neoway

Get connected Get smart

N723-EA

AT Commands Manual

Issue 1.0 Date 2022-04-29

Notice

This document provides guide for users to use N723-EA.

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

THIS GUIDE PROVIDES INSTRUCTIONS FOR CUSTOMERS TO DESIGN THEIR APPLICATIONS. PLEASE FOLLOW THE RULES AND PARAMETERS IN THIS GUIDE TO DESIGN AND COMMISSION. NEOWAY WILL NOT TAKE ANY RESPONSIBILITY OF BODILY HURT OR ASSET LOSS CAUSED BY IMPROPER OPERATIONS.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE DUE TO PRODUCT VERSION UPDATE OR OTHER REASONS.

EVERY EFFORT HAS BEEN MADE IN PREPARATION OF THIS DOCUMENT TO ENSURE ACCURACY OF THE CONTENTS, BUT ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT DO NOT CONSTITUTE A WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

Neoway provides customers complete technical support. If you have any question, please contact your account manager or email to the following email addresses:

Sales@neoway.com

Support@neoway.com

Website: http://www.neoway.com

Contents

About This Document

Scope

This document is applicable to N723-EA.

Audience

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

Change History

Issue	Date	Change	Changed by
1.0	2022-03	Initial draft	Songhaiqing LiZhengfu An Fangfang Zhang Hao Zhao Guiqiang An Fangfang Li Chuyu Yin Jianjun Xue Jianyun

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_N723-EA_Datasheet

Neoway_N723-EA_Product_Specifications

Neoway_N723-EA_HW_User_Guide

Neoway_N723-EA_EVK_User_Guide

1 Boot LOG Instruction

The UART outputs **+PBREADY** after the phonebook is available.

If the module is booted in automatic baudrate detection mode, send **AT** 10 seconds after the module is powered up to check if the AT function is initialized. The UART responds with **OK** if AT is initialized and outputs **+PBREADY** after the phonebook is available.

Network indicator status

- Off: No network found.
- On: the module finds a network and the PDP context is not activated.
- Blinks (on for 0.2 seconds and off for 1.8 seconds): the PDP context is activated successfully.

2 AT Syntax

2.1 Symbols

- <CR>: carriage return character
- <LF>: linefeed character
- <..>: parameter name, the angle brackets do not appear in the command line.
- [..]: optional parameter, the square brackets do not appear in the command line.
- : space

2.2 Description

Prefix

AT or at

Command Line

Standard commands, in compliance with 3GPP 27007, 27005and ITU-T Recommendation V.250.

Extended commands, defined by Neoway

Joint Mark

+ or $, used between the prefix and a command line

Termination Character

<CR>, i.e. 0x0D

Response Syntax

<CR><LF>response<CR><LF>

Response can be one or multiple messages.

Result Syntax

<CR><LF>OK<CR><LF> indicates that a command is executed successfully.

<CR><LF>ERROR<CR><LF> indicates that a command fails to be executed.

For the error codes, see Appendix A.

2.3 Command Types

Type	Syntax	Response	Function
Set	AT+CMD=<VALUE><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>	Store a value or values for later use
Execute	AT+CMD[=<VALUE>]<CR>	[<CR><LF>response] <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>	Invoke a function of the module.
Test	AT+CMD=?<CR>	[<CR><LF>response] <CR><LF>OK<CR><LF>	Determine the range of parameter values or parameter lengths that are supported
Query	AT+CMD?<CR>	[<CR><LF>response] <CR><LF>OK<CR><LF>	Determine the current value or values stored
Unsolicited result code	<CR><LF>+CMD: <VALUE><CR><LF>	N/A	Report the status change and data receiving
Remarks	Symbols are not displayed in AT commands. All commands comply with the rules in this chapter.		

2.4 Command Response Time-Out

Every command issued to the Neoway module returns a result response and 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+CLCK	15
3	AT+CMGR	10
4	AT+CMGL	15
5	AT+CMGS	60
6	AT+CMGW	6
7	AT+CMSS	60
8	AT+CMGD	20
9	AT$MYNETACT	120
10	AT$MYNETCON	5
11	AT$MYNETOPEN	30
12	AT$MYNETWRITE	30
13	AT$MYNETCREATE	30
14	AT$UDPSEND(S)	30
15	AT$MYFTPOPEN	Customized
16	AT$MYFTPCLOSE	5
17	AT$MYFTPSIZE	Customized
18	AT$MYFTPGET	Customized
19	AT$MYFTPPUT	Customized
20	AT+HTTPSETUP	30
21	AT+HTTPACTION	10
22	AT+HTTPSSETUP	30
23	AT+HTTPSACTION	10
24	AT+CIPGSMLOC	60
25	AT+UPDATETIME	60
26	AT$MYBCCH	5
27	AT$MYCGED	3
28	AT$MYSYSINFO	3
29	AT$MYNETINFO	3
30	AT^SYSCONFIG	3
31	AT$MYTIMEUPDATE	10
32	AT+NRSP	3
33	AT$MYANTAUX	3
34	AT+NETMSG	3

35	AT+NETSEL	3
36	AT+NETDMSG	3
37	AT+NEOFTPFOTA	120
38	AT+MYDATAONLY	10
39	AT+BANDLOCK	3

3 General Commands

3.1 ATI – Querying Manufacturer Information

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

Format

Type	Command	Response
Execute	ATI<CR>	<CR><LF><module_info> <CR><LF>OK<CR><LF>

Parameter

<module_info> module manufacturer information

Example

```
ATI
NEOWAY                                  Manufacturer
N723                                    Module model
REVISION V001                           Version
OK
```

3.2 AT+GMR – Querying Software Version

To query the software version

Format

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

Parameter

<reversion>	software version

Example

```
AT+GMR
+GMR: N723-A12-EA-DL-001                    Query the software version
OK
```

3.3 AT+CGSN – Querying IMEI

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

Format

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

Parameter

<IMEI>	International Mobile Equipment Identity, a character string of 15 digits. The return value of this command is a 8-bit ESN on a 3GPP2 network.

Example

```
AT+GSN
+GSN: "860998021170687"
OK
```

3.4 AT+CGMM – Querying Module Model

To query the module model

Format

Type	Command	Response

Execute	AT+CGMM<CR>	<CR><LF>+CGMM:<model> <CR><LF>OK<CR><LF>

Parameter

<model> module model

Example

```
AT+CGMM                                          Query the product model.
+CGMM: N723
OK
```

3.5 AT+GMM – Querying Module Model

To query the the module model.

Format

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

Parameter

<model> Module model.

Example

```
AT+GMM
+GMM: N723                                       Query the product model.
OK
```

3.6 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
N723
V001
030920
V1.0
090916
OK
```

3.7 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
```

3.8 AT+CCLK – Clock

To set and query the real-time clock.

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>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: "08/07/01,14:54:0"                      Query the setting of the real-time clock.
OK
AT+CCLK=14/07/02,10:48:50
ERROR                                          Incorrect command syntax.
```

3.9 AT+SETTZ – Setting Clock Offset Time

To set the local clock offset time. Display the setting values through AT+CCLK?.

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

Format

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

Parameter

<n> Offset time, 15 minutes as a unit, ranging from -96 to 96.

Example

```
AT+SETTZ?                                      Query the current offset time.
+SETTZ: +32
OK
AT+SETTZ=+32                                   Query current status.
OK
AT+CCLK?
```

```
+CCLK: "80/01/06,00:56:50+32"
OK
```

3.10 ATE1/ATE0 – Enabling & Disabling 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.

ATE equals to ATE1.

Format

Type	Command	Response
Set	ATE[<value>]<CR>	<CR><LF>OK<CR><LF>

Parameter

<value>	Switch of the terminal display function of the AT commands 0: off 1: on (default)

Example

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

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

3.11 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 off.

Format

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

Parameter

<onoff> integer type
0: disable automatic time zone update (default)
1: enable automatic time zone update

<sync> Timing success flag
0: no synchronized
1: synchronize

Example

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

4 UART Control

4.1 AT+IPR – Setting Baud Rate

To set the baud rate of the module.

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

- The default baud rate is 0, indicating automatic baud rate detection.
- When the module is in automatic baud rate detection mode, the baud rate can be 9600, 19200, 38400, 57600, and 115200 bit/s.

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>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=?                                        Query the range of the baud rate.
```

```
+IPR: 0, 300, 600, 1200, 2400, 4800, 9600,
19200, 38400, 57600, 115200, 230400, 460800,
921600
OK
AT+IPR=100                                          worng baud rate.
+IPR: wrong baudrate
```

4.2 AT+CMUX – Activating Multiplexing Mode

To activate multiplexing mode.

To enable multiplexing function, send AT+CMUX=0.

Only channel 1 supports Internet access through dial-up.

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: 9600 bit/s 2: 19200 bit/s 3: 38400 bit/s 4: 57600 bit/s 5: 115200 bit/s

	6: 230400 bit/s 7: 460800 bit/s 8: 921600 bit/s
<N1>	Maximum frame size. Integer type, ranging from 1 to 32768. The default value is 31.
<T1>	Acknowledgement timer in unit of ten milliseconds, integer type, ranging from 1 to 255, where 10 (100ms) is the default value.
<N2>	Maximum number of re-transmissions, integer type. (Not supported)
<T2>	Response timer for the multiplexer control channel in units of ten milliseconds, integer type. (Not supported)
<T3>	Wakeup response timer, integer type. (Not supported)
<k>	Window size, integer type. (Not supported)

Example

```
AT+CMUX=0
OK                                                  Basic option

AT+CMUX=?
+CMUX:(0-1),(0-1),(1-8),(1-32768),(1-
255),(0-100),(2-255),(1-255),(1-7)                  Query the available range of parameters.
OK
```

5 SIM Card Operating

5.1 AT+CIMI – Querying IMSI

To query the international mobile subscriber identification (IMSI).

Format

Type	Command	Response
Execute	AT+CIMI<CR>	<CR><LF><IMSI> <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
460022201575463                          Obtain the IMSI number.
OK
AT+CIMI                                  Query the IMSI.
ERROR                                    No SIM card is installed.
```

5.2 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.
```

5.3 AT+CPIN – Entering PIN Code

To query the PIN status and enter PIN code.

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.

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>OK<CR><LF>

Parameter

<pin>,<newpin>	Character string type
<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?                                    Query whether PIN code is required.
+CPIN: READY                                No password is required.
OK
AT+CPIN="0000"                              Input correct PIN code.
ERROR
AT+CPIN="1234"
OK
```

5.4 AT+CLCK – Locking/Unlocking Module

To lock, unlock or interrogate module.

Format

Type	Command	Response
Execute	AT+CLCK=<fac>,<mode>[,<passwd>[,<class>]]<CR>	<CR><LF>+GMR: <reversion> <CR><LF>OK<CR><LF>
Test	AT+CLCK=?<CR>	<CR><LF>+CLCK: (list of supported <fac> values) <CR><LF>OK<CR><LF>

Parameter

<fac>	A pair of quotation marks are required for the value. "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": 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",    Query the network information related to the
"NT","NM","NS","NA","AB","AG","AC","FD","PN","    module.
PU","PP","PC")
OK
AT+CLCK="SC",1,"1234"                             Lock the current SIM card. "1234" is the PIN
OK                                                code of current SIM card.
AT+CLCK="SC",0,"1234"                             Unlock the current SIM card. "1234" is the PIN
OK                                                code of current SIM card.
AT+CLCK="SC",1,"2222"
+CME ERROR: 16                                    The PIN code is incorrect.
```

5.5 AT+CPWD – Modifying Password

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

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

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>OK<CR><LF>

Parameter

<fac>	A pair of quotation marks is required for the value. "P2":SIM PIN2 "OI": Outgoing international calls "AI": All incoming calls "IR": Incoming calls when roaming outside the home 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
<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.

5.6 AT+CNUM – Reading My Number

To read my number

Before reading "my number", use the AT+CPBS="ON" to set the storage of "my number"and use the AT+CPBW to store "my number".

Format

Type	Command	Response
Execute	AT+CNUM=<index>[,<number>,<type>,<text>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CNUM<CR>	<CR><LF>+CNUM:[<text>],<number>,<type><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<index>	index in phonebook, integer type
<number>	phone number in character string type
<type>	type of phone number
<text>	item name of phonebook

Example

```
AT+CNUM                                    Query my number.
+CNUM:"t","13651445684",129

OK
AT+CNUM=1,"13651445684",129,"t"            Store my number.
OK
```

5.7 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

<SIM_CCID>	Character string type, CCID of the SIM card.

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.
```

5.8 AT+SIMHOTSWAP – Setting SIM Card Hotswap

To enable hotswap of SIM card.

This function must be supported by hardware.

The setting takes effect immediately after the command is executed.

Format

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

Parameter

<onoff>	0: Disable (default)

1: Enable

Example

```
AT+SIMHOTSWAP=1                                 enable hotswap of SIM card.
OK
AT+SIMHOTSWAP=?                                 Query the range of the parameter.
+SIMHOTSWAP: (0-1)
OK
AT+SIMHOTSWAP?                                  Query the current hotswap status.
+SIMHOTSWAP: 1
OK
```

6 Network Server

6.1 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>OK<CR><LF>

Parameter

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

	signal	Rssi
0	<4 or 99	<-107 dBm 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: 1,99                                  Query the strength of the current signal.
OK
```

6.2 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> Or <CR><LF>+CREG:<stat>
Query	AT+CREG?<CR>	<CR><LF>+CREG: <n><stat>[,<lac>,<ci>[,<Act>]]<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CREG=?<CR>	<CR><LF>+CREG: (list of supported <n>s) <CR><LF>OK<CR><LF>
URC	When n=1, +CREG: <stat> When n=2, +CREG: <stat>[,[<lac>],[<ci>],[<AcT>]]	

Parameter

<n> Specifies whether to enable unsolicited result codes for network registration.
0: disable network registration unsolicited result code (default).
1: enable network registration unsolicited result code `+CREG: <stat>`.
2: enable network registration unsolicited result code with location information (Cell ID, Local ID) `+CREG: <stat>[,[<lac>],[<ci>],[<AcT>]]`

<stat> Network 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

<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+

Example

```
AT+CREG=1               Enable network registration unsolicited codes.
OK
AT+CREG?                Query the network registration status of the module
+CREG: 0,1              The module is registered with a home network
OK
AT+CREG=?               Query the value range of the network registration status parameter.
+CREG: (0-2)
OK
```

6.3 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>	When <n>=0,1,2 <CR><LF>+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>] <CR><LF>OK<CR><LF> When <n>=4 <CR><LF>+CEREG: <n>,<stat>[,[<tac>],[<ci>],[<AcT>][,[<Active-Time>],[<Periodic-TAU>]]]]<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CEREG=?<CR>	<CR><LF>+CEREG: (value range of<n>) <CR><LF>OK<CR><LF>

URC	When <n>=1 <CR><LF>+CEREG: <stat><CR><LF> When <n>=2 <CR><LF>+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>]]<CR><LF> When <n>=4 <CR><LF>+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]<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. 2: enable network registration and location information (Cell ID, Local ID) unsolicited result code
<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
<lac>	Two-byte tracking 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 HSUPA 7: E-UTRAN

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 status
+CEREG: (0-2)                      parameter.

OK
```


6.4 AT+COPS – Selecting and Registering 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>]]
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

It the network selection mode is automation, omit <AcT>.

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
OK                                                   46000
AT+COPS?
+COPS: 0,0,"CHINAUNICOM",7                           China Unicom
OK
AT+COPS?
+COPS: 0,2,"46001",7                                 If it is set to digital mode, then get the
OK                                                   number 46001.
AT+COPS?
+COPS: 0,0,"CHINA TELECOM",7                         China Telecom
OK
AT+COPS=?
+COPS: (2,"CHINA                                     Query the parameter ranges.
MOBILE","CMCC","46000"),(1,"CHN-CT","CT",
"46011"),(1,"CHN-
UNICOM","UNICOM","46001"),,(0,1,2,3,4),(0,1,2)

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

6.5 AT+CSQEX – Obtaining RX Singal Information

To obtain signal information.

There is a delay to update the RX signal inforamtion when the network changes.

Format

Type	Command	Response
Execute	AT+CSQEX<CR>	<CR><LF>+CSQEX: <mode>,<effective_signal>,<real_rssi>,<ber><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<index>	Network mode 0: GSM 1: WCDMA 2: LTE 3: TD-SCDMA 4: CDMA 5: HDR/EvDO 6: 1XLTE (mix mode of CDMA and LTE)
<effective_signal>	Effective RX signal strength, an absolute value, varying with network modes: GSM: RX power, ranging from 0 to 199, unit dBm 199 indicates null WCDMA: RSCP, ranging from 28 to 121, unit dBm LTE/1XLTE:RSRP, ranging from 40 to 140, unit dBm TD-SCDMA/CDMA/HDR: Ec/Io, ratio between the RX signal strength and the jamming level of neighbor cells, ranging from 0 to 32, unit dBm
<real_rssi>	Real RSSI value, an absolute value. The conversion between this parameter and the RSSI value in +CSQ is shown as follows: GSM/WCDMA/LTE/1XLTE: rssi = 57 - 0.5*real_rssi(round down) TD-SCDMA: rssi = 34 - real_rssi*31/382 (round) CDMA/HDR/EvDO: rssi =31*(125 - real_rssi)/50 (round down) If the converted value is beyond the range, reserve the threshold value. e.g. +CSQEX:0,87,90,99 wherein, real_rssi=90 rssi=57-0.5*90=12 The RSSI grade is 12. The effective signal strength is -87 dB. The real RSSI indicator is -90 dB.
<ber>	Bit error rate

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+CSQEX                                                    Query the RX signal information
+CSQEX: 0,87,90,99
OK
```

6.6 AT$MYBCCH – Locking the BCCH Channel

To lock the BCCH channel.

This command is used only for modules that support GSM900/1800.

Format

Type	Command	Response
Execute	AT$MYBCCH=<mode>[,<bcch1>],[<bcch2>],[<bcch3>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>$MYBCCH: +BA(num): <CR><LF><bcch1>,<mcc1>,<mnc1>,<lac1>,<cell-id1> <CR><LF><bcch2>,<mcc2>,<mnc2>,<lac2>,<cell-id2> <CR><LF><bcch3>,<mcc3>,<mnc3>,<lac3>,<cell-id3> ... <CR><LF>OK<CR><LF> Or <CR><LF>$MYBCCH: UNLOCKED <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYBCCH?<CR>	<CR><LF>$MYBCCH: <bcch1>,<mcc1>,<mnc1>,<lac1>,<cell-id1> <CR><LF>OK<CR><LF>
Test	AT$MYBCCH=?<CR>	<CR><LF>$MYBCCH: (list Of <mode>s),<bcch1>,…<bcch3> <CR><LF>OK<CR><LF>

Parameter

<mode>	Integer. The settings of locked channel can be saved after the module is powered off only by running **AT$MYBCCH=1,XX**. If the BCCH locked does not exist or its signal is weak, the module cannot register network. 0: Unlock 1: Lock the BCCH cell 2: List IDs of seven BCCH channels that have the strongest signals at current place.
<bcch>	channel ID
<num>	the number of BCCH channels in the BA list (7 at most)
<mcc>	Mobile country code
<mnc>	Mobile network code
<lac>	Location ID, four-byte characters in hexadecimal format
<cell-id>	Cell ID, hexadecimal

Example

```
AT$MYBCCH=2                                  List IDs of seven BCCH channels that have the
$MYBCCH: +BA(7):                             strongest signals at current place.
120, 460, 1, 2543, A85D
734, 460, 1, 2543, AB12
118, 460, 1, 2543, A8AD
115, 460, 1, 2543, A9A7
124, 460, 1, 2543, A85F
731, 460, 1, 2543, B046
123, 460, 1, 2543, A8A5
OK
AT$MYBCCH=1,120                              Lock channel 120.
OK
AT$MYBCCH=?                                  Query the value range of parameters in this
$MYBCCH: (0,1,2), 115,113,111                command. Channels 120 can be locked.
OK
AT$MYBCCH?                                   Query channels that are locked currently.
$MYBCCH: 734, 460, 1, 2543, AB12
OK
AT$MYBCCH=0                                  Unlock
OK
```

6.7 AT$MYBAND – Locking GSM Band

To lock the specified GSM band.

This command is used only for modules that support GSM900/1800.

Format

Type	Command	Response
Execute	AT$MYBAND=<band><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR: 980<CR><LF>
Query	AT$MYBAND?<CR>	<CR><LF>$MYBAND: <band> <CR><LF>OK<CR><LF>

Parameter

<band> Band to be locked.
0: Select a band automatically
1: GSM_EGSM_900
2: GSM_DCS_1800

Example

```
AT$MYBAND=1
OK
AT$MYBAND?
$MYBAND: 1
OK
```

6.8 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: 910C,BCE5C03
OK
```

6.9 AT$MYCGED – Querying Current Channel, RX Power and TX Power

To query current channel, RX power and TX power.

Format

Type	Command	Response
Execute	AT$MYCGED<CR>	<CR><LF>$MYCGED: <Channel>,<Rx power>,<Tx power> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<channel>	channel
<Rx power>	RX power, unit dBm (199 indicates that the value is invalid currently)
<Tx power>	TX power, unit dBm (199 indicates that the value is invalid currently)

Example

```
AT$MYCGED
$MYCGED: 48,-75,25
OK
AT$MYCGED
$MYCGED: 48,-44,199
OK
```

```
The current channel is 48, RX power is -75
dBm, and TX power is 25dBm.

The current channel is 48, RX power is -44
dBm, and no TX power.
```

6.10 AT$MYSYSINFO – Querying or Locking the Network Mode

To query or lock the network mode.

The module is in AUTO mode by default after starting up.

When the network mode is set to any non-AUTO mode, the module will automatically check the network status. If the module does not register to the network within three minutes (the default value is 3, which can be modified with the $MYNETAUTO command), the network is forced to switch to 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 1: AUTO (default) 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)
<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)
<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
```

6.11 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$MYSYSINFOURC=1
OK
AT$MYSYSINFOURC?
$MYSYSINFOURC: 1
OK
```

6.12 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$MYURCSYSINFO=1
OK
AT$MYURCSYSINFO?
$MYURCSYSINFO: 1
OK
```

6.13 $MYURCSYSINFO – Notifying the Network Mode

To notify the current network mode of the module.

The module might go offline when switching between different network modes. The module will not switch networks during the TCP connection unless the carrier forces switching or the network conditions trigger switching (signal strength changes or network carrying capacity changes, etc.) .

Format

Type	Command
Unsolicited result code	<CR><LF>$MYURCSYSINFO:<SysMode>,<mnc><CR><LF>

Parameter

<SysMode>	Network mode 0: No service 2: 2G (including GSM, EDGE, and CDMA) 3: 3G (including WCDMA, TD-SCDMA, and 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 module is registered on the 3G network of China Mobile.
OK

$MYURCSYSINFO: 3,01
```

6.14 AT$MYNETINFO – Setting Network Mode Choices

To set network mode choices.

This command is applicable to the communication interface protocol of China Southern Power Grid.

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: (value range of <mode>) <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=1
OK
```

```
AT$MYNETINFO?
$MYNETINFO: 1
OK
AT$MYNETINFO=?
$MYNETINFO: 1-7
OK
```

6.15 AT$MYNETAUTO – Enabling/Disabling the Default AUTO Network Modes during Startup

To enable/disable the default AUTO network mode during startup.

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 after the period specified in **<CYCLE>** passes. This value ranges 3 to 1440 minutes, 2 minutes by default.

Example

```
AT$MYNETAUTO=0
OK
AT$MYNETAUTO?
$MYNETAUTO: 0,2
OK
```

6.16 AT^SYSCONFIG – Setting Network Mode Choices

To set network mode choices.

Format

Type	Command	Response
Execute	AT^SYSCONFIG=<mode>,<acqorder>, <roam>,<srvdomain><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT^SYSCONFIG?<CR>	^SYSCONFIG:<mode>,<acqorder>,<roam>, <srvdomain><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode> Network mode
2: AUTO
13: GSM
14: WCDMA
16: No change
17: LTE
18: GSM/TDS-CDMA
19: GSM/LTE
20: TDS-CDMA/LTE
21: GSM/WCDMA
22: TDS-CDMA/WCDMA
23: WCDMA/LTE
24: TDS-CDMA/GSM/LTE
25: TDS-CDMA/WCDMA/LTE
26: TDS-CDMA/WCDMA/GSM
27: WCDMA/GSM/LTE
28: other (setting this value is not supported but it can be queried.)

<acqorder> Network access order
0: AUTO
1: GSM, TD-SCDMA, LTE
2: TD-SCDMA, GSM, LTE
3: No change
4: GSM, LTE, TD
5: TD-SCDMA, LTE, GSM
6: LTE, TD-SCDMA, GSM
7: LTE, GSM, TD-SCDMA

	8: GSM, WCDMA, LTE 9: WCDMA, GSM, LTE 10: GSM, LTE, WCDMA 11: WCDMA, LTE, GSM 12: LTE, WCDMA, GSM 13: LTE, GSM, WCDMA 14: TD-SCDMA, WCDMA, LTE 15: WCDMA, TD-SCDMA, LTE 16: TD-SCDMA, LTE, WCDMA 17: WCDMA, LTE, TD-SCDMA 18: LTE, WCDMA, TD-SCDMA 19: LTE, TD-SCDMA, WCDMA 20: TD-SCDMA, WCDMA, GSM 21: WCDMA, TD-SCDMA, GSM 22: TD-SCDMA, GSM, WCDMA 23: WCDMA, GSM, TD-SCDMA 24: GSM, WCDMA, TD-SCDMA 25: GSM, TD-SCDMA, WCDMA
<roam>	Roam 0: Not support 1: Roam 2: No change
<srvdomain>	Domain setting 0: CS_ONLY 1: PS_ONLY 2: CS_PS 3: ANY 4: No change

Example

```
AT^SYSCONFIG=13,2,1,2
OK
AT^SYSCONFIG?
^SYSCONFIG:2,2,0,2

OK
```

6.17 AT+NETDMSG – Querying Network Registration Information

To query the network registration information.

This command works only after the module is registered with a network.

The value of <TA> is valid only when the module is in a GSM call or transmitting or receiving GPRS data.

On a 3GPP network, there will be valid values in the fields of LAC, CELL_ID, and BSIC and the value in the SID, NID, and BID fields is 0.

On a 3GPP2 CDMA1X network, there will be valid values in the fields of SID, NID, and BID and the value in the LAC, CELL_ID, and BSIC fields is 0. For HDR only, the values of the SID, NID, and BID fields is 0 since the HDR standard base stations do not issue information indicating the network.

Format

Type	Command	Response
Execute	AT+NETDMSG<CR>	+NETDMSG: <MCC+MNC>,[<LAC>/SID>/<TAC>],<CELL_ID>, [<BSIC>/<Phy_cellid>/<PSC>],<net_mode>,<BAND>, <ARFCN>,<RX_dBm>,<TX_dBm>,<SID>,<NID>,<BID>, <RSRP>,<RSRQ>,<SINR> <CR><LF>OK<CR><LF>

Parameter

<MCC+MNC>	Mobile Country Code + Mobile Network + Code
[<LAC>/<SID>/<TAC>]	Hexadecimal SID on CDMA1X mode TAC on LTE mode Location Area Code on other modes
[<CELL_ID>/<NID>]	Hexadecimal NID on CDMA1X mode CELL_ID on other modes
[<BSIC>/<Phy_cellid>/<PSC>]	BSIC (Base Station Identity Code) on GSM mode, decimal BID on CDMA1X mode, Hexadecimal PSC on WCDMA mode, decimal Phy_cellid on LTE mode, decimal NULL on other modes
<net_mode>	Network Mode NONE CDMA1X CDMA1X and HDR CDMA1X and EHRPD CDMA1X and LTE HDR

	HDR REV0
	HDR REVA
	HDR REVB
	HDR EMPA EHRPD
	HDR MMPA EHRPD
	GSM
	GPRS
	EDGE
	WCDMA
	TDSCDMA
	HSDPA
	HSUPA
	HSDPA and HSUPA
	HSDPA+
	HSDPA+ and HSUPA
	DC HSDPA+
	DC HSDPA+ and HSUPA
	64QAM HSDPA+
	64QAM HSDPA+ and HSUPA
	DC HSDPA+ and DC HSUPA
	TDD LTE
	FDD LTE
<BAND>	Network Bands
	CDMA BC0
	...
	CDMA BC19
	GSM 850
	GSM 900
	GSM 1800
	GSM 1900
	WCDMA 2100
	WCDMA 1900
	WCDMA 1800
	WCDMA 1700 US
	WCDMA 850
	WCDMA 800
	WCDMA 2600
	WCDMA 900
	WCDMA 1700 JAPAN
	WCDMA 1500
	WCDMA 850 JAPAN
	LTE BAND 1
	…
	LTE BAND 43
	TDSCDMA BAND A
	...

	TDSCDMA BAND F
<ARFCN>	Absolute Radio-Frequency Channel Number
<RX dBm>	Absolute Radio-Frequency Channel Number
<TX dBm>	TX power, 199 indicates invalid
<SID>	TX power, 199 indicates invalid
<NID>	Network Identity Number on a CDMA1X network
<BID>	BID on a CDMA1X network
<RSRP>	Reference Signal Received Power, unit 0.1 dBm, valid on an LTE network
<RSRQ>	Reference Signal Received Quality, unit 0.1 dB, valid on an LTE network
<SINR>	Signal-to-Interference-Plus-Noise Ratio, unit 0.1 dB, valid on an LTE network.

Example

```
AT+NETDMSG                                          Query the network registration information on
+NETDMSG: "460+11", 0x7757, 0x77E9B30, 219,         an LTE network.
"CDMA1X and LTE", LTE BAND 3, 1825, -58, 11,
0x0, 0x0, 0x0, -848, -83, 159
OK
AT+NETDMSG                                          Query the network registration information on
+NETDMSG: "460+11", 0x0, 0x0, 0, "HDR REVA",        a non-LTE 3GPP2 network.
CDMA BC 0, 37, -60, 199, 0x0, 0x0, 0x0, 0, 0,
0
OK
AT+NETDMSG                                          Query the network registration information on
+NETDMSG: "460+00", 0xA57B, 0xAC1474C, 0,           a non-LTE 3GPP network.
"HSDPA", TDSCDMA BAND A, 10080, -69, 199, 0x0,
0x0, 0x0, 0, 0, 0
OK
AT+NETDMSG                                          The module has not been registered with any
+NETMSG: "0",0,0,0,"NONE",0,0,0,0,0,0,0,0,0,0       network or the network encountered
OK                                                  abnormalities.
```

6.18 AT^SYSINFO – Obtaining System Information

To obtain system information.

Format

Type	Command	Response
Set	AT^SYSINFO<CR>	<CR><LF>^SYSINFO: <srv_status>,<srv_domain>, <roam_status>,<sys_mode>,<sim_state> [,[<reserve>],<sys_submode>]<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<srv_status>	0: No service
	1: Limited service
	2: Service
	3: Service with area limit
	4: power saving
<srv_domain>	0: No service
	1: CS only
	2: PS only
	3: PS + CS
	4: EPS service
<roam_status>	0: Non-roaming
	1: Roaming
<sys_mode>	0: No service
	1: AMPS mode
	2: CDMA mode
	3: GSM mode
	4: EVDO mode or TDS mode
	5: WCDMA mode
	6: GPS mode
	7: GSM and WCDMA mode
	8: CDMA Hybrid mode
	9: LTE mode
	10: GSM, WCDMA, and LTE mode
<sim_state>	0: invalid
	1: valid
	255: No SIM card or PIN code is required
<reserve>	Reserved field
<sys_submode>	System sub mode
	0: No service
	1: GSM mode
	2: GPRS mode
	3: EDGE mode
	4: WCDMA mode
	5: HSDPA mode
	6: HSUPA mode
	7: HSUPA and HSDPA mode
	8: TD-SCDMA mode
	9: TDD_SUBACT
	10: FDD_SUBACT

Example

```
AT^SYSINFO
^SYSINFO: 2,3,0,4,1
```



```
OK
```

6.19 AT+NETREJURC – Controlling URC of 3GPP Network Reject Cause

To control the URC that notifies the 3GPP network reject cause.

This command is applicable only to LTE/WCDMA/GSM.

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

Format

Type	Command	Response
Set	AT+NETREJURC=<ONOFF><CR>	<CR><LF>OK<CR><LF> or <CR><LF>ERROR<CR><LF>
Query	AT+NETREJURC?<CR>	<CR><LF>+NETREJURC: <ONOFF> <CR><LF>OK<CR><LF>
Test	AT+NETREJURC=?<CR>	<CR><LF>+ NETREJURC: (range of supported <ONOFF>) <CR><LF>OK<CR><LF>

Timeout

The command times out if the module does not respond in 300ms.

Parameter

<ONOFF> Integer type
0: disable the URC of 3GPP network reject cause (default)
1: enable the URC of 3GPP network reject cause

Example

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

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

6.20 +NETREJCAUSE – Notifying 3GPP Network Reject Cause

To notify the 3GPP network reject cause.

The module reports the cause only when it fails to be registered on a 3GPP network or encounters a dial-up failure.

Format

Type	Response
Unsolicited result code	<CR><LF>+NETREJCAUSE: <reject_cause>,<string_cause><CR><LF>

Parameter

<reject_cause> Reject cause, integer type
See 3GPP TS24.301 and TS24.008

<string_cause> Reject cause, character string type
See 3GPP TS24.301 and TS24.008

Example

```
+NETREJCAUSE: 19,”ESM failure”                    The module reports a reject cause
+NETREJCAUSE: 6,"Illegal ME"                      The module reports a reject cause
```

6.21 AT+MYDATAONLY – Disabling CS Services

To disable voice call and SMS services.

The setting of this command takes effect after the module is restarted.

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 services
OK
AT+MYDATAONLY=1                          Disable voice call and SMS services
OK
AT+MYDATAONLY?                           Query the status of voice call and SMS services
+MYDATAONLY: 1
OK
```

7 Data Dialing Control

7.1 AT+CGDCONT – Defining PDP Context

To set the packet data protocol (PDP) format of GPRS/3G.

The APN of each operator is different.

Format

Type	Command	Response
Set	AT+CGDCONT=<cid>,<type>,<APN><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>,<pd1> ,…,<pdN><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CGDCONT=?<CR>	<CR><LF>+CGDCONT: value range of <cid>, value range of <PDP_type>, value range of <APN>, value range of <PDP_addr>, value range of <d_comp>, value range of <h_comp>, value range of <pd1> ,…, value range of pdN <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 (IPv6, IPv4v6,PPP). IP: Internet Protocol (IETF STD 5) IPv6: Network protocol IPv6 IPv4v6: Network protocol IPv4v6 PPP: Point to Point Protocol (IETF STD 51)
<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_addr>	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 0 - off (default if value is omitted)
<h_comp>	numeric parameter that controls PDP header compression. 0 - off (default if value is omitted)
<pd1>, … <pdN>	zero to N string parameters whose meanings are specific to the <PDP_type>

Example

```
AT+CGDCONT=1,”IP”,”CMNET”                                    Set PDP type to IP and APN to CMNET.
OK
AT+CGDCONT=1,”IP”,”UNINET”                                   Set PDP type to IP and APN to UNINET.
OK
AT+CGDCONT?
+CGDCONT: 1,"IPV6","CMNET","",,,                             Query the set APN
OK
AT+CGDCONT=?
+CGDCONT: (0-15),"IP",,,(0-3),(0-4),(0-1),(0-1)
+CGDCONT: (0-15),"PPP",,,(0-3),(0-4),(0-1),(0-1)
+CGDCONT: (0-15),"IPV6",,,(0-3),(0-4),(0-1),(0-1)            Query the parameter ranges
+CGDCONT: (0-15),"IPV4V6",,,(0-3),(0-4),(0-1),(0-
1)
OK
```

7.2 AT+CGATT – Setting GPRS Attach and Detach

To set GPRS attach and detach.

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
OK                                  The URC "$MYURCACT" will be prompted to notify the GPRS
                                    status after the AT$MYNETACT command is executed.
$MYURCACT=0,0
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
```

7.3 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 is registered on the network and the APN is set before dialing any number.

Format

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

Parameter

N/A.

Example

```
ATD*99#
CONNECT
```

7.4 AT$MYNETCON – Setting Initialization Parameters of Network Connections

To set parameters before network connection is set up.

This is a mandatory step before using the TCP/IP function.

Format

Type	Command	Response
Execute	AT$MYNETCON=<channel>,<type>,<type_name><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR:<err><CR><LF>
Query	AT$MYNETCON?<CR>	<CR><LF>$MYNETCON:<Channel>,<Type>,<type_name> <CR><LF>OK<CR><LF>

Parameter

<channel> Channel number, ranging from 0 to 5

<type> and <type_name>:

- USERPWD: User name and password, in format of "user,passwd"
- APN: Character string type
- CFGT: waiting time to send each packet, ranging from 1 to 65535. The default value is 100 and the unit is ms. This parameter is used in transparent transmission mode.
- CFGP: The length threshold of data to be transmitted, ranging from 1 to 1460. The default value is 1024. This parameter is used in transparent transmission mode.
- AUTH: Authentication type.
  0: NONE
  1: PAP (default)
  2: CHAP

Example

```
AT$MYNETCON=0,"USERPWD","user,pwd"
OK
AT$MYNETCON?
$MYNETCON:0,"USERPWD",","
$MYNETCON:0,"APN",""
$MYNETCON:0,"CFGT",1000
$MYNETCON:0,"CFGP",1024
$MYNETCON:0,"AUTH",0
OK
```

7.5 AT$MYNETACT – Activating/Deactivating the Network Connection

To activate or deactivate GPRS network connection.
This is a mandatory step before using the TCP/IP function.

If the command is executed successfully, the IP address is obtained and the local IP address is actively reported.

Format

Type	Command	Response
Execute	AT$MYNETACT=<channel>,<action><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYNETACT?<CR>	<CR><LF>$MYNETACT:<channel>,<status>,<IP> <CR><LF>OK<CR><LF>
Test	AT$MYNETACT=?<CR>	<CR><LF>$MYNETACT: (value range of <action>), (value range of <channel>) <CR><LF>OK<CR><LF>

Parameter

<action>	Specifies whether to activate network connections. 0: Deactivate network connections 1: Activate network connections
<channel>	Channel number, ranging from 0 to 5
<status>	0: The network connection has not been activated. 1: The network connection has been activated.

<IP> The local IP address displayed after the network connection is activated.

Example

```
AT$MYNETACT=0,1                          Activate the network connection.
OK                                       The command is executed successfully.
$MYURCACT:0,1,"10.92.220.73"             The URC "$MYURCACT" is prompted to notify the obtained IP
                                         address.
AT$MYNETACT=0,1                          Dual-stack activation
OK
$MYURCACT: 0,1,"10.67.163.14"
$MYURCACT:
0,1,"24e0:bf:d206:6ed:1:2:fd94:4982"
AT$MYNETACT?                             Query the local IP address.
$MYNETACT:0,1,"10.92.220.73"
OK
AT$MYNETACT=0,1                          Duplicate activation.
OK
$MYURCACT:0,1,"10.67.163.14"
AT$MYNETACT=0,1
ERROR:902
```

7.6 AT$MYIPFILTER – Configuring IP Access

To configure the IP addresses of clients that can access the module.

This command is valid only when the communication module works in Server mode.

If the IP address of the client that request connection is within the IP address range, the client can connect to the module.

At most 5 groups of IP address can be configured for a module.
All the clients with any IP address can connect to the module by default if the IP address is not configured.

Format

Type	Command	Response
Execute	AT$MYIPFILTER=<id>,<action>,<ip_address>,<net_mask><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR:<err><CR><LF>
Query	AT$MYIPFILTER?<CR>	<CR><LF>$MYIPFILTER:<id>,<ip_address>,<net_mask> <CR><LF>$MYIPFILTER:<id>,<ip_address>,<net_m

		ask> … <CR><LF>OK<CR><LF>
Test	AT$MYIPFILTER=?<CR>	<CR><LF>$MYIPFILTER: (value range of supported <id>),(value range of supported <action>)," "," " <CR><LF>OK<CR><LF>

Parameter

<id>	The ID of data, ranging from 0 to 4. 5 groups data in total.
<action>	Delete or add specified IP authentication channel 0: Delete specified IP authentication channel. 1: Add specified IP authentication channel. 2: Delete all IP authentication channels.
<ip_address>	IP address of legal client, char type, in format of XXX.XXX.XXX.XXX.
<net_mask>	subnet mask, char type, in format of XXX.XXX.XXX.XXX.
<err>	Error code

Example

```
AT$MYIPFILTER=0,1,"192.168.0.23","255.255.255.255"   Add the specified IP authentication channel.
OK
AT$MYIPFILTER=0,0,"192.168.0.23","255.255.255.255"   Delete the specified IP authentication channel.
OK
AT$MYIPFILTER=0,1,"192.168.0.23","255.255.255.0"     Add allowed IP address queue (192.168.0.0 to 192.168.0.255).
OK
AT$MYIPFILTER?                                       Query the specified IP authentication channels.
$MYIPFILTER:0,"192.168.0.23","255.255.255.255"
$MYIPFILTER:1,"192.168.0.23","255.255.255.255"
$MYIPFILTER:2,"",""
$MYIPFILTER:3,"",""
$MYIPFILTER:4,"",""
OK
```

7.7 AT$MYNETURC – Controlling the Unsolicited Result Code

To control the the unsolicited result code.

This command is valid only when the MT uses internal protocol stack.

Format

Type	Command	Response
Execute	AT$MYNETURC=<ONOFF><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYNETURC?<CR>	<CR><LF>$MYNETURC:<ONOFF> <CR><LF>OK<CR><LF>

Parameter

<ONOFF> Integer type
0: Disable unsolicited result code of the internal protocol stack
1: Enable unsolicited result code of the internal protocol stack (default)

Example

```
AT$MYNETURC=1
OK
```

7.8 $MYURCACT – Indicating Network Connection Status

To indicate the status of the network connection.

Format

Type	Command
Unsolicited result code	<CR><LF>$MYURCACT:<channel>,<type>,<IP><CR><LF>

Parameter

<channel> Channel ID
<type> Network connection status
0: Network disconnection
1: Network connection
<IP> Local IP address

Example

```
AT$MYNETACT=0,1                          Activate the network connection.
OK                                       Indicate the network connection status.
$MYURCACT:0,1,"10.91.102.62"             Network connection is activated successfully. The
                                         obtained local IP address is 10.91.102.62.
```

7.9 AT+DNSSERVER – Setting the DNS Server

To set the preferred DNS server and alternate DNS server.

Generally, you do not need to manually set the DNS server. During the PPP negotiation phase, the base station controller will assign the DNS server.

Format

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

Parameter

<n> ID of the DNS server. The values are 1 (preferred DNS server) and 2 (alternate DNS server).

<dns -ip> IP address of the DNS server.

Example

```
AT+DNSSERVER=1,114.114.114.114                Set the DNS server.
+DNSSERVER:OK
AT+DNSSERVER?
+DNSSERVER:dns1:114.114.114.114;dns2:0.0.0.0  Query the DNS server.
```

7.10 AT$MYNETDNS – Setting DNS Servers

To set the primary and secondary DNS servers

The DNS settings become valid immediately after you set them but they are not saved after the module is powered down.

If you do not set the DNS servers by sending this command but the network assigns DNS server addresses automatically, the assigned addresses can be queried by **$MYNETDNS?**.

The DNS server addresses set by this command are not valid for PPP dialup.

This command works only afer <channel> is activated.

Format

Type	Command	Response
Set	AT$MYNETDNS=<channel>,<DNS1>,<DNS2><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYNETDNS?<CR>	<CR><LF>AT$MYNETDNS:<Channel>,<DNS1>,<DNS2> <CR><LF>OK<CR><LF>

Parameter

<channel>	Channel number, integer type, 0-5
<DNS1>	Primary DNS server address
<DNS2>	Secondary DNS serve address

Example

```
AT$MYNETDNS=0,114.114.114.114,8.8.8.8                    Setting DNS server addresses
OK
AT$MYNETDNS?                                             Query DNS server addresses
$MYNETDNS: 0,202.96.134.33,202.96.128.166
OK
```

7.11 AT$MYNETDTS – Collecting Traffic Statistics

To collect statistics on module traffic.

This command is used to collect statistics on the total traffic in bytes that passes through the module.

The statistics can be saved upon a power failure. After traffic statistics is enabled, the statistics is periodically saved at the backend. You can also run AT$MYNETDTS=1,1 to manually save the statistics.

Traffic statistics is supported when the internal or external protocol stack is used for dial-up, but the two dial-up modes cannot be used concurrently.

Format

Type	Command	Response
Execute	AT$MYNETDTS=<onoff>,<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYNETDTS?<CR>	<CR><LF>$MYNETDTS: <onoff>,<bytes_sent>,<bytes_recv> <CR><LF>OK<CR><LF>
Test	AT$MYNETDTS=?<CR>	<CR><LF>$MYNETDTS: <onoff> <CR><LF>OK<CR><LF>

Parameter

<onoff> Integer type
0: Disable traffic statistics. (Default)
1: Enable traffic statistics.

<mode> Integer type. This parameter is set for traffic statistics.
0: Clear the traffic statistics of the communication module.
1: Save the traffic statistics of the communication module.

<bytes_sent> Data in bytes sent by the communication module. The value ranges from 0 to 999999999999.

<bytes_recv> Data in bytes received by the communication module. The value ranges from 0 to 999999999999.

Example

```
AT$MYNETDTS=1                          Query the traffic.
OK
AT$MYNETDTS?
$MYNETDTS:1,420,680
OK
AT$MYNETDTS=1,0                        Clear the traffic statistics.
OK
AT$MYNETDTS=1,1                        Manually save the traffic statistics to the Flash.
```

```
OK
AT$MYNETDTS=?                              Query the traffic statistics switch status.
$MYNETDTS:0
OK
```

8 SMS Commands

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

8.2 AT+CPMS – Setting Preferred SMS Storage

To set preferred SMS storage.

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

Format

Type	Command	Response
Set	AT+CPMS=<mem1>[,<mem2>,<mem3>]<CR>	<CR><LF>+CPMS: <used1>, <total1>, <used2>, <total2>, <used3>, <total3> <CR><LF>OK<CR><LF>
Query	AT+CPMS?<CR>	<CR><LF>+CPMS: <mem1>, <used1>, <total1>, <mem2>, <used2>, <total2>, <mem3>, <used3>, <total3> <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>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 (not supported on CDMA mode)
<mem2>	memory for writing and sending, String type, for example, "SM", "ME".
<mem3>	priority memory for storing SMS, String type, for example, "SM", "ME".
<used>	Used quantity
<total>	Total capacity of the storage

Example

```
AT+CPMS="SM"                                          Set the SMS storage to "SM", that is, store SMS
+CPMS: 50,50,50,50,50,50                              messages in SIM card.
OK
AT+CPMS?                                              Query the capacity of current SMS storage.
+CPMS: "SM",50,50,"SM",50,50,"SM",50,50
OK
AT+CPMS=?                                             Query the available storages.
+CPMS: ("SM","ME"),("SM","ME"),("SM","ME")
OK
AT+CPMS="SM"                                          No SIM card is installed.
+CMS ERROR: 500
```

8.3 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>OK<CR><LF>
Test	AT+CMGF=?<CR>	<CR><LF>+CMGF: (range of supported <mode>s) <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
```

8.4 AT+CSCS – Setting 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>OK<CR><LF>
Test	AT+CSCS=?<CR>	<CR><LF>+CSCS: (list of supported <chset>s) <CR><LF>OK<CR><LF>

Parameter

<chset>

- "GSM": default GSM alphabet (GSM03.38.6.2.1)
- "IRA" international reference alphabet (ITU-T T.50) (default)
- "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. The rest of the commands and responses are still in IRA alphabetical format.

Example

```
AT+CSCS="IRA"                               Set IRA character set.
OK
AT+CSCS?                                    Query the format of current character set.
+CSCS: "IRA"
OK
AT+CSCS=?                                   Query the character set formats that the module
+CSCS: ("IRA","GSM","UCS2")                 supports.
OK
```

8.5 AT+CNMI – Setting SMS Indication Mode

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

On CDMA mode, <bm> and <ds> are not supported.

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>OK<CR><LF>
Test	AT+CNMI=?<CR>	<CR><LF>+CNMI: (list of supported <mode>s),(list of supported <mt>s),(list of supported <bm>s),(list of supported <ds>s),(list of supported <bfr>s) <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.

<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
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** to 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.
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. 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: <memr>,<index>
<bfr>	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 **2, 1, 0, 0, 0**.
- The recommended setting is **+CNMI: 2,1,0,0,0** (new messages are stored on SIM card rather than displayed directly.

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: (0,1,2),(0,1,2,3),(0,2),(0,1,2),(0,1)
OK
AT+CNMI?                                            Query the current setting of the parameters.
+CNMI: 1,1,0,0,0
OK
```

8.6 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 status of the message is received unread, the 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)

	<CR><LF>+CMGR: <stat>,<oa>,[<alpha>],<scts>[,<tooa>,<fo>,<pid>,<dcs>, <sca>,<tosca>,<length>] <CR><LF><data><CR><LF><CR><LF>OK<CR><LF> • PDU mode (+CMGF=0) <CR><LF>+CMGR: <stat>,[<alpha>],<length><CR><LF><pdu><CR><LF><CR><LF>OK<CR><LF>

Parameter

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

<stat> Status of SMS messages in the storage

- When set **AT+CMGF=1**,
  "REC UNREAD": received unread
  "REC UNREAD": received read
  "STO UNSENT": stored unsent
  "STO SENT": stored sent
- When set AT**+CMGF=0**,
  0: received unread
  1: received read
  2: stored unsent
  3: stored sent

<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).

<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>**.

<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)

<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=33                                                          Read the message in text mode
+CMGR: "REC READ","13510895077",,"15/07/23,20:14:55+32"
123

OK
AT+CMGR=33                                                          Read the message in PDU mode
+CMGR: 1,,22
0891683110808805F0240BA13115805970F70000517032024155230331D90C

OK
AT+CMGR=50                                                          The 50th message does not exist in the memory.
+CMS ERROR: 321
```

8.7 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) <CR><LF>+CMGL:<index>,<stat>,<oa/da>,[<alpha>],[<scts>][,<tooa/toda>,<length>]<CR><LF> <data><CR><LF> [<CR><LF>+CMGL: <index>,<stat>,<da/oa>,[<alpha>],[<scts>][,<tooa/toda>,<length>]<CR><LF> <data><CR><LF>>[...]] • PDU mode (+CMGF=0) <CR><LF>+CMGL:<index>,<stat>,[<alpha>],<length> <CR><LF><pdu> [<CR><LF> +CMGL:<index>,<stat>,[<alpha>],<length><CR><LF> <pdu><CR><LF> [...]]
Test	AT+CMGL=?<CR>	• Text mode (+CMGF=1) <CR><LF>+CMGL: (list of supported <stat> values) <CR><LF> OK<CR><LF> • PDU mode (+CMGF=0) <CR><LF>+CMGL: (value range of <stat>) <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"
0500034F0302672C77ED4FE14E2D768452694F596D4191CF5305542B53E052A053056D4191CFFF0C8BF76CE8610F533
A52064F7F7528FF093002672C6B2167E58BE27ED3679C5B5857285EF665F6FF0C8BF74EE551FA8D264E3A51C6300276
7B96468054901A624B673A84254E1A53850020007700610070002E00310030003000310030002E0063006F006D
+CMGL: 2,"REC READ","10010",,"14/06/23,14:42:27+32"
0500034F03016E2999A863D0793AFF0C622A6B62003667080032003265E5FF0C60A85F536708595799105185 6D4191C
F5DF24F7F752800340033002E00360031004D0042FF0C52694F596D4191CF003200350036002E00330039004D0042FF
08598260A88BA28D2D4E867EA256F4811662164E9196C0621660A6005400566D4191CF53E052A05305FF0C5219

OK
AT+CMGL=?                                          Query in text format (AT+CMGF=1).
+CMGL: ("REC UNREAD", "REC READ", "STO UNSENT",
"STO SENT", "ALL")
OK
AT+CMGL=?                                          Query in PDU format (AT+CMGF=0).
+CMGL: (0-4)
OK
AT+CMGL=ALL                                        A pair of quotation marks ("") is required
ERROR                                              for the parameter.
AT+CMGF=1                                          The parameter should be set to 0.
OK
AT+CMGL=4
ERROR
AT+CMGF=1                                          The parameter should be set to 1.
OK
AT+CMGL=4
ERROR
```

8.8 AT+CMGS – Sending SMS Messages

To send an SMS message from the module to the network.

If you use a serial port tool to send PDU SMS messages, input a carriage return manually after the AT+CMGS command, or send <CR> in hexadecimal.

For details of sending PDU SMS message, see Appendix A.1 .

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>OK<CR><LF> • PDU mode (+CMGF=0): <CR><LF>+CMGS: <mr>[,<ackpdu>] <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→
+CMS ERROR: 500

AT+CMGS=33<CR>                                        PDU mode (+CMGF=0)
>0891683108705505F001000B815118784271F20008146DF1
57335E025B9D5B89533A59276D6A80545EFA→
+CMGS: 119
OK
```

8.9 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>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
+CMS ERROR: 304                                          required for the number in text mode.
AT+CMGW=31<CR>                                           PDU mode (+CMGF=0)
>0891683108705505F001000B813124248536F3000812004
00026002A535A53D153A653C1532052C7<Ctrl+Z>
+CMGW: 1
```

```
OK
```

8.10 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>OK<CR><LF> • PDU mode (+CMGF=0): <CR><LF>+CMSS: <mr>[,<ackpdu>] <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.
```

8.11 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: (value range of <index>),(value range of <delflag>) <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>**.

Maximum value of the recording number of the stored SMS messages varies from different SIM cards. It is not a fixed value.

The value of <index> cannot be set to 0, when only **<index>** is configured.

Example

```
AT+CMGD=1,3                                          Delete all read, sent, and unsent SMS
OK                                                   messages.

AT+CMGD=?                                            Query the value ranges of parameters.
+CMGD:
(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,
22,23,24,25,26,27,28,29,30,31,32,33),(0-4)
OK
```

8.12 AT+CSCA – Setting SMS Center Number

To set the SMS center number.

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>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
ERROR                                       mandatory for SMSC number.
AT+CSCA?                                    Query the SMSC number.
+CSCA: "+8613800755500",145
OK
```

8.13 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>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)

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: 17,167,0,0
OK
```

8.14 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**.

This command is not support on CDMA mode.

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>OK<CR><LF>
Test	AT+CSDH=?<CR>	<CR><LF>+CSDH: (value range of <show>) <CR><LF>OK<CR><LF>

Parameter

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

Example

```
AT+CSDH=0                                                    Set the header information to
OK                                                           not display
AT+CMGR=1                                                    Read the 1th 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=1
+CMGR:"REC READ","13510895077",,"15/07/23,20:58:28+32",      Read the 1th 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
```

8.15 AT+CSAS – Saving the Setting

To save the setting.

This command can only be used to save the parameter settings of the AT+CSCA and AT+CSMP commands.

The command function is currently not supported, and the compatibility process is performed on the returned code.

Format

Type	Command	Response
Execute	AT+CSAS[=<profile>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CSAS=?<CR>	<CR><LF>+CSAS:<profile> <CR><LF>OK<CR><LF>

Parameter

<profile> Value
0: Save the settings (or default parameters).

Example

```
AT+CSAS                         Save the setting.
OK
AT+CSAS=0                       Save the setting.
OK
AT+CSAS=1
ERROR
AT+CSAS=?                       Query the valid parameter values for the command.
+CSAS:0
OK
```

9 TCP/UDP Commands

9.1 AT$MYNETSRV – Setting Parameters for the Non-transparent Transmission

To set the parameters of TCP/IP service.

Format

Type	Command	Response
Set	AT$MYNETSRV=<Channel>,<SocketID>,<nettype>,<viewMode>,<ip:port><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYNETSRV?<CR>	<CR><LF>$MYNETSRV:<Channel>,<SocketID>,<nettype>,<viewMode>,<ip:port> <CR><LF>OK<CR><LF>
Test	AT$MYNETSRV=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<Channel>	Channel ID, integer type, ranging from 0 to 5. Each channel has 2 KB sending buffer and 10 KB receiving buffer.
<SocketID>	Socket ID, integer type, ranging from 0 to 14
<nettype>	Network type. 0: TCP Client 1: TCP Server. In this mode, at most 4 listening services can be set up. 2: UDP
<viewMode>	Data display mode in non-transparent transmission mode 0: HEX (Default) 1: TEXT
<ip:port>	IP address and port number, e.g. 59.40.29.34:4988If the IP address is 127.0.0.1, listening service has been set up using this SocketID.
<ip_type>	Domain name parsing mode 0: IPv4 (Default) 1: IPv6
<err>	error code

Example

```
AT$MYNETSRV=0,0,0,0,"59.40.29.34:4988"      Set TCP client service parameters for channel 0.
OK                                          (The IP address and port number of the remote server
                                            is 59.40.29.34: 4988.)
```

9.2 AT$MYNETLOCALPORT – Setting the Local Port

To set the local TCP/IP port.

Format

Type	Command	Response
Execute	AT$MYNETLOCALPORT=<SocketID>,<local_port><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR: <err><CR><LF>
Query	AT$ MYNETLOCALPORT?<CR>	<CR><LF>$MYNETLOCALPORT:<SocketID>,<local_port> <CR><LF>OK<CR><LF>
Test	AT$ MYNETLOCALPORT=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<SocketID> Socket ID. Int type, range from 0 to 14.
<local_port> Local port, int type, range from 0 to 65535.

Example

```
AT$MYNETLOCALPORT=0,6800        Set the local number of the TCP connection on socket 0 to 6800.
OK                              successful

AT$MYNETLOCALPORT=0,0           Set the local number of the TCP connection on socket 0 to a
OK                              random number.
                                successful
```

9.3 AT$MYNETOPEN – Enabling TCP/UDP Service

To enable TCP/UDP connections

Format

Type	Command	Response
Execute	AT$MYNETOPEN=<SocketID><CR>	<CR><LF>$MYNETOPEN:<SocketID>[,<Remote_MSS>] <CR><LF>OK<CR><LF> Or <CR><LF>ERROR:<err><CR><LF>
Query	AT$MYNETOPEN?<CR>	<CR><LF>$MYNETOPEN:<SocketID>,<LocalIP>,<local_port>,<gate>,<DNS1>,<DNS2>,<type>,<dest_ip>,<dest_port> <CR><LF>OK<CR><LF>
Test	AT$MYNETOPEN=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<SocketID>	Socket ID, integer type, ranging from 0 to 14
<Remote_MSS>	MSS value negotiated by client and server after the TCP connection is enabled, integer type
<LocalIP>	Local IP address, character type
<local_port>	Local port (varying with channels)
<gate>	Gateway
<DNS1>	Primary DNS server
<DNS2>	Standby DNS server
<type>	TCP Client/TCP Server/UDP 0: TCP Client 1: TCP Server 2: UDP
<dest_ip>	The IP address to be linked
<dest_port>	The number of the port to be linked

Example

```
AT$MYNETOPEN=0                                                    Enable socket 0.
$MYNETOPEN: 0,2000
OK
AT$MYNETOPEN?                                                     Query sockets that are in
$MYNETOPEN: 0,10.92.220.73,5000,0,dns1,dns2,0,59.40.29.34,4988    OPEN state. Socket 0 is in
OK                                                                OPEN state.
AT$MYNETSRV=5,1,0,0,"58.60.184.213:10188"                         Fail to open the socket.
OK
AT$MYNETOPEN=2
ERROR: 913
```

9.4 AT$MYNETREAD – Reading Received Data

To read received data.

Format

Type	Command	Response
Execute	AT$MYNETREAD=<Socket ID>,<data_len><CR>	<CR><LF>$MYNETREAD:<SocketID>,<data_length> <CR><LF><data> <CR><LF>OK<CR><LF>

Parameter

<SocketID>	Socket ID, integer type, ranging from 0 to 14
<data_len>	The maximum length of data expected to be read, ranging from 1 to 2048.
<data_length>	The length of data actually read, ranging from 0 to 2048.
<data>	Data. No value for <data> is displayed if data_length is 0.

Example

```
AT$MYNETREAD=0,2048                    Read 2048-byte data from socket 0.
$MYNETREAD:0,0                         No data is in buffer or all data has been read.
OK

AT$MYNETREAD=0,2048                    Read 2048-byte data from socket 0.
$MYNETREAD:0,10                        10-byte data is actually read.
1234567890                             The data content is 1234567890.
OK
```

9.5 AT$MYNETWRITE – Sending Data

To send data.

It is recommended to send less than 1472 bytes every time through a UDP link to decrease the packet loss rate.

Format

Type	Command	Response
Execute	AT$MYNETWRITE=<SocketI	<CR><LF>$MYNETWRITE:<SocketID>,<data_len>

	D>,<data_len><CR>	<CR><LF><input_data> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR:<err><CR><LF>
Test	AT$MYNETWRITE=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<SocketID>	Socket ID, integer type, ranging from 0 to 14
<data_len>	The length of data to be sent, ranging from 1 to 2048
<input_data>	Content of sent data
<err>	Error code

Example

```
AT$MYNETWRITE=0,10                    Socket 0 sends 10-byte data.
$MYNETWRITE:0,10                      Indicates that you can input data to be sent.
1234567890                            Write data to be sent: 1234567890.
OK                                    The command is executed successfully.
```

9.6 AT$MYNETCLOSE – Closing Socket Connection

To close a socket connection.

Format

Type	Command	Response
Execute	AT$MYNETCLOSE=<SocketID><CR>	<CR><LF>$MYNETCLOSE:<SocketID> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR:<err><CR><LF>

Parameter

<SocketID>	Socket ID, integer type, ranging from 0 to 14

Example

```
AT$MYNETCLOSE=0                          Close socket 0.
$MYNETCLOSE:0
OK
```

9.7 AT$MYNETACK – Querying TCPACK Information

To query the quantity of TCP data sent by the MT but not acknowledged and the size of remaining buffer for the built-in protocol stack

Format

Type	Command	Response
Execute	AT$MYNETACK=<SocketID><CR>	<CR><LF>$MYNETACK:<SocketID>,<unAcked_dataLen>,<rest_bufferLen> <CR><LF>OK<CR><LF>

Parameter

<SocketID>	Socket ID, integer type, ranging from 0 to 14
<unAcked_dataLen>	The quantity of data sent but not acknowledged, 2^32-1 at most
<rest_bufferLen>	The size of remaining buffer for built-in protocol, ranging from 0 to 10240

Example

```
AT$MYNETACK=0                    Query the ACK that socket 0 sends data
$MYNETACK:0,0,10240              The data that has been sent but not acknowledged is 0.
OK                               The size of remaining buffer for built-in protocol is 10240.
```

9.8 AT$MYNETACCEPT – Accepting the Listening Request

To accept listening request from the client in TCP server mode.

After the command is executed successfully, if transportMode is 0, the module enters the non-transparent transmission mode. If transportMode is 1, the module enters the transparent transmission mode.

To close connections to the server in transparent transmission modem, you must exit data mode using the +++ command and enter command mode. Then you can run AT$MYNETCLOSE=SocketID to close SocketID listened by server and all connections using the SocketID.

Format

Type	Command	Response
Execute	AT$MYNETACCEPT=<SocketID>,<action>,<transportMode><CR>	<transportMode>=0: <CR><LF>OK<CR><LF> <transportMode>=1: <CR><LF>CONNECT<CR><LF> Or <CR><LF>ERROR: <err><CR><LF>
Query	AT$MYNETACCEPT?<CR>	<CR><LF>OK<CR><LF>
Test	AT$MYNETACCEPT=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<SocketID>	New socket ID that is reported by $MYURCCLIENT, ranging from 0 to 14
<action>	Specifies whether to accept listening request 0: Accept 1: Reject
<transportMode>	The transmission mode after the listening service is enabled 0: Non-transparent transmission mode 1: Transparent transmission mode

Example

```
AT$MYNETSRV=0,0,1,0,"127.0.0.1:5000"        Set the local listening port to 5000 for socket 0.
OK
AT$MYNETOPEN=0                              Enable socket 0 to start the server mode.
$MYNETOPEN:0
OK

$MYURCCLIENT:1,"10.90.72.235",51614         Report connection of client 1
AT$MYNETACCEPT=1,0,0                        Accept connection of client 1.
OK
$MYURCCLIENT:2,"10.90.72.235",55469         Report connection of client 2
AT$MYNETACCEPT=2,1,0                        Reject connection of client 2.
OK
```

9.9 AT$MYNETCREATE – Enabling Transparent Transmission Service

To enable the transparent transmission service of the module.

In TCP/UDP client mode, the module starts to connect the server after executing this command. If the connection to the server is set up successfully, the module enters transparent transmission mode.

In TCP server mode, the module returns OK after executing this command. Only listening socket is set up successfully and the module is still in the command mode; it enters transparent transmission mode only after the connection is set up on the client.

Before you execute AT$MYNETCLOSE to exit the transparent transmission mode, you must use the +++ command to switch to the command mode.

+++ must be entered in character string format in order to exit the transparent transmission mode. If a single "+" is entered and the next "+" is entered 50ms later, "+" will be sent as data.

Format

Type	Command	Response
Execute	AT$MYNETCREATE=<channel>,<mode>,<Socket ID>,<ip>,<port>[,<local_port>][,<ip_type>]<CR>	<mode>=1: <CR><LF>OK<CR><LF> <mode>=0 or 2: <CR><LF>CONNECT<CR><LF> Or <CR><LF>ERROR: <err><CR><LF>
Query	AT$MYNETCREATE?<CR>	<CR><LF>OK<CR><LF>
Test	AT$MYNETCREATE=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<channel>	Channel number, ranging from 0 to 5
<mode>	TCP Client/TCP Server/UDP 0: TCP Client 1: TCP Server 2: UDP
<SocketID>	Socket ID, integer type, ranging from 0 to 5
<ip>	The IP address to be linked

<port>	The port to be connected when <mode> is set to client; the listening port when <mode> is set to server.
<local_port>	It can be omitted
<ip_type>	Domain name parsing mode, it can be omitted. 0: IPv4 (default) 1: IPv6

Example

```
AT$MYNETCREATE=0,0,0,"59.40.29.34",4988,3000      Set up channel for transparent TCP data transmission.
CONNECT                                           The destination IP address and port are "59.40.29.34",4988.

OK                                                CONNECT indicates that the transparent transmission service is available.
                                                  Enter +++.
                                                  The module returns OK, indicating that it enters command mode again.
T$MYNETCREATE=0,0,0,"2400:da00:2::29",4988,3000   Establish a TCPIPv6 transparent transmission channel.
CONNECT

OK
```

9.10 $MYURCSRVPORT – Notifying Server Port Opened

To report that the module connects to the server successfully.

This URC is used for TCP/UDP clients in non-transparent transmission mode.

Format

Type	Response
Unsolicited Result code	<CR><LF>$MYURCSRVPORT: <PortNum><CR><LF>

Parameter

<PortNum>	Number of the server port that is opened, ranging from 0 to 65535, integer type

Example

```
AT$MYNETOPEN=0
$MYNETOPEN: 0,2000                          The module connects to the server successfully.
```

```
OK

$MYURCSRVPORT: 1234                          The module reports a server port of 1234.
```

9.11 $MYURCREAD – Notifying Data Received

To notify the received data.

The URC is displayed once the module receives a packet of data.

Format

Type	Command
Unsolicited result code	<CR><LF>$MYURCREAD:<SocketID><CR><LF>

Parameter

<SocketID> Socket ID, integer type, ranging from 0 to 14.

Example

```
AT$MYNETSRV=0,0,0,0,"59.40.29.34:4988"     Set socket 0 as the TCP client.
OK
AT$MYNETOPEN=0                             Enable socket 0 to set up TCP connection.
$MYNETOPEN:0,2000
OK

$MYURCREAD:0                               Receive data from socket 0.
AT$MYNETREAD=0,2048                        Read 2048 bytes data.
$MYNETREAD:0,10                            Socket 0 read 10-byte data.
0123456789                                 The data content is 0123456789.
OK
```

9.12 $MYURCCLOSE – Notifying the Disconnection of the Current Socket ID

To notify the disconnection of the current Socket ID.

Data in the buffer for the socket will be cleared after this command is displayed.

Format

Type	Command
Unsolicited result code	<CR><LF>$MYURCCLOSE:<SocketID><CR><LF>

Parameter

<SocketID> ID of disconnected socket, ranging from 0 to 14.

Example

```
AT$MYNETSRV=0,0,0,0,"59.40.29.34:4988"     Set socket 0 as the TCP client.
OK
AT$MYNETOPEN=0                             Enable socket 0 to set up TCP connection.
$MYNETOPEN:0,2000
OK


$MYURCCLOSE:0                              Socket 0 is disconnected.
```

9.13 $MYURCCLIENT – Notifying the Client Connection

To notify the client connection after listening service is started.

This command indicates that some client applies for connection.

Format

Type	Command
Unsolicited result code	<CR><LF>$MYURCCLIENT:<SocketID>,<IP>,<port><CR><LF>

Parameter

<SocketID> Socket ID of new client
<IP> IP address of new client
<port> Client port

Example

```
AT$MYNETSRV=0,0,1,0,"127.0.0.1:5000"        Set socket 0 as the TCP server. The listening port
OK                                          is 5000.
AT$MYNETOPEN=0                              Enable socket 0 to start listening.
$MYNETOPEN:0
OK

$MYURCCLIENT:1,"10.90.72.235",51614         Indicate client connection.
AT$MYNETACCEPT=1,0,0
OK
```

9.14 AT$UDPLISTEN – Creating UDP Listener as a Server

To create a UDP listener as a server.

At most 14 clients are supported. This command is valid only after a PPP connection is set up.

V012 and the later versions support this command.

Format

Type	Command	Response
Set	AT$UDPLISTEN=<port>[,<recv_mode>][,<ip_type>]<CR>	<CR><LF>$UDPLISTEN: <socket>,OK<CR><LF> Or <CR><LF>$UDPLISTEN: bind error<CR><LF> Or <CR><LF>Listening<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$UDPLISTEN?<CR>	<CR><LF>$UDPLISTEN: listening status<CR><LF> Or <CR><LF>$UDPLISTEN: not listening<CR><LF>
Test	AT$UDPLISTEN=?<CR>	<CR><LF>$UDPLISTEN: (range of <port> value),(range of recv_mode> value)<CR><LF>

Parameter

<port> port number, ranging from 1 to 65535

<recv_mode> Data receiving mode

0: output the data after receiving it (default)

1: save the data to the buffer after receiving it and send **$IPNETREAD** to read it

	later
<ip_type>	IP type 0: IPv4 (default) 1: IPv6

Example

```
AT$UDPLISTEN=6000                       Set a listening socket on port 6000.
$UDPLISTEN: 0,OK                        The server starts to listen.
Or
$UDPLISTEN: bind error                  Fails to bind
AT$UDPLISTEN=6000                       Set a listening socket on channel 0.
ERROR                                   No PPP connection is set up yet.
AT$UDPLISTEN=6000
                                        A listening socket is set up already.
Listening…
AT$UDPLISTEN=?                          Query the value range of the parameters.
$UDPLISTEN: (1-65535),(0-1)
OK
AT$UDPLISTEN?                           Query the listening status. Here the server is in the
$UDPLISTEN: listening status            listening status.
AT$UDPLISTEN?                           Query the listening status. Here the server is not in
$UDPLISTEN: not listening               the listening status.
```

9.15 AT$CLOSEUDPLISTEN – Closing Listening Socket

To close the listening socket and close all connections to clients.

V012 and the later versions support this command.

Format

Type	Command	Response
Execute	AT$CLOSEUDPLISTEN<CR>	<CR><LF>$CLOSEUDPCLIENT: <socket>,remote link closed<CR><LF> <CR><LF>$CLOSEUDPLISTEN: <socket>,local link closed<CR><LF>

Parameter

<socket>	Socket ID

Example

```
AT$CLOSEUDPLISTEN=0                                  Close the listening socket.

$CLOSEUDPLISTEN: 0,local link closed

                                                     The connections to the server are
$CLOSEUDPCLIENT: 1,remote link closed                closed.
```

9.16 AT$CLOSEUDPCLIENT – Closing UDP Remote Socket

To close remote sockets

The sockets can be used for other connections after they are closed.

V012 and the later versions support this command.

Format

Type	Command	Response
Execute	AT$CLOSEUDPCLIENT[=<socket>]<CR>	<CR><LF>$CLOSEUDPCLIENT: <socket>,remote link closed<CR><LF>

Parameter

<Socket> Socket ID

Example

```
AT$CLOSEUDPCLIENT
                                                     There is no parameter in this command.
$CLOSEUDPCLIENT: 1,remote link closed                All remote sockets are closed successfully.

$CLOSEUDPCLIENT: 2,remote link closed
AT$CLOSEUDPCLIENT=1                                  Close socket 1.

$CLOSEUDPCLIENT: 1,remote link closed
AT$CLOSEUDPCLIENT=1                                  No client is connected to socket 1.
ERROR
AT$CLOSEUDPCLIENT
$CLOSEUDPCLIENT: All remote link closed              All client sockets are closed.
```

9.17 $UDPRECV(S) – Notifying UDP Data Received from Clients

To notify UDP data received from clients.

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

V012 and the later versions support this command.

Format

Type	Response
Unsolicited result code	$UDPRECV(S): <socket>,<length>,<data>

Parameter

<socket> Socket ID
<length> length of the data received
<data> data received
Add **0x0d 0x0a** to the end of the data. We can identify the end based on <**length**>.

Example

```
$UDPRECV(S): 1,10,1234567899    Socket 1 receives 10-byte data in char format from the client.
```

9.18 AT$UDPSEND(S) – Transmitting Data to Client

To transmit data to the client after the module receives data from it.

Ensure that the UDP connection has been set up before sending UDP data.

The server can transmit data to a client only after receiving the data from the client.

V012 and the later versions support this command.

Format

Type	Command	Response

Execute	AT$UDPSENDS=<socket>[,<length>]<CR>	<CR><LF>> <CR><LF>OK<CR><LF> <CR><LF>$UDPSENDS:<socket>[,<length>]<CR><LF> Or <CR><LF>> <CR><LF>$UDPSENDS: Error!TimeOut<CR><LF> Or <CR><LF>$UDPSENDS: SOCKET ID NOT ACTIVE<CR><LF> Or <CR><LF>$UDPSENDS: DATA LENGTH ERROR<CR><LF> Or <CR><LF>$UDPSENDS: ERROR<CR><LF>

Parameter

<socket>	value of **AcceptSocket**, that is, the socket that the client sets up with the module.
<length>	length of the data to be sent, value ranges from **1** to **1024**, unit: byte.
<err>	ERROR SOCKET ID NOT ACTIVE DATA LENGTH ERROR

Example

```
AT$UDPSENDS=0,10                                    10-byte data is successfully sent through
>1234567890                                         socket 0.
OK

$UDPSENDS: 0,10
AT$UDPSENDS=0,10                                    No connection is set up on socket 0.
OK

$UDPSENDS: SOCKET ID NOT ACTIVE
AT$UDPSENDS=0,10                                    No UDP connection is set up on socket 0
$UDPSEND: ERROR                                     through listening
AT$UDPSENDS=0,10
>

$UDPSENDS: Error!TimeOut                            Data input times out
AT$UDPSENDS=0,5120                                  The length is incorrect.
$UDPSENDS: DATA LENGTH ERROR
```

9.19 AT$UDPCLIENTSTATUS – Querying Client Connection Status

To query the connection status of the client.

V012 and the later versions support this command.

Format

Type	Command	Response
Execute	AT$UDPCLIENTSTATUS=<socket><CR>	<CR><LF>$UDPCLIENTSTATUS: <socket>,<status>, <type> <CR><LF>OK<CR><LF>

Parameter

<socket>	value of **AcceptSocket**, that is, the socket that the client sets up with the module.
<status>	socket status, CONNECT or DISCONNECT
<type>	socket type, UDP or INVALID If the socket is invalid, it may be the listen socket of TCP/UDP client or server.

Example

```
AT$UDPCLIENTSTATUS=0                    A UDP connection is set up on the remote socket 0.
$UDPCLIENTSTATUS: 0,CONNECT,UDP
OK
AT$UDPCLIENTSTATUS=4                    No connection is set up on socket 4.
$UDPCLIENTSTATUS: 4,DISCONNECT
OK
AT$UDPCLIENTSTATUS=1                    The socket 1 is used for server listening.
$UDPCLIENTSTATUS: 1,CONNECT,INVAL ID
OK
```

9.20 AT$IPNETREAD – Reading UDP Data

To read UDP data that the server receives and saves in the buffer
If the <length> is greater than the actual length of the data, the module read the data of the actual length. The buffer for each socket is 10 KB.

V012 and the later versions support this command.

Format

Type	Command	Response
Execute	AT$IPNETREAD=<n>,<length><CR>	<CR><LF>$IPNETREAD: <n>,<len> <CR><LF><data> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	socket ID, ranging from 0 to 14
<len>	length of data to be read, ranging from 1 to 2048
<data>	Data that is read

Example

```
$IPURCREAD: 0                          Socket 0 receives data.
AT$IPNETREAD=0,5                       Read 5-byte data.
$IPNETREAD: 0,5                        The data read is 12345.
12345
OK
AT$IPNETREAD=0,1024                    No data received by socket 0 in the buffer.
$IPNETREAD: 0,0

OK
AT$IPNETREAD=1,10                      The socket is not set or the data is not set
ERROR                                  to receive manually.
```

10 FTP Commands

10.1 AT$MYFTPOPEN – Starting FTP Service

To start the FTP service

The module can set up only one FTP connection every time.FTP service and transparent transmissions are mutual exclusive. They cannot be used at the same time.

Format

Type	Command	Response
Execute	AT$MYFTPOPEN=<Channel>,<destination_ip/url:port>,<username>,<password>,<mode>,<Tout>,<FTPtype>[,<ip_type>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR: <err><CR><LF>
Query	AT$MYFTPOPEN?<CR>	<CR><LF>$MYFTPOPEN: <connection_status> <CR><LF>OK<CR><LF>
Test	AT$MYFTPOPEN=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<Channel>	The number of channels that have been activated. The FTP service is enabled based on channel type.
<destination_ip/url:port>	The IP address or URL of the remote FTP server. IP address is in format of XXX.XXX.XXX.XXX.The URL consists of at most 64-byte ASCII characters. If you do not set <port>, the default FTP port is 21.
<username>	User name to log in to the FTP server, ASCII characters, 32 bytes at most.
<password>	Password for the user name to log in to the FTP server, ASCII characters, 32 bytes at most.
<mode>	FTP mode. 0: active FTP mode; 1: passive FTP mode (default mode)
<Tout>	Timeout period of FTP command or data idle, ranging from 5 to 180. The default value is 30 and the unit is second.
<FTPtype>	FTP type. 0 indicates decimal mode (default); 1 indicates text mode.
<connection_status>	1: successful 0: failure

Example

```
AT$MYFTPOPEN=0,"neowaysvr.demo.net:21","neoway","neoway",1,30,1
OK
AT$MYFTPOPEN=0,"240e:980:9900::e1d:f8a9:21","csb004","wuyanjie123",1,30,1,1
OK
```

10.2 AT$MYFTPCLOSE – Closing the FTP Service

To close the FTP service

Format

Type	Command	Response
Execute	AT$MYFTPCLOSE<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

N/A.

Example

```
AT$MYFTPCLOSE
OK
```

10.3 AT$MYFTPSIZE – Obtaining the FTP File Size

To obtain the file length on the FTP service

Format

Type	Command	Response
Execute	AT$MYFTPSIZE=<File_Name><CR>	<CR><LF>$MYFTPSIZE:<File_length> <CR><LF>OK<CR><LF> Or

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

Parameter

<File_Name>	Name of the file to be downloaded, including file path
<File_length>	The FTP file length, ranging from 0 to 2^32-1

Example

```
AT$MYFTPSIZE="ftp_demo.txt"
$MYFTPSIZE:10
OK
```

10.4 AT$MYFTPGET – Downloading FTP Files

To download files from the FTP server

It is recommended that you use the default values of <data_offset> and <data_Length>.

Format

Type	Command	Response
Execute	AT$MYFTPGET=<File_Name>[,<data_offset>,<data_Length>]<CR>	<CR><LF>CONNECT <CR><LF><file_content> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<File_Name>	Name of the file to be downloaded.
<data_offset>	Offset of the file data
<data_Length>	The length of file data to be read.
<file content>	Content of the FTP file.

Example

```
AT$MYFTPOPEN=0,"neowaysvr.demo.net:21",   Start the FTP service.
"neoway","neoway",1,30,1
```

```
OK
AT$MYFTPGET="ftp_demo.txt"                Send the download command.
CONNECT
0123456789                                File content.
OK
                                          Report the disconnection state.
$MYURCFTP:1
```

10.5 AT$MYFTPPUT – Uploading Files to the FTP Server

To upload files to the FTP server

You can use +++ to exit FTP PUT mode.The FTP PUT command does not support upload starting from the interrupted point.

Format

Type	Command	Response
Execute	AT$MYFTPPUT=<File_Name>,<data_length>,<EOF><CR>	<CR><LF>CONNECT<CR><LF><file_content><CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<File_Name>	Name of the file name to be downloaded, including file path.
<data_length>	The length of data to be uploaded, ranging from 1 to 3072.
<EOF>	End of the file to be uploaded, integer type 0: The starting packet of the file or the intermediate packet. 1: The last packet of the file.
<file content>	Content of the file to be uploaded. If the entered file length is equal to the data_length configured in the command, the module will send the file data to the FTP server. OK is returned after all data is sent successfully. Otherwise, the module will return to the command mode and report ERROR.

Example

```
AT$MYFTPOPEN=0,"neowaysvr.demo.net:21","neoway"   Start the FTP service.
,"neoway",1,30,1
OK
                                                  Send the file upload command.
AT$MYFTPPUT="ftp_demo.txt",10,1
```

```
CONNECT                                          Enter the content.
ABCDEFGHIJ
OK                                               Report the disconnection state.

$MYURCFTP:1
```

10.6 $MYURCFTP – Notifying the FTP Disconnection Status

To notify the FTP connection status.

Format

Type	Command
Unsolicited result code	<CR><LF>$MYURCFTP:<Status><CR><LF>

Parameter

<Status> FTP connection status, integer type
0: Disconnection of FTP commands socket
1: Disconnection of FTP data socket

Example

```
AT$MYFTPOPEN=0,"neowaysvr.demo.net:21","neoway","neoway",1,30,1  Start the FTP service.
OK

$MYURCFTP:0                                                      Indicates the disconnection of
                                                                 FTP commands socket.
```

11 Network Application

11.1 AT+CIPGSMLOC – Obtaining the Location of the Module

To obtain the location information of the module.

The obtained data is GPS coordinates.

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)
<fail_string>	Failure string CONTACT FAIL LINK FAIL LINK NOT FREE
<result_string>	string including longitude and latitude
<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$MYNETACT=0,1
OK
$MYURCACT: 0,1,"183.42.232.191"
AT+CIPGSMLOC                                        Single-BS positioning request is executed
OK                                                  successfully in State Grid mode.

+CIPGSMLOC:                                         The module reports location coordinates.
{"location":{"lat":22.682403116613813,"lng":113
.99042272056249},"accuracy":0.0}
+CIPGSMLOC: OK
AT+CIPGSMLOC                                        PPP connection is not established.
GPRS DISCONNECTION
AT+CIPGSMLOC                                        The server domain name fails to be
+CIPGSMLOC: CONTACT FAIL                            translated.
AT+CIPGSMLOC                                        The connection to the server fails to be set
+CIPGSMLOC: LINK FAIL                               up.
AT+CIPGSMLOC
OK                                                  The location request is sent successfully,
                                                    but the queried BS is not included.
+CIPGSMLOC: 404
+CIPGSMLOC: FAIL
AT+CIPGSMLOC=0                                      Close GPS positioning. "OK" is returned.
OK
AT$MYNETACT=0,1
OK
$MYURCACT: 0,1,"10.151.44.13"
AT+CIPGSMLOC=1                                      Request multi-BS positioning in standard mode
OK                                                  is executed successfully.

+CIPGSMLOC:                                         The module reports its location coordinates.
{"location":{"lat":22.682403116613813,"lng":113
.99042272056249},"accuracy":0.0}
+CIPGSMLOC: OK
```


11.2 AT$MYTIMEUPDATE – Network Time Update

To update the module time to the network time.

- It is an internal protocol command. The settings by this command will not be saved after the module is powered off.
- You must enable PPP link (AT$MYNETACT) before sending this command.
- You can send AT+CCLK? to query whether RTC is synchronized to the current network time after this command is sent successfully.

The following time servers support time update: time.windows.com, time.nist.gov, etc.

Format

Type	Command	Response
Execute	AT$MYTIMEUPDATE=<url><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYTIMEUPDATE?<CR>	<CR><LF>$MYTIMEUPDATE:<state> <CR><LF>OK<CR><LF>
Test	AT$MYTIMEUPDATE=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<url>	The settings by this command will not be saved after the module is powered off.
<err>	The settings by this command will not be saved after the module is powered off.
<state>	0: The module time is not synchronized after the module is powered on. 1: The module time is synchronized after the module is powered on.

Example

```
AT$MYTIMEUPDATE=""                          Synchronize with the Network Time of the default
OK                                          server.
AT$MYTIMEUPDATE="time.nist.gov"             Synchronize with the network time of
OK                                          time.windows.com
AT$MYTIMEUPDATE="time.nist.gov"             Synchronize with the network time.
ERROR:901                                   PDP is not activated.
AT$MYTIMEUPDATE?                            Check whether the time is synchronized after
$MYTIMEUPDATE:1                             powering on.
OK                                          Time is synchronized.
AT$MYTIMEUPDATE=?
OK
AT+CCLK?                                    Read time before synchronization via standard
+CCLK:"05/01/01,00:00:45"                   command
```

```
OK

AT$MYNETCON=0,"APN","CMNET"
OK

AT$MYNETURC=1
OK

AT$MYNETACT=0,1
OK
$MYURCACT:0,1,"10.141.49.251"
AT$MYTIMEUPDATE="time.nist.gov"            Synchronize network time
OK
AT+CCLK?                                   Read time
+CCLK: "14/05/07,15:31:36"
OK
```

11.3 AT+NCUSTSWITCH – Controlling Extended Functions

To enable of disable extended functions

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

Execute this command before the TCP connection is established.

The type parameter can be set to 1, 3, and 4 by now. Other values are for subsequent extended functions.

Format

Type	Command	Response
Execute	AT+NCUSTSWITCH=<typeX>,<modeX><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NCUSTSWITCH?<CR>	<CR><LF>+NCUSTSWITCH: <type1>,<mode1> <CR><LF>OK<CR><LF>
Test	AT+NCUSTSWITCH=?<CR>	<CR><LF>+NCUSTSWITCH: (value range of <type>) <CR><LF>OK<CR><LF>

Parameter

<typeX> Extended function
1: TCPnagle Algorithm switch (To solve the sticky packet issue during TCP transmissions)
3: TCP sub-packet reading
4: Configure the wake-up source
Currently, only type3 and type4 are supported.

<modeX> Frequency band supported.
Mode1:
0 disable the TCP Nagle Algorithm
1 disable the TCP Nagle Algorithm (default)
Mode3:
0: sub-packet reading (default)
1: Automatic packet combination
Mode4:
Bit[X]=1, enable
Bit[X]=0, disable

Flag	Bit[15~3]	Bit[2]	Bit[1]	Bit[0]
Ring report type	Undefined	DATA	CALL	SMS
Enable	1	1	1	1
Disable	0	0	0	0

0x0001 enable SMS status indication
0x0002 enable CALL status indication
0x0003 enable CALL+SMS status indication
0x0004 enable DATA status indication (DATA is only used for the internal protocol stack services.)
0x0005 enable DATA+SMS status indication
0x0006 enable DATA+CALL status indication
0x0007 enable DATA+CALL+SMS status indication (default)

Example

```
AT+NCUSTSWITCH=1,0                          Disable the Nagle algorithm.
OK
AT+NCUSTSWITCH?                             The TCP Nagle algorithm is disabled.
+NCUSTSWITCH: 1,0                           The data packets are read after merged.
+NCUSTSWITCH: 3,1
OK
AT+NCUSTSWITCH=?                            Currently, only type1 is supported.
+NCUSTSWITCH: (1-99)
OK
```


12 SSL Certificate

12.1 AT+CERTADD – Writing an 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>

Parameter

<file_name>	Certificate file name
<length>	file length

Example

```
AT+CERTADD=ca_cert.pem,1428                    Add the alipay.crt certificate of 1428 bytes
CONNECT                                        to the module.

+CERTADD: 1428,OK
```

12.2 AT+CERTCHECK – Checking the SSL Certificate

To check the SSL certificate.

Format

Type	Command	Response
Execute	AT+CERTCHECK=<file_name><CR>	<CR><LF>+CERTCHECK: <file_name>,OK<CR><LF> Or <CR><LF>+CERTCHECK: ERROR<CR><LF>
Query	AT+CERTCHECK?<CR>	<CR><LF><file_name><CR><LF> …… <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+CERTCHECKT?                              Check the added file.
cacert.pem
keycert.pem
OK
```

12.3 AT+CERTDEL –Deleting an SSL Certificate

To delete an SSL certificate.

Format

Type	Command	Response
Set	AT+CERTDEL[=<file_name>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT+CERTDEL<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<file_name>	Certificate file name to be deleted.

Example

```
AT+CERTDEL=ca_cert.pem                          Delete the ca_cert.pem certificate.
OK
AT+CERTDEL                                      Delete all the added certificates.
OK
```

13 HTTP/HTTPS Commands

13.1 AT+HTTPPARA – Setting HTTP Parameters

To set HTTP parameters

Format

Type	Command	Response
Set	AT+HTTPPARA=<para>,<para_value>[,<ip_type>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<para>	HTTP parameters, supporting the following two parameters: url: Destination path port: Destination port ID (no default value) keepalive: set long-time connection recvmode: receive mode timeout: timeout period, the timeout parameter is supported by part of the variants.
<para_value>	The value of **<para>**. The value of url contains at most 512 bytes and url supports domain name translation.
<ip_type>	IP type, when <para>=URL, <ip_type> is IPv4 by default. IPv4: IPv4 connection IPv6: IPv6 connection

Example

```
AT+HTTPPARA =url,www.neoway.com.cn/en/index.aspx          Set the Neoway homepage as the URL. The
OK                                                        URL supports domain name translation.
AT+HTTPPARA=url,121.15.200.97/Service1.asmx/GetNote       Set URL.
OK
AT+HTTPPARA=url,                                          The AT command is not complete.
ERROR
AT+HTTPPARA=port,80                                       Set the destination port ID to 80.
OK
AT+HTTPPARA=port,8080                                     Set the destination port ID to 8080.
OK
```

```
AT+HTTPPARA=recvmode,1                          Set the received data to display in the
OK                                              +HTTPRECV: <length>,<data> format
AT+HTTPPARA=timeout,60                          Set the time-out period to 60s.
OK
```

13.2 AT+HTTPSETUP – Setting up an HTTP Connection

To set up an HTTP connection

The connection is set up successfully only after setting the destination address and port ID correctly. Ensure that a network connection has been set up successfully before setting an HTTP connection.

Format

Type	Command	Response
Execute	AT+HTTPSETUP<CR>	<CR><LF>OK<CR><LF> <CR><LF>+HTTPSETUP OK<CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+HTTPSETUP: <result><CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<result> ERROR: DNS resolution fails
FAIL: fail to establish an HTTP connection.

Example

```
AT+HTTPSETUP                        Set up an HTTP connection
OK                                  Successful

HTTP SETUP OK
AT+HTTPSETUP
OK                                  Set up an HTTP connection.
                                    Failed: DNS resolution fails.
+HTTPSETUP: ERROR
AT+HTTPSETUP                        Set up an HTTP connection.
ERROR                               Failed: no port is set before the HTTP connection is set up.
AT+HTTPSETUP
OK                                  Fail to set up an HTTP connection.
```

```
+HTTPSETUP: FAIL

AT+HTTPSETUP
OK
                                    Incorrect parameters.

+HTTPCLOSE: Link Closed
```

13.3 AT+HTTPACTION – Initiating an HTTP Request

To initiate an HTTP request

Comply with the HTTP protocol when defining packets.

Add a carriage return to the end of the packets if the HTTP request is set to custom packet mode.

Format

Type	Command	Response
Execute	AT+HTTPACTION=<mode>[,<length>[,<type>]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>><post_content><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+HTTPACTION: SOCKET ID OPEN FAILED<CR><LF>

Parameter

<mode>	HTTP request mode, available value can be 0, 1, 2, 99 0: GET 1: HEAD 2: POST 99: OPEN_MODE, custom packet mode
<length>	POST content length or custom packet length; mandatory when <mode> is set to POST or OPEN_MODE, 2048 at most.
<type>	data type of POST request 0: x-www-form-url encoded 1: text 2: json 3: xml 4: html

<offset>	Offset in GET mode
<size>	Size of file to be downloaded in GET mode
<post_content>	Content sent through HTTPPOST

Example

```
AT+HTTPPARA=url,www.neoway.com.cn/en/index.aspx                Set the destination path.
OK                                                             Set up an HTTP connection.
AT+HTTPPARA=port,80                                            Set the port to 80.
OK

AT+HTTPSETUP                                                   Set up an HTTP connection.
OK                                                             Get request

                                                               Successful
HTTP SETUP OK
AT+HTTPACTION=0
OK

                                                               Receive the response from the
+HTTPRECV:                                                     HTTP server.
HTTP/1.1 200 OK
Cache-Control: private
Content-Type: text/html; charset=utf-8
Server: Microsoft-IIS/7.5
Set-Cookie: ASP.NET_SessionId=rh3fjg554ufzb145aevgzz45;
path=/; HttpOnly
X-AspNet-Version: 2.0.50727
X-Powered-By: ASP.NET
X-UA-Compatible: IE=EmulateIE7
Date: Wed, 02 Mar 2016 06:52:35 GMT
Connection: close
Content-Length: 13842
                                                               The server finishes the response
                                                               and disconnects the connection.
/*neoway homepage, html format, 13842 bytes*/
……..
/* neoway homepage*/
+HTTPCLOSED: HTTP Link Closed


AT+HTTPPARA =url,www.neoway.com.cn/en/index.aspx               Set the destination path.
OK
AT+HTTPPARA=port,80
OK
AT+HTTPSETUP
OK

                                                               Set up an HTTP connection
HTTP SETUP OK                                                  HEAD request
AT+HTTPACTION=1
OK

                                                               The HTTP server responds.
+HTTPRECV:
HTTP/1.1 200 OK
Cache-Control: private
Content-Length: 13842
Content-Type: text/html; charset=utf-8
Server: Microsoft-IIS/7.5
```



```
Set-Cookie: ASP.NET_SessionId=znt4fqabqsuclz55pvfufn55;
path=/; HttpOnly
X-AspNet-Version: 2.0.50727
X-Powered-By: ASP.NET
X-UA-Compatible: IE=EmulateIE7
Date: Thu, 28 Nov 2013 03:32:35 GMT
Connection: close


+HTTPCLOSED: Link Closed
AT+HTTPPARA=url,121.15.200.97/Service1.asmx/GetNote          Set destination path
OK
AT+HTTPPARA=port,8080                                        Set the destination port ID as
OK                                                           8080.
AT+HTTPSETUP                                                 Set up an HTTP connection
OK
AT+HTTPACTION=2,23                                           POST request.
>MAC=NEOWAY&DATA=0123456                                     Send 23 bytes; enter the contents
OK                                                           to be uploaded after > is
                                                             displayed.
+HTTPRECV:
HTTP/1.1 200 OK                                              Receive the response from the
Cache-Control: private, max-age=0                            HTTP server.
Content-Type: text/xml; charset=utf-8
Server: Microsoft-IIS/7.5
X-AspNet-Version: 4.0.30319                                  The server replies an XML file
X-Powered-By: ASP.NET                                        containing the uploaded content
Date: Thu, 28 Nov 2013 03:41:52 GMT                          NEOWAY and 0123456.
Connection: close
Content-Length: 98

<?xml version="1.0" encoding="utf-8"?>                       The server disconnected with the
<string xmlns="http://wsliu.cn/">NEOWAY+0123456              module after it finished
</string>                                                    responding.

+HTTPCLOSED: HTTP Link Closed
AT+HTTPPARA=url,www.neoway.com.cn/en/index.aspx              Set destination path
OK
AT+HTTPSETUP                                                 The HTTP connection is set up
OK                                                           through port 80.
AT+HTTPACTION=99,76
>HEAD /en/index.aspx HTTP/1.1
connection: close
HOST: www.neoway.com.cn
                                                             Send 76-byte user-defined packets
OK

+HTTPRECV:
HTTP/1.1 200 OK
Cache-Control: private                                       Receive the response from the
Content-Length: 13842                                        HTTP server.
Content-Type: text/html; charset=utf-8
Server: Microsoft-IIS/7.5
Set-Cookie: ASP.NET_SessionId=pvlaai3fizxg44eyvyqsyenk;
path=/; HttpOnly
```

```
X-AspNet-Version: 2.0.50727
X-Powered-By: ASP.NET
X-UA-Compatible: IE=EmulateIE7
Date: Thu, 28 Nov 2013 05:40:24 GMT                    The server disconnects with the
Connection: close                                      module after it finishes
                                                       responding.

+HTTPCLOSED: Link Closed
AT+HTTPACTION=0                                        PPP is not enabled or SOC
+HTTPACTION: SOCKET ID OPEN FAILED                     connection encountered an error.
AT+HTTPACTION=0                                        Failed to send data.
+HTTPSETUP: ERROR
AT+HTTPACTION=2,adasd                                  Other errors
ERROR
```

13.4 AT+HTTPCLOSE – Closing an HTTP Socket

To close an HTTP socket

After the **+HTTPCLOSE** command is sent, the HTTP socket is closed and the setting of +HTTPPARA is cleared.

Only **OK** is returned after running this command if the HTTP socket is not connecting.

Format

Type	Command	Response
Execute	AT+HTTPCLOSE<CR>	<CR><LF>OK<CR><LF> [<CR><LF>+HTTPCLOSE: OK]

Parameter

N/A.

Example

```
AT+HTTPCLOSE                    Close the HTTP socket.
OK

+HTTPCLOSE: HTTP Link Closed    URC notifies the socket is closed successfully.
AT+HTTPCLOSE                    Only "OK" is returned because the socket has been closed already.
OK
```

13.5 +HTTPRECV – Notifying HTTP Data Received

To notify the received HTTP data.

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

Format

Type	Command
Unsolicited result code	<CR><LF>+HTTPRECV: <datas><CR><LF>

Parameter

<datas> Data received through the HTTP socket

Example

```
+HTTPRECV: HTTP/1.1 200 OK
Cache-Control: private
Content-Length: 13842
Content-Type: text/html; charset=utf-8
Server: Microsoft-IIS/7.5
Set-Cookie: ASP.NET_SessionId=pvlaai3fizxg44eyvyqsyenk; path=/;
HttpOnly                                                    URC notifies the received
X-AspNet-Version: 2.0.50727                                 HTTP data.
X-Powered-By: ASP.NET
X-UA-Compatible: IE=EmulateIE7
Date: Thu, 28 Nov 2013 05:40:24 GMT
Connection: close

+HTTPCLOSE: Link Closed
```

13.6 +HTTPCLOSED – Notifying HTTP Socket Closed

To notify that the HTTP socket is closed.

Format

Type	Command
Unsolicited result code	<CR><LF>+HTTPCLOSE: Link Closed<CR><LF>

Parameter

N/A

Example

```
+HTTPCLOSED: Link Closed                    the HTTP socket is closed.
```

13.7 AT+HTTPSCFG – Configuring SSL Parameters for HTTPS

To configure SSL parameters for HTTPS.

Format

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

Parameter

<type>	SSL options **sslversion**: SSL version **authmode**: authentication mode **ciphersuite**: Cipher suite **cacert**: CA certificate **clientcert**: Client certificate **clientkey**: Client key
<type_name>	setting of 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
ciphersuites: Cipher suite
Standard value of the TLS cipher suite, hexadecimal
0XFFFF: all suites the module supports (default)
Cacert: string type, CA certificate
Clientcert: string type, client certificate
Clientkey: string type, client key 443

Example

```
AT+HTTPSCFG =”sslversion”,0                    Set SSL version to ssl3.0.
OK
AT+HTTPSCFG =”authmode”,0                      Set no authentication.
OK
AT+ HTTPSCFG?
+HTTPSCFG: 0,1,,ca.pem,cc.pem,ck.pem           Query current SSL settings.
OK
AT+HTTPSCFG=?
+HTTPSCFG: <type>,<type_name>                  Query the value range of parameters.
OK
```

13.8 AT+HTTPSPARA – Setting HTTPS Parameters

To set HTTPS parameters.

Set new HTTPS parameters for new HTTPS requests.

After the **+HTTPSCLOSE** command is sent, the connection is closed and parameter settings will be cleared.

Format

Type	Command	Response
Execute	AT+HTTPSPARA=<para>,<para_value><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<para>	HTTPS parameters, supporting the following two parameters: url: Destination path port: Destination port ID, 443 when omitted. keepalive: set long-time HTTP connection
<para_value>	The value of **<para>**. The value of **url** contains at most 512 bytes and **url** supports domain name translation. para_value=0 default mode, only one +HTTPRECV: header is included in one HTTP response para_value=1 (long connection is enabled) data is displayed in format of +HTTPRECV: <length>,<data>
<ip_type>	IP type, when <para>=URL, <ip_type> is IPv4 by default. IPv4: IPv4 connection IPv6: IPv6 connection

Example

```
AT+HTTPSPARA=url,www.alipay.com/index.html          Set the alipay homepage as the URL. The URL
OK                                                  supports domain name translation.
AT+HTTPSPARA=url,support.cdmatech.com/index.html    Set URL.
OK
AT+HTTPSPARA=url,
ERROR                                               incomplete parameters
AT+HTTPSPARA=port,443                               Set the destination port ID to 443.
OK
AT+HTTPSPARA=port,443
OK                                                  Set the destination port to 443
AT+HTTPSPARA=url,www.taobao.com,ipv6                 Set the IPv6 URL
```

13.9 AT+HTTPSSETUP – Setting up an HTTPS Connection

To set up an HTTPS connection.

Format

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

Parameter

N/A.

Example

```
AT+HTTPSSETUP                                   Set up an HTTPS connection
OK                                              Successful

HTTPS SETUP OK
AT+HTTPSSETUP                                   Set up an HTTPS connection
+HTTPSETUP: ERROR                               Failed.
```

13.10 AT+HTTPSACTION – Executing an HTTPS Request

To execute an HTTPS request.

Comply with the HTTP protocol when defining packets.

When customizing the message, pay attention to whether the tool used comes with the carriage return and line feed characters.

Format

Type	Command	Response
Execute	AT+HTTPSACTION=<mode>[,<length>[,<type>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>><post_content><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode>	HTTPS request mode, available value can be 0, 1, 2, 99 0: GET 1: HEAD 2: POST 99: OPEN_MODE, custom packet mode
<length>	POST content length, ranging from 1 to 2048; or custom packet length when **<mode>** is set to **POST** or **OPEN_MODE**
<type>	data type of POST request 0: x-www-form-urlencoded 1: text

	2: json 3: xml 4: html
<post_content>	HTTP content sent in POST method

Example

```
AT+HTTPSPARA=url,support.cdmatech.com/login/                    Set the destination path.
OK
AT+HTTPSPARA=port,443                                           Set the destination port.
OK
AT+HTTPSSETUP                                                   Set up an HTTPS connection.
OK

HTTPS SETUP OK
AT+HTTPSACTION=0
OK
+HTTPSRECV:
HTTP/1.1 200 OK                                                 GET request.
Server: QUALCOMM
X-Powered-By: Servlet/2.5 JSP/2.1                               Receive the request from the HTTPS
Content-Type: text/html; charset=ISO-8859-1                     server.
Date: Sat, 15 Feb 2014 05:58:54 GMT
Content-Length: 7630
Connection: close
Set-Cookie:
JSESSIONID=8V1dS1CpzlPcyNl2LzJZLQgDxWclpMJzP3FHZhVhpGb8         The server responded and then
3GVM02sn!1955538012; path=/; HttpOnly                           disconnected the connection.
/*homepage, html format*/
……..
/*homepage*/
+HTTPSCLOSED: HTTPS Link Closed
AT+HTTPSPARA=url,support.cdmatech.com/login/                    Set the destination path.
OK
AT+HTTPSPARA=port,443                                           Set the destination port.
OK
AT+HTTPSSETUP                                                   Set up an HTTPS connection.
OK
AT+HTTPSACTION=1
OK
+HTTPSRECV:                                                     HEAD request
HTTP/1.1 200 OK
Server: QUALCOMM                                                The HTTPS server responds.
X-Powered-By: Servlet/2.5 JSP/2.1
Content-Type: text/html; charset=ISO-8859-1
Date: Sat, 15 Feb 2014 06:05:39 GMT
Content-Length: 0
Connection: close
Set-Cookie:
JSESSIONID=qyNVS1DSmnjS9cvh72yW1xz1jtjBBRj0yv0zTmMy2LVy
BG7HK02b!1955538012; path=/; HttpOnly

+HTTPSCLOSED: HTTPS Link Closed
AT+HTTPSPARA=url,mybank.icbc.com.cn/icbc/perbank/index.         Set the destination path.
jsp
```

```
OK                                                          Set the port.
AT+HTTPSPARA=port,443                                       Set up an HTTPS connection.
OK
AT+HTTPSSETUP
OK

HTTPS SETUP OK
AT+HTTPSACTION=99,69
>HEAD /index.html HTTP/1.1                                 Send 69-byte custom packets.
HOST:www.alipay.com
connection: close                                           The HTTPS server responded.

OK

+HTTPSRECV:                                                 The server closed the link after
HTTP/1.1 200 OK                                             responding.
Server: spanner/1.0.6
Date: Sat, 02 Aug 2014 06:06:21 GMT
Content-Type: text/html; charset=gbk
Content-Length: 56059
Connection: close
Last-Modified: Fri, 01 Aug 2014 07:45:49 GMT
Strict-Transport-Security: max-age=31536000
Accept-Ranges: bytes
Set-Cookie:
spanner=LBKsxiiZAaTeM3wRYcCaUtMjpheSwnH+;path=/;secure;


+HTTPSCLOSE: Link Closed
```

13.11 AT+HTTPSCLOSE – Closing an HTTPS Socket

To close an HTTPS socket.

After the **+HTTPSCLOSE** command is sent, the HTTPS socket is closed and the setting of **+HTTPPARA** is cleared.

Format

Type	Command	Response
Execute	AT+HTTPSCLOSE<CR>	<CR><LF>OK<CR><LF> <CR><LF>+HTTPSCLOSE: Link Closed<CR><LF>

Parameter

N/A.

Example

```
AT+HTTPSCLOSE                                   Close the HTTPS socket.
OK

+HTTPSCLOSE: HTTPS Link Closed
```

13.12 +HTTPSCLOSED – Notifying HTTP Socket Closed

To notify the HTTPS socket that is closed.

Format

Type	Command
Unsolicited result code	<CR><LF>+HTTPSCLOSE: Link Closed<CR><LF>

Parameter

N/A

Example

```
+HTTPCLOSED: HTTP Link Closed              Unsolicited result code of the HTTP socket closing
```

14 ECM Commands

14.1 AT$MYUSBNETACT – Activating/Deactivating the Network Connection

To activate or deactivate the network connection.

Format

Type	Command	Response
Set	AT$MYUSBNETACT=<channel>,<action><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR: <code><CR><LF>
Query	AT$MYUSBNETACT?<CR>	<CR><LF>$MYUSBNETACT: <channel>,<status>,<ip> <CR><LF>OK<CR><LF>
Test	AT$MYUSBNETACT=?<CR>	<CR><LF>$MYUSBNETACT: (list of supported <channel>s),(list of supported <action>s) <CR><LF>OK<CR><LF>

Parameter

<action> 0: deactivate network connection
1: activate network connection
<channel> channel number (0-5)
<status> 0: deactivated
1: activated
<IP> IP address
Display the local IP after the network is activated.

Example

```
AT$MYUSBNETACT=0,1                                  The address is 10.223.204.160
OK
$MYURCUSBACT: 0,1,"10.223.204.160"
```

```
AT$MYUSBNETACT?                                          Query the local IP address
$MYUSBNETACT: 0,1,"10.223.204.160"
OK
AT$MYUSBNETACT=0,1                                       "ERROR" is displayed after Repeat
OK                                                       activation.
AT$MYUSBNETACT=0,1
ERROR: 902
AT$MYUSBNETACT?                                          When IPv4v6 is used.
$MYUSBNETACT: 0,0,"0.0.0.0"
$MYUSBNETACT: 0,0,"0:0:0:0:0:0:0:0"
AT$MYUSBNETACT=0,1
OK
AT$MYUSBNETACT?
$MYUSBNETACT: 0,1,"10.195.120.239"
$MYUSBNETACT:0,1,"240e:454:2bc:77cd:8df5:f1c4:d6f2:e3fe"
```

14.2 AT$MYUSBNETADDR – Setting USB Network Address

To set the USB network address.

Format

Type	Command	Response
Set	AT$MYUSBNETADDR=<network address>,<subnet mask><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYUSBNETADDR?<CR>	<CR><LF>$MYUSBNETADDR: <network address>,<subnet mask> <CR><LF>OK<CR><LF>
Test	AT$MYUSBNETADDR=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<network address> Network address, character type. Format: XXX.XXX.XXX.XXX.
<subnet mask> Subnet mask, character type. Format: XXX.XXX.XXX.XXX.

Example

```
AT$MYUSBNETADDR=192.168.53.1,255.255.255.0    Set the USB network address.
```

```
OK
AT$MYUSBNETADDR?                                Query the current settings.
$MYUSBNETADDR:"192.168.53.1","255.255.255.0"
OK
```

14.3 AT$MYUSBNETURC – Switch of the URC

Switch of the URC.

Format

Type	Command	Response
Set	AT$MYUSBNETURC=<onoff><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYUSBNETURC?<CR>	<CR><LF>$MYUSBNETURC: <onoff> <CR><LF>OK<CR><LF>
Test	AT$MYUSBNETURC=?<CR>	<CR><LF>$MYUSBNETURC: (list of supported <onoff>s) <CR><LF>OK<CR><LF>

Parameter

<onoff> Integer
0: off
1: on (default)

Example

```
AT$MYUSBNETURC=1                                Enable the "MYUSBNETURC" URC
OK
AT$MYUSBNETURC?                                 Query the current setting.
$MYUSBNETURC: 1
OK
```

14.4 $MYURCUSBACT – Notifying the Current Network Status

Format

Type	Command
Unsolicited result code	<CR><LF>$MYURCUSBACT: <channel>,<type><CR><LF>

Parameter

<channel> Channel number
<type> Network status
0: disconnected
1: connected

Example

```
$MYURCUSBACT: 0,0                    The network is disconnected.
```

14.5 AT$MYUSBIPMODE – Controling the Network IP Mode

To control the network IP mode.

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

Format

Type	Command	Response
Execute	AT$MYUSBIPMODE=<ip_mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYUSBIPMODE?	<CR><LF>$MYUSBIPMODE: <ip_mode><CR><LF> <CR><LF>OK<CR><LF>
Test	AT$MYUSBIPMODE=?	<CR><LF>$MYUSBIPMODE: (list of

	supported <ip_mode>s) <CR><LF>OK<CR><LF>

Parameter

<ip_mode> Integer
0: non-transparent IP mode (default)
the computer that controls the module obtains the private network IP address.
1: transparent IP mode
the computer that controls the module obtains the air-interface IP address.

Example

```
AT$MYUSBIPMODE=1                          Set the IP mode to non-transparent IP mode
OK
AT$MYUSBIPMODE?                           Query the network IP mode.
$MYUSBIPMODE: 0
OK
AT$MYUSBIPMODE=?                          Query the parameter range.
$MYUSBIPMODE: (0-1)
OK
```

15 FOTA Update

15.1 AT+NEOFTPFOTA – FTP FOTA Upgrade

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

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

This function only supported by some variants of the N723-EA series.

Format

Type	Command	Response
Execute	AT+NEOFTPFOTA=<url>,<port>,<user>,<pwd>,<pkt_name><CR>	<CR><LF>OK<CR><LF> <CR><LF>+NEODOWNLOAD: SUCCESS<CR><LF> <CR><LF>+NEOUPDATE: START<CR><LF> <CR><LF>+NEOUPDATE: <result1><CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+NEODOWNLOAD: <result2><CR><LF>
Query	AT+NEOFTPFOTA?<CR>	<CR><LF>+NEODOWNLOAD: <download progress> <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
<result1>	SUCCESS: upgrade is successful. FAIL: upgrade is failed.
<result2>	FAIL: the upgrade fails to be downloaded or fail to query the download result of the upgrade package.

Example

```
AT+NEOFTPFOTA=58.60.184.213,12008,neoway,neoway,update-V013C-V013B.zip   Start to perform an upgrade.
OK                                                                       The upgrade package is downloaded successfully.
+NEODOWNLOAD: SUCCESS
                                                                         Upgrade starts
+NEOUPDATE: START
                                                                         Firmware is upgraded successfully.
+NEOUPDATE: SUCCESS
AT+NEOFTPFOTA?
+NEODOWNLOAD: 21%                                                        Query the current download progress.
OK
```

16 Power Management

16.1 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.

- After the sleep mode is enabled, the module can enter the sleep mode only after the **DTR** signal is externally driven to **low (or high)**, and circuits inside the module are allowed to enter the sleep mode.
- When the DTR signal of the module is driven to the high (or low) level, the module restores its normal working state.

Do not use this command together with +CFUN.

Format

Type	Command	Response
Set	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.
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                          Set the module to allow sleep mode.
OK
AT+ENPWRSAVE?                           Query current sleep mode status.
+ENPWRSAVE:1
```

```
OK
```

16.2 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
```

17 Hardware Commands

17.1 AT$MYSOCKETLED – Controlling the Indicator of Socket Status

To control the indicator of socket status.

This command is used to control the LED indicator when the socket is connected complying with external protocol stack.

Format

Type	Command	Response
Execute	AT$MYSOCKETLED=<ONOFF><CR>	<CR><LF>OK<CR><LF>

Parameter

<ONOFF>	Indicates the socket status, integer type. 0: The LED status when the socket is not connected. 1: The LED status when the socket is connected.

Example

```
AT$MYSOCKETLED=1
OK
```

17.2 AT$MYADCTEMP – Querying Environment Temperature

To query the environment temperature. This command is not supported yet.

Refer to Neoway temperature sensing scheme.

Format

Type	Command	Response
Execute	AT$MYADCTEMP=<mode><CR>	<CR><LF>$MYADCTEMP: <temp>,<volt> <CR><LF>OK<CR><LF> Or <CR><LF>$MYADCTEMP: OVL <CR><LF>OK<CR><LF> Or <CR><LF>$MYADCTEMP: OVH <CR><LF>OK<CR><LF>

Parameter

<mode>	Selection of the temperature sensing scheme. 0: temperature and voltage at ADC port 1: ADC value
<temp>	Temperature, °C
<volt>	Voltage, mV OVL: The temperature exceeds the lower threshold -40°C OVH: The temperature exceeds the higher threshold 95°C

Example

```
AT$MYADCTEMP=0
$MYADCTEMP:10,350
OK
```

17.3 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 off.

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),(list of supported <rst>s) <CR><LF>OK<CR><LF>

Parameter

<fun> power saving function mode
0: turn off radio and SIM power
1: Full functionality (default)
<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
OK
AT+CFUN?
+CFUN: 1
OK
AT+CFUN=?
+CFUN: (0,1),(0-1)
OK
```

18 LWM2M Function Commands

This chapter lists the AT commands ralated to the LWM2M function and it is only supported by some variants of the N723 series.

18.1 AT+NLWSERV – LWM2M Server Parameter Configuration

To configure the URL and port of the connected LWM2M server.

Format

Type	Command	Response
Execute	AT+NLWSERV=<url>,<port><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NLWSERV?<CR>	<CR><LF>+NLWSERV: <url>,<port><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+NLWSERV=?<CR>	<CR><LF>+NLWSERV: <url>,<port><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<url> URL of the connected LWM2M server.
<port> Port of the connected LWM2M server.

Example

```
AT+NLWSERV=58.60.184.213,12009                  Configure the URL and port of the connected LWM2M
OK                                              server.
AT+NLWSERV?
+NLWSERV: "58.60.184.213",12009                 Configure parameters of the connected LWM2M server.

OK
AT+NLWSERV=?                                    Query the parameter ranges.
+NLWSERV: <url>,<port>
```

```
OK
```

18.2 AT+NLWCONF – Configuring the LWM2M Connection Parameters

To configure the LWM2M connection parameters.

Format

Type	Command	Response
Execute	AT+NLWCONF=<endpointname>,<lifetime><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NLWCONF?<CR>	<CR><LF>+NLWCONF: <endpointname>,<lifetime> <CR><LF>OK<CR><LF>
Test	AT+NLWCONF=?<CR>	<CR><LF>+NLWCONF: <endpointname>[,<lifetime>]<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<endpointname> Endpoint name of the device.
<lifetime> Lifetime of the device connection.

Example

```
AT+NLWCONF=neo_lwm2m_client2,60                  Configure the LWM2M connection parameters.
OK
AT+NLWCONF?
+NLWCONF: "lwm2m_client2",60                     Query the set parameters.

OK
AT+NLWCONF=?
+NLWCONF: <endpointname>[,<lifetime>]            Query the parameter ranges.

OK
```

18.3 AT+NLWDTLSCFG – Configuring the LWM2M Encrypted Connection Parameters

To configure the LWM2M encrypted connection parameters.

When adding a certificate for encryption, you must first use AT+CERTADD to add the certificate to the module.

Format

Type	Command	Response
Execute	AT+NLWDTLSCFG=<dtls_mode>[,dtls_val,…]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NLWDTLSCFG?<CR>	<CR><LF>+NLWDTLSCFG: <dtls_mode>[,dtls_val,…]<CR><LF> <CR><LF>ERROR<CR><LF>
Test	AT+NLWDTLSCFG=?<CR>	<CR><LF>+NLWDTLSCFG: <dtlsmode>[,<cert_val1>,<cert_val2>,<cert_val3>]<CR><LF><CR><LF>OK<CR><LF>

Parameter

<dtls_mode> Encrypted connection mode.
0: Pre-Shared Key mode
1: Raw Public Key mode
2: Certificate mode
3: NoSec mode (default)

<dtls_val> Parameters required for the encryption mode.
When dtls_mode = 0:
AT+NLWDTLSCFG =0,<psk_identity>,<psk _key>
Where, psk _key must be in HEX format.
When dtls_mode = 1:
AT+NLWDTLSCFG =1,<cli_prikey>,<cli_pubkey>,<ser_pubkey>
It is currently not supported.
When dtls_mode = 2:
AT+NLWDTLSCFG =2,<cli_prikey>,<cli_cert>,<ser_cert>,
<cli_prikey>: client private key name.
<cli_cert>: client certificate name.
<ser _cert>: server certificate name.
The certificate is added by the AT+CERTADD command.
When dtls_mode = 3:

AT+NLWDTLSCFG =3

Example

```
AT+NLWDTLSCFG=0,public_pn,3031323334                Configure PSK encryption for the LWM2M DTLS
OK                                                  connection.
AT+NLWDTLSCFG=2,clikey,clicert,sercert              Configure certificate encryption for the
OK                                                  LWM2M DTLS connection.
AT+NLWDTLSCFG?
+NLWDTLSCONF: 2,"clikey","clicert","sercert"

                                                    Query the LWM2M DTLS connection parameters.
OK
AT+NLWDTLSCFG=?
+NLWDTLSCFG:
<dtlsmode>[,<cert_val1>,<cert_val2>,<cert_val3>]    Query the parameter ranges.

OK
```

18.4 AT+NLWADDOBJ – Adding an LWM2M Object

To add an LWM2M object.

The three objects, including security, server, and devices, are added by default when the client connection is set up.

This command can be added before and after the connection is set up.

Format

Type	Command	Response
Execute	AT+NLWADDOBJ=<obj_id>,[<ins_id>[,<res_id>]]<CR>	<CR><LF>OK<CR><LF> <CR><LF>+NLWADDOBJ: OK<CR><LF> Or <CR><LF>ERROR<CR><LF> <CR><LF>+NLWADDOBJ: FAIL,<lwm2m_err><CR><LF>
Query	AT+NLWADDOBJ?<CR>	<CR><LF>+NLWADDOBJ:<CR><LF>(list of “/<obj_id>/<ins_id><CR><LF>”)<CR><LF>OK<CR><LF>
Test	AT+NLWADDOBJ=?<CR>	<CR><LF>+NLWADDOBJ: <obj_id>[,<ins_id>[,<res_id>]]<CR><LF><CR><LF>OK<CR><LF>

Parameter

<obj_id>	Int type, Object ID.
<ins_id>	Int type, Instance ID.
<res_id>	Int type, Resources ID number.
err	Error code -13: adding this object is not supported. -14: failed to add an object or failed to add an instance of an object. Others: other errors

Example

```
AT+NLWADDOBJ=0
OK

+NLWADDOBJ: OK                                          Object is added successfully.

AT+NLWADDOBJ=65535
OK                                                      Fail to add object.

+NLWADDOBJ: FAIL,<-13>

AT+NLWADDOBJ?
+NLWADDOBJ:
/0/0                                                    Query the added objects.

OK

AT+NLWADDOBJ=?
+NLWADDOBJ: <obj_id>[,<ins_id>[,<res_id>]]              Query the parameter range.

OK
```

18.5 AT+NLWDELOBJ－Deleting an LWM2M Object

To delete an LWM2M object.

This command can be added before and after the connection is set up.

Format

Type	Command	Response
Execute	AT+NLWDELOBJ=<obj_id><CR>	<CR><LF>OK<CR><LF> <CR><LF>+NLWDELOBJ: OK<CR><LF> Or

	<CR><LF>ERROR<CR><LF> <CR><LF>+NLWDELOBJ: FAIL,<lwm2m_err><CR><LF>

Parameter

<obj_id>	Int type, Object ID.
err	Error code -132: deleting this object is supported but the object does not exist. -13: deleting this object is supported. Others: other errors

Example

```
AT+NLWDELOBJ=0
OK                                          Delete the object successfully.

+NLWDELOBJ: OK
AT+NLWDELOBJ=0
OK

+NLWDELOBJ: FAIL,<-132>
                                            Fail to delete the object.
AT+NLWADDOBJ=65535
OK

+NLWADDOBJ: FAIL,<-13>
AT+NLWDELOBJ?
+NLWDELOBJ: 0
                                            Query the deleted object.

OK
AT+NLWDELOBJ=?
+NLWDELOBJ: <obj_id>
                                            Query the parameter range.

OK
```

18.6 AT+NLWOPEN – Setting up an LWM2M Connection

To set an LWM2M data receiving mode and set up the connection.

Format

Type	Command	Response
Execute	AT+NLWOPEN=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>OK<CR><LF><CR><LF>+NLWOPEN: FAIL,<err><CR><LF> Or <CR><LF>OK<CR><LF><CR><LF>+NLWOPEN: OK<CR><LF>
Query	AT+NLWOPEN?<CR>	<CR><LF>+NLWOPEN: <mode><CR><LF><CR><LF>OK<CR><LF>
Test	AT+NLWOPEN=?<CR>	<CR><LF>+NLWOPEN: (range of <mode>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode>	LWM2M data receiving mode. 0: Direct push mode 1: Buffer access mode
err	Error code -6: the connection times out. -7: failed to connect to the server. -17: no dial-up connection is established. Others: other errors

Example

```
AT+NLWOPEN=0
OK
                              Set the LWM2M data receiving mode and set up the connection.

+NLWOPEN: OK
AT+NLWOPEN?
+NLWOPEN: 0
                              The LWM2M data reception mode is direct push mode.

OK
AT+NLWOPEN=?
+NLWOPEN: (0-1)
                              Query the parameter range.

OK
```

18.7 AT+NLWUPDATE – Updating LWM2M Data

To initiate an LWM2M data update request.

Format

Type	Command	Response
Execute	AT+NLWUPDATE<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>OK<CR><LF><CR><LF>+NLWUPDATE: FAIL,<err><CR><LF> Or <CR><LF>OK<CR><LF><CR><LF>+NLWUPDATE: OK<CR><LF>

Parameter

err	Error code -8: the module does not connect to the server Others: other errors

Example

```
AT+NLWUPDATE
OK
                                        LWM2M data updated successfully

+NLWUPDATE: OK
AT+NLWUPDATE
OK


+NLWUPDATE: FAIL,<-8>
```

18.8 AT+NLWDATASEND – Sending LWM2M Data

To send LWM2M data.

Format

Type	Command	Response
Execute	AT+NLWDATASEND=<obj_id>,<ins_id>,<res_id>,<date><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>OK<CR><LF><CR><LF>+NLWDATASEND: FAIL,<err><CR><LF> Or <CR><LF>OK<CR><LF><CR><LF>+NLWDATASEND: OK<CR><LF>

Parameter

<obj_id>	Int type, Object ID.
<ins_id>	Int type, Instance ID.
<res_id>	Int type, Resources ID number.
<date>	Data to be sent, ranging from 1 to 1024.

Example

```
AT+NLWDATASEND=3,0,9,50
OK
                                        Send data "50" to the /3/0/9 resource.

+NLWDATASEND: OK
```

18.9 AT+NLWDATASTATUS – Querying LWM2M Sending Status

To query the LWM2M sending status.

The initial NLWDATASTATUS is 4.

Format

Type	Command	Response
Query	AT+NLWDATASTATUS?<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or

	<CR><LF>+NLWDATASTATUS: <status><CR><LF><CR><LF>OK<CR><LF>

Parameter

<status>	LWM2M sending status. 0: Not sent. 1: Waiting for a response after sending. 2: Failed to send. 3: Sending timeout. 4: Sent successfully.

Example

```
AT+NLWDATASTATUS?
+NLWDATASTATUS: 4                                   Query the LWM2M sending status.

OK
```

18.10 AT+NLWRD – Reading LWM2M Data

To read the LWM2M data received in Buffer access mode.

Format

Type	Command	Response
Execute	AT+NLWRD=<read_len><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+NLWRD: <read_actual_length>,<remain_length>[<CR><LF><data>]<CR><LF><CR><LF>OK<CR><LF>
Test	AT+NLWRD=?<CR>	<CR><LF>+NLWRD: <read_len><CR><LF><CR><LF>OK<CR><LF>

Parameter

<read_len>	The length of data read.
<data>	Data read

Example

```
+NLWDATARECV: 1,0,2,2

AT+NLWRD =2

                                    After the uploaded data is received, read 2 bytes of data.
+NLWRD: 2,0
60

OK
AT+NLWRD=?
+NLWRD: <read_len>
                                    Query the parameter range.

OK
```

18.11 AT+NLWSTATUS – Querying LWM2M Connection Status

To query the LWM2M connection status.

Format

Type	Command	Response
Query	AT+NLWSTATUS?<CR>	<CR><LF>+NLWSTATUS:<status><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<status> LWM2M connection status
0: INITIAL
1: BOOTSTRAP_REQUIRED
2: BOOTSTRAPPING
3: REGISTER_REQUIRED
4: REGISTERING
5: READY (The connection is successfully set up.)

Example

```
AT+NLWSTATUS?
+NLWSTATUS: 5                       Query the LWM2M status
```

```
OK
```

18.12 AT+NLWCLOSE – Closing an LWM2M Connection

To close an LWM2M connection.

Format

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

Parameter

N/A.

Example

```
AT+NLWCLOSE
OK
                                        Close the LWM2M connection.

+NLWCLOSE: OK
```

18.13 AT+NLWCFG – Configuring LWM2M Optional Parameters

To configure the LWM2M optional parameters.

If the configured data sending type is HEX, the data sent by the +NLWDATASEND command must be in HEX format.

Format

Type	Command	Response
Execute	AT+NLWCFG=<type>,<type_value1>[,<type_value2>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NLWCFG?<CR>	<CR><LF>+NLWCFG: <CR><LF>(list of configuration) <CR><LF>OK<CR><LF>
Test	AT+NLWCFG=?<CR>	<CR><LF>+NLWCFG: <type>,<type_value1>[,<type_value2>]<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<type>	Optional parameter type.
<type_value1>[,<type_value2>]	Optional parameter value. The following types are supported currently: Dataformat: Sent and received data format. Dataformat parameters: <send_data_format>, <recv_data_format> 0: text format, 1: hex format bootstrap: Whether to connect to the Bootstrap server. family: 4: Use the IPv4 mode for auto connection. 6: Use the IPv6 mode for auto connection. factory_save: 2: Save the connection parameters to the EFS file system.

Example

```
AT+NLWCFG=DATAFORMAT,1,0                Set the data sending format to the hex format and
OK                                      the data receiving format to the text format.
AT+NLWCFG=BOOTSTRAP,1
OK                                      Open a connection to the Bootstrap server.
```

18.14 +NLWOBSERVE – Notifying the Observation Mode

To notify the current observation mode.

Format

Type	Command

Unsolicited result code	+NLWOBSERVE:<flag>,<obj_id>,<ins_id>,<res_id><CR>

Parameter

<flag>	Indicates whether the observe mode is enabled. 0: Enable the observe mode. 1: Disable the observe mode.
<obj_id>	Int type, Object ID.
<ins_id>	Int type, Instance ID.
<res_id>	Int type, Resources ID. When res_id=65535, the operated object is the entire instance, including all resources of the instance.

Example

```
+NLWOBSERVE: 0,1,0,0                    Start to observe the /1/0/0 resource.
+NLWOBSERVE: 0,1,0,65535                Start to observe all the /1/0 resources.
```

18.15 +NLWDATARECV – Notifying Data Received

To notify the received data.

If the buffer access mode is configured upon setup of the connection, only the latest data record is cached, and the previous records are cleared.

The display format of <data> is determined by the setting of data format.

Format

Type	Command
Unsolicited result code	+NLWDATARECV:<obj_id>,<ins_id>,<res_id>,<length>[,<data>] <CR>

Parameter

<obj_id>	Int type, Object ID.
<ins_id>	Int type, Instance ID.
<res_id>	Int type, Resources ID.
<length>	Length of the data received.
<data>	Received data. If the direct push mode is configured upon setup of the connection, this data will be displayed. If the buffer access mode is configured upon setup of the connection, this data will be cached and can be read by using the +NLWRD command.

Example

```
+NLWDATARECV: 3,0,13,10,"1554346580"        Receive the report indicating that the resource
                                            /3/0/13 observation starts.
```

18.16 AT+NLWDATASET – Setting Data

To set the resource value/setting reporting conditions.

Format

Type	Command	Response
Execute	AT++NLWDATASET=<data>,<type>,<obj_id>,<obj_ins_id>,<res_id>[,<res_ins_id>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NLWDATASET=?<CR>	<CR><LF>+NLWDATASET: <data>,<type>,<obj_id>,<obj_ins_id>,<res_id>[,<res_ins_id>]<CR><LF><CR><LF> OK<CR><LF>

Parameter

<obj_id> Int type, Object ID.

<obj_ins_id> Int type, Object Instance ID.

<res_id> Int type, Resources ID.

<res_ins_id> Int type, Resources Instance ID.

<type> Data type:

1 to 2: NLWDATASET is used to fill the value of the resource. <type> represents the data type of <data>:

1) Integer

2) String

4 to 9: Set the conditions for reporting the resource value or setting:

4) Set the reporting condition as follows: If the time since the previous check of the resource value is less than <data> (unit: second), the current change of the resource value will be reported; otherwise it will not be reported.

5) Set the reporting condition as follows: If the time since the previous check of the resource value is greater than <data>, the current change of the resource value will be reported; otherwise it will not be reported.

6) Set the reporting condition as follows: If the resource value is greater than <data>, the current resource value will be reported; otherwise it will not be reported.

7) Set the reporting condition as follows: If the resource value is less than <data>, the current resource value will be reported; otherwise it will not be reported.

	8) Set the reporting condition as follows: If the change amount of resource value changed compared with that in the previous check is greater than <data>, the current change of the resource value will be reported; otherwise it will not be reported. 9) Set the condition as reporting unconditionally. -4 to -8: Cancel the configured reporting conditions.
<data>	Fill in the resource value (integer/string) or set the reporting standard (integer).

Example

```
AT+ NLWDATASET=1                          The command format is incorrect.
ERROR
AT+ NLWDATASET=160,1,1,0,1                The resource 1/0/1 resource is successfully set to 160.
OK
AT+NLWDATASET=?
+NLWDATASET:
<data>,<type>,<obj_id>,<obj_ins_i
d>,<res_id>[,<res_ins_id>]


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

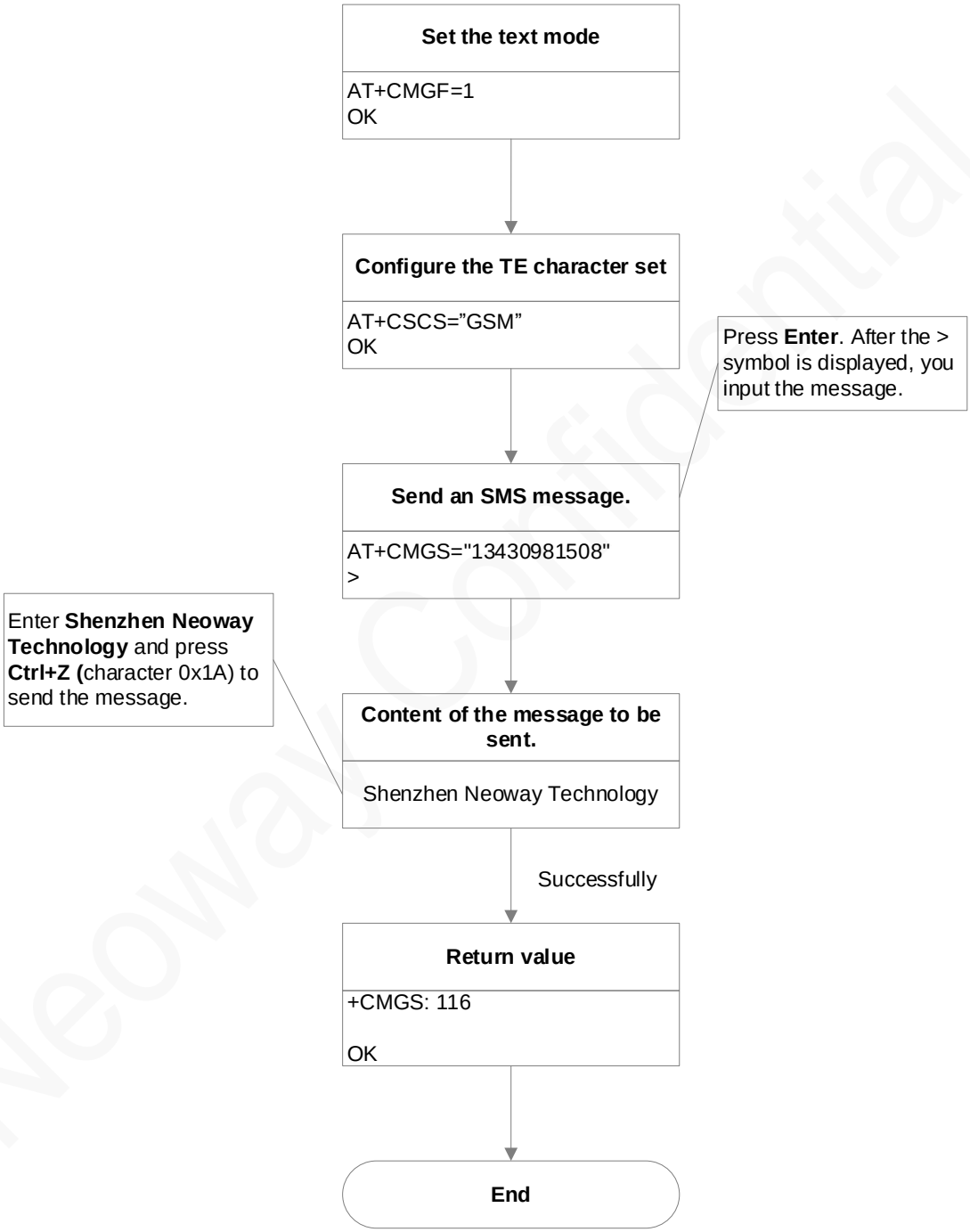
00680065006C006C006F: SMS message content

SMS message content: hello

The SMS message content starts from 0100, so the value of LENGTH in **AT+CMGS=LENGTH** is **23**.

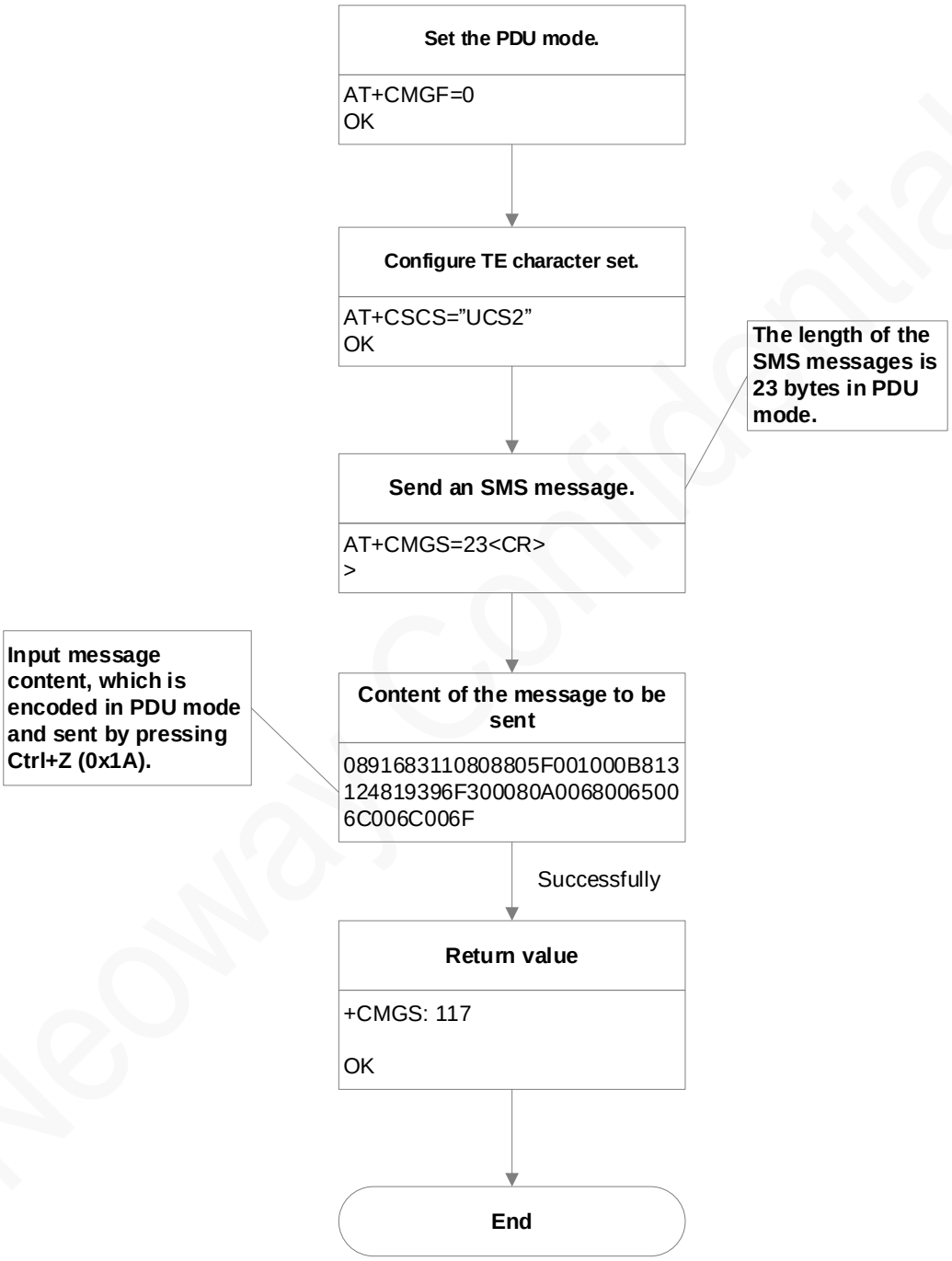
A.2 Flowchart of Sending Text SMS Messages (Through UART)

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