202.96.134.133

AT+DNSSERVER=1,240E:1F:1::1

+DNSSERVER: OK
AT+DNSSERVER=2,240E:1F:1::33

+DNSSERVER: OK                                          Set DNS server.(IPV4V6)
AT+DNSSERVER?

+DNSSERVER:
dns1:114.114.114.114;dns2:202.96.134.133
+DNSSERVER:
ip6dns1:240E:1F:1::1;ip6dns2:240E:1F:1::33
```

5.18 AT+PDPKEEPALIVE – Setting the PDP Keepalive Heartbeat

To set the PDP keepalive heartbeat.

Activate PDP context before sending this command.

When PDP keepalive heartbeat is enabled, values will be returned to the AT+PDPKEEPALIVE? Command quickly.

When PDP keepalive heartbeat is disabled, the query status will have some delay, 200 ms to 10000 ms, depending on the network environment.

Format

Type	Command	Response
Set	AT+PDPKEEPALIVE=<onoff>,<inerval><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+PDPKEEPALIVE?<CR>	<CR><LF>+PDPKEEPALIVE=<onoff>,<inerval> <CR><LF>OK<CR><LF>
Test	AT+ PDPKEEPALIVE=?<CR>	<CR><LF>+PDPKEEPALIVE: (value range of<n>),(value range of<interval>)<CR><LF>

Parameter

<onoff>	heartbeat switch 0: disable (default) 1: enable
<inerval>	heartbeat interval, ranging from 1 to 65535. Unit: second.

Example

```
AT+PDPKEEPALIVE?
+PDPKEEPALIVE: 1,5                          Query heartbeat setting.
OK

AT+PDPKEEPALIVE=1,60                        Enable heartbeat and set the interval to 60
OK                                          seconds.

AT+PDPKEEPALIVE=?
+PDPKEEPALIVE: (0-1),(1-65535)              Query the value range of heartbeat function
OK
```

5.19 AT+PDPSTATUS – Querying PDP Status

To query the status of PDP.

Format

Type	Command	Response
Execute	AT+PDPSTATUS<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<ICCID>	SIM card ID The ICCID number is a string of 20 digits.

Example

```
AT+PDPSTATUS
                                            PDP connected
+PDPSTATUS: CONNECT
```

```
AT+PDPSTATUS                                PDP disconnected
+PDPSTATUS: DISCONNECT
AT+PDPSTATUS                                PDP activated, but in pseudo_connect state
+PDPSTATUS: PSEUDO_CONNECT
```

5.20 AT+TCPKEEPALIVE – Keep TCP link Settings

The TCP continuous online function takes effect only after the KEEPALIVE parameter is set and a TCP link is established

类型	命令	响应格式
Set	AT+TCPKEEPALIVE=<onoff>,<time>,<interval><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+TCPKEEPALIVE?<CR>	<CR><LF>+TCPKEEPALIVE: <mode>,<time>,<interval><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+TCPKEEPALIVE=?<CR>	<CR><LF>+TCPKEEPALIVE: (range of supported <onoff>),(range of supported <time>),(range of supported <interval>) <CR><LF> OK<CR><LF>

Parameter

<onoff> heartbeat switch
0: disable (default)
1: enable

<time> Idle interval (how long it takes for TCP to send KEEPALIVE packets to the remote server during idle time), The value ranges from 1S to 7200S. The default value is 20.

<interval> Retransmission interval (how long it takes for TCP to send KEEPALIVE packets without receiving replies from the remote server, the KEEPALIVE packet is sent again) The value is 1S to 1800S. The default value is 5.

Example

```
AT+TCPKEEPALIVE=1,30,40                 Enable and set the idle time of KEEPALIVE to 30S, and the interval
                                        for resending to 40S
```

```
OK

AT+TCPKEEPALIVE=?                          Query the range of KEEPALIVE parameters
+TCPKEEPALIVE: (0-1),(1-7200),(1-
1800)                                      Close KEEPALIVE function
AT+TCPKEEPALIVE=0,20,5
OK

AT+TCPKEEPALIVE?                           Read current KEEPALIVE parameters
+TCPKEEPALIVE: 0,20,5
OK
```

6 Transparent TCP/UDP Commands

6.1 AT+TCPTRANS – Setting up a Transparent TCP Connection

To set up a transparent TCP connection

TCP data can be transparently transmitted after the transparent TCP connection is set up successfully and **+TCPTRANS:OK** is returned. At most 4096-byte data can be sent or received in transparent mode.

The UART does not display the data transmitted to the server after the transparent TCP connection is set up successfully.

Use +++ to switch the server to command mode and ATO to switch it to data mode.

The module will disconnect the transparent link if a call or message is incoming.

In IPv6 single stack, ip_type is IPv6 by default. In IPv4 single stack and IPv4v6 double stack, ip_type is IPv4 by default.

Parameter [,<ip_type>] is temporarily not supported. <ip> can recognize ip_type automatically.

Format

Type	Command	Response
Execute	AT+TCPTRANS=<ip>,<port>[,<ip_type>]<CR>	<CR><LF>OK<CR><LF> <CR><LF>+TCPTRANS: <result><CR><LF> Or <CR><LF>+TCPTRANS: ERROR<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<ip>	destination IP address, in xx.xx.xx.xx format or domain name format (www.*XXXXXX*.com)
<port>	destination port ID in decimal ASCII code
<result>	OK: connection established successful.

Fail: connection established failed.

<ip_type>: Destination IP address: (Not supported temporarily)
IPv4
IPv6

Example

```
AT+TCPTRANS=220.199.66.56,6800          A transparent TCP connection is set up successfully.
OK

+TCPTRANS: OK

AT+TCPTRANS=2408:84fb:213:bdc1:7d7
b:b58b:2a92:d564,25800,IPV6
(Not supported temporarily)             Set up a transparent IPv6 TCP connection.
OK                                      Successful

+TCPTRANS: OK

AT+TCPTRANS=neowayjsr.oicp.net,600      A transparent TCP connection is set up successfully by
10                                      using domain name.
OK

+TCPTRANS: OK

AT+TCPTRANS=220.199.66.56,              The command format is incorrect.
ERROR

AT+TCPTRANS=220.199.66.56,6800          Fails to set up a transparent TCP connection.
OK

+TCPTRANS: FAIL

AT+TCPTRANS=220.199.66.56,6800          A transparent (TCP, UDP, TCP server) connection has been
+TCPTRANS: ERROR1                       set up.
```

6.2 AT+TCPACK – Querying Status of Data Sent by TCP Socket

To query the size of data successfully sent by the TCP socket and the size of the data successfully received.

Format

Type	Command	Response
Execute	AT+TCPACK<CR>	<CR><LF>+TCPACK: <data_sent>,<acked_recv><CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+TCPACK: DISCONNECT Or <CR><LF>+TCPACK: NO TCP LINK

Parameter

<data_sent > Size of data successfully sent through this socket, unsigned 64-bit integer in decimal ASCII. The unit is byte

<acked_recv> Size of data acknowledged by the receiver, unsigned 64-bit integer in decimal ASCII. The unit is byte

Example

```
AT+TCPACK                       Transparently transmit TCP data
+TCPACK: 1024,1024              1024-byte data has been transmitted and the receiver
                                acknowledged 1024-byte data.

AT+TCPACK                       No transparent connection is set up.
+TCPACK: DISCONNECT

AT+TCPACK                       A transparent UDP connection is set up.
+TCPACK: NO TCP LINK
```

20-byte data has been transmitted from socket 0 and the receiver acknowledged 20-byte data.

6.3 AT+UDPTRANS – Setting up a Transparent UDP Connection

To set up a transparent UDP link

UDP data can be transparently transmitted after the transparent UDP connection is set up successfully

and **+UDPTRANS:OK** is returned. At most 4096-byte data can be sent or received in transparent mode.

The UART does not display the data transmitted to the server after the transparent UDP connection is set up successfully.

Use +++ to switch the server to the command mode and ATO to switch it to the data mode.

The module will disconnect the transparent link if a call or message is incoming.

In IPv6 single stack, ip_type is IPv6 by default. In IPv4 single stack and IPv4v6 double stack, ip_type is IPv4 by default.

Parameter [,<ip_type>] is temporarily not supported. <ip> can recognize ip_type automatically.

Format

Type	Command	Response
Execute	AT+UDPTRANS=<ip>,<port>[,<ip_type>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<ip>	destination IP address, in xx.xx.xx.xx format or in domain name format (www.*XXXXX*.com).
<port>	destination port ID in decimal ASCII code
<port>	Destination port ID in decimal ASCII code.
<ip_type>:	Type of the destination IP address. (Not supported temporarily) IPv4 IPv6

Example

```
AT+UDPTRANS=220.199.66.56,6800
OK                                              A transparent UDP connection is set up
                                                successfully.

+UDPTRANS: OK

AT+UDPTRANS=2408:84fb:213:bdc1:7d7b:b58b:2a92:d
564,25801,IPV6 (Not supported temporarily)
OK                                              Set up a transparent IPv6 UDP connection.
                                                Successful

+UDPTRANS: OK

AT+UDPTRANS=neowayjsr.oicp.net,60010            A transparent UDP connection is set up by
```

```
OK                                                  using domain name successfully.

+UDPTRANS: OK

AT+UDPTRANS=220.199.66.56,6800
OK                                                  Fails to set up a transparent UDP connection.

+UDPTRANS: FAIL

AT+UDPTRANS=220.199.66.56,6800                      A transparent (TCP, UDP, TCP server)
+UDPTRANS: ERROR1                                   connection has been set up.
```

6.4 AT+IPSTATUS – Querying TCP/UDP Socket Status

To query the TCP/UDP socket status.

Format

Type	Command	Response
Execute	AT+IPSTATUS<CR>	<CR><LF>+IPSTATUS: <CONNECT or DISCONNECT>[,<TCP or UDP>,<send-buffer-size>] Or <CR><LF>+IPSTATUS: DISCONNECT<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<CONNECT or DISCONNECT>	Socket status, CONNECT or DISCONNECT
<TCP or UDP>	Socket type, TCP or UDP
<send-buffer-size>	The size of the available send buffer on the module, in decimal ASCII mode, unit: byte.

Example

```
AT+IPSTATUS                                         The AT command is not complete.
ERROR

AT+IPSTATUS                                         A transparent TCP connection has been set up and
+IPSTATUS: CONNECT,TCP,10240                        the available buffer size is 10240 bytes.

AT+IPSTATUS                                         A transparent UDP connection has been set up and
                                                    the available buffer size is 10240 bytes.
```

```
+IPSTATUS: CONNECT,UDP,10240

AT+IPSTATUS                                  No any TCP or UDP connections are set up.
+IPSTATUS: DISCONNECT
```

6.5 AT+TRANSCLOSE – Closing the Transparent Socket

To close the transparent socket.

Format

Type	Command	Response
Execute	AT+TRANSCLOSE<CR>	<CR><LF>+TRANSCLOSE: 1,OK Or <CR><LF>ERROR<CR><LF>

Parameter

N/A.

Example

```
AT+TRANSCLOSE                                A transparent TCP socket is closed
+TRANSCLOSE: 0,OK                            successfully.

AT+TRANSCLOSE                                No transparent TCP/UDP socket is set up.
ERROR

AT+TRANSCLOSE                                A transparent UDP socket is closed
+TRANSCLOSE: 1,OK                            successfully.

+TCPTRANS: Link Closed                       The transparent TCP socket is closed.

+UDPTRANS: Link Closed                       The transparent UDP socket is closed.
```

7 Call Control Commands

7.1 ATD – Dialing Command

To initialize a data, fax, or voice link. For a voice link, the dialing string consists of numbers and modifiers and must end with a semicolon.

There are two types of calls: voice call, data and fax call. Semicolon is required for voice calls and not required for data or fax calls.

Format

Type	Command	Response
Execute	• ATD<dial string>[;]<CR> • ATD><n>;<CR> • ATD>"name";<CR>	<CR><LF>OK<CR><LF> <CR><LF><result><CR><LF>

Parameter

<dial string>	Phone number.
<n>	The location of phone number.
name	name in current phonebook.

Example

```
ATD10010;                                   Make a call.
OK

CONNECT                                     The callee answers.
ATD>4;                                      Dial up the number listed as 4 in the
OK                                          phonebook.

CONNECT
ATH
OK
ATD>"Comneon";                              Dial up the number according to the name in
```

```
OK                                                  the phonebook.
                                                    The callee answers and then ends the call.
CONNECT

NO CARRIER
```

7.2 ATA – Call Answering

To answer the call and establish a call connection

The return codes containing **RING** or **+CRING** indicate an incoming call.

Format

Type	Command	Response
Execute	ATA<CR>	<CR><LF>OK<CR><LF> If the callee hangs up during a call, the command returns: <CR><LF>BUSY<CR><LF>

Parameter

N/A.

Example

```
ATA                                                 Answer a call (voice)
OK
```

7.3 ATH – Hanging Up Calls

To hang up all calls.

Format

Type	Command	Response
Execute	ATH<CR>	<CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
ATH                                          End the call connection.
OK

ATH                                          Refuse the incoming call. Hang up the call.
OK
```

7.4 AT+CLIP – Caller ID

To enable or disable caller ID.

The caller ID function is enabled by default.

Format

Type	Command	Response
Set	AT+CLIP=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CLIP?<CR>	<CR><LF>+CLIP: <n>,<m><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CLIP=?<CR>	<CR><LF>+CLIP: (value range of <n>)<CR><LF> <CR><LF>OK<CR><LF>
URC	<CR><LF>+CLIP: <phone number>,<tosca>	

Parameter

<n>	0: disable caller ID 1: enable caller ID (default value)
<m>	0: CLIP not provisioned 1: CLIP provisioned 2: unknown (no connection, etc.)

Example

```
AT+CLIP=1                                    Enable the caller ID function.
OK

RING                                         An incoming call from 136********.

+CLIP: "136********",161

AT+CLIP?                                     Query the setting of the caller ID.
+CLIP: 1,1

OK

AT+CLIP=?                                    Query the value range of caller ID function.
+CLIP: (0-1)

OK
```

7.1 AT+CLVL – Setting the Voice Volume

To set the level of the voice volume, which is valid before a call or during a call.

The settings by this command are not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+CLVL=<level><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CLVL?<CR>	<CR><LF>+CLVL: <level> <CR><LF>OK<CR><LF>
Test	AT+CLVL=?<CR>	<CR><LF>+CLVL: (range of <level> value)

	<CR><LF>OK<CR><LF>

Parameter

<level> Integer type, ranging from 0 to 100; the smaller the parameter value, the smaller the level. The default value is 75.

Example

```
AT+CLVL=60                          Set the level of the voice volume to 60.
+clvl: OK
OK

AT+CLVL?                            Query the level of voice volume of the module.
+CLVL: 60
OK

AT+CLVL=?                           Query the valid voice volume level for the module.
+CLVL: (0-100)
OK
```

7.2 AT+CMUT – Mute Control

To set mute control of the voice calls.

The setting is only valid during a call. Not effective will be returned in any other situations.

Format

Type	Command	Response
Set	AT+CMUT=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CMUT?<CR>	<CR><LF>+CMUT: <n>,<m> <CR><LF>OK<CR><LF>
Test	AT+CMUT=?<CR>	<CR><LF>+CMUT: (value range of <n>),(value range of <m>) <CR><LF>OK<CR><LF>

Parameter

<n>	0: Mute off output (default value) 1: Mute on output
<m>	0: Mute off input (default value) 1: Mute on input

Example

```
AT+CMUT=0,0                          Disable the mute mode.
OK

AT+CMUT=1                            Enable mute control before a call.
OK

AT+CMUT?                             Query whether the mute mode is enabled.
+CMUT: 0,0
OK

AT+CMUT=?                            Query the value range of mute mode function.
+CMUT: (0-1)
OK
```

7.3 AT+MICL – Setting the Microphone Gain

To set the microphone volume gain of the module. The setting is valid during a call.

The settings by this command are not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+MICL=<level><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MICL?<CR>	<CR><LF>+MICL: <level><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+MICL=?<CR>	<CR><LF>+MICL: (range of <level> value)<CR><LF>

Parameter

<level>	Microphone gain level, ranging from 0 to 100. The default value is 75.

Example

```
AT+MICL=60
+MICL: OK                                   Set the level of the MIC volume to 60.
OK
AT+MICL?
+MICL: 60                                   Query the current level of the MIC volume.
OK
```

8 Audio Recording Commands

8.1 AT+RECMODE – Setting Recording Mode

To set the buffer mode of record data.

Execute this command before starting audio record. The setting is not saved after the module is powered off.

The default value is 0. Set the record mode before starting an audio recording.

The file extension does not have to be in this format, the file format is based on setting the third parameter.

Format

Type	Command	Response
Set	AT+RECMODE=<mode>[,<file_name>[,<format>]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+RECMODE?<CR>	<CR><LF>+RECMODE:<mode> <CR><LF>OK<CR><LF>
Test	AT+RECMODE=?<CR>	<CR><LF>+RECMODE: (range of <mode> value),<file_name>,(range of<format> value) <CR><LF>OK<CR><LF>

Parameter

<mode>	0: record local audio (default) 1: record downlink audio of voice (not supported) 2: record uplink audio of voice (not supported) 3: record mixed audio of voice
<file_name>	saved file name for audio record. Record files are saved from 001.pcm.
<format>	Record format 1: wav (default)

Example

```
AT+RECMODE=0,audio.wav,1
+RECMODE: OK                                        Set to record local audio
OK
```

8.2 AT+RECF – Starting and Stopping an Audio Recording

To start of stop an audio recording.

Set record mode before starting to record audios.

If a file with the same name exists in the module already, the new record file overwrites the previous one.

Record File is stored in the “/data” directory by default.

Format

Type	Command	Response
Execute	AT+RECF=<val><CR>	<CR><LF>OK<CR><LF>

Parameter

<val> 1: stop an audio recording
1: start an audio recording

Example

```
AT+RECF=1
+RECF: OK                                           Recording
OK
AT+RECF=0
+RECF: OK                                           To stop recording audio
OK
```

8.3 AT+RECP – Playing the Record File

To play the record file.

Audio records can be played when the module is idle or in a call.

Only .wav files can be played by this command.

Format

Type	Command	Response
Execute	AT+RECP=<sta>[,<file_name0>……[,<file_name15>]]<CR>	<CR><LF>OK<CR><LF>

Parameter

<sta>	Audio play switch 0: stop playing 1: start to play
<file_name>	audio record file saved in the module, supports playing 16 wav files at most.

Example

```
AT+RECP=1,001.wav                                   Play the record file 001.wav.
+RECP: OK
OK
AT+RECP=0,001.wav                                   Stop playing the audio record.
+RECP: OK
OK
```

9 FTP Commands

9.1 AT+FTPLOGIN－Logging in to the FTP Server

To log in to the FTP server

The FTP functions cannot be used together with the internal protocol stack TCP/UDP function.

Data can be read or written on the FTP server only after login.

Format

Type	Command	Response
Execute	AT+FTPLOGIN=<ip>,<port>,<user>,<pwd>[,<ftpmode>]<CR>	<CR><LF>OK<CR><LF> <CR><LF>+FTPLOGIN: <result> Or <CR><LF>OK<CR><LF> <CR><LF>+FTP: Server Control Link Disconnect<CR><LF> Or <CR><LF>ERROR<CR><LF>
Unsolicited result code	+FTPLOGIN: <result>	

Parameter

<ip>	FTP server address
<port>	Port ID of the FTP server, 21
<user>	The user name to log in to the FTP server. The length of the user name cannot exceed 100 ASCII codes and cannot contain comma (,).
<pwd>	The password for the user account to log in to the FTP server. The length of the password cannot exceed 100 bytes in ASCII code and the password cannot contain comma (,)
<result>	• Error: The format of the AT command is incorrect

- Have Logged In: The user has logged in to the FTP server.

A Busy: Last FTP AT command has not been executed completely.

- User logged in: The user logged in to the FTP server successfully.
- 530 Not logged in: The user failed to log in to the FTP server because the user account or password is incorrect.
- GPRS DISCONNECTION: The user logged in to the FTP server before a PPP

link is set up.

Example

```
At+FTPLOGIN=219.134.179.52,21,user1,pwd2009          user1 logs in to the server 219.134.179.52
OK                                                   through port 21 successfully. And the
                                                     password
+FTPLOGIN: User logged in                            for user1 is pwd2009.
AT+FTPLOGIN=58.60.184.213,21,neoway,neoway           Fails to log in to the FTP server using
OK                                                   neoway
                                                     because the connection times out.
+FTPLOGIN: Error Connect Server Fail
AT+FTPLOGIN=58.60.184.213,21,neowayftp,neowayftp     Fails to log in to the FTP server.
OK

+FTP: Server Control Link Disconnect
```

9.2 AT+FTPLOGOUT – Logging out from the FTP Server

To log out from the FTP server

Format

Type	Command	Response
Execute	AT+FTPLOGOUT<CR>	<CR><LF>+FTPLOGOUT: User logged out <CR><LF>OK<CR><LF> Or <CR><LF>+CME ERROR: INVALID SOCKET ID<CR><LF> <CR><LF>ERROR<CR><LF>

Parameter

N/A

Example

```
AT+FTPLOGOUT                                    Log out from the FTP server
+FTPLOGOUT: User logged out
OK
AT+FTPLOGOUT                                    Log out of the FTP server when the FTP server
+CME ERROR: INVALID SOCKET ID                   is offline.

ERROR
```

9.3 AT+FTPGET–Downloading Data from the FTP Server

To download data from the FTP server

Format

Type	Command	Response
Execute	AT+FTPGET=<dir&filename>,<type>,<Content or Info>[,offset[,lenth]]<CR>	<CR><LF>+FTPGET: Error Not Login<CR><LF> Or <CR><LF>+FTPGET: Error Not Login<CR><LF> Or <CR><LF>+FTPGET: Error TimeOut<CR><LF> Or <CR><LF>+FTPGET: <length>,<data><CR><LF> Or <CR><LF>+FTPGET: OK.total length is <n><CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	+FTPSTATE:	

Parameter

<dir&filename>	Path and name of the file to be read. The file directory is under the FTP root directory.
type>	File transfer mode 1: ASCII 2: Binary
<Content or Info>	File content or file (or specified directory) information 1: Obtain the file content 2: Obtain the information of the file or the specified path 3: Obtain the file size
<offset>	Specifies offset of file content.
<lenth>	Length of file downloaded from the start point, ranging from 1 to 8192

Example

```
AT+FTPGET=test.txt,1,2                          Obtain the information about test.txt.
+FTPGET: 65,-rw-rw-rw- 1 user group 10 Jan 15
15:01 test.txt

+FTPGET: OK.total length is 65

+FTP: Server Data Link Disconnect
AT+FTPPUT=test.txt,1,2,10                       Obtain the information in test.txt.
>
+FTPPUT: OK,10
AT+FTPGET=test.txt,1,1
+FTPGET: 10,0123456789

+FTPGET: OK.total length is 10

+FTP: Server Data Link Disconnect
AT+FTPGET=test.txt,1,1,2                        Read all data after the first byte
+FTPGET: 8,23456789

+FTPGET: OK.total length is 8

+FTP: Server Data Link Disconnect
AT+FTPGET=test.txt,1,1,2,4                      Read 4-byte data after the first byte
+FTPGET: 4,2345

+FTPGET: OK.total length is 4

+FTP: Server Data Link Disconnect
AT+FTPGET=test.txt,1,3                          Read the file size of test.txt
+FTPGET: OK.the file size is 10
```


9.4 AT+FTPPUT–Uploading Data to the FTP Server

To upload data to the FTP server.

Use +++ to exit from transparent transmission mode and end uploading.

In transparent mode, it is recommended to use buffer mode or APPE mode for large files. Otherwise, the port will be occupied all the time and this affects other commands

Format

Type	Command	Response
Execute	AT+FTPPUT=<filename>,<type>,<mode>[,<size>]<CR>	None-transparent <CR><LF>+FTPPUT: OK,<size><CR><LF> Transparent <CR><LF>CONNECT <CR><LF>+FTPPUT: OK,<size><CR><LF> Or <CR><LF>+FTPPUT: Error Not Login<CR><LF> Or <CR><LF>+FTPPUT: AT Busy<CR><LF> Or <CR><LF>+FTPPUT: SIZE Error (None-transparent)) Or <CR><LF>+FTPPUT: OK,<n><CR><LF> Or <CR><LF>+FTPPUT: Delete File OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<filename>	The name of the file to be uploaded, the file directory under the FTP root directory
<type>	File transfer mode 1: ASCII 2: Binary
<mode>	Operation mode 1: STOR mode. Create a file on the FTP server and write the data to the file. If the file exists, the original file is overwritten. 2: APPE mode. Create a file on the FTP server and write the data to the file. If the file exists, the data is attached to the end of the file. 3: DELE mode. Delete a file.

<SIZE>	File size

Example

```
AT+FTPPUT=test.txt,1,1,10                    Upload the text.txt file, which is 10 bytes.
>                                            The file is transferred in ASCII and the
+FTPPUT: OK,10                               operated in STORE.
AT+FTPPUT=test.txt,1,2,10                    Upload the text.txt file, which is 10 bytes.
>                                            The file is transferred in ASCII and the
+FTPPUT: OK,10                               operated in APPE
AT+FTPPUT=test.txt,1,3,0                     Delete the test.txt file.
+FTPPUT: Delete File OK
AT+FTPPUT=test.txt,1,1                       Transparent mode
CONNECT
+FTPPUT: OK,10
AT+FTPPUT=test.txt,1,2                       Transparent mode
CONNECT
+FTPPUT: OK,10
AT+FTPPUT=test.txt,1,3                       Transparent mode
+FTPPUT: Delete File OK
```

8.5 AT+FTPSTATUS–Querying FTP Link Status

To query the FTP link status

Format

Type	Command	Response
Execute	AT+FTPSTATUS<CR>	<CR><LF>+FTPSTATUS: <status>,<ip>, <port><CR><LF>

Parameter

<status>	0: The FTP link has not been set up. 1: The FTP link has been set up.
<ip>	The IP address of the FTP server
<port>	The port of the FTP server

Example

```
AT+FTPSTATUS                                 Not logged in
+FTPSTATUS: 0,0.0.0.0,21
AT+FTPSTATUS                                 The module is successfully connected to the
+FTPSTATUS: 1,119.139.221.66,21              FTP server.
```

10 SSL TCP Data Service

10.1 AT+SSLTCPCFG – Configuring SSL Parameters for TCP

To configure SSL parameters for TCP data service.

If the **authmode** is set to 0, you do not have to set other parameters, such as **cacert**, **clientcert**, and **clientkey**.

Format

Type	Command	Response
Execute	AT+SSLTCPCFG=<type>,<type_name><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+SSLTCPCFG?<CR>	<CR><LF>+SSLTCPCFG:<sslversiontype_name>,<authmodetype_name>,<ciphersuitetype_name>,<cacerttype_name>,<clientcerttype_name>,<clientkeytype_name> <CR><LF>OK<CR><LF>
Test	AT+SSLTCPCFG=?<CR>	<CR><LF>+SSLTCPCFG: <type>,<type_name> <CR><LF>OK<CR><LF>

Parameter

<type>	SSL parameter options. **sslversion**: SSL protocol version **authmode**: authorization mode **ciphersuite**: Cupher cuite **cacert**: CA certificate **clientcert**: Client certificate **clientkey**: Client key
<type_name>	parameter settings for SSL sslversion 0: SSL3.0

1: TLS1.0
2: TLS1.1
3: TLS1.2
authmode
0: No authentication
1: Manage server authentication
2: Manage server and client authentication if requested by the remote server
Cacert: string, CA certificate
Clientcert: string, client certificate
Clientkey: string, client key

Example

```
AT+SSLTCPCFG=”sslversion”,0                    Set SSL version to SSL3.0.
OK
AT+SSLTCPCFG=”authmode”,0                      Set authmode to no authentication.
OK
AT+SSLTCPCFG?                                  Query the current SSL settings.
+SSLTCPCFG: 0,1,,ca.pem,cc.pem,ck.pem
OK
AT+SSLTCPCFG=?                                 Query the value range of the parameters.
+SSLTCPCFG: <type>,<type_name>
OK
```

10.2 AT+SSLTCPSETUP –Setting up TCP Connection over SSL

To set up a TCP connection over SSL.

Activate PPP and apply one IP address before setting up a TCP connection over SSL.

Format

Type	Command	Response
Execute	AT+SSLTCPSETUP=<n>,<ip>,<port>,<mode><CR>	<CR><LF>OK<CR><LF> <CR><LF>+SSLTCPSETUP: <n>,<status> Or <CR><LF>CONNECT Or <CR><LF>+SSLTCPSETUP: ERROR Or <CR><LF>+SSLTCPSETUP: GPRS DISCONNECTION
Query	AT+SSLTCPSETUP?	<CR><LF>+SSLTCPSETUP: <socket_id>,<ip>,

		<port>,<mode> [<CR><LF>+SSLTCPSETUP: <socket_id>,<ip>, <port>,<mode>]… <CR><LF>OK<CR><LF>
Test	AT+SSLTCPSETUP=?	<CR><LF>+SSLTCPSETUP:(value range of supported <socket_id>),<ip>,<port>,<mode><CR><LF>

Parameter

<n>	socket ID, ranging from 0 to 5, used to identify the connection to the server
<ip>	IP address or domain name of the server
<port>	server port
<mode>	transmission mode 0: non-transparent 1: transparent
<status>	OK ERROR1 AUTHFAIL FAIL

Example

```
AT+SSLTCPSETUP=0,183,239.240,45,4451,0          Set up a non-transparent connection to
OK                                              183.239.240.45 on socket 0. The port number is
                                                4451.
+SSLTCPSETUP: 0,OK
AT+SSLTCPSETUP=0,183,239.240,45,4451,1          Set up a transparent connection to
CONNECT                                         183.239.240.45 on socket 0. The port number is
                                                4451.
AT+SSLTCPSETUP=0, www.alipay.com,443,0          Set up a non-transparent connection to
OK                                              www.alipay.com on socket 0. The port number is
                                                443.
+SSLTCPSETUP: 0,FAIL                            Fails because of timeout.
AT+SSLTCPSETUP=0, www.alipay.com,443,0          Set up a non-transparent connection to
OK                                              www.alipay.com on socket 0. The port number is
                                                443.
+SSLTCPSETUP: 0,AUTHFAIL                        Fails to authenticate.
AT+SSLTCPSETUP?                                 Query the connection status.
+SSLTCPSETUP: 0,183.239.240.45,4451,0           A transparent TCP connection has been set up
+SSLTCPSETUP: 1,183.239.240.45,4452,0           on socket 0 and socket 1.
OK
AT+SSLTCPSETUP=0,183,239.240,45,4451,0          A connection has been set up on socket 0
OK

+SSLTCPSETUP: 0,ERROR1
```

10.3 AT+SSLTCPCLOSE – Closing TCP Connection over SSL

To close a TCP connection over SSL.

Format

Type	Command	Response
Execute	AT+SSLTCPCLOSE=<socket_id>	<CR><LF>+SSLTCPCLOSE: <socket_id>,<result> Or <CR><LF>+SSLTCPCLOSE: ERROR<CR><LF>
URC	+SSLTCPCLOSE: <socket_id>,Link Closed	

Parameter

<socket_id> socket ID, ranging from 0 to 5.

Example

```
AT+SSLTCPCLOSE=0                          Close the TCP connection on socket 0.
+SSLTCPCLOSE: 0,OK
AT+SSLTCPCLOSE=0
+SSLTCPCLOSE: ERROR                       Socket ID is incorrect.
+SSLTCPCLOSE: 0,Link Closed                The connection on socket 0 is closed.
```

10.4 AT+SSLTCPSEND – Sending TCP Data over SSL

To send TCP data over SSL.

Format

Type	Command	Response
Execute	AT+SSLTCPSEND=<socket_id>,<data_length>	<CR><LF>> <CR><LF>+SSLTCPSEND: <socket_id>,OK Or <CR><LF>+SSLTCPSEND: Data length error<CR><LF> Or <CR><LF>+SSLTCPSEND: <socket_id>,FAIL<CR><LF>

Test	AT+SSLTCPSEND=?	<CR><LF>+SSLTCPSEND: (value range of<n>),(value range of<data_length>)<CR><LF>

Parameter

<socket_id> ranging from 0 to 5, used to identify the connection to the server.
<data_length> data length, ranging from 1 to 4096.

Example

```
AT+SSLTCPSEND=0,20                      Send 20-byte data to the server over socket 0.
>
+SSLTCPSEND: 0,OK
AT+SSLTCPSEND=0,1024                    Send 1024-byte data to the server over socket 0.
>                                       Failed because buffer is full.
+SSLTCPSEND: 0,FAIL
AT+SSLTCPSEND=0,4097                    Send 4097-byte data to the server.
+SSLTCPSEND: Data length error          Failed because the data length exceeds the threshold.
AT+SSLTCPSEND=?                         Query the value range of the parameters.
+SSLTCPSEND: (0-5),(1-4096)
OK
```

10.5 +SSLTCPRECV – Notifying SSLTCP Data Output

To notify SSLTCP data output.

When the module receives SSLTCP data from the network, the UART prints the data automatically.

Format

Type	Command
URC	+SSLTCPRECV: <socket_id>,<data_length>,<data><CR><LF>

Parameter

<socket_id> ranging from 0 to 5, used to identify the connection to the server
<data_length> data length, ranging from 1 to 4096.
<data> data received

Example

```
+SSLTCPRECV: 0,20,1234567890abcdefghjk                Received 20-byte data over socket 0
```

10.6 AT+CERTADD – Writing SSL Certificate

To write an SSL certificate to the module.

The writing process can be interrupted by +++.

Format

Type	Command	Response
Execute	AT+CERTADD=<file_name>,<length><CR>	<CR><LF>CONNECT<CR><LF> <CR><LF>+CERTADD: <length>,OK<CR><LF> Or <CR><LF>+CERTADD: ERROR<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<filen_ame> Certificate file name
<length> file length

Example

```
AT+CERTADD=ca.crt,1188              Add the ca.crt certificate of 1158 bytes to the module.
CONNECT

+CERTADD: 1188,OK
AT+CERTADD=client.crt,1200          Add the client.crt certificate of 1200 bytes to the module.
CONNECT

+CERTADD: 1200,OK
AT+CERTADD=client.key,1675          Add the client.crt certificate of 1675 bytes to the module.
CONNECT

+CERTADD: 1675,OK
```

10.7 AT+CERTCHECK – Checking SSL Certificate

To check the SSL certificate.

Format

Type	Command	Response
Execute	AT+CERTCHECK=<file_name><CR>	<CR><LF>+CERTCHECK: <file_name>,OK Or <CR><LF>+CERTCHECK: ERROR
Query	AT+CERTCHECK?<CR>	<CR><LF><file_name> [<CR><LF><file_name>] <CR><LF>OK<CR><LF>

Parameter

<file_name> Certificate file name

Example

```
AT+CERTCHECK=ca_cert.pem                    Check the ca_cert.pem certificate.
+CERTCHECK: ca_cert.pem,OK
AT+CERTCHECK=clent_cert.pem                 Check the clent_cert.pem certificate.
+CERTCHECK: clent_cert.pem,OK
AT+CERTCHECK=client_key.pem                 client_key.pem does note exist.
+CERTCHECK: ERROR
AT+CERTCHECKT?                              Check the added file.
cacert.pem
keycert.pem
OK
```

10.8 AT+CERTDEL – Deleting SSL Certificate

To delete an SSL certificate.

Format

Type	Command	Response
Execute	AT+CERTDEL=<file_name><CR>	<CR><LF>OK<CR><LF> Or

	<CR><LF>ERROR<CR><LF>

Parameter

<file_name> Certificate file name

Example

```
AT+CERTDEL=ca_cert.pem                          Delete ca_cert.pem.
OK
AT+CERTDEL=client_cert.pem                      Delet client_cert.pem
OK
AT+CERTDEL=client_key.pem                       Delete cilent_key.pem
OK
AT+CERTDEL                                      Delete all added files.
OK
```

10.9 AT+SSLTCPCFGA – Configuring SSL Parameters for TCP

To configure SSL parameters for TCP data service.

Certificate must be imported before setting.

Certificate can be set empty.

Format

Type	Command	Response
Execute	AT+SSLTCPCFGA=<sslversion>,<authmode>,<cacert>,<clientcert>,<clientkey><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+SSLTCPCFGA?<CR>	<CR><LF>+SSLTCPCFGA: <sslversion>,<authmode>,<cacert>,<clientcert>, <clientkey> <CR><LF>OK<CR><LF>
Test	AT+SSLTCPCFGA=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<sslversion>	0: SSL3.0 1: TLS1.0 2: TLS1.1 3: TLS1.2
<authmode>	0: No authentication 1: Manage server authentication 2: Manage server and client authentication if requested by the remote server
<Cacert>	string, CA certificate
<Clientcert>	string, client certificate
<Clientkey>	string, client key

Example

```
AT+SSLTCPCFGA=3,1,"ca.pem","",""                    Set SSL TCP parameters
OK
AT+SSLTCPCFGA?                                      Query the current SSL settings.
+SSLTCPCFGA: 0,1,ca.pem,cc.pem,ck.pem
OK
```

10.10 AT+SSLTCPREAD－Reading SSL TCP Data

To read SSL TCP data.

This command is sent after executing **AT+RECVMODE=0** to modify receive mode.

Format

Type	Command	Response
Execute	AT+SSLTCPREAD=<id>,<length><CR><LF>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<id>	socket ID, ranging from 0 to 5.
<length>	data length, ranging from 1 to 4096.

Example

```
AT+SSLTCPSETUP=0,58.60.184.213,12004,0
OK
+SSLTCPSETUP: 0,OK
AT+SSLTCPSEND=0,10
>
+SSLTCPSEND: 0,OK

+SSLTCPRECV: 0

AT+SSLTCPREAD=0,4096
+SSLTCPREAD: 0,10,1111111111
OK
```

11 GPS Commands

11.1 AT+NWGNSSPWR – Switching GPS

To switch GPS.

GPS service occupies a lot of resources. If it is not used, disable the service.

After the request is sent successfully, it will take the module some time to enable the GPS service and get GPS position. It depends on the circumstance. Ensure that a matched antenna is used.

Format

Type	Command	Response
Execute	AT+NWGNSSPWR=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT+NWGNSSPWR=<?><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	0: disable GPS service 1: enable GPS service

Example

```
AT+NWGNSSPWR=1                          Enable GPS service
OK
AT+NWGNSSPWR=0                          Disable GPS service
OK
AT+NWGNSSPWR?                           Querying GPS status
+NWGNSSPWR: 0

OK
```

11.2 AT+NWGNSSMODE – Set GPS Mode

To set GPS mode.

Format

Type	Command	Response
Execute	AT+NWGNSSMODE=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT+NWGNSSMODE?	<CR><LF>+NWGNSSMODE: n<CR><LF>
Execute	AT+NWGNSSMODE=?	<CR><LF>+NWGNSSMODE: <1-15><CR><LF>

Parameter

<n> GNSS supports positioning mode, integer type
The corresponding Bit bit represents 1: open, 0: closed
Bit 0: GPS
Bit 1: GLONASS
Bit 2: Beidou (not support)
Bit 3: GALILEO
Bit 4: QZSS

Example

```
AT+NWGNSSMODE=1                          Enable GPS
OK                                       Query GPS status.
AT+NWGNSSMODE?
+NWGNSSMODE: 1

OK
```

11.3 AT+NWGNSSNMEA – Obtaining NMEA Data

To obtain NMEA data.

The obtained data is GPS coordinates.

It takes time to fix the position for the first time after GPS is enabled.

Format

Type	Command	Response
Execute	AT+NWGNSSNMEA=<TYPE>[,<LOOP>]<CR>	example for GPS 0:NMEA$GPGGA <CR><LF>+NWGNSSPOS: $GPGGA,<1>,<2>,<3>,<4>,<5>, <6>,<7>,<8>,<9>,<10>,<11>,<12>,<13>,<14>*<hh> <CR><LF>OK<CR><LF> 1:NMEA$GPGSA <CR><LF>+NWGNSSPOS: $GPGSA,<1>,<2>,<3>,<4>,<5>, <6>,<7>,<8>,<9>,<10>,<11>,<12>,<13>,<14>,<15>,<16>,<17>,*<hh><CR><LF>OK<CR><LF> 2:NMEA$GPGSV <CR><LF>+NWGNSSPOS: $GPGSV, <1>,<2>,<3>,<4>,<5>, <6>,<7>[,<4>,<5>,<6>,<7>...]*<hh> <CR><LF>$GPGSV, <1>,<2>,<3>,<4>,<5>,<6>,<7>[,<4>, <5>,<6>,<7>...]*<hh> <CR><LF>... <CR><LF>OK<CR><LF> 3:NMEA$GPRMC <CR><LF>+NWGNSSPOS: $GPRMC,<1>,<2>,<3>,<4>,<5>, <6>,<7>,<8>,<9>,<10>,<11>,<12>*<hh> <CR><LF>OK<CR><LF> 4:NMEA$GPVTG <CR><LF>+NWGNSSPOS: $GPVTG,<1>,T,<2>,M,<3>,N, <4>,K,<5>*<hh> <CR><LF>OK<CR><LF> 5:NMEA$GPGLL <CR><LF>+NWGNSSPOS: $GPGLL,<1>,<2>,<3>,<4>,<5>, <6>*<hh> <CR><LF>OK<CR><LF> 6:NMEA$GPZDA <CR><LF>+NWGNSSPOS: $GPZDA,<1>,<2>,<3>,<4>,<5>, <6>*<hh> <CR><LF>OK<CR><LF>

Parameter

<TYPE>	GPS data type, integer 0: NMEA$GPGGA 1: NMEA$GPGSA 2: NMEA$GPGSV 3: NMEA$GPRMC 4: NMEA$GPVTG 5:NMEA$GPGLL 6: NMEA$GPZDA 7: NMEA$GLDTM (GLONASS)

0: NMEA$GPGGA

<1>	UTC time of position fix, hh mm ss
<2>	latitude of position, dd mm mmmm
<3>	North & South Hemispheres N: north S: south
<4>	longitude of position, ddd mm mmmm
<5>	Western & Eastern Hemispheres E: east W: west
<6>	GPS quality indicator 0: no fix 1: GPS fix 2: differential GPS fix 3: not valid 6: estimated
<7>	Number of satellites in use
<8>	Horizontal Dilution of Precision (HDOP)
<9>	altitude above mean sea level (geoid)
<10>	units of altitude M: Meter
<11>	geoidal height
<12>	unit of geoidal height
<13>	time since last DGPS update
<14>	DGPS reference station ID
<hh>	checksum

1: NMEA$GPGSA

<1>	Mode
	A: Automatic
	M: Manual
<2>	fix mode
	1: invalid fix
	2: 2D fix
	3: 3D fix
<3>	PRN number of satellites used for fix
<4>	PRN number in second channel
<5>	PRN number in third channel
<6>	PRN number in forth channel
<7>	PRN number in fifth channel
<8>	PRN number in sixth channel
<9>	PRN number in seventh channel
<10>	PRN number in eighth channel
<11>	PRN number in ninth channel
<12>	PRN number in tenth channel
<13>	PRN number in eleventh channel
<14>	PRN number in twelfth channel
<15>	position (3D) dilution of precision (PDOP) (0.5 - 99.9)
<16>	HDOP (0.5 - 99.9)
<17>	VDOP (0.5 - 99.9)
<hh>	checksum

2: NMEA$GPGSV

<1>	Message number
<2>	Message number
<3>	Total number of satellites in view, ranging from 00 to 12.
<4>	SV PRN number, ranging from 01 to 32.
<5>	Evaluation in degrees, ranging from 00 to 90 degree.
<6>	Azimuth, degrees from true north, 000 to 359
<7>	SNR (C/No), 00-99 dB (null when not tracking)
<hh>	checksum
	8-11: Information about second SV, same as field 4-7
	12-15: Information about third SV, same as field 4-7
	16-19: Information about fourth SV, same as field 4-7
	The number of $GPGSV sentences is same as the value of <1>.

3: NMEA$GPRMC

<1>	UTC time of position fix, hhmmss
<2>	status of position fix
	A: valid
	V: invalid
<3>	latitude of fix, ddmm.mmmm
<4>	N or S
	N: North
	S: South
<5>	longitude, dddmm.mmmm
<6>	E or W
	E: East
	W: West
<7>	Track made good in degrees, 000.0~359.9, True
<8>	Track made good in degrees, 000.0~359.9, True
<9>	UTC date, ddmmyy
<10>	Magnetic variation degrees (Easterly var. subtracts from true course), 000.0~180.0
<11>	E or W
	E: Eastern
	W: Western
<12>	GPS quality indicator
	A: Automatic
	D: DGPS
	E: Estimated
	N: invalid
<hh>	Checksum

4: NMEA$GPVTG

<1>	True track made good (degrees)
<2>	Magnetic track made good
<3>	Ground speed, knots
<4>	Ground speed, Kilometers per hour
<5>	GPS quality indicator
	A: Autonomous
	D: DGPS
	E: Estimated
	N: Invalid
<hh>	Checksum

5: NMEA$GPGLL

<1> Latitude, ddmm.mmmm
<2> N or S
N: North
S: South
<3> Longitude, dddmm.mmmm
<4> E or W
E: East
W: West
<5> UTC time of position fix, hhmmss.sss
<6> Status
A: Fix
V: No fix
<hh> Checksum
6: NMEA$GPZDA
<1> UTCtime
<2> day
<3> month
<4> year
<5> tzh,Not supported,fixed to 00
<6> izn,Not supported,fixed to 00
<hh> Checksum
7: NMEA$GLDTM (GLONASS)
<1> Local coordinate system code W84
<2> Empty coordinate system subcode
<3> Latitude Offset
<4> Latitude Hemisphere N (Northern Hemisphere) or S (Southern Hemisphere)
<5> Longitude offset
<6> Longitude Hemisphere E (East longitude) or W (West longitude)
<7> Height Offset
<8> Coordinate System Code W84
<hh> Checksum

[<loop>] GNSS loop output NMEA data
<TYPE> Must be set to 99
0: Output all NMEA data only once
1: Loop output NMEA data
2: Pause loop output of NMEA data

Example

```
AT+NWGNSSMODE=0                                                    Obtain the position
+NWGNSSPOS:                                                        information.
$GPGGA,060239.00,2241.170914,N,11359.187225,E,2,16,2.5,116.6,M,,,,*39

OK
```

```
AT+NWGNSSMODE=1                                                    Obtain GPGSA
+NWGNSSPOS: $GPGSA,A,2,01,11,16,22,,,,,,,,,2.4,2.2,1.0*32          message.

OK
AT+NWGNSSMODE=2                                                    Obtain GPGSV
+NWGNSSPOS:                                                        message.
$GPGSV,5,1,19,01,57,160,50,11,84,151,31,16,09,090,41,22,10,140,49*74
$GPGSV,5,2,19,04,40,184,,07,57,314,,08,87,050,,10,03,164,*7B
$GPGSV,5,3,19,15,15,270,,19,,,,20,,,,21,50,071,*78
$GPGSV,5,4,19,24,13,196,,26,05,309,,28,,,,32,,,*7A
$GPGSV,5,5,19,41,,,41,42,,,43,50,,,43*72

OK
AT+NWGNSSMODE=3                                                    Obtain GPRMC
+NWGNSSPOS:                                                        message.
$GPRMC,074855.00,A,2241.207019,N,11359.188919,E,0.0,78.5,050517,2.3,W,A*16

OK
AT+NWGNSSMODE=4                                                    Obtain GPVTG
+NWGNSSPOS: $GPVTG,78.5,T,80.8,M,0.0,N,0.0,K,A*29                  message

OK
AT+NWGNSSMODE=5                                                    Obtain GPGLL
+NWGNSSPOS: $GPGLL,2241.207179,N,11359.188345,E,074856.00,A*0F     message

OK
AT+NWGNSSMODE=6                                                    Obtain GPZDA
+NWGNSSPOS: $GNZDA,075721.000,15,02,2023,00,00*4B                  message

OK
AT+NWGNSSNMEA=99,1                                                 Loop output of NMEA
OK                                                                 data
AT+NWGNSSNMEA=99,2                                                 Pause loop output
OK                                                                 of NMEA data
```

11.4 AT+NWGNSSCFG – Set GPS selecttion parameters

Set NMEA output frequency, switch coordinate systems, etc

Format

Type	Command	Response
Execute	AT+NWGNSSCFG=<n>,[<type>]<CR>	<CR><LF>OK <CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	GNSS output frequency 1000: 1Hz 500: 2Hz 200: 5Hz 100: 10Hz
[<type>]	0: WGS84 1: PZ90

Example

```
AT+NWGNSSCFG=1000                       GNSS output frequency 1Hz
OK
AT+NWGNSSCFG=1000,1                     GNSS output frequency 1Hz, Switch to PZ90
OK                                      coordinate system
```

11.5 AT+NWGNSSAUX – Save ephemeris information

To set GPS mode.

Format

Type	Command	Response
Execute	AT+NWGNSSAUX=<n>[,angle]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT+NWGNSSAUX=?	<CR><LF>+NWGNSSAUX: <0-1>[,<5,15>]<CR><LF>

Parameter

<n>	GNSS save ephemeris information 0: Delete saved ephemeris information 1: Save ephemeris information
angle	Navigation satellite cutoff angle, Optional parameters 5: The pitch cutoff angle is 5 degrees 15: The pitch cutoff angle is 15 degrees

Exampl

```
AT+NWGNSSAUX=1                                  Save ephemeris information
OK                                              Query GPS status.
AT+NWGNSSAUX=?
+NWGNSSAUX: <0-1>[,<5,15>]

OK
AT+NWGNSSAUX=0,15                               Set not to save ephemeris information, with a
OK                                              cutoff angle of 15 degrees
AT+NWGNSSAUX?                                   Query Configuration Status
+NWGNSSAUX: 0, 15

OK
```

11.6 AT+AGPSOPEN * – Setting APN of AGPS

To set the APN of AGPS.

Format

Type	Command	Response
Execute	AT+AGPSOPEN=<pdp_type>,<apnname><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<pdp_type> 1: IPV4
2: IPV6
3: IPV4V6

<apnname> APN, length ranging from 1 to 100.

Example

```
AT+AGPSOPEN=1,3GNET
                                                Set the APN of AGPS.
OK
```

11.7 AT+SETSERVER * – Setting AGPS Server

To set the server of AGPS.

SETSERVER must be set before GPS is enabled.

Format

Type	Command	Response
Execute	AT+SETSERVER =<n>,<url>,<port><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	current network mode 1: 3GPP2 3: 3GPP (only GSM, WCDMA, and LTE are supported)
<url>	Server address, IPV4, IPV6, and URL are supported
<port>	Server port

Example

```
AT+SETSERVER=3,supl.qxwz.com,7276                Set the server address of AGPS

OK
```

11.8 AT+NWGNSSDEL – Clearing GPS Data

To clear GPS data or cold start.

It is recommended to use this command before enabling GPS or after disabling GPS.

Format

Type	Command	Response
Execute	AT+NWGNSSDEL=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	0: GNSS cold start and clear NMEA data 1: Clear NMEA data

Example

```
AT+NWGNSSDEL=0                                  Clear NMEA data and cold start
OK
```

12 Network Sharing Commands

12.1 AT+NETSHAREACT –Enabling Network Sharing

To enable network sharing.

Format

Type	Command	Response
Execute	AT+NETSHAREACT=<cid>,<action>,<auto>[,<APN>[,<username>[,<passwd>[,<authtype>[,<ip_family>]]]]]	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NETSHAREACT?	<CR><LF>+NETSHAREACT:<stat>,<auto>,<err_code>,<wx_stat>,<PDP_type>,<share_mode> <CR><LF>OK<CR><LF>
Test	AT+NETSHAREACT=?	<CR><LF>+NETSHAREACT:(range of <cid> value),(range of <action> value),(range of <auto> value),<APN>,<username>,<passwd>,(range of <authtype> value) <CR><LF>OK<CR><LF>

Parameter

<cid>	PDP context, ranging from 1 to 11
<action>	Action 0: stop 1: start
<auto>	Specify whether to enable network sharing automatically after the module is powered on. 0: manual 1: auto
<APN>	Access Point Name, ranging from 0 to 99
<user_name>	User name, ranging from 0 to 255
<passwd>	password, ranging from 0 to 255
<auth_type>	Authentication type 0: NONE

	1: PAP (default)
	2: CHAP
	3: PAP and CHAP
<ip_family>	IP family
	4: IPv4 (Default)
	6: IPv6
	10: IPv4_6
<share_mode>	RNDIS/ECM
<stat>	0: disconnect
	1: connect
<atuo>	0: manual
	1: auto
<err_code>	Not supported
<wx_stat>	Notsupported
<PDP_type>	IPv4

Example

```
AT+NETSHAREACT=2,1,0,ctnet,card,card,1          Enable network sharing.
OK                                              Disable automatic sharing after power on
AT+NETSHAREACT=1,1,0,ctnet,card,card,1,6        IPv6 dialup
OK                                              network sharing is not supported
AT+NETSHAREACT=?                                Query the parameter ranges.
+NETSHAREACT: (0-11),(0-1),(0-1),"apn","user",
"passwd",(0-3)
OK
AT+NETSHAREACT?                                 Query the network sharing status.
+NETSHAREACT: 0,0,,,"IPV4",RNDIS
OK
```

12.2 AT$QCRMCALL – RMNET Dialup

To enable network sharing.

Format

Type	Command	Response
Execute	AT$QCRMCALL=<Action>,<Instance>[,<IPType>[,<Tech Pref>[,<umts profilenumber>[,<cdma profile number>[,<APN>]]]]]	<CR><LF>$QCRMCALL: <Instance>,<IPType><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF><NO CARRIER><CR><LF> Or <CR><LF><ERROR><CR><LF

Query	AT$QCRMCALL?	<CR><LF>$QCRMCALL: <Instance>,<IPType><CR><LF <CR><LF>OK<CR><LF>
Test	AT$QCRMCALL=?	<CR><LF>$QCRMCALL: (value range of supported <action>),(value range of supported <Instance>),(value range of supported <IP Type>),(value range of supported <Tech Pref>),(value range of supported <profile num>)<CR><LF <CR><LF>OK<CR><LF>

Parameter

<Action>	specifies whether to enable RMNET dialup 0: Stop 1: Start
<Instance>	1 - 8
<IP Type>	IP type 1: IPv4 2: IPv6 3: IPv4v6
<Tech Pref>	technology standard 1: 3GPP2 2: 3GPP
<umts_profile>	profile number, 1 to 24
<cdma profile number>	
<APN>	APN name maximum of 100 bytes

Example

```
AT$QCRMCALL=1,1,1                       Enable the RMNet dialup.
$QCRMCALL: 1,V4

OK
AT$QCRMCALL=1,1,1                       Failed to activate or network abnormality.
NO CARRIER
AT$QCRMCALL?                            Query the status of RMNet dialup.
$QCRMCALL: 1,V4                         The instance number and IP type are returned. RMNet
                                        dialup is enabled successfully.
OK
AT$QCRMCALL?                            RMNet dialup is not enabled successfully.
OK
```



```
AT$QCRMCALL=1,1,4                          Parameters are set incorrectly.
ERROR
AT$QCRMCALL=?                              Query the value ranges of parameters.
$QCRMCALL: (0-1),(1,2,3,4,5,6,7,8),(1-
3),(1-2),(1-24,100-179),,
OK
```

12.3 AT+NETSHAREMODE－Selecting Network Sharing Mode

To select the network sharing mode.

The setting by this command is saved after the module is powered off.

Format

Type	Command	Response
Execute	AT+NETSHAREMODE=<share_mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NETSHAREMODE?	<CR><LF>+NETSHAREMODE: <share_mode> <CR><LF>OK<CR><LF>

Parameter

<share_mode> Integer type
0: RNDIS (PID=8241)
1: ECM (PID=8249)
2: RMNET (PID = 8247 or 8243)

Example

```
AT+NETSHAREMODE=1              Set the network sharing mode to ECM.
OK
AT+NETSHAREMODE?               Query what the network sharing mode is set.
+NETSHAREMODE: 1               The current network sharing mode is ECM.
OK
```

13 SIM Card Operating Commands

13.1 AT+CCHO – Enabling Logical Channel of SIM Card

To enable the logical channel of SIM card.

Do not use the default AID.

Format

Type	Command	Response
Execute	AT+CCHO=<dfname><CR>	<CR><LF>+CCHO: <sessionid><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CCHO=?<CR>	<CR><LF>+CCHO=<dfname><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<dfname> AID, a string of 12 to 32 characters (the length must be an event number)

Example

```
AT+CCHO=a0000000871002ff86ffff89ffffffff
+CCHO: 1                                        Enable logical channel.

OK
AT+CCHO=?
+CCHO=<dfname>                                  Query the setting.

OK
```

13.2 AT+CCHC – Disabling Logical Channel of SIM Card

To disable the logical channel of the SIM card.

Format

Type	Command	Response
Execute	AT+CCHC=<sessionid><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CCHC=?<CR>	<CR><LF>+CCHC: <sessionid><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<sessionid> session ID returned to +CCHO.

Example

```
AT+CCHC=1

OK                                          Disable logical channel.


AT+CCHC=?
+CCHC=<sessionid>                           Query the setting.

OK
```

13.3 AT+CGLA – Sending APDU Command

To send APDU command through the logical channel of the SIM card.

For details about the format of APDU commands, see the 3GPP protocol.

Format

Type	Command	Response
Execute	AT+CGLA=<sessionid>,<lengt	<CR><LF>+CGLA:

	h>,<command><CR>	<sessionid>,<length>,<command><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CGLA=?<CR>	<CR><LF>+CGLA: <sessionid>,<length>,<command> <CR><LF><CR><LF>OK<CR><LF>

Parameter

<sessionid> session ID returned to +CCHO.
<length> command length
<command> APDU command

Example

```
AT+CGLA=1,10,81F2000000
+CGLA:118,"6237820278218410A0000000871002FF86FFFF89FFFFF
FFFA50C8001718201FF830400004B5C8A01058B032F0607C60990014
08301018301819000"                                          Send APDU command.

OK
AT+CGLA=?
+CGLA: <sessionid>,<length>,<command>
                                                            Query the setting.

OK
```

14 NITZ Commands

14.1 AT+CTZU – Automatic Time Zone Update

To enable and disable automatic time zone update on the device.

The setting by this command is saved after the module is powered down.

Format

Type	Command	Response
Set	AT+CTZU=[<onoff>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF>
Query	AT+CTZU?<CR>	<CR><LF>+CTZU: <onoff><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF>

Parameter

<onoff> integer type

0: disable automatic time zone update (default)

1: enable automatic time zone update

Example

```
AT+CTZU=1                                   Enable automatic time zone update.
OK
AT+CTZU?                                    Query the NITZ status.
+CTZU: 1                                    Automatic time zone update is enabled.
```

```
OK
```

14.2 AT+CTZR – Reporting Time Zone Changes

To report the time zone changes.

To update local time, execute +CCLK.

Format

Type	Command	Response
Execute	AT+CTZR=<reporting><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF>
Query	AT+CTZR?<CR>	<CR><LF>+CTZR: <reporting><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF>
URC	+CTZV: <tz><CR> +CTZE: <tz>, <dst>, [<time>]<CR>	

Parameter

<reporting>	integer type 0: disable report of time zone changes. (default) 1: enable Unsolicited result code of time zone changes by +CTZV: <tz>. +CTZV: <tz>. 2: enable unsolicited time zone report by +CTZE:<tz>, <dst>, [<time>]. +CTZE:<tz>, <dst>, [<time>].
<tz>	local time zone (every 15 minutes as a unit), a character string in format of ±zz. zz indicates two integrals, and the value ranges from -96 to +96. Keep these values in a fixed length. For example, use “-09” or “+09” to represent -9 or +9.
<dst>	indicates whether to adjust the day light saving time, integer type. 0: no adjustment 1: +1 hour adjustment 2: +2 hour adjustment
<time>	local time, a character string in the format YYYY/MM/DD,hh:mm:ss. MT obtains the

local time from the information that the network provides.

Example

```
AT+CTZR=1                                          Enable reporting
OK
AT+CTZR?
+CTZR: 1                                           Query the report status
OK                                                 Unsolicited result code is enabled.


+CTZV: “+32”                                       The time zone is UTC+8. Automatic reporting, not
                                                   instruction code back
+CTZE: “+32”,0,2018/10/13,15:32:33                 Report the local time. Automatic reporting, not
                                                   instruction code back
```

15 File System Commands

15.1 AT+FSWF –Writing Data to File

To write data to a file.

Format

Type	Command	Response
Execute	AT+FSWF=<file_name>,<mode>,<size>,<time><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+FSWF: Timeout!<CR><LF>

Parameter

<file_name>	File name, at most 50 characters
<mode>	0: If the file already exists, new data is written to the start of the file and then it overwrites original data. 1: If the file already exists, new data is written to the end of the file.
<size>	Data size, ranging from 0 to 16384, unit: byte. The data to be written should not exceed <size>.
<time>	timeout period, ranging from 0 to 60000, unit: ms

- The data to be written should not exceed <size>.
- The size of the user disk is 512 KB, i.e. the total size of the files that can be stored is 512 KB.

Example

```
AT+FSWF="test.txt",1,1024,10000          Write 1024-byte data to the test.txt file.
>
OK
```

```
AT+FSWF="test.txt",1,1024,10000          The command times out after 10 seconds.
>
+FSWF: Timeout!
AT+FSWF="test.txt",1,1024,60001          ERROR is returned because the set value exceeds the
ERROR                                    parameter range.
```

15.2 AT+FSRF－Reading Data from a File

To read data from a file

Format

Type	Command	Response
Execute	AT+FSRF=<file_name>,<mode>,<size>[,<position>]<CR>	<CR><LF>+FSRF: <size>,<content><CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<file_name>	File name, at most 50 characters
<mode>	0: read data from the beginning of the file. 1: read data starting from <position> of the file.
<size>	data size, not exceed the size of the file, 0 is valid.
<position>	the position in the file, where data to be read starts, valid when <mode> is set to 1. <Size> and <position> are determined by the file size.
<content>	Content of the file to be read.

- the specified data size cannot exceed the total size of the file.
- <size> and <position> are determined by the file size.

Example

```
AT+FSRF="test.txt",0,10                  Read 10-byte data from the beginning of the test.txt
+FSRF: 10,start01234                     file.
OK
AT+FSRF="test.txt",0,0                   Read 0-byte data from the beginning of the test.txt file.
+FSRF: 0,
OK
```

```
AT+FSRF="test.txt",0,1025                ERROR is returned because <size> exceeds the file size.
ERROR
AT+FSRF="test.txt",1,20,2                Read 20-byte data from the second byte of the test.txt
+FSRF: 20,tart0123456789012345           file.
OK                                       The data is read successfully.
AT+FSRF="test.txt",1,0,2                 Read 0-byte data from the second byte of the test.txt
+FSRF: 0                                 file.
OK
AT+FSRF="test.txt",1,10,0                Read 10-byte data from the begining of the test.txt file.
+FSRF: 10,start01234
OK
```

15.3 AT+FSDF – Deleting a File

To delete a file.

Format

Type	Command	Response
Execute	AT+FSDF=<file_name><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<file_name> File name, the file length does not exceed 50 characters.

Example

```
AT+FSDF="test.txt"                       Delete the text.txt file.
OK
AT+FSDF="123.txt"                        Delete the 123.txt file. ERROR is returned because the
ERROR                                    file does not exist.
```

15.4 AT+FSLIST – Obtaining File List

To obtain the list of files in the file system.

Format

Type	Command	Response
Query	AT+FSLIST?<CR>	<CR><LF>OK<CR><LF> Or <CR><LF><file_name>,<size> <CR><LF><file_name>,<size> …… <CR><LF>OK<CR><LF>

Parameter

<file_name> Specify the
<size> File size

Example

```
AT+FSLIST?                          The size of file.txt is 6000 bytes.
i.amr,6181
file.txt,6000
OK
AT+FSLIST?                          No file in the file system
OK
```

15.5 AT+FSFS – Obtaining the Size of a File

To obtain the size of a file.

Format

Type	Command	Response
Execute	AT+FSFS=<file_name><CR>	<CR><LF>+FSFS: <size> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<file_name> File name, the file length does not exceed 50 characters.

Example

```
AT+FSFS="test.txt"                    Obtain the size of the text.txt file.
+FSFS: 1024                           The size is 1024 bytes.
OK
AT+FSFS="123.txt"                     Obtain the size of the 123.txt file.
ERROR                                 ERROR is returned because the file does not exist.
```

15.6 AT+FSLS – Obtaining the Remaining Storage Size of User Disk

To obtain the remaining storage size of user disk.

Format

Type	Command	Response
Query	AT+FSLS?<CR>	<CR><LF>+FSLS: <size> <CR><LF>OK<CR><LF> Or <CR><LF>+FSLS: DiskInfo Not Right!<CR><LF>

Parameter

<size> Remaining storage size of user disk.

- The size of user disk is 512 Kb in total.
- The block size of the file system is 512 bytes. For example, if the total size of the user disk is 100 bytes, the query result will be 65024 (65536-512), and the actual remaining storage size is 65436 (65536-100).

Example

```
AT+FSLS?                  The remaining storage size of user disk is 48128 bytes.
+FSLS: 48128              48128=512*94, that is, there are 94 data blocks (512 bytes/block).
OK
AT+FSLS?                  The user disk is not ready.
+FSLS: DiskInfo Not
Right!
```

15.7 AT+FSFAT – Formating the User Disk

To format the user disk.

Format

Type	Command	Response
Execute	AT+FSFAT<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Example

```
AT+FSFAT                    Format the user disk.
OK
AT+FSLS?                    The remaining storage size of user disk is 65536 bytes.
+FSLS: 65536
OK
AT+FSFAT                    The user disk is not ready.
ERROR
```

16 eCall Commands

16.1 AT+NWECALLMSD – Writing MSD Data Directly

This command is used to set/get the whole MSD in hex bytes.
The max length of <msd_data> is 280 hex characters, which represents 140 bytes of MSD. Spaces in <msd_data> will be skipped; and characters out of the range of '0'~'f' will be regarded as '0'

The setting by this command is not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+NWECALLMSD=<MSD Data><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWECALLMSD=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<MSD Data> MSD message in hexadecimal format

Example

```
AT+NWECALLMSD="02251C0680E30A51439E2955D43800800437F
80A3105669023F8A71166932185B004150043C040"          Write complete MSD data into the module.
OK

AT+NWECALLMSD=?                                     Test mode
OK
```

16.2 AT+NWECALLCFG–Setting eCall Parameters

This command is used to set the eCall-related parameter of IVS.
The setting by this command is not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+NWECALLCFG=<param>,<value>[,<testnum>][,<reconfignum>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+ NWECALLCFG=?<CR>	<CR><LF>+NWECALLCFG: <param>(value range of supported <value>)<CR><LF> <CR><LF>OK<CR><LF>
Query	AT+ NWECALLCFG?<CR>	<CR><LF>+NWECALLCFG: <param><value><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<param>

"T3": IVS INITIATION signal duration

- START: T3 is started as soon as the IVS-NAD starts sending the INITIATION signal
- STOP: T3 stops when the IVS-NAD receives a SEND MSD signal from the PSAP,at which time the IVS-NAD shall stop sending the INITIATION signal
- EXPIRY: Upon expiry of T3 the IVS-NAD shall stop sending the INITIATION signal

"T5": IVS wait for SEND MSD period

- START: T5 starts as soon as the IVS-NAD received notification that the call is first answered
- STOP: T5 stops when the IVS-NAD detects a SEND MSD signal sent by the PSAP
- EXPIRY: Upon expiry of T5 the IVS-NAD shall reconnect the IVS audio system and terminate eCall specific behavior (i.e.it shall not proceed with the sending of MSD data) until requested to do otherwise

"T6": IVS wait for AL-ACK period

- START: T6 starts as soon as the IVS-NAD has received LL-ACK
- STOP: T6 stops when the IVS-NAD receives an AL-ACK message
- EXPIRY: Upon expiry of T6 the IVS-NAD shall mark the transfer of the MSD as unsuccessful and reconnect the IVS audio system and terminate eCall specific behavior until requested to do otherwise

"T7": IVS MSD maximum transmission time

- START: T7 starts as soon as the IVS-NAD starts sending the MSD data
- STOP: T7 stops when the IVS-NAD receives an LL-ACK message
- EXPIRY: Upon expiry of T7 the IVS-NAD shall mark the transfer of the MSD as unsuccessful and reconnect IVS audio system and terminate eCall specific behavior until requested to do otherwise

	“ecallonly”: Activate or disable eCall only mode “ecallmutespk”: Enable or disable to mute IVS speaker in MSD transmission. “ecallmode”: Configure eCall mode
<value>	<T3>: (2-255) <T5>: (5-255) <T6>: (5-255) <T7>: (20-255) <ecallonly> 0: Disable eCall only mode (returns to normal operation based on the used SIM) 1: Activate eCall only mode on a normal SIM or a SIM that supports both eCall and normal call mode <ecallmutespk>: 0: Do not automatically mute speaker during MSD transmission 1: Automatically mute speaker during MSD transmission Default :1 <ecallmode>: 0: for EU mode 1: for ERA-GLONASS mode Default :0 if<param>=”ecallonly”, <ecallonly>=0; <testnum> and <reconfignum> can be set
<testnum>	Specifies the eCall test number, which will be used when AT+CECALL=0.
<reconfignum>	Specifies the eCall reconfiguration number, which will be used when AT+CECALL=1

Example

```
AT+NWECALLCFG=?
+NWECALLCFG: "T3",(2-255)
+NWECALLCFG: "T5",(5-255)
+NWECALLCFG: "T6",(5-255)
+NWECALLCFG: "T7",(20-255)
+NWECALLCFG: "ecallonly",(0-1)                    Test mode
+NWECALLCFG: "ecallmutespk",(0-1)
+NWECALLCFG: "ecallmode",(0-1)

OK
AT+NWECALLCFG?
+NWECALLCFG: "T3",2
+NWECALLCFG: "T5",5
+NWECALLCFG: "T6",5                               Query mode
+NWECALLCFG: "T7",20
+NWECALLCFG: "ecallonly",0
+NWECALLCFG: "ecallmutespk",1
+NWECALLCFG: "ecallmode",0
```

```
OK
AT+NWECALLCFG=”T6”,6                                    Set T6 = 6s
OK
AT+NWECALLCFG="ecallonly",0,13609289008                 Disable eCall-only mode and set the
OK                                                      test number to 13609289008.
AT+NWECALLCFG="ecallonly",0,,18812345678                Set reconfignum = 18812345678
OK
AT+NWECALLCFG="ecallmutespk",0                          Do not automatically mute speaker
OK                                                      during MSD transmission
AT+NWECALLCFG="ecallmode",1                             Set ERA eCall mode
OK
```

16.3 AT+CECALL–Establishing/Releasing eCall

The eCall function can be initiated automatically by the vehicle sensor, or be initiated manually.

The setting by this command is not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+CECALL=<type_of_eCall>[,<format>,<msd_data_length>,<msd_data>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CECALL?<CR>	<CR><LF>+CECALL: <eCall_type><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CECALL=?<CR>	<CR><LF>+CECALL: (value range of supported <eCall_type>)<CR><LF> <CR><LF>OK<CR><LF>

Currently, only <type_of_eCall> of the command is supported.

Parameter

<eCall_type> eCall type:
0: Test ecall mode, used to call testnum

1: Re-configuration ecall mode, used to call reconfignum.
2: Initiate eCall manually.
3: Initiate eCall automatically.
4: eCall not activated.
5: Release eCall

Example 1

```
AT+CECALL=?
+CECALL: (0-5)                                  Query the value range of supported
                                                eCall_type

OK
AT+CECALL?
+CECALL: 4
                                                Query the value of eCall_type parameter

OK
```

Example 2

Disable eCall-only mode, make a test call, and then release the call.

```
AT+NWECALLCFG="ecallonly",0,13279345260         Disable eCall-only mode and set the test
OK                                              number to 13279345260
AT+CECALL=0
OK                                              Establish a test eCall

+CLCC: 1, 0, 2, 0, 0, "13279345260", 129, "",, 0    2:dialing (MO call)

+CLCC: 1, 0, 3, 0, 0, "13279345260", 129, "",, 0    3:alerting (MO call)

+CLCC: 1, 0, 0, 0, 0, "13279345260", 129, "",, 0    0:active
AT+CECALL=5
OK                                              Release eCall

NO CARRIER
```

Example 3

Disable eCall only mode and set reconfiguration number to 13609289008

```
AT+NWECALLCFG="ecallonly",0,,13609289008
                                                Set reconfignum to 13609289008.
OK
AT+CECALL=1
                                                Establish a reconfiguration ecall
OK
```



```
+CLCC: 1, 0, 2, 0, 0, "13609289008", 129, "",, 0        2:dialing (MO call)

+CLCC: 1, 0, 3, 0, 0, "13609289008", 129, "",, 0        3:alerting (MO call)

+CLCC: 1, 0, 0, 0, 0, "13609289008", 129, "",, 0        0:active
```

16.4 AT+NWECALLPUSH–Pushing MSD to PSAP

To push MSD to PSAP. This command must be executed after AT+CECALL.

The setting by this command is not saved after the module is powered down.

Format

Type	Command	Response
Execute	AT+NWECALLPUSH<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWECALLPUSH=?<CR>	<CR><LF> +NWECALLPUSH <CR><LF> <CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT+NWECALLPUSH                                          Push MSD to PSAP
OK

AT+NWECALLPUSH=?
+NWECALLPUSH                                            Test mode

OK
```

16.5 +NWECALLIND–eCall Related URCs

eCall related URCs

Format

Type	Command
Execute	+NWECALLIND: <report_id>[,<report_data>]

Parameter

<report_id> ID of the URC message
<1>URC of failed eCall
24. T5 expired
25. T6 expired
26: T7 expired
<2>URC of Successful eCall MSD Transmission and the Request for Updating MSD
9: specify the IVS and PASP to initiates the PULL process and requests MSD data.
17: Higher Layer ACK received (similar to AL-ACK(#0) but without data)
<3>URC of eCall Process Information
0: AL-ACK signal received
<report_data>: four information bits received through AL-ACKs, returned within a single byte of hex data (for example: +NWECALLIND: 0,"0B". This means that 0xB is the value of the 4 information bits); that can indicate positive AL-ACK; clear-down, etc.
1: eIM lost synchronization with PSAP (A resynch occurs after the synchronization
is lost.)
2: NACK signal received
6: eIM is terminated: fatal error (3GPP Abort function)
10: Start Send MSD: eIM starts sending MSD
11: The eIM starts sending the Initiation signal (in PUSH mode only).
12: SEND MSD signal received
14: IVS eIM is synced with PSAP
15: IVS eIM is synced with PSAP eIM
16: Link layer ACK received
18: The eIM transmitter idles after a full reset. The eIM transmitter is reset after
sync loss (for example, because of the end of PSAP transmission).
21: MSD update is done. The IVS transmitter is ready to send the new MSD at the next PULL request. That is, the update was done while receiving "Update on idle" and the IVS was Idle or the update was done after sending notification #20 and the IVS became Idle and sent the MSD.
22: MSD update request cannot be processed. Occurs when an update is attempted when currently processing a previous request. That is, URC 20 or 21 was not sent yet

	23: T3 expired 27: Reserved
<report_data>	The URC message (optional)

Example

Pull mode, PSAP requests MSD data from IVS, IVS-side process (Execute the +NWECALLMSD command to update data)

```
+NWECALLIND: 9                                          The IVS side receives the message of
                                                        requesting MSD data from the PSAP side.
AT+NWECALLMSD="02251C0680E30A51439E2955D43800800437F8
0A3105669023F8A71166932185B004150043C040"               Update MSD data and send it to PSAP.
OK
```

16.6 Examples of eCall AT Process

1 MO eCall: Establish a test eCall and Transfer MSD in Push Mode

Before sending AT commands, complete the PSAP configuration. Then send AT commands accroding to the following process.

The following example shows one possible transmission

```
AT+NWECALLCFG="ecallonly",0,13609289008                 Set the PSAP number to a test number,
OK                                                      such as 13609289008
AT+CECALL=0
OK                                                      Establish a test eCall

+CLCC: 1, 0, 2, 0, 0, "13279345260", 129, "",, 0        2:dialing (MO call)

+CLCC: 1, 0, 3, 0, 0, "13279345260", 129, "",, 0        3:alerting (MO call)

+CLCC: 1, 0, 0, 0, 0, "13279345260", 129, "",, 0        0:active
AT+NWECALLMSD="02251C0680E30A51439E2955D
43800800437F80A3105669023F8A71166932185B
004150043C040"                                          Write complete MSD data into the module
OK
AT+NWECALLPUSH                                          Push MSD to PSAP
OK

+NWECALLIND: 21
                                                        MSD updata is done
```

```
+NWECALLIND: 11                                    eIM starts sending the Initiation signal

+NWECALLIND: 14                                    IVS eIM is synced with PSAP

+NWECALLIND: 23                                    T3 expired

+NWECALLIND: 14                                    IVS eIM is synced with PSAP

+NWECALLIND: 14                                    IVS eIM is synced with PSAP

+NWECALLIND: 15                                    IVS eIM is synced with PSAP eIM

+NWECALLIND: 12                                    SEND MSD request received

+NWECALLIND: 10                                    eIM starts sending MSD

+NWECALLIND: 2                                     NACK signal received

+NWECALLIND: 2                                     NACK signal received

+NWECALLIND: 2                                     NACK signal received

+NWECALLIND: 2                                     NACK signal received

+NWECALLIND: 16                                    Link layer ACK received

+NWECALLIND: 0,00                                  eIM completes sending MSD

+NWECALLIND: 17                                    Higher Layer ACK received

+NWECALLIND: 27

+NWECALLIND: 1                                     eIM lost synchronization with PSAP

+NWECALLIND: 18                                    eIM transmitter idles after a full reset

NO CARRIER                                         PSAP hangs up eCall
```

2 MO eCall: Automatically initiate eCall and Transfer MSD in Push Mode

If user automatically initiates eCall, AT+CECALL=3 should be used

To enable push mode, AT+NWECALLPUSH should be used (MUST be after AT+CECALL).

The following example shows one possible transmission

```
AT+NWECALLMSD="02251C0680E30A51439E2955D
43800800437F80A3105669023F8A71166932185B
004150043C040"                                  Set whole MSD
OK
AT+CECALL=3                                     Automatically initiate eCall
OK
AT+NWECALLPUSH                                  Push MSD to PSAP
OK

+CLCC: 1, 0, 2, 0, 0, "112", 129, "",, 0        2:dialing (MO call)

+CLCC: 1, 0, 3, 0, 0, "112", 129, "",, 0        3:alerting (MO call)

+CLCC: 1, 0, 0, 0, 0, "112", 129, "",, 0        0:active

+NWECALLIND: 21                                 MSD updata is done

+NWECALLIND: 11                                 eIM starts sending the Initiation signal

+NWECALLIND: 23                                 T3 expired

+NWECALLIND: 14                                 IVS eIM is synced with PSAP

+NWECALLIND: 14                                 IVS eIM is synced with PSAP

+NWECALLIND: 14                                 IVS eIM is synced with PSAP

+NWECALLIND: 15                                 IVS eIM is synced with PSAP eIM

+NWECALLIND: 12                                 SEND MSD request received

+NWECALLIND: 10                                 eIM starts sending MSD

+NWECALLIND: 2                                  NACK signal received

+NWECALLIND: 2                                  NACK signal received

+NWECALLIND: 2                                  NACK signal received

+NWECALLIND: 16                                 Link layer ACK received

+NWECALLIND: 0,00                               eIM completes sending MSD

+NWECALLIND: 17                                 Higher Layer ACK received
```

```
+NWECALLIND: 27

+NWECALLIND: 1                                          eIM lost synchronization with PSAP

+NWECALLIND: 18                                         eIM transmitter idles after a full reset

NO CARRIER                                              PSAP hangs up eCall
```

3 MO eCall: Manually initiate eCall and Transfer MSD in Pull Mode

If user manually initiates eCall, AT+CECALL=2 should be used.

If application does not execute push command, after eCall is established PSAP should request MSD and IVS sends MSD correspondingly.

The following example shows one possible transmission

```
AT+CECALL=2
OK                                                      Manually initiate eCall

+CLCC: 1, 0, 2, 0, 0, "112", 129, "",, 0                2:dialing (MO call)

+CLCC: 1, 0, 3, 0, 0, "112", 129, "",, 0                3:alerting (MO call)

+CLCC: 1, 0, 0, 0, 0, "112", 129, "",, 0                0:active

+NWECALLIND: 21                                         MSD update is done

+NWECALLIND: 14                                         IVS eIM is synced with PSAP

+NWECALLIND: 14                                         IVS eIM is synced with PSAP

+NWECALLIND: 14                                         IVS eIM is synced with PSAP

+NWECALLIND: 15                                         IVS eIM is synced with PSAP eIM

+NWECALLIND: 12                                         SEND MSD request received

+NWECALLIND: 9                                          Request MSD from eCall application
AT+NWECALLMSD="02251C0680E30A51439E2955D
43800800437F80A3105669023F8A71166932185B
004150043C040"                                          Write complete MSD data into the module
OK
+NWECALLIND: 21
                                                        MSD updata is done

+NWECALLIND: 1                                          eIM lost synchronization with PSAP
```

```
+NWECALLIND: 12                                    SEND MSD request received

+NWECALLIND: 10                                    eIM starts sending MSD

+NWECALLIND: 2                                     NACK signal received

+NWECALLIND: 2                                     NACK signal received

+NWECALLIND: 2                                     NACK signal received

+NWECALLIND: 16                                    Link layer ACK received

+NWECALLIND: 0,00                                  eIM completes sending MSD

+NWECALLIND: 17                                    Higher Layer ACK received

+NWECALLIND: 27

+NWECALLIND: 1                                     eIM lost synchronization with PSAP

+NWECALLIND: 18                                    eIM transmitter idles after a full reset

NO CARRIER                                         PSAP hangs up eCall
```

4 MT eCall: Transfer MSD in Pull Mode

After the completion of the eCall,eCall Daemon could automatically accept PSAP call back.

When eCall is established, MSD can be transferred in push mode or pull mode:

(1)If user wants to transfer MSD in push mode, AT+NWECALLPUSH command could be used.

(2)If“+NWECALLIND: 9” is reported, it indicates PSAP requests MSD. After application updates MSD using AT+NWECALLMSD, the MSD will be transmitted to PSAP.

The following example shows one possible transmission

```
+CLCC: 1, 1, 6, 0, 0, "112", 129, "",, 0           6: offering(MT call,PSAP call back)

+CLCC: 1, 1, 4, 0, 0, "112", 129, "",, 0           4: incoming (MT call)

+CLCC: 1, 1, 0, 0, 0, "112", 129, "",, 0           0:active(ecall_daemon automatically
                                                   answer call)
+NWECALLIND: 14                                    IVS eIM is synced with PSAP

+NWECALLIND: 14                                    IVS eIM is synced with PSAP
```

+NWECALLIND: 14	IVS eIM is synced with PSAP
+NWECALLIND: 15	IVS eIM is synced with PSAP eIM
+NWECALLIND: 12	SEND MSD request received
+NWECALLIND: 9	Request MSD from eCall application
AT+NWECALLMSD="02251C0680E30A51439E2955D43800800437F80A3105669023F8A71166932185B004150043C040" OK	Write complete MSD data into the module
+NWECALLIND: 21	MSD updata is done
+NWECALLIND: 1	eIM lost synchronization with PSAP
+NWECALLIND: 12	SEND MSD request received
+NWECALLIND: 10	eIM starts sending MSD
+NWECALLIND: 2	NACK signal received
+NWECALLIND: 2	NACK signal received
+NWECALLIND: 2	NACK signal received
+NWECALLIND: 16	Link layer ACK received
+NWECALLIND: 0,00	eIM completes sending MSD
+NWECALLIND: 17	Higher Layer ACK received
+NWECALLIND: 27	
+NWECALLIND: 1	eIM lost synchronization with PSAP
+NWECALLIND: 18	eIM transmitter idles after a full reset
NO CARRIER	PSAP hangs up eCall

17 FOTA Commands

17.1 AT+NEOFTPFOTA–FTP FOTA Command

To control the firmware-over-the-air of the module.

Do NOT power down or restart the module during the upgrade.

If the baud rate is set to automatic detection, send AT\r\n to detect the baud rate and then the module returns the upgrade result.

Format

Type	Command	Response
Set	AT+NEOFTPFOTA=<url>,<port>,<user>,<pwd>,<pkt_name><CR>	<CR><LF>OK<CR><LF> Or Failed to download the upgrade package. <CR><LF>+NEODOWNLOAD: FAIL<CR><LF> Or The upgrade package is downloaded successfully. <CR><LF>+NEODOWNLOAD: SUCCESS<CR><LF> Or Start to upgrade. <CR><LF>+NEOUPDATE: START<CR><LF> Or Upgrade successfully. <CR><LF>+NEOUPDATE: SUCCESS<CR><LF> Failed to upgrade. <CR><LF>+NEOUPDATE: FAIL<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NEOFTPFOTA=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<url>	IP address of the server
<port>	Port number of the server

<user>	User name
<pwd>	Password
<pkt_name>	File name of the upgrade package

Example

```
AT+NEOFTPFOTA=58.60.184.213,12008,neoway,neoway,upda   Start to upgrade.
te-V013C-V013B.zip
OK

                                                       The upgrade package is
+NEODOWNLOAD: SUCCESS                                  downloaded successfully.

+NEOUPDATE: START                                      Start to upgrade.

+NEOUPDATE: SUCCESS                                    Upgrade successfully.

AT+NEOFTPFOTA?                                         Test mode
OK
```

18 MQTT Commands

18.1 AT+MQTTTLS –Configuring TLS Parameters

To configure MQTT TLS parameters.

Execute **AT+CERTADD** to import certificate.

Format

Type	Command	Response
Execute	AT+MQTTTLS=<type>,<type_name><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MQTTTLS?<CR>	<CR><LF>+MQTTTLS:<sslmode>,<authmode>,<rootca_name>,<clientcert_name>,<clientkey_name> <CR><LF>OK<CR><LF>
Test	AT+MQTTTLS=?<CR>	<CR><LF>+MQTTTLS: <type>,<value> <CR><LF>OK<CR><LF>

Parameter

<type> parameter type
sslmode: SSL mode
authmode: authentication mode
rootca: CA certificate
clientcert: client certificate
clientkey: client key

<type_name> setting of SSL
sslmode:
0: not authentication
1: authentication
Authmode:
0: verify optional
1: verify required
rootca: string, CA certificate

clientcert: string, client certificate
clientkey: string, client key

Example

```
AT+MQTTTLS=authmode,1                                  Set authentication mode to verify
OK                                                     required.
AT+MQTTTLS?
+MQTTTLS: 1,1,ca.pem,cc.pem,ck.pem                      Query SSL configuration.
OK
AT+MQTTTLS=?
+MQTTTLS: <type>,<type_name>                            Query value range of parameters.
OK
```

18.2 AT+MQTTCONNPARAM – Setting User Parameters

To set ID, user name, and password.

Format

Type	Command	Response
Execute	AT+MQTTCONNPARAM=<clientdID>,<username>,<password><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MQTTCONNPARAM?<CR>	<CR><LF>+MQTTCONNPARAM:<"clientID">,<"username">,<"password"> <CR><LF>OK<CR><LF>
Test	AT+MQTTCONNPARAM=?<CR>	<CR><LF>+MQTTCONNPARAM: <cliendid>,<username>,<password> <CR><LF>OK<CR><LF>

Parameter

<"clientID"> Device ID, max. 256 bytes.
<"username"> User name, max. 512 bytes.
<"password"> Password, max. 256 bytes.

Example

```
+MQTTCONNPARAM="C_201801021127","lixytest/thing01","0lSoY/e
YnlSqUeAsbAKKQ/ACmipZwEw9H7Ff0h1kOps="                    Parameters are set
                                                          successfully.
```

18.3 AT+MQTTWILLPARAM – Will Settings

To set will parameters.

Format

Type	Command	Response
Execute	AT+MQTTWILLPARAM=<retained>,<qos>,<topicname>,<message><CR>	<CR><LF>+GNSSSTATE:<status><CR><LF> Or <CR><LF>OK<CR><LF>
Query	AT+MQTTWILLPARAM?<CR>	<CR><LF>+MQTTWILLPARAM:<retained>,<qos>,<"topicname">,<"message"> <CR><LF>OK<CR><LF>
Test	AT+MQTTWILLPARAM=?<CR>	<CR><LF>+MQTTWILLPARAM: <retained>,<qos>,<topicname>,<messag e> <CR><LF>OK<CR><LF>

Parameter

<retained>	Retain symbol, digit type, 0 and 1.
<qos>	Quality of service, 0 and 1 supported
<"topicname">	Will topic, max. 128 bytes. The parameter must be marked by quotation marks.
<"message">	Will Message, max. 1024 bytes

Example

```
AT+MQTTWILLPARAM=0,1,"lixytopic",byby
OK                                                The will is set successfully.
```

18.4 AT+MQTTWILLMSG – Setting Long Will Messages

To set long will messages or will messages of non-character string by specifying retained, qos, topic, and message length.

If the message is not input completely within 30 seconds, the UART returns timeout.

Format

Type	Command	Response
Execute	AT+MQTTWILLMSG=<retained>,<qos>,<"topic">,<msg_length><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<retained>	Retain flag, digit type, 0 or 1.
<qos>	QoS of publish message
<"topic">	Publish topic
<willmsg_length>	Length of will messages, maximum 10240 bytes. After the UART returns >, input message content of the specified length.

Example

```
AT+MQTTWILLMSG =1,1,"neoway02",10 >
OK                                          Set will message. Successfully
AT+MQTTWILLMSG=1,1,"neoway02",10 >
+MQTTWILLMSG: Timeout!                      Set will message. Failed
```

18.5 AT+MQTTCONN – Connection Command

To connect to the MQTT server.

Format

Type	Command	Response

Execute	AT+MQTTCONN=<host>,<clean>,<keep_alive><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MQTTCONN?<CR>	<CR><LF>+MQTTCONN:<"ip:port">,<clean>,<keep_alive> <CR><LF>OK<CR><LF>
Test	AT+MQTTCONN=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<Host>	Server address (URL:port)
<clean>	whether to clean session, digit type, 0-Not clean (by default) 1-Clean
<Keep_alive>	keepAlive time,ranging from 20 to 180,unit: second

Example

```
AT+MQTTCONN=121.43.166.63:1883,0,60                Connect to the MQTT server
OK                                                 successfully.
```

18.6 AT+MQTTSUB – Subscription

To subscribe a topic.

Topic name must be marked by quotation marks.

Format

Type	Command	Response
Execute	AT+MQTTSUB=<topicname>,<qos><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MQTTSUB?<CR>	<CR><LF>+MQTTSUB:<"topicname">,<qos> <CR><LF>OK<CR><LF>
Test	AT+MQTTSUB=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<topicname>	Topic to subscribe to, max. 128 bytes
<Qos>	Quality of service, 0, 1, and 2 are supported

Example

```
AT+MQTTSUB="neoway02",1
OK                                          Subscribe to the topic successfully. The
                                            server issues the topic retained last
                                            time.
+MQTTSUB:9,"neoway02",11,neoway mqtt

AT+MQTTSUB=“neoway02”,1                     Subscribed to the topic successfully.
OK
```

18.7 AT+MQTTUNSUB – Unsubscription

To unsubscribe a topic.

Format

Type	Command	Response
Execute	AT+MQTTUNSUB=<topicname><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<topicname>	Topic to subscribe to, max. 128 bytes

Example

```
AT+MQTTUNSUB="neoway02"
OK                                          Unsubscribe to a topic
```

18.8 AT+MQTTPUB – Publishing Topic

To publish a topic.

Format

Type	Command	Type
Execute	AT+MQTTPUB=<retained>,<qos>,<topicname>,<message><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<Retained>	Retain mark, digit type, 0 and 1.
<Qos>	Quality of service, 0 and 1 supported.
<topicname>	Topic, max. 128 bytes
<Message>	Message, max. 1024 bytes

Example

```
AT+MQTTPUB=1,1,"neoway02", neoway mqtt
                                          The topic is published successfully.
OK
AT+MQTTPUB=1,1,"neoway02", neoway mqtt
                                          The server does not support QoS=2. After the
OK                                        command times out, the device disconnects to
                                          the MQTT server.
                                          Developers need to reconnect manually.
+MQTTSUB:5,"neoway02",11, neoway mqtt
```

18.9 AT+MQTTPUBS – Publishing Topic

To publish topic.

Format

Type	Command	Response
Execute	AT+MQTTPUBS=<retained>,<topic>,<qos>,<msg_length><CR>	<CR><LF>> <CR><LF>OK<CR><LF> Or <CR><LF>> <CR><LF>+MQTTPUBS: Timeout!<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<retained>	retain symbol, digit type, 0 and 1.
<topic>	topic to be published.
<qos>	Quality of service, 0 and 1 are supported
<msg_length>	the length of message published, max. 10240 bytes

Example

```
AT+MQTTPUBS=1,1,"lixytopic",10
>                                          Topic publishes 10-byte data Input 10-byte
OK                                         data.


AT+MQTTPUBS=0,1,"lixytopic",12
>                                          Times out
+MQTTPUBS: Timeout!
```

18.10 AT+MQTTDISCONN – Disconnecting to the MQTT Server

To disconnect to the MQTT server and release resources.

The device disconnects to the MQTT server proactively and releases the MQTT resources.

To publish messages after disconnecting, set up the connection again.

Format

Type	Command	Response
Execute	AT+MQTTDISCONN<CR>	<CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT+MQTTDISCONN                                  Disconnect to the MQTT server.
OK
```

18.11 +MQTTSUB – Notifying Topic Content

To notify the content of the received topic sent by the server.

Format

Type	Command
URC	+MQTTSUB:<message_id>,<”topicname”>,<message_len>,<message><CR>

Parameter

<message_id> Message ID

<"topicname">	Topic name, topic name must be marked by quotation marks.
<message_len>	The length of the data received
<message>	Data received

Example

```
+MQTTSUB:"neoway02",5,12345        Receive messages Published by the topic subscribed to.
```

18.12 AT+MQTTSTATE – Querying MQTT Connection Status:

To query the status of the MQTT connection.

Format

Type	Command	Response
Query	AT+MQTTSTATE?<CR>	<CR><LF>+MQTTSTATE: <state><CR><LF> <CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT+MQTTSTATE?
+MQTTSTATE: 1                      The device is connected to the MQTT server.
OK
AT+MQTTSTATE?
                                   Query the MQTT connection status.
+MQTTSTATE: 0
                                   The device is disconnected to the MQTT server.
OK
```

19 Other AT Commands

19.1 AT+CIPGSMLOC–Obtaining the Location of the Module

To obtain the location information of the module.

The obtained location information is the GPS coordinates.

If the server does not reply in 10 seconds after the request is submitted successfully, the module returns +CIPGSMLOC: TIMEOUT. TIMEOUT.

The current coordinates of latitude and longitude are valid and precision is reserved (0.0 by default).

Format

Type	Command	Response
Execute	AT+CIPGSMLOC<CR> AT+CIPGSMLOC=<n><CR>	<CR><LF>+CIPGSMLOC: <fail_string><CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: {<result_string>} <CR><LF>+CIPGSMLOC: OK<CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: <code><CR><LF> <CR><LF>+CIPGSMLOC: FAIL<CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: TIMEOUT<CR><LF>

Parameter

<n>	request selection 0: Close location request (default) 1: multi-BS positioning request (reserved)
<module_model>	Failure string CONTACT FAIL LINK FAIL

	LINK NOT FREE <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: {<result_string>} <CR><LF>+CIPGSMLOC: OK<CR><LF>
<fail_string>	Firmware version, 4 bytes in ASCII format.
<result_string >	string including longitude and latitude <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: <code> <CR><LF>+CIPGSMLOC: FAIL <CR><LF>
<code >	return code after request is submitted successfully but not longitude or latitude information is returned. 401: No right 400: error occurs during request parsing. 404: legal request, but the queried BS is not included. 408: parsing times out. 500: internal error of server

Example

```
AT+CIPGSMLOC
OK
                                                    Single-BS positioning request is executed
+CIPGSMLOC:{"location":{"lat":22.6824031166         successfully in State Grid mode.
13813,"lng":113.99042272056248},"accuracy":         The module reports location coordinates.
0.0}
+CIPGSMLOC: OK

AT+CIPGSMLOC
+CIPGSMLOC: CONTACT FAIL                            The server domain name fails to be translated.
The server domain name fails to be translated.



AT+CIPGSMLOC                                        The connection to the server fails to be set
+CIPGSMLOC: LINK FAIL                               up. Query the current location of the user



AT+CIPGSMLOC
OK                                                  queried BS is not included.
+CIPGSMLOC: 404
+CIPGSMLOC: FAIL



AT+CIPGSMLOC=0                                      End the requset
OK
```

```
AT+CIPGSMLOC=1
OK

+CIPGSMLOC:{"location":{"lat":22.689646628671216,
"lng":113.98586121790129},"accuracy":0.0}
+CIPGSMLOC: OK
```

Request multi-BS positioning in standard mode is executed successfully. The module reports its location coordinates.

19.2 AT+UPDATETIME–Updating Time to Network

To update the module time to the network time

Set up a PPP link (AT+XIIC=1) before sending this command. Send AT+CCLK? to query whether RTC is synchronized to the current network time after this command is sent successfully.

The settings by this command are not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+UPDATETIME=<mode>[,<serv_ip>, <time>[[,<TZ>][,<DST>]]]<CR>	<mode>=0 <CR><LF>+UPDATETIME: Last Update Time yyyy-mm-dd,hh: mm: ss<CR><LF> <CR><LF>OK<CR><LF> <mode>=1 <CR><LF>OK<CR><LF> <CR><LF>+UPDATETIME: <result code><CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>Time Updating,Please Wait... <CR><LF> <CR><LF>+UPDATETIME: <result code><CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>Time Updating,Please Wait... <CR><LF> <CR><LF>+UPDATETIME: Update To yyyy-mm-dd,hh: mm: ss <CR><LF>
Query	AT+UPDATETIME? <CR>	<CR><LF>+UPDATETIME: <serv_ip>,<time>,<TZ>,<DST><CR><LF>

		<CR><LF>OK<CR><LF>
Test	AT+UPDATETIME=?<CR>	<CR><LF>+UPDATETIME: (value range of <mode>),,(value range of <time>),,(value range of <DST>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode>	0: query mode. Query when the time was updated to the network time last time. 1: setting mode. Synchronize the time to the network time.
<serv_ip>	IP address of the time server, in form of xx.xx.xx.xx or domain name The following time servers support time update: time.windows.com, time.nist.gov, etc.
<time>	Timeout period, ranging from 1 to 30, unit: second
<TZ>	Time zone, in format of E/W+digits; E8 by default E: east time zone, 0 to 13 W: west time zone, 0 to 12 0: zero time zone
<DST>	Daylight Saving Time 1: select DST auto-adjustment 0: not select (by default)
<result code>	No PPP Link Time Updating,Please Wait... • Time Out • Time Data Is Null • Send Request Fail • Domain Name Invalid • Socket Error

Example

```
AT+UPDATETIME=0                                  Query when the time was updated last time
+UPDATETIME: Last Update Time 2014-03-31,
11:10:26
OK
AT+UPDATETIME=0                                  Query when the time was updated last time.
+UPDATETIME: Last Update Time 0000-00-00,         The time was not updated.
00:00:00
OK
AT+UPDATETIME=1,210.72.145.44,10                 Set up a PPP link
+UPDATETIME: No PPP Link
AT+UPDATETIME=1,120.25.108.11,10,"E8",0          Time out
OK

+UPDATETIME: Time Out
AT+UPDATETIME=1,128.138.141.172,10,"E8",0        Update the time to that of the server
OK                                               128.138.141.172.
```

```
                                                    Timeout period is 10 seconds.
Time Updating,Please Wait...                        East time zone 8 by default
                                                    No DST
+UPDATETIME: Update To 2014-03-31,11:32:55          Time is updated successfully.
AT+UPDATETIME?                                      Query the IP address of the server to which
+UPDATETIME: 128.138.141.172,10,"E8",0              the time is updated and the timeout period,
                                                    time zone, and DTS.
OK
AT+UPDATETIME=?                                     Query the value range of parameters.
+UPDATETIME: (0-1),,(1-30),,(0-1)
OK
AT+UPDATETIME=1,time.windows.com,10,"W12",1         Socket error.
OK                                                  The possible reason might be network
                                                    congestion.
+UPDATETIME: Socket Error
```

19.3 AT$MYPOWEROFF – Powering Off the Module

To power off the module.

Format

Type	Command	Response
Execute	AT$MYPOWEROFF<CR>	<CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT$MYPOWEROFF                                       Power off the module.
OK
```

19.4 AT$MYGMR – Obtaining the Version Information of the Module

To obtain the version information of the module firmware and hardware.

Format

Type	Command	Response
Execute	AT$MYGMR<CR>	<CR><LF><module_manufacture> <CR><LF><module_model> <CR><LF><firmware_version> <CR><LF><firmware_release_date> <CR><LF><module_hardware_version> <CR><LF><module_hardware_release_date> <CR><LF>OK<CR><LF>

Parameter

<module_manufacture>	Manufacture code, 4 bytes in ASCII format.
<module_model>	Module mode, 8 bytes in ASCII format.
<firmware_version>	Firmware version, 4 bytes in ASCII format.
<firmware_release_date>	Firmware release date, DDMMYY, 6 bytes in ASCII format.
<module_hardware_version>	Hardware version, 4 bytes in ASCII format.
<module_hardware_release date>	Hardware release date

Example

```
AT$MYGMR
NEO6
N725
V003
100815
V1.0
260815
OK
```

19.5 AT$MYCCID – Obtaining the CCID of the SIM Card

To obtain the circuit card identifier (CCID) of the SIM card.

Format

Type	Command	Response
Execute	AT$MYCCID<CR>	<CR><LF>$MYCCID: <SIM_CCID> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<reversion> software version

Example

```
AT$MYCCID
$MYCCID: "89860112965403839541"
OK
AT$MYCCID                          Query the CCID of the SIM card.
ERROR                              Error is returned because no SIM card is installed.
```

19.6 AT$MYTYPE – Querying the Type of the Module

To query the software version.

Format

Type	Command	Response
Execute	AT$MYTYPE?<CR>	<CR><LF>$MYTYPE: <mode>,<network_type>,<extended_feature> <CR><LF>OK<CR><LF>

Parameter

<mode> Work mode. Extensible bytes are indicated by bit.
1 at bit 0: supporting transparent mode
1 at bit 1: supporting non-transparent mode

<network_type> Network type. Extensible bytes are indicated by bit.
1 at bit 0: GSM network supports GPRS
1 at bit 1: WCDMA network
1 at bit 2: TD-SCDMA network
1 at bit 3: CDMA 2000
1 at bit 4: CDMA EVDO
1 at bit 5: LTE
1 at bit 6: PSTN
1 at bit 7: extending one byte

<extended_feature> Extended feature. Extensible bytes are indicated by bit.
1 at bit 0: supporting GPS positioning function
1 at bit 1: supporting BeiDou positioning function
1 at bit 7: extending one byte

Example

```
AT$MYTYPE?
$MYTYPE: 03,23,00
OK
```

19.7 AT$MYMODEM (Standards for Canton) – Querying the Type of the Module

To query the type of the module.

Format

Type	Command	Response
Execute	AT$MYMODEM?<CR>	<CR><LF>$MYMODEM: <mode>,<network_type> <CR><LF>OK<CR><LF>

Parameter

<mode>	Work mode. Extensible bytes are indicated by bit. 1 at bit 0: supporting transparent mode 1 at bit 1: supporting non-transparent mode
<network_type>	Network type. Extensible bytes are indicated by bit, 16-bit 1 at bit 0: GPRS 1 at bit 1: CDMA network 1 at bit 2: TD-SCDMA network 1 at bit 3: WCDMA 1 at bit 4: CDMA EVDO 1 at bit 5: TDD-LTE 1 at bit 6: FDD-LTE Bits 8 to 15: Reserved

Example

```
AT$MYMODEM?
$MYMODEM: 03,0069
OK
```

19.8 AT$MYLACID – Querying Current LAC and CELL_ID

To query current LAC and CELL_ID.

This command is used only for modules that support GSM900/1800.

Format

Type	Command	Response
Execute	AT$MYLACID<CR>	<CR><LF>$MYLACID: <LAC>,<CELL_ID> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<LAC>	Location Area Code, hexadecimal
<CELL_ID>	CELL_ID, hexadecimal

Example

```
AT$MYLACID                                  Query the current location of the user.
$MYLACID: 90F3,07828C01
OK
```

19.9 AT$MYSYSINFO – Querying or Locking the Network Mode

To query or lock the network mode.

When the network mode is set to any non-AUTO mode, the module will automatically check the network status.

- If the module registers a network, it will determine whether the network is the mode set by the command. The module will switch to the set network if the registered network is not the mode set by the command.
- If the module fails to register any network within interval set by $MYNETAUTO, the module will automatically switch to the AUTO mode.

Format

Type	Command	Response
Set	AT$MYSYSINFO=<SysMode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT$MYSYSINFO<CR	<CR><LF>$MYSYSINFO: <Sys_Mode>,<mnc> <CR><LF>OK<CR><LF>
Test	AT$MYSYSINFO=?<CR>	<CR><LF>$MYSYSINFO: value range of <SysMode> <CR><LF>OK<CR><LF>

Parameter

<Sys_Mode> Network mode
0: No service
2: 2G (including GSM, EDGE, CDMA)
3: 3G (including WCDMA, TD-SCDMA, EVDO)
4: 4G (including FDD-LTE, TDD-LTE)
5: 2G+3G (including GSM,EDGE,CDMA,WCDMA,TD-SCDMA,EVDO)
6: 2G+4G (including GSM,EDGE,CDMA,FDD-LTE,TDD-LTE)
7: 3G+4G (including WCDMA,TD-SCDMA,EVDO,FDD-LTE,TDD-LTE)

<mnc> Network carrier code
00: Fail to register
01: China Mobile
02: China Unicom
03: China Telecom
04: Unknown

Example

```
AT$MYSYSINFO                          The module registered to the 4G network of
$MYSYSINFO: 4,01                      China Mobile.
OK
AT$MYSYSINFO=1                        Set network mode to auto.
OK
AT$MYSYSINFO=?                        Query the value range of the parameter.
$MYSYSINFO: 1-7
OK
```


19.10 AT$MYSYSINFOURC – Enabling/Disabling Network Mode Report

To enable/disable the network mode report of the module.

Format

Type	Command	Response
Set	AT$MYSYSINFOURC=<ONOFF><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYSYSINFOURC?<CR>	<CR><LF>$MYSYSINFOURC: <ONOFF> <CR><LF>OK<CR><LF>

Parameter

<ONOFF>	integer type 0: Disable network status report (default) 1: Enable network status report

Example

```
AT$MYSYSINFOURC=1
OK
```

19.11 AT$MYURCSYSINFO – Enabling/Disabling Network Mode Report

To enable/disable the network mode report of the module.

Format

Type	Command	Response
Execute	AT$MYURCSYSINFO=<ONOFF><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Query	AT$MYURCSYSINFO?<CR>	<CR><LF>$MYURCSYSINFO: <ONOFF> <CR><LF>OK<CR><LF>

Parameter

<ONOFF>	integer type 0: Disable network status report (default) 1: Enable network status report

Example

```
AT$MYURCSYSINFO=1
OK
```

19.12 $MYURCSYSINFO – Notifying Current Network Mode

To notify the current network mode.

Format

Type	Command
URC	<CR><LF>$MYURCSYSINFO: <mnc>,<SysMode><CR><LF>

Parameter

<SysMode>	Network mode 0: No service 2: 2G (including GSM, EDGE, CDMA) 3: 3G (including WCDMA, TD-SCDMA, EVDO) 4: 4G (including FDD-LTE, TDD-LTE)
<mnc>	Network carrier code 00: Fail to register 01: China Mobile 02: China Unicom 03: China Telecom

Example

```
AT$MYSYSINFOURC=1                       The network mode of that the module registered on is a
OK                                      3G network of China Mobile.

$MYURCSYSINFO: 01,3
```

19.13 AT$MYNETINFO – Setting Network Mode Choices

To set network mode choices.

When the network mode is set to any non-AUTO mode, the module will automatically check the network status.

- If the module registers a network, it will determine whether the network is the mode set by the command. The module will switch to the set network if the registered network is not the mode set by the command.
- If the module fails to register any network within interval set by $MYNETAUTO, the module will automatically switch to the AUTO mode.

Format

Type	Command	Response
Set	AT$MYNETINFO=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR:<err><CR><LF>
Query	AT$MYNETINFO?<CR>	<CR><LF>$MYNETINFO: <mode> <CR><LF>OK<CR><LF>
Test	AT$MYNETINFO=?<CR>	<CR><LF>$MYNETINFO: (list of supported<mode>s) <CR><LF>OK<CR><LF>

Parameter

<mode> Network mode
1: AUTO
2: 2G (including GSM, EDGE, CDMA)
3: 3G (including WCDMA, TD-SCDMA, EVDO)
4: 4G (including FDD-LTE, TDD-LTE)
5: 2G+3G (including GSM, EDGE, CDMA, WCDMA, TD-SCDMA, EVDO)
6: 2G+4G (including GSM, EDGE, CDMA, FDD-LTE, TDD-LTE)
7: 3G+4G (including WCDMA, TD-SCDMA, EVDO, FDD-LTE, TDD-LTE)

Example

```
AT$MYNETINFO=4
OK
AT$MYNETINFO?
$MYNETINFO: 4
OK
AT$MYNETINFO=?
$MYNETINFO: 1-7
OK
```

19.14 AT$MYNETAUTO – Enabling/Disabling the Default AUTO Network Modes during Startup

To query the software version

When the network mode is set to non-auto mode through this command, the module will start the function of automatic detection of network status, that is, real-time monitoring whether the module can register the network. If the network is not registered within three minutes (the default value is 3, which can be modified through the $MYNETAUTO command), it will be forced to switch to AUTO mode.

Format

Type	Command	Response
Set	AT$MYNETAUTO=<ONOFF>[,<CYCLE>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYNETAUTO?<CR>	<CR><LF>$MYNETAUTO: <ONOFF> <CR><LF>OK<CR><LF>

Parameter

<ONOFF>	0: Disable the default AUTO network modes during startup 1: Enable the default AUTO network modes during startup
<CYCLE>	the period that the module restores to auto mode after the network mode is locked and the module failed to register with the network. This value ranges 3 to 1440 minutes, 3 minutes by default.

Example

```
AT$MYNETAUTO=1
OK
```

```
AT$MYNETAUTO?
$MYNETAUTO: 1,3
OK
AT$MYNETAUTO=?
$MYNETAUTO: (0-1),(3-1440)
OK
```

19.15 AT+XISP – Selecting Internal or External Protocol Stack

To select internal or external protocol stack.

To use external protocol stack, send **AT+XISP=1** to set before using.

Format

Type	Command	Response
Execute	AT+XISP=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+XISP?<CR>	<CR><LF>+XISP: <n> <CR><LF>OK<CR><LF>

Parameter

<n>	0: Select internal TCP/IP protocol stack (default) 1: Select external protocol stack

Example

```
AT+XISP=0                   Specify internal protocol stack.
OK
AT+XISP?                    Query which protocol stack is used.
+XISP: 0                    Internal protocol stack is used.
OK
AT+XISP=?
+XISP: (0,1)
OK
```

19.16 AT+MYDATAONLY–Disabling CS Services

To disable voice call and SMS message services.

The setting of this command takes effect after the module is restarted.

The setting is invalid on LTE networks.

Format

Type	Command	Response
Execute	AT+MYDATAONLY=<onoff><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MYDATAONLY?<CR>	<CR><LF>+MYDATAONLY: <onoff> <CR><LF>OK<CR><LF>

Parameter

<onoff>	0: On 1: OFF

Example

```
AT+MYDATAONLY=0                          Enable voice call and SMS message services
OK
AT+MYDATAONLY=1                          Disable voice call and SMS message
OK                                       services
AT+MYDATAONLY?                           Query the status of voice call service
+MYDATAONLY: 1
OK
```

A Reference Process of AT Command Programming

A.1 Content of PDU SMS Messages

<PDU> SMS message sending format:

1>: 0891

08: indicates the length of the SMSC address information

91: indicates the format of the SMSC address

2>: Inversion of every two bits (add F if the bits are not sufficient) in SMSC number, fixed. For example, China Unicom 8613010888500 should be 683108705505F0 here.

3>: 0100

01: Indicates basic parameters

00: indicates message baseline value

4>: Convert the receiving number into hexadecimal. For example, the number length is 11 bits and then the hexadecimal length should be 0B.

5>: 81 (Receiving mode) there are multiple receiving modes. 81 indicates that the receiving mode is unknown.

6>: Inversion of every two bits (add F if the bits are not sufficient) in the recipient number. For example, 13421839693 should be 3124819396F3 after conversion.

7>: 0008

8>: The hexadecimal length of the SMS message content. For example, the UCS2 code of hello is 00080A00680065006C006C006F, that is 10 bits and the hexadecimal length is 0A.

9>: Message content, for example, the USC2 code of hello is 00080A00680065006C006C006F.

One PDU message contains the above 9 parts and the parameter values are determined by the actual situation.

If the SMSC address length is 0, replace 08 with 00 and the SMSC type and address fields must be omitted.

The following is an example of the PDU message whose SMSC address length is not 0:

0891683110808805F001000B813124819396F300080A00680065006C006C006F

Wherein,

0891

683108705505F0: SMSC number of China Unicom

0100

0B: the length of the recipient number

81: Receiving mode

3124819396F3: The number of recipient

0008

0A: The length of the content

00680065006C006C006F: SMS message content

Message content: hello

The SMS message content starts from 0100, so the value of LENGTH in **AT+CMGS=LENGTH** is **23**.

The following is an example of the PDU message whose SMSC address length is **0**:

0001000B813124819396F300080A00680065006C006C006F

Wherein,

00: SMSC address information length

SMSC number is not needed.

0100

0B: the length of the recipient number

81: Receiving mode

3124819396F3: The number of recipient

0008

0A: The length of the content

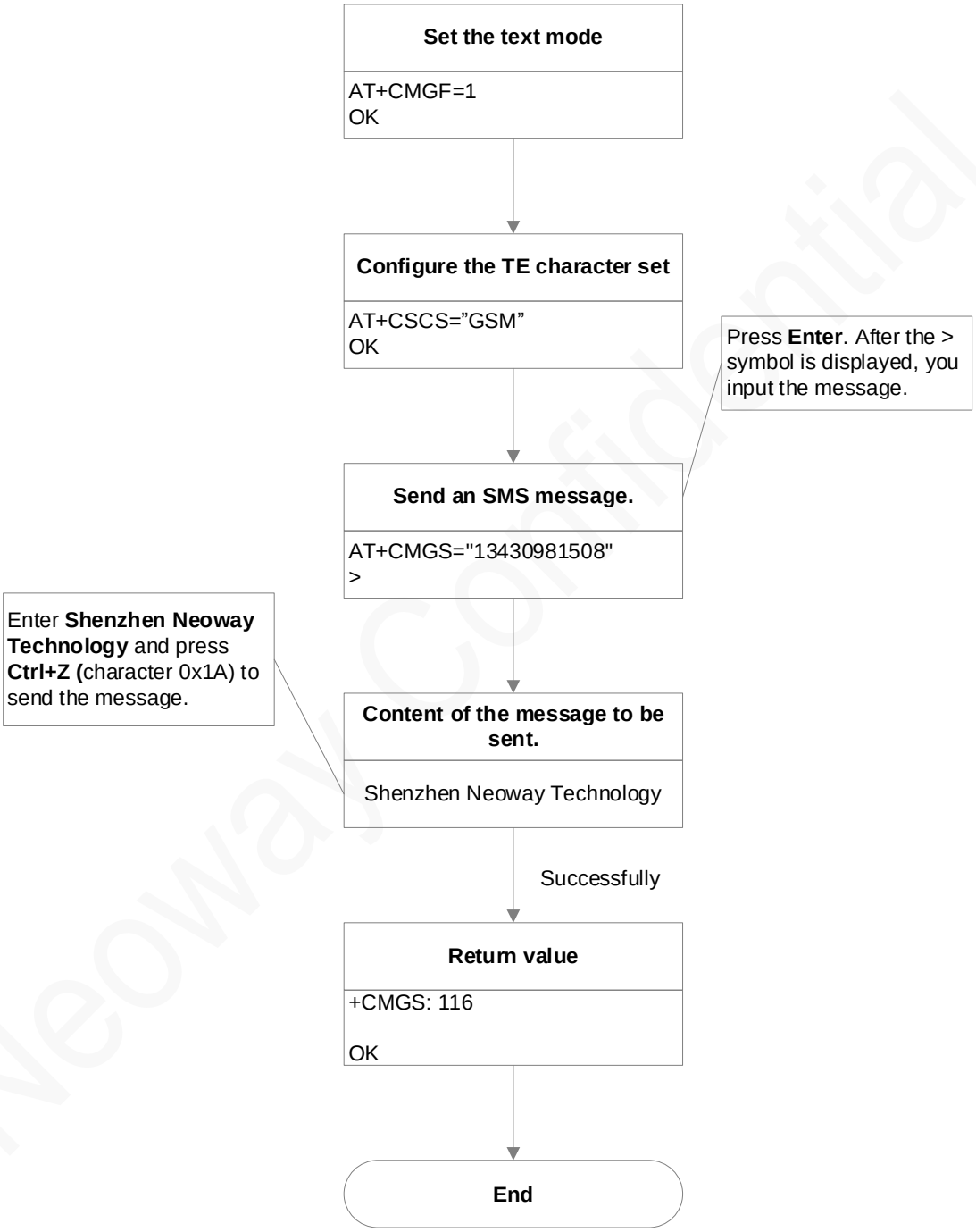
00680065006C006C006F: SMS message content

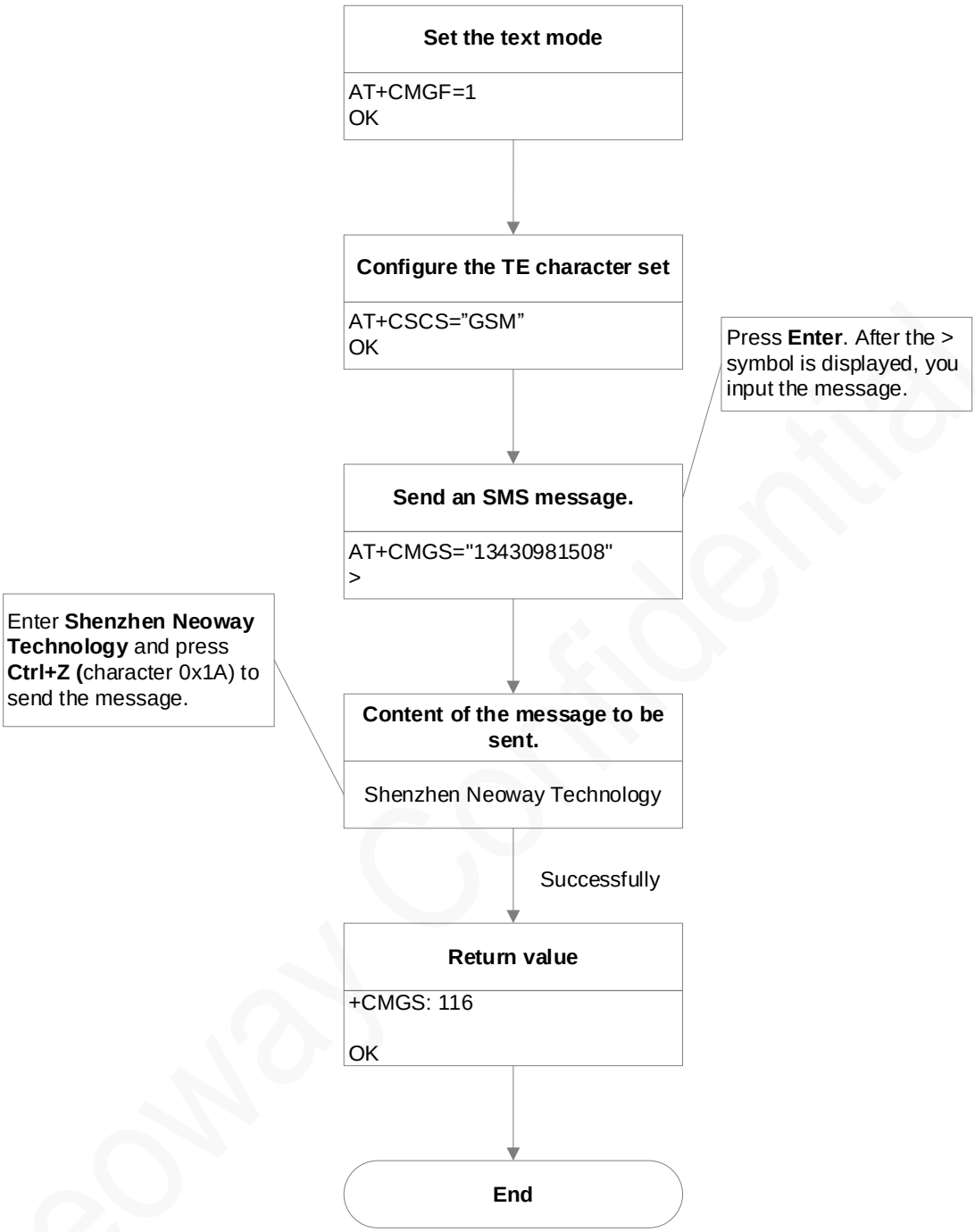
SMS message content: hello

The SMS message content starts from 0100, so the value of LENGTH in **AT+CMGS=LENGTH** is **23**.

A.2 Flowchart of Sending Text SMS Messages (Through UART)

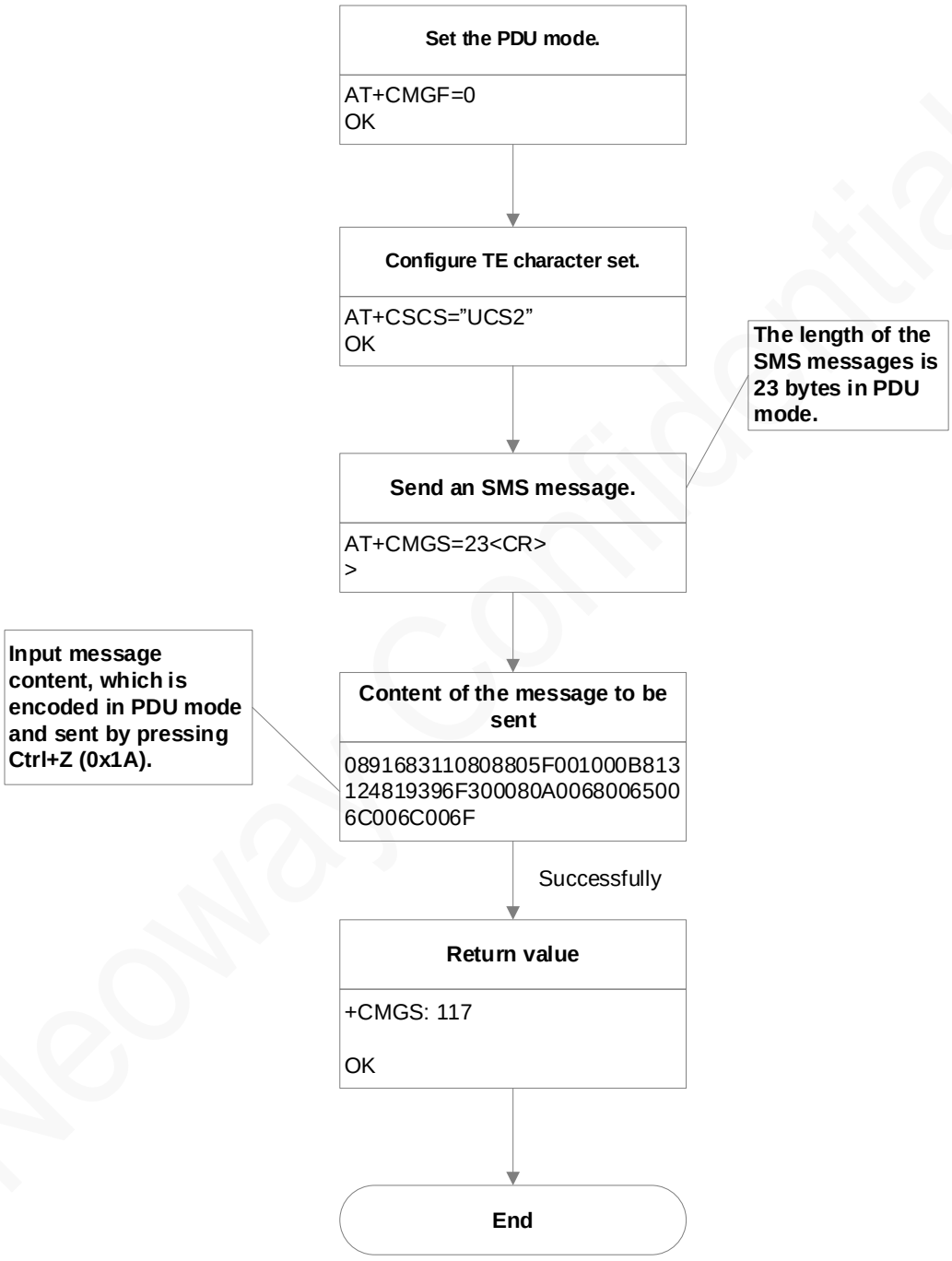
Figure A-1 Flowchart of sending text format SMS messages

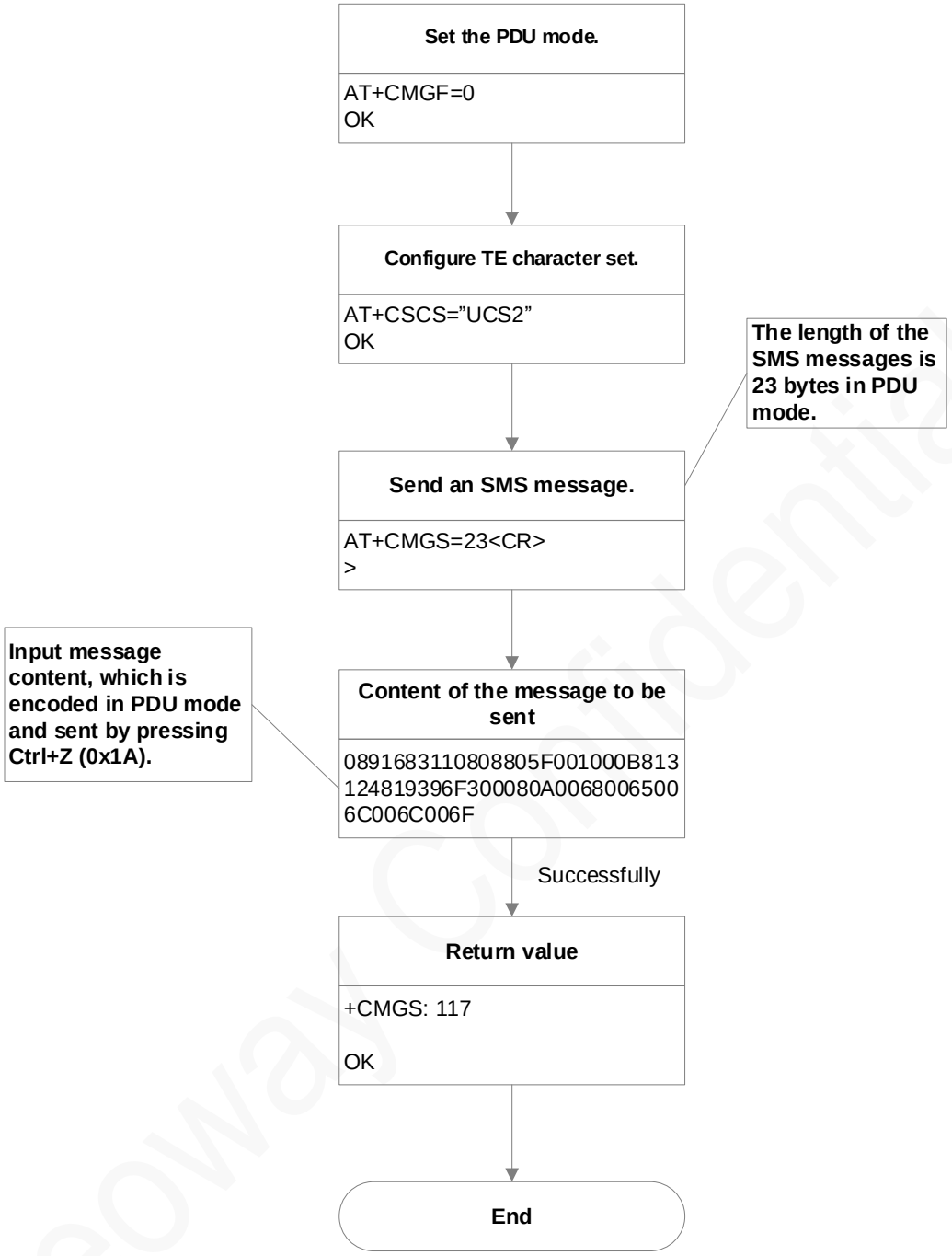




A.3 Flowchart of Sending PDU SMS Messages (Through UART)

Figure A-2 Flowchart of Sending PDU SMS messages





B Support Band List

C/G/W Band Name	Bit	HEX Band Mask	LTE Band Name	Bit	HEX Band Mask
CDMA BC0	1&2	3	EUTRAN_BAND1	1	1
CDMA BC1	3	4	EUTRAN_BAND2	2	2
GSM 850	20	80000	EUTRAN_BAND3	3	4
EGSM 900	9	100	EUTRAN_BAND4	4	8
PGSM 900	10	200	EUTRAN_BAND5	5	10
DCS 1800	8	80	EUTRAN_BAND6	6	20
PCS 1900	22	200000	EUTRAN_BAND7	7	40
GSM Railway 900	21	100000	EUTRAN_BAND8	8	80
GSM 450	17	10000	EUTRAN_BAND9	9	100
GSM 480	18	20000	EUTRAN_BAND10	10	200
GSM 750	19	40000	EUTRAN_BAND11	11	400
IMT 2000(W B1)	23	400000	EUTRAN_BAND12	12	800
PCS 1800(W B2)	24	800000	EUTRAN_BAND13	13	1000
III 1700(W B3)	25	1000000	EUTRAN_BAND14	14	2000
IV 1700 (W B4)	26	2000000	EUTRAN_BAND17	17	10000
V 850 (W B5)	27	4000000	EUTRAN_BAND33	33	100000000
VI 800 (W B6)	28	8000000	EUTRAN_BAND34	34	200000000
VII 2600(W B7)	49	1000000000000	EUTRAN_BAND35	35	400000000
VIII 900(W B8)	50	2000000000000	EUTRAN_BAND36	36	800000000
IX 1700 (W B9)	51	4000000000000	EUTRAN_BAND37	37	1000000000
XIX 800 (W B10)	61	1000000000000000	EUTRAN_BAND38	38	2000000000
XI 950 (W B11)	62	2000000000000000	EUTRAN_BAND39	39	4000000000
TDS Band Name	Bit	HEX Band Mask	EUTRAN_BAND40	40	8000000000
B34	1	1	EUTRAN_BAND41	41	10000000000
B39	6	20	EUTRAN_BAND42	42	20000000000
B40	5	10	EUTRAN_BAND43	43	40000000000
NV Define			EUTRAN_BAND44	44	80000000000

Band Config	CGW	1877	EUTRAN_BAND65	51	4000000000000
Band Config	LTE	6828	EUTRAN_BAND66	52	8000000000000
Band Config	TDS	22605	EUTRAN_BAND71	60	800000000000000
Band PREF	CGW	441/946/2954	EUTRAN_BAND252	61	1000000000000000
Band PREF	LTE	65633	EUTRAN_BAND253	62	2000000000000000
Band PREF	TDS	EFS:tds_bandpref	EUTRAN_BAND255	64	8000000000000000

C Message Service Failure Result Code +CMS ERROR

Final result code +CMS ERROR: <err> indicates an error related to mobile equipment or network. The operation is similar to ERROR final result code. None of the following commands in the same command line is executed. Neither ERROR nor OK final result code shall be returned.

Defined Values

<err> values used by common messaging commands:

0...127	3GPP TS 24.011 [6] clause E.2 values
128...255	3GPP TS 23.040 [3] clause 9.2.3.22 values.
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	(U)SIM not inserted
311	(U)SIM PIN required
312	PH-(U)SIM PIN required
313	(U)SIM failure
314	(U)SIM busy
315	(U)SIM wrong
316	(U)SIM PUK required
317	(U)SIM PIN2 required
318	(U)SIM 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
...511	other values in range 256...511 are reserved
512...	manufacturer specific

D Mobile Termination Error Result Code +CME ERROR

Final result code +CME ERROR: <err> indicates an error related to mobile equipment or network. The
operation is similar to ERROR result code. None of the following commands in the same command line is
executed. Neither ERROR nor OK result code shall be returned.

<err> values are mostly used by common message commands. The following table lists most of general
and GRPS related ERROR codes. For some GSM protocol failure cause described in GSM specifications,
the corresponding ERROR codes are not included.

General errors

Numeric	Text
0	phone failure
1	no connection to phone
2	phone-adaptor 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 (See NOTE 1)
11	SIM PIN required
12	SIM PUK required
13	SIM failure (See NOTE 1)
14	SIM busy (See NOTE 1)
15	SIM wrong (See NOTE 1)
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 calls 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
100 unknown