neoway

Get connected Get smart

N21

AT Commands Manual

Issue 1.8 Date 2022-06-13

Notice

This document provides guide for users to use N21.

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 N21.

Audience

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

Change History

Issue	Date	Change	Changed By
1.0	2018-05	Initial draft	Tao Wenhong
1.1	2018-08	• Added PSM&eDRX commands • Added +CPWROFF • Added +CSCLK • Added +NVSETPM	Tao Wenhong
1.2	2018-10	• Added +CEREG, +CPIN, +CCLK, +IPR • Added +NETSTATE, +TUESTATS, +CESTATUS • Added +SETSCMODE • Added +IPADDR, +CSCON, PSMEVENT • Added +RRCRLSREQ • Added +NCDP, +NMSTATUS • Added +NEONBIOTCFG, +LEDMODE • Added MQTT commands	Tao Wenhong
1.3	2018-12	• Added FOTA commands • Added +TCPACK and +TCPKEEPALIVE	Tao Wenhong
1.4	2019-01	• Added +UARTRSPMODE • Added +WIFIAPSCAN	Tao Wenhong
1.5	2019-04	• Added +SIMSELECT • Added +CMEE • Added +NROAMEN • Added +NVSETBAND	Tao Wenhong

		• Added MQTTDISCONNED • Added MOTA commands • Modified the description of some commands • Added timeout periods	
1.6	2019-08	Added +MIPLCFG	Tao Wenhong
1.7	2020-08	• Added the timeout period of +CFUN • Added the related description of +CPSMS • Added the related description of +CEDRXS	Li Xuanting
1.8	2022-05	• Added +CFGDFTPDN • Added +CGAUTH • Added +NVSETRELEASEVERSION	Tao Wenhong

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_N21_Datasheet

Neoway_N21_Product_Specifications

Neoway_N21_HW_User_Guide

Neoway_N21_EVK_User_Guide

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.

1 AT Syntax

1.1 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

1.2 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 .

1.3 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 (URC)	<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.		

1.4 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+CCID	5
2	AT+CGDCONT	1
3	AT+XIIC	30
4	AT+CGATT	90
5	AT+TCPSETUP	60
6	AT+TCPSEND	60
7	AT+TCPCLOSE	60
8	AT+UDPSETUP	60
9	AT+UDPSEND	60
10	AT+TCPTRANS	60
11	AT+UDPTRANS	60
12	AT+TRANSCLOSE	60
13	AT+TCPLISTEN	60
14	AT+CLOSELISTEN	60
15	AT+CLOSECLIENT	60
16	AT+TCPSENDS	60
17	AT+FTPLOGIN	30
18	AT+FTPSIZE	30
19	AT+FTPGET	30
20	AT+FTPPUT	30
21	AT+HTTPSETUP	30
22	AT+HTTPACTION	30
23	AT+HTTPSSETUP	30
24	AT+HTTPSACTION	30
25	AT+HTTPSCLOSE	30
26	AT+NCDPOPEN	180

27	AT+NCDPCLOSE	180
28	AT+NMGS	60
29	AT+NSOTAVERDL	60
30	AT+CTM2MREG	60
31	AT+CTM2MUPDATE	180
32	AT+CTM2MDEREG	180
33	AT+CTM2MSEND	60
34	AT+WSOTAVERDL	60
35	AT+IMQTTAUTH	TLS connected: 60s TLS unconnected: 60s
36	AT+IMQTTCONN	TLS connected: 60s TLS unconnected: 60s
37	AT+IMQTTPUB	60
38	AT+IMQTTPUBS	60
39	AT+IMQTTPUBIN	60
40	AT+IMQTTSUB	30
41	AT+IMQTTUNSUB	30
42	AT+IMQTTRCVPUB	30
43	AT+IMQTTDISCONN	30
44	AT+ICOAPAUTH	90
45	AT+ICOAPSENDREQ	60
46	AT+ICOAPSENDBIN	60
47	AT+ICOAPSENDREQN	60
48	AT+ICOAPSENDNBIN	60
49	AT+MQTTCONN	60
50	AT+MQTTSUB	30
51	AT+MQTTUNSUB	30
52	AT+MQTTPUB	30
53	AT+MQTTPUBIN	30
54	AT+MQTTDISCONN	30
55	AT+MIPLOPEN	Customized, ranging from 0 to 60s, 60s by default
56	AT+MIPLCLOSE	60
57	AT+MIPLNOTIFY	60
58	AT+MIPLREADRSP	60
59	AT+MIPLWRITERSP	60
60	AT+MIPLEXECUTERSP	60
61	AT+MIPLOBSERVERSP	60

62	AT+MIPLDISCOVERRSP	60
63	AT+MIPLPARAMETERRSP	60
64	AT+MIPLUPDATE	60
65	AT+MOTADL	60
66	AT+MOTASTAT	3
67	AT+MOTATRANS	60
68	AT+BLEOPEN	3
69	AT+BLESEND	1
70	AT+PING	Customized, 1-255
71	AT+NEOFOTA	30

2 General Commands

2.1 ATI – Querying the Manufacturer Information

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

Format

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

Parameter

<module_info> module manufacturer, module model, software version

Example

```
ATI
NEOWAY                                  Manufacturer
N21                                     Module model
V001                                    Version

OK
```

2.2 AT+CGMR – Querying the Software Version

To query the software version

Format

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

Parameter

<version> software version

Example

```
AT+CGMR
+CGMR: N21_RDD0CM_BZ_V001
OK
```

2.3 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

Example

```
AT+CGSN
+CGSN: 355910044336974

OK
```

2.4 AT+CIMI – Querying the IMSI

To query the international mobile subscriber identification (IMSI)

Format

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

Parameter

<IMSI> International mobile subscriber identification, a character string of 15 digits that starts with 3-bit MCC and 2-bit MNC.
It is used to authenticate the SIM card.

Example

```
AT+CIMI                                    Obtain the IMSI number.
+CIMI: 460022201575463

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

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

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

Format

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

Parameter

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

Example

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

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

3 UE Control and Status Report

3.1 AT+CREG – Querying Network Registration Status

To query the network registration status of the module

Format

Type	Command	Response
Set	AT+CREG=[<n>]<CR>	<CR><LF>OK<CR><LF> or <CR><LF>ERROR<CR><LF>
Query	AT+CREG?<CR>	<CR><LF>+CREG: <n><stat>[,<lac>,<ci>[,<Act>]] <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 <CR><LF>+CREG: <stat><CR><LF> When n=2 <CR><LF>+CREG: <stat>[,<lac>,<ci>[,<Act>]]<CR><LF>	

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: ECGSM
9: NB-IoT

Example

```
AT+CREG=1                                   Enable the URC of network registration status.
OK
AT+CREG?                                    Query the network registration status of the
+CREG: 0,1                                  module.

OK
AT+CREG=?                                   Query the value range of the network
+CREG: (0-2)                                registration status parameter.

OK
```

3.2 AT+CEREG – Querying EPS Network Registration Status

To query the EPS network registration status of the module

Format

Type	Command	Response
Execute	AT+CEREG=[<n>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CEREG?<CR>	<CR><LF>+CEREG: <n>,<stat>[,[<tac>],[<ci>], [<AcT>] [,[<cause_type>], [<reject_cause>][,[<Active-Time>], [<Periodic-TAU>]]]]

		<CR><LF>OK<CR><LF>
Test	AT+CEREG=?<CR>	<CR><LF>+CEREG: (list of supported <n>s) <CR><LF>OK<CR><LF>
URC	• <n>=1 <CR><LF>+CEREG: <stat><CR><LF> • <n>=2 <CR><LF>+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>]]<CR><LF> • <n>=4 • <CR><LF>+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<Active-Time>],[<Periodic-TAU>]]]]<CR><LF> • <n>=5 <CR><LF>+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,[<cause_type>],[<reject_cause>][,[<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 4: enable network registration unsolicited result codes containing **Active-Time** and **Periodic-TAU**
<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
<tac>	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 8: ECGSM 9: NB-IOT
<Active-	8-bit unibyte

Time>	Requested Active Time on GERAN/UTRAN network (T3324) Bit8-Bit6: unit 000 – 2 seconds 001 – 1 minute 010 – 6 minutes 111 - T3324 invalid **Bit5-Bit1**: binary-code time e.g. 00000001 indicates two seconds
<Periodic-TAU>	8-bit unibyte Requested periodic-TAU cycle on GERAN/UTRAN network (T3412) Bit8-Bit6: unit 000 – 10 minutes 001 – 1 hour 010 – 10 hours 011 – 2 seconds 100 – 30 seconds 101 – 1 minute 110 – 320 hours 111 - T3412 invalid Bit5-Bit1: binary-code time e.g. 00100010 indicates 2 hours

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: (0-2,4)          parameter.

OK
```

3.3 ATE1/ATE0 – Enabling & Disabling the Terminal Display

To enable or disable the terminal display function of the AT commands

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

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

ATE is equal to ATE1.

Format

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

Parameter

<value> 0: disable the terminal display
1: enable the terminal display (default)

Example

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

3.4 ATQ – Setting the Code Result Suppression Mode

To set the mode whether to suppress the code result.

- After mode set to the code result suppression mode, the module does not output **OK** or **ERROR** to commands.
- The setting by this command is not saved after the module is powered down. The setting is valid only for the GSM commands and invalid for customized commands.
- ATQ is equal to ATQ1.

Format

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

Parameter

<value> 0: Output the code result (default)
1: Suppress the code result

Example

```
ATQ1                    Set to code result suppress mode. (The module does not return OK
                        after this command is executed successfully.)
AT+CSQ                  After the mode is set, the return value to AT+CSQ does not
+CSQ: 31,99             contain the code result OK.
ATQ0
OK                      Set to the code result output mode.
                        After the mode is set, the return value to AT contains the code
AT                      result OK.
OK
```

3.5 ATV – Setting the Response Format of the Device

To set the response format of the device ATE is equal to ATE1.

- ATV is equal to ATV1.
- After **ATV0** is executed, the module returns 0 if a command is sent in correct format (default setting is OK) and returns 4 if a command is sent in incorrect format (default setting is ERROR).
- The settings by this command are not be saved after the module is powered down.

Format

Type	Command	Response
Set	ATV[<value>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<value> 0: Set the response format to output with only some header, footer, and digit text.
1: Set the response format to output with all headers, footers, and detailed response text (default).

Example

```
ATV1                    Set the response format to output with all headers, footers,
OK                      and detailed response text.

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

```
OK
ATV0                        Set the response format to output with only some header,
0                           footer, and digit text. The module returns 0 after the format
                            is set successfully.
AT+CSQ
+CSQ: 31,99

0
```

3.6 AT&W – Saving the Setting

To save the settings of BASIC type commands.

Format

Type	Command	Response
Execute	AT&W	<CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT&W
OK
```

3.7 AT&F – Resetting the Module to Factory Settings

To reset the module to the factory settings

- If the module is set to the code result suppression mode (ATQ1), reset it to factory settings by sending this command.
- The following commands are supported: ATE, +CMEE, ATV, ATQ, CEREG, IPR.

Format

Type	Command	Response
Execute	AT&F<CR>	<CR><LF>OK<CR><LF>

Parameter

N/A

Example

```
AT&F0                                          Reset the module to factory settings.
OK
AT&F                                           Reset the module to factory settings.
OK
```

3.8 AT+CMUX – Activating Multiplexing Mode

To activate multiplexing mode

The multiplexing protocol allows two or more virtual ports to be created on a physical port. Generally, three virtual ports are created: one is used for dialing up to network and the other two are used for AT command sending and receiving.

It is recommended to send AT+CMUX=0 to activate the multiplexing mode.

Format

Type	Command	Response
Set	AT+CMUX=<mode>[,<subset>[,<port_speed>[,<N1>[,<T1>[,<N2>[,<T2>[,<T3>[,<k>]]]]]]]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CMUX?<CR>	<CR><LF>+CMUX: <mode>,[<subset>], <port_speed>, <N1>,<T1>,<N2>,<T2>,<T3> [,<k>]<CR><LF> <CR><LF>OK<CR><LF>

Test	AT+CMUX=?<CR>	<CR><LF>+CMUX: (list of supported <mode >s),(list of supported <subset>s),(list of supported <port_speed>s),(list of supported <N1>s),(list of supported <T1>s),(list of supported <N2>s),(list of supported <T2>s),(list of supported <T3>s),(list of supported <k>s) <CR><LF><CR><LF>OK<CR><LF>

Parameter

<mode>	The mode of MUX that is enabled, integer type 0: basic option (default) 1: advanced option (not supported)
<subnet>	Subset of frame format, integer type 0: UIH frames used only (default) 1: UI frames used only (not supported currently)
<port_speed>	UART port rate, integer type 1: 9600bit/s 2: 19200bit/s 3: 38400bit/s 4: 57600bit/s 5: 115200bit/s (not supported currently) 6: 230400bit/s (not supported currently)
<N1>	Maximum frame size, integer type, ranging from 1 to 32768. The value that are supported currently ranges from 1 to 2048.
<T1>	Acknowledgement timer in unit of ten milliseconds, integer type, ranging from 1 to 255, where 10 is default (equal to 100 ms).
<N2>	Maximum number of re-connections, integer type, ranging from 0 to 100. The default value is 3 and the value that are supported currently ranges from 0 to 5.
<T2>	Response timer for the multiplexer control channel in unit of ten milliseconds, integer type, ranging from 2 to 255. The default value is 30. Indicating 300 ms. <T2> must be greater than <T1>
<T3>	Wake up response timer in seconds, integer type, ranging from 1 to 255. The default value is 10, indicating 10 seconds.
<k>	Window size, integer type, ranging from 1 to 7. The default value is 2. This parameter is used in advanced mode in which error restore is supported.

Example

```
AT+CMUX?                                    Query the current settings
```

```
+CMUX: 0,0,4,127,10,3,30,10,2
OK
AT+CMUX=?                                   Query the available range of parameters.
+CMUX: (0,1),(0),(1-4),(1-2048),(1-255),
(0-100),(2-255),(1-255),(1-7)
OK
AT+CMUX=6                                   The parameter value exceeds available range
ERROR
AT+CMUX=0,0,4,512,254,5,255                 Enable the CMUX function
OK
```

3.9 AT+CFUN – Setting Module Functionality

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

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

Do NOT use this command if you send +ENPWRSAVE already.

Format

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

Parameter

<fun>	Power saving function mode 0: turn off radio and SIM power 1: full functionality (default) 4: turn off the RF circuit (flight mode)
<rst>	Specifies whether to restart the module 0: do not reset the module before setting it to **<fun>** power level 1: reset the module before setting it to **<fun>** power level

Example

```
AT+CFUN=1                                   Set full functionality.
OK
AT+CFUN?                                    Query current function level.
+CFUN: 1                                    Full functionality

OK
AT+CFUN=?                                   Query available parameter value ranges.
+CFUN: (0,1,4),(0,1)
OK
```

3.10 AT+CCLK – Clock

To set and query the real-time clock

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

Format

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

Parameter

<time> Character string in format of "YY/MM/DD,hh:mm:ss[+TZ]".
TZ The time lag between the local time and the GMT time, two digits.

Example

```
AT+CCLK="18/07/01,14:54:01"                 Set the real-time clock of the module.
OK
AT+CCLK?                                    Query the setting of the real-time clock.
+CCLK: "18/07/01,14:54:10"
OK
AT+CCLK=14/07/02,10:48:50                   Incorrect command syntax.
ERROR
```

3.11 AT+IPR – Setting Baud Rate

To set the baud rate of the module

The baud rate is detected automatically by default. The setting by this command is not saved after the module is powered down. You can save the settings by executing AT&W.

Format

Type	Command	Response
Set	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, 2400, 4800, 9600, 19200, 38400, 57600.
The default value is 0, indicating automatic baud rate detection.

Example

```
AT+IPR=57600                                          Set the baud rate to 115200 bps.
OK
AT+IPR?                                               Query the current baud rate.
+IPR: 57600
OK
AT+IPR=?                                              Query the available baud rate
+IPR: (0,2400,4800,9600,19200,38400,57600)            range.
OK
AT+IPR=100                                            Set the baud rate to 100.
ERROR                                                 The value is not allowed
```

3.12 AT+CPIN – Entering PIN Code

To query the PIN status and enter PIN code

To enter PIN code, lock current SIM card (send AT+CLCK="SC",1,"1234") and then restart the module.

If the PIN code is input incorrectly for three times, PUK is required to unlock the SIM card.

The module reports +CPIN: NO SIM after detecting SIM offline.

Format

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

Parameter

<pin>, <newpin>	String type
<code>	**READY**: no password **SIMPIN**: enter PIN code. **SIMPUK**: enter PUK code. **SIMPIN2**: enter PIN2 code. **SIMPUK2**: enter PUK2 code.

Example

```
AT+CPIN?
+CPIN: READY                          Query whether PIN code is required.
OK
AT+CPIN="0000"
ERROR                                 Incorrect pin code
AT+CPIN="1234"
OK                                    Input correct PIN code.
AT+CPIN?
ERROR                                 No SIM card is inserted.
```

3.13 CMEE – Setting Error Information

To enable or disable the **+CME ERROR:<err>** result code

- The settings by this command are not be saved after the module is powered down.
- AT+CMEE=2 is recommended when debugging.

For error codes, refer to Appendix A .

Format

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

Parameter

<n> 0: Disable the **+CME ERROR:<err>** result code and display **ERROR**. (default)
1: Enable the **+CME ERROR:<err>** result code and use the numeric **<err>** value.
2: Enable the **+CME ERROR:<err>** result code and use verbose **<err>** values. Set AT+CMEE=2 when commissioning the module.

Example

```
AT+CMEE=1                                   Enable the result code in digit format.
OK
AT+CMEE?
+CMEE: 1                                    Query the status of the current result code.
OK
AT+CMEE=?
+CMEE: (0-2)                                Query the status range of error code.
OK
```

4 Network Service

4.1 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> <CR><LF>OK<CR><LF>
Test	AT+CSQ=?<CR>	<CR><LF>+CSQ: (list of supported <rssi>s),(list of supported <ber>s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

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

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

<ber>

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

Example

```
AT+CSQ
+CSQ: 1,99                                    Query the strength of the current signal.

OK
```



```
AT+CSQ=?
+CSQ: (0-31,99),(0-7,99)                                    Query the value ranges of parameters.

OK
```

4.2 AT+COPS – Selecting and Registering With a Network

To select and register with a network

Format

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

Parameter

<mode> Specifies the mode of network 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> 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 of the network selected manually
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
8: ECGSM
9: NB-IOT

Example

```
AT+COPS=0,0                                 Automatic network selection is enabled. Long
OK                                          alphanumeric mode.
AT+COPS=0,2                                 Set to digital mode
OK
AT+COPS?
+COPS: 0,0,"China Mobile",9                 China Mobile

OK
AT+COPS?                                    If it is set to numeric mode, get the number
+COPS: 0,2,"46000",9                        46000

OK
AT+COPS?
+COPS: 0,0,"China Unicom",9                 China Unicom

OK
AT+COPS?                                    If it is set to numeric mode, then get the
+COPS: 0,2,"46001",9                        number 46001.

OK
AT+COPS=?                                   Query the values supported for network
+COPS:                                      selection.
(2,"ChinaMobile","CMCC","46000"),
(1,"ChinaTelecom","CT","46011"),
(1,"ChinaUnicom","UNICOM","46001"),,
(0,1,2,3,4),(0,1,2)

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

4.3 AT+NETSTATE – Querying Network Registration State

To query the current network registration state

Format

Type	Command	Response
Query	AT+NETSTATE?<CR>	<CR><LF>+NETSTATE: <net_type>,<net_band><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<net_type> Registered network mode
0: No network
1: CAT NB1

<net_band> Hexadecimal registered network band
The following values are returned if **<net_type>** is set to NB-IoT:
0x0: Not registered to the network
0x1: LTE B1
0x2: LTE B2
0x4: LTE B3
0x8: LTE B4
0x10: LTE B5
0x80: LTE B8
0x800: LTE B12
0x1000: LTE B13
0x20000: LTE B18
0x40000: LTE B19
0x80000: LTE B20
0x2000000: LTE B26
0x8000000: LTE B28

Example

```
AT+NETSTATE?
+NETSTATE: 1,0x80                  The current registered network is NB-IoT and the
                                   frequency band is band8.

OK
```

4.4 AT+TUESTATS – Querying Network Status Information

To query network status information.

Format

Type	Command	Response
Execute	AT+TUESTATS=<type><CR>	<CR><LF>TUESTATS:<type>,<value> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<type> type. A pair of quotation marks are required.
RADIO: radio specific information
CELL: per-cell information for the top 5 cells
BLER: block error rate information
THP: throughput
ALL: all information. The value of <type> output is the correct one for each data type.
<type> = RADIO
<signal power in centibels>
<total power in centibels>
<current TX power level in centibels >
<total TX time since last reboot in millisecond>
<total RX time since last reboot in millisecond>
<last SIB1 cell ID>
<last ECL value>
<last snr value>
<last earfcn value>
<last pci value>
<rsrq in centibels>
<type> = CELL, supporting at most 5 cells.
<earfcn> absolute radio-frequency channel number
<physical cell id> physical id of the cell
<primary cell> 1 indicates the current serving cell
<rsrp> reference signal received power
<rsrq> reference signal received quality
<rssi> received signal strength indicator
<snr> signal to noise ratio
<type> = BLER
<rlc_ul_bler> RLC layer block error rate (uplink). Integer %
<rlc_dl_bler> RLC layer block error rate (downlink). Integer %
<mac_ul_bler> physical layer block error rate (uplink). Integer %
<mac_dl_bler> physical layer block error rate (downlink). Integer %
<total bytes transmitted>
<total bytes received>
<transport blocks sent>

<transport blocks received>
<transport blocks retransmitted>
<total ack/nack messages received>
<type> = THP
<rlc_ul> RLC layer throughput (uplink). Integer bps
<rlc_dl> RLC layer throughput (downlink). Integer bps
<mac_ul> physical layer throughput (uplink). Integer bps
<mac_dl> physical layer throughput (downlink). Integer bps

Example

```
AT+TUESTATS="RADIO"
TUESTATS:RADIO,Signal power,13
TUESTATS:RADIO,Total power,45
TUESTATS:RADIO,Tx power,-1
TUESTATS:RADIO,TX time,288
TUESTATS:RADIO,RX time,44
TUESTATS:RADIO,Cell ID,197756455
TUESTATS:RADIO,ECL,0
TUESTATS:RADIO,SNR,9
TUESTATS:RADIO,EARFCN,1640
TUESTATS:RADIO,PCI,245
TUESTATS:RADIO,RSRQ,255
OK

AT+TUESTATS="CELL"
TUESTATS:CELL,2506,269,1,53,255,-
65,-1
OK

AT+TUESTATS="BLER"
TUESTATS:BLER,RLC UL BLER,0
TUESTATS:BLER,RLC DL BLER,0
TUESTATS:BLER,MAC UL BLER,0
TUESTATS:BLER,MAC DL BLER,0
TUESTATS:BLER,Total TX bytes,77
TUESTATS:BLER,Total RX bytes,77
TUESTATS:BLER,Total TX blocks,1
TUESTATS:BLER,Total RX blocks,1
TUESTATS:BLER,Total RTX blocks,0
TUESTATS:BLER,Total ACK/NACK RX,0
OK

AT+TUESTATS="THP"
TUESTATS:THP,RLC UL,600
TUESTATS:THP,RLC DL,844
TUESTATS:THP,MAC UL,2156
TUESTATS:THP,MAC DL,2464
OK
```


The query result is valid only during data reception and transmission.

4.5 AT+CESTSTUS – Querying CE Status

To query the CE status

Format

Type	Command	Response
Execute	AT+CESTSTUS<CR>	<CR><LF>+CESTATUS: <status> <CR><LF>OK<CR><LF>

Parameter

<status>
0: No CE
1: CE level 0
2: CE level 1
3: CE level 2

Example

```
AT+CESTATUS
+CESTATUS: 1
OK
```

4.6 AT+SETSCMODE – Setting SC Mode

To set scrambling code mode of the module.

It must be the same as that on the base station so that the module can register with network. The new scrambling code is used by default. The settings by this command are saved after the module is powered off.

Format

Type	Command	Response
Set	AT+SETSCMODE=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Query	AT+SETSCMODE?<CR>	<CR><LF>+SETSCMODE: <mode> <CR><LF>OK<CR><LF>

Parameter

<mode> 0: Old SC
1: New SC

Example

```
AT+SETSCMODE=1
OK                                          Use new SC.

AT+SETSCMODE?
+SETSCMODE: 1                               Query SC mode.
OK
```

4.7 AT+NROAMEN – Switching Roaming of the Module

To enable or disable the roaming function of the module The settings by this command are saved after the module is powered off.

Format

Type	Command	Response
Set	AT+NROAMEN=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NROAMEN?<CR>	<CR><LF>+NROAMEN: <mode> <CR><LF>OK<CR><LF>
Test	AT+NROAMEN=?<CR>	<CR><LF>+NROAMEN: (list of supported <n>) <CR><LF>OK<CR><LF>

Parameter

<mode> 0: disabled (default)
1: enabled

Example

```
AT+NROAMEN=1                          Enable the roaming function of the module.
OK
AT+NROAMEN?
+NROAMEN: 1                           Query the status of the roaming function.
OK
AT+NROAMEN=?
+NROAMEN: (0-1)                       Query the value range of the parameter.
OK
```

4.8 AT+NVSETBAND – Setting Frequency Band

To set the number of frequency bands and band value. The settings by this command are saved after the module is powered off.

Format

Type	Command	Response
Set	AT+NVSETBAND=<band_num>, <band_value>,…<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NVSETBAND?<CR>	<CR><LF><band_num> band in total: <band_value1>,…<band_valueN><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+NVSETBAND=?<CR>	<CR><LF>+NVSETBAND: <band_num>, (list of supported <band_value>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<band_num> allowed band quantity, ranging 1 to 10.
<band_value> allowed frequency bands (1,2,3,5,8,12,18,19,20,26,28)

Example

```
AT+NVSETBAND=3,3,5,8                  Set the number of frequency bands to 3 and the bands are
OK                                    respectively 3, 5, and 8.
AT+NVSETBAND?
3 band in total: 3,5,8                Query the number of frequency bands and band values.
OK
```


5 EGPRS Commands

5.1 AT+CFGDFTPDN – Setting the Default Bearer

To set the default APN and PDN type.

The setting by this command takes effect after the module is restarted and is saved after the module is powerred off.

Format

Type	Command	Response
Set	AT+CFGDFTPDN=<defaultPdnType>[,<apn>] <CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CFGDFTPDN?<CR>	<CR><LF>+CFGDFTPDN: defaultPdnType=<defaultPdnType>; [0]pdnType=1,apn=<APN>; [1]pdnType=2,apn=<APN>; [2]pdnType=3,apn=<APN>; [3]pdnType=5,apn=<APN>;<CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CFGDFTPDN=?<CR>	<CR><LF>+CFGDFTPDN: pdnType=[1,2,3,5], apn="string" <CR><LF> <CR><LF>OK<CR><LF>

Parameter

<defaultPdnType>	Default PDN type: 1: IPv4 2: IPv6 3: IPv4v6 5: NonIP
<APN>	(Access Point Name) string format. It is a logic name, used to select GGSN or External Packet Data Network.

Example

```
AT+CFGDFTPDN=1,"CMIOT"                                  Set the default PDN type.
OK
AT+CFGDFTPDN?
+CFGDFTPDN: defaultPdnType=1;
[0]pdnType=1,apn=CMIOT;
[1]pdnType=2,apn=;  [2]pdnType=3,apn=;                  Query the settings.
[3]pdnType=5,apn=;
OK
```

5.2 AT+CGDCONT – Setting PDP Format

To set the packet data protocol (PDP) format of the GPRS. Only one APN can be set.

Format

Type	Command	Response
Set	AT+CGDCONT=[<cid>[,<PDP_type>[,<APN>[,<PDP_addr>[,<d_comp>[,<h_comp>[,<pd1>[,…[,pdN]]]]]]]]<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 supported <cid>, value range of supported <PDP_type>, value range of supported <APN>, value range of supported <PDP_addr>, value range of supported <d_comp>, value range of supported <h_comp>, value range of supported <pd1> ,…, value range of supported pdN<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<cid> (PDP Context Identifier) a numeric parameter that specifies a particular PDP context definition. The parameter is local to the TE-MT interface and is used in

	other PDP context-related commands. The range of permitted values (minimum value = 1) is returned by the test form of the command.
<PDP_type>	(Packet Data Protocol type) a string parameter. IP Internet Protocol (IETF STD 5)
<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. If the value is null or omitted, then the subscription value is requested.
<PDP_address>	a string parameter that identifies the MT in the address space applicable to the PDP. If the value is null or omitted, then a value maybe provided by the TE during the PDP startup procedure or, failing that, a dynamic address is requested. The read form of the command continues to return the null string even if an address has been allocated during the PDP startup procedure. The allocated address may be read using the +CGPADDR command.
<d_comp>	a numeric parameter that controls PDP data compression (applicable for SNDCP only) 0: off (default if value is omitted)
<h_comp>	a 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","CMIOT"                    Set APN.
OK
AT+CGDCONT=1,IP,CMIOT                        A pair of quotation marks are required for each
ERROR                                        parameter.
AT+CGDCONT?
+CGDCONT:1,"IP","CMIOT","0.0.0.0",0,         Query the current parameter value.
0,0,0
OK
AT+CGDCONT=?
+CGDCONT: (1),(IP,IPV6,IPV4V6),,,(0-         Query the value range of parameters.
3), (0-4),(0-1),(0-4)
OK
```

5.3 AT+CGAUTH – User Authentication

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

Execute this command after AT+CGDCONT.

Add this command execution operation into your code when using the internal protocol stack since the PDP authentication is required for the application that uses the private network.

Format

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

Parameter

<cid>	Integer type. Specifies a particular PDP context definition. This parameter corresponds to the <cid> parameter of the +CGDCONT command.
<auth>	Integer type. Authentication protocol used for this PDP context. 0: NONE 1: PAP 2: CHAP 3: PAP or CHAP (this option is not implemented by the 3GPP standard and depends on the platform.)
<name>	String type. User name for access to the IP network.
<pwd>	String type. Password for access to the IP network.

Example

```
AT+CGAUTH=1,1,”card”,”card”                    Set the first PDP context authentication parameters.
OK

AT+CGAUTH=?
+CGAUTH: (1-2),(0-3),,                         Query the range of parameter values.

OK
```


5.4 AT+XIIC – Setting up a PPP Link

To set up a PPP link.

Ensure that the module registers the network before using the AT+XIIC=1 command to set up PPP link. Use AT+CEREG? to check the network registration of the module. If +CEREG: 0,1 or +CEREG: 0,5 is returned, the module is not registered with the network.

Format

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

Parameter

<act> 0: Deactivate the PPP link
1: Activate the PPP link.

<ip> IP address

Example

```
AT+XIIC=1                          Set up a PPP link.
OK
AT+XIIC?                           The PPP link is set up successfully and the IP address is
+XIIC:    1,10.107.216.162         10.107.216.162.
OK                                 There are four spaces before 1.
AT+XIIC?
+XIIC:    0,0.0.0.0                The PPP link is not set up successfully.
OK                                 There are four spaces before 0.
```

5.5 AT+CGATT – Setting GPRS Attach and Detach

To set GPRS attach and detach.

By default, the module can automatically perform PS attach.
Ensure that the GPRS attach is set before the PPP connection is set up. Add AT+CGATT? 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>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CGATT?<CR>	<CR><LF>+CGATT: <state><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CGATT=?<CR>	<CR><LF>+CGATT: (range of supported <state>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<state> 0, 1
0: indicates detach
1: indicates attach

Example

```
AT+CGATT=1
OK                                  GPRS attach is set successfully.

AT+CGATT=0
OK                                  GPRS detach is set successfully.

AT+CGATT=0
GPRS DISCONNECTION                  Send this command after setting a PPP connection.
OK
AT+CGATT=0
ERROR                               ERROR is returned because no SIM card is installed.

AT+CGATT?                           Query the GPRS status.
+CGATT: 0
OK

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

5.6 +IPADDR – Reporting IP Address

To report IP address allocated by network after PPP is activated. This function is disabled by default. Use AT+NEONBIOTCFG to enable or disable it.

Format

Type	Command
URC	+IPADDR=<ip><CR>

Parameter

<ip> IP address.

Example

```
+IPADDR: 10.100.45.2
```

5.7 +CSCON – Reporting RRC Status

To report current RRC status. This function is disabled by default. Use AT+NEONBIOTCFG to enable or disable it.

Format

Type	Command
URC	+CSCON=<status><CR>

Parameter

<status> 0: IDLE
1: CONNECTED

Example

```
+CSCON: 1
```

5.8 +PSMEVENT – Reporting PSM Status

To report current PSM status. This function is disabled by default. Use AT+NEONBIOTCFG to enable or disable it.

Format

Type	Command
URC	+PSMEVENT: <status><CR>

Parameter

<status>: ENTER PSM
PSM status PSM WAKEUP

Example

```
+PSMEVENT: ENTER PSM
```

5.9 AT+RRCRLSREQ – Releasing RRC Connection

To release RRC connection.

Module state changes from CONNECTED to IDLE.

Format

Type	Command	Response
Execute	AT+RRCRLSREQ<CR>	<CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT+RRCRLSREQ
OK
```


6 TCP/UDP Client Commands

6.1 AT+RECVMODE – Setting Receive Mode

To set the receive mode of TCP and UDP data

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

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

The settings will be saved in PSM mode.

Format

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

Parameter

<n> Receive mode
0: buffer the TCP or UDP data received and the MCU sends command to read the data
1: print the TCP or UDP data received to UART directly (default)
2: When the platform issues multiple data, the module caches all the data received and report the first data only; then the module reports the next data after the MCU reads the first data. This mode is valid only in UDP mode.

<mode> Specifies whether to report in hexadecimal format
0: report in ASCII format (default)
1: report in hexadecimal format

Example

```
AT+RECVMODE=0                                    Set data receive mode.
OK
```

6.2 AT+TCPKEEPALIVE – Setting Keepalive Heartbeat

To set TCP keepalive heartbeat

Send this command before setting up a TCP connection. And it is valid for all connections. The settings by this command are not saved after the module is powered down.

Note that this function consumes data traffic.

Format

Type	Command	Response
Set	AT+TCPKEEPALIVE=<mode>[,<time>[,<interval>]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+TCPKEEPALIVE? <CR>	<CR><LF>+TCPKEEPALIVE: <mode>,<time>,<interval> <CR><LF>OK<CR><LF>
Test	AT+TCPKEEPALIVE=?<CR>	<CR><LF>+TCPKEEPALIVE: (value range of supported <onoff>),(value range of supported <time>),(value range of supported <interval>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<onoff>	Heartbeat switch 0: disable (default) 1: enable
<time>	Heartbeat interval at which the module sends data packets to the server, ranging from 30 to 7200. Unit: second. The default value is 120. The recommended value ranges from 30 to 300.
<interval>	Retransmission interval at which the module sends heartbeat data packet again if it does not receive the response from the server, ranging from 1 to 1800, unit: second. The default value is 75. The recommended value ranges from 40 to 100.

Example

```
AT+TCPKEEPALIVE=1                                          Enable the KEEPALIVE
OK                                                         function.
AT+TCPKEEPALIVE=1,120,75                                   Enable and set the
OK                                                         KEEPALIVE parameters
AT+TCPKEEPALIVE=0                                          Disable TCP KEEPALIVE
OK
AT+TCPKEEPALIVE?                                           Query the parameter range
+TCPKEEPALIVE: 1,120,75
OK
AT+TCPKEEPALIVE=?                                          Query the value range of
+TCPKEEPALIVE: (0-1),(30-7200),(1-1800)                    the network registration
OK                                                         status parameter.
```

6.3 AT+TCPSETUP – Setting up a TCP Connection

To set up a TCP Connection

Use the **AT+XIIC=1** command to set up a PPP link before sending this command.

Format

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

Parameter

<n>	Socket ID, ranging from 0 to 4
<ip>	Destination IP address, in xx.xx.xx.xx or domain name format
<port>	Destination port ID in decimal ASCII code
<result>	Result code OK FAIL LINK ALREADY OPENED

Example

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

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

+TCPSETUP:0,OK                          Successful
+TCPCLOSE:0,Link Closed                 The socket is closed.
AT+TCPSETUP=1,192.168.20.6,7000         Fails to set up a connection to
OK                                      192.168.20.6,7000 on socket 1. The server is
                                        not started, the IP address is incorrect, or
+TCPSETUP: 1,FAIL                       the SIM card is out of credit.
AT+TCPSETUP=0,neowayjsr.oicp.net,60010  A TCP/UDP connection has been set up on socket
OK                                      0.

+TCPSETUP: 0,ERROR1
AT+TCPSETUP=5,192.168.20.6,7000         Parameters are set incorrectly.
+TCPSETUP:ERROR
AT+TCPSETUP=0.58.60.184.213.10012       Parameters are set incorrectly.
+TCPSETUP:ERROR
AT+TCPSET=0,58.60.184.213,10012         The AT command is not complete.
ERROR
```

6.4 AT+TCPSEND – Sending TCP Data

To send TCP data

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

Ensure that a TCP connection is set up before sending TCP data. The **AT+IPSTATUS** command is recommended to check the buffer size before sending data.

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

Format

Type	Command	Response
Execute	AT+TCPSEND=<n>,<length><CR>	<CR><LF>> <CR><LF>OK<CR><LF> <CR><LF>+TCPSEND: <n>,<length><CR><LF> Or <CR><LF>>> <CR><LF>+TCPSEND: ERROR<CR><LF> Or <CR><LF>> <CR><LF>+TCPSEND: <n>, OPERATION EXPIRED<CR><LF>

	Or <CR><LF>+TCPSEND: <result><CR><LF>

Parameter

<n>	Socket ID, ranging from 0 to 5. A TCP connection is established on the socket.
<length>	Length of the data to be sent, ranging from 1 to 4096, unit: byte
<content>	Data to be sent in command mode. It supports escape mode.
<mode>	0: send data in ASCII mode. the escaped character is supported (default). 1: send data in HEX mode.
<result>	SOCKET ID OPEN FAILED DATA LENGTH ERROR

Example

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

+TCPSEND: 0,1,SUCCESS
AT+TCPSEND=0,1024                       Network congestion occurs when 1024-byte data is
>                                       sent. Only some data is sent successfully.
+TCPSEND: ERROR
+TCPSEND: 0,1024,FAIL
AT+TCPSEND=0,10                         After the data sending command is input and > is
>                                       returned, no more data is entered in 60 seconds. Then
+TCPSEND: 0,OPERATION EXPIRED           the expiration information is displayed.
AT+TCPSEND=0,1                          One-byte data fails to be sent on socket 0 because
+TCPSEND: SOCKET ID OPEN FAILED         the connection is not established.
AT+TCPSEND=0,4097                       4097-byte data fails to be sent on socket 0 because
+TCPSEND: DATA LENGTH ERROR             data length exceeds the limit.
```

6.5 AT+TCPACK – Querying Status of Data Sent

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

Format

Type	Command	Response
Execute	AT+TCPACK=<n><CR>	<CR><LF>+TCPACK: <n>,<data_sent>, <acked_recv> <CR><LF> Or <CR><LF>ERROR<CR><LF>

	Or <CR><LF>+TCPACK: <n>, DISCONNECT<CR><LF> Or <CR><LF>+TCPACK: NO TCP LINK <CR><LF>

Parameter

<n>	Socket number, ranging from 0 to 4.
<data_sent>	Data successfully sent through this socket.
<acked_recv>	Data acknowledged by the receiver.

Example

```
AT+TCPACK=0                          20-byte data is transmitted from socket 0 and the receiver
+TCPACK: 0,20,20                     acknowledges 20-byte data.
AT+TCPACK=0                          128-byte data is transmitted from socket 0 and the receiver
+TCPACK: 0,128,120                   acknowledges 120-byte data.
AT+TCPACK=1                          No connection is set up on socket 1.
+TCPACK: 1,DISCONNECT
AT+TCPACK=2                          A UDP connection is set up on socket 2.
+TCPACK: NO TCP LINK
AT+TCPACK=6                          The socket number in the command is incorrect.
ERROR
```

The values of <data_sent> and <acked_recv> are unsigned 64-bit integers in decimal ASCII. The unit is byte.

6.6 +TCPRECV – URC Notifying TCP Data Received

To notify the received TCP data.

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

Format

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

Parameter

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

Example

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

6.7 AT+TCPREAD – Reading TCP Data

To read TCP data from the buffer.

If the length parameter is not included in the command, the length of data read is 1 by default.

Format

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

Parameter

<n>	Socket ID, ranging from 0 to 4
<length>	Maximum length of data allowed to read, ranging from 1 to 1024, byte. Length of data is 1 by default if the <length> parameter is not contained in this command.
<data>	Data that is read

Example

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


6.8 AT+TCPCLOSE – Closing TCP Connection

To close a TCP connection.

Formats

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

Parameter

<n>	Socket ID, ranging from 0 to 4
<result>	OK Link Closed

Example

```
AT+TCPCLOSE=1                 Close the TCP connection.
+TCPCLOSE: 1,OK               The TCP connection on socket 1 is closed successfully.
AT+TCPCLOSE=5                 Socket number error
ERROR
+TCPCLOSE: 0,Link             The server sends command to close TCP connection or the
Closed                        network encounters abnormality or weak signals.
```

6.9 AT+UDPSETUP – Setting up a UDP Connection

To set up a UDP connection

Use the **AT+XIIC=1** command to set up a PPP link before executing this command.

Format

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

	<CR><LF>+UDPSEND: <n>,<length><CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+UDPSETUP: ERROR<CR><LF> Or <CR><LF>+UDPSETUP: 0,ERROR1<CR><LF> Or <CR><LF>+UDPSEND: DATA LENGTH ERROR<CR><LF>

Parameter

<n>	Socket ID, ranging from 0 to 4
<ip>	Destination IP address, in xx.xx.xx.xx format or domain name format (www.*XXXX*.com)
<port>	Destination port ID in decimal ASCII code
<result>	OK FAIL

Example

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

+UDPSETUP:1,OK
AT+UDPSETUP=0,neowayjsr.oicp.net,60010    Set up a connection to neowayjsr.oicp.net,60010 on
OK                                        socket 0 Successful

+UDPSETUP: 0,OK
AT+UDPSETUP=0,58.60.184.213,11008
OK
                                          A TCP/UDP connection is already set up on socket 0.
+UDPSETUP: 0, ERROR1
AT+UDPSETUP=1,192.168.20.6,7000           Fail to set up the connection to 192.168.20.6,7000
OK                                        on socket 1 because it is unavailable.

+UDPSETUP: 1,FAIL
AT+UDPSETUP=5,192.168.20.6,6800           Socket ID is set incorrectly.
+UDPSETUP: ERROR
AT+UDPSETUP=0.58.60.184.213.10012         Punctuation mark is used incorrectly.
+UDPSETUP: ERROR
AT+UDPSET=0,58.60.184.213,10012           The AT command is not complete.
ERROR
```


6.10 AT+UDPSEND – Sending UDP Data

To send UDP data

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

Ensure that the UDP connection is set up before sending UDP data. Send AT+IPSTATUS to query the buffer size before sending UDP data.

Format

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

Parameter

<n>	Socket ID, ranging from 0 to 4. A UDP connection is established on the socket.
<length>	The length of the data to be sent, ranging from 1 to 1024, unit: byte.
<content>	data to be sent
<mode>	0: send data in ASCII mode. The escaped character is not supported currently. 1: send data in HEX mode.

Example

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

+UDPSEND: 0,2                           Data is sent successfully.
AT+UDPSEND=0,1025                       Fail to send 1025-byte data on socket 0 because
+UDPSEND: DATA LENGTH ERROR             data length exceeds the limit.
```

```
AT+UDPSEND=0,10                          After the data sending command is input and > is
>                                        returned, no data is entered in 1 minute. Then the
+UDPSEND: 0,OPERATION EXPIRED            expiration information is displayed.
```

6.11 +UDPRECV – URC Notifying UDP Data Received

To notify the received UDP data.

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

Format

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

Parameter

<n>	Socket ID, ranging from 0 to 4
<length>	Length of the data received
<data>	Data received, Add 0x0d 0x0a to the end of the data. Identify the end based on **<length>**.

Example

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

6.12 AT+UDPREAD – Reading UDP Data

To read UDP data

Format

Type	Command	Response
Execute	AT+UDPREAD=<n>[,<length>]<CR>	<CR><LF>+UDPREAD: <n>,<length>,<data> <CR><LF>OK<CR><LF>

Parameter

<n> Socket ID, ranging from 0 to 5
<length> Maximum length of data allowed to read, ranging from 1 to 1024 bytes.

Example

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

OK
```

6.13 AT+UDPCLOSE – Closing UDP Connection

To close the UDP connection

Format

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

Parameter

<n> socket ID, ranging from 0 to 4

Example

```
AT+UDPCLOSE=1                          The UDP link on socket 1 is closed successfully.
OK

+UDPCLOSE:1,OK
AT+UDPCLOSE=5                          Socket number error
+UDPCLOSE:ERROR
```

6.14 AT+IPSTATUS – Querying TCP/UDP Socket Status

To query the TCP/UDP socket status

Format

Type	Command	Response
Execute	AT+IPSTATUS=<n><CR>	<CR><LF>+IPSTATUS: <n>,<status>[,<type>,<send-buffer-size>] <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	Socket ID, ranging from 0 to 4
<status>	Socket status, CONNECT or DISCONNECT, or CONNECTING or DISCONNECTING
<type>	Socket type, TCP or UDP
<send-buffer-size>	The size of the available send buffer on the module, in decimal ASCII mode, unit: byte.

Example

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

OK
AT+IPSTATUS=1                          A UDP connection has been set up on socket 1.
+IPSTATUS: 1,CONNECT,UDP,4096

OK
AT+IPSTATUS=0                          No TCP or UDP connection is set up on socket 1.
+IPSTATUS: 0,DISCONNECT

OK
```


7 Transparent TCP/UDP Commands

7.1 AT+TCPTRANS – Setting up a Transparent TCP Connection

To set up a transparent TCP connection

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

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

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

Format

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

Parameter

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

Example

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

+TCPTRANS:OK
```

```
AT+TCPTRANS=neowayjsr.oicp.net,60010          A transparent TCP connection is set up
OK                                            successfully by using domain name.

+TCPTRANS:OK
AT+TCPTRANS=220.199.66.56,
+TCPTRANS: ERROR                              The command format is incorrect.
AT+TCPTRANS=220.199.66.56,6800                Fails to set up a transparent TCP
OK                                            connection.

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

7.2 AT+UDPTRANS – Setting up a Transparent UDP Connection

To set up a transparent UDP link

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

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

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

Format

Type	Command	Response
Execute	AT+UDPTRANS=<ip>,<port><CR>	<CR><LF>OK<CR><LF> <CR><LF>+UDPTRANS: <result><CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<ip>	destination IP address, in xx.xx.xx.xx format or in domain name format (www.*XXXXX*.com).
<port>	destination port ID in decimal ASCII code
<result>	OK FAIL ERROR

Example

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

+UDPTRANS:OK
AT+UDPTRANS=neowayjsr.oicp.net,60010          A transparent UDP connection is set up by using
OK                                            domain name successfully.

+UDPTRANS:OK
AT+UDPTRANS=220.199.66.56,                    The command format is incorrect.
+UDPTRANS:ERROR
AT+UDPTRANS=220.199.66.56,6800                Fails to set up a transparent UDP link.
OK

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

7.3 AT+TRANSCLOSE – Closing Transparent Socket

To close the transparent socket.

Format

Type	Command	Response
Execute	AT+TRANSCLOSE<CR>	<CR><LF>+TRANSCLOSE: <n>,OK Or <CR><LF>ERROR<CR><LF>
URC	+TCPTRANS: Link Closed	

Parameter

N/A.

Example

```
AT+TRANSCLOSE
+TRANSCLOSE: 0,OK                             A transparent TCP socket is closed successfully.
AT+TRANSCLOSE
ERROR                                         No transparent TCP/UDP connection is set up.
AT+TRANSCLOSE
+TRANSCLOSE: 1,OK                             A transparent UDP socket is closed successfully.

+TCPTRANS: Link Closed                        The transparent TCP socket is closed by the server or
                                              because of network abnormality.
```

7.4 AT+IPSTATUS – Querying TCP/UDP Socket Status

To query the TCP/UDP socket status

For UDP socket, this command can query only whether the module sets up a connection through an AT command.

Format

Type	Command	Response
Execute	AT+IPSTATUS<CR>	<CR><LF>+IPSTATUS: <STATUS>[,<TYPE>,<send-buffer-size>]<CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

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

Example

```
AT+IPSTATUS                                    Query TCP socket status
+IPSTATUS: CONNECT,TCP,4096
OK
AT+IPSTATUS                                    Query UDP socket status.
+IPSTATUS: CONNECT,UDP,0
OK
```


8 TCP Server Commands

8.1 AT+TCPLISTEN – Setting TCP Listening on the Server

To set the TCP listening function of the server.

Format

Type	Command	Response
Set	AT+TCPLISTEN=<port><CR>	<CR><LF>+TCPLISTEN: <socket>,OK Or <CR><LF>+TCPLISTEN: bind error Or <CR><LF> Listening... <CR><LF> Or <CR><LF>+TCPLISTEN: ERROR
Query	AT+TCPLISTEN?<CR>	<CR><LF>+TCPLISTEN: listening status<CR><LF> <CR><LF>+TCPLISTEN: not listening<CR><LF>

Parameter

<port>	Port number
<socket>	Socket number

Example

```
AT+TCPLISTEN=6800                    Listening port ID: 6800
+TCPLISTEN: 0,OK                     The server starts listening
AT+TCPLISTEN=6800                    Listening port ID: 6800
+TCPLISTEN: bind error               Fails to bind
AT+TCPLISTEN=6800
Listening...                         Transparent listening is set already.
AT+TCPLISTEN=0
+TCPLISTEN: ERROR                    The port ID is invalid.
AT+TCPLISTEN?                        Query the listening status. Here the server is in
+TCPLISTEN: listening status         the listening status.
AT+TCPLISTEN?                        Query the listening status. Here the server is not
+TCPLISTEN: not listening            in the listening status.
Connect                              Receive the connection request from the client.
AcceptSocket=1,ClientAddr=119.123    AcceptSocket indicates the socket ID on the module,
.77.133,ClientPort=8000              and 119.123.77.133 is the IP address of the client.
```


8.2 AT+CLOSELISTEN – Closing the Listening Socket

To close the listening socket and close all connections.

Format

Type	Command	Response
Execute	AT+CLOSELISTEN<CR>	<CR><LF>+CLOSELISTEN: <Socket>,local link closed
URC	+CLOSELISTEN: <Socket>,local link closed	

Parameter

N/A.

Example

```
+CLOSELISTEN: 0,local link closed          The server closes the socket or network
                                           abnormalities occur.
AT+CLOSELISTEN                             The local socket is closed if there is any
+CLOSELISTEN: 0,local link closed          connection to the client.
```

8.3 AT+CLOSECLIENT – Closing Connections with the Client

To close all connections with the client.

Format

Type	Command	Response
Execute	AT+CLOSECLIENT[=<socket>]<CR>	<CR><LF>+CLOSECLIENT: <socket>,remote link closed<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CLOSECLIENT: All remote link closed<CR><LF>

Parameter

<socket> Socket number

Example

```
AT+CLOSECLIENT
+CLOSECLIENT: 1,remote link closed        All connections with the client are closed
                                         successfully.
+CLOSECLIENT: 2,remote link closed
AT+CLOSECLIENT=1
+CLOSECLIENT: 1,remote link closed        Close the connection with the client on socket 1.
AT+CLOSECLIENT=1
ERROR                                    No client on socket 1.
AT+CLOSECLIENT
+CLOSECLIENT: All remote link            All clients are closed.
closed
```

8.4 +TCPRECV(S) – Receiving Data from the Client

To receive data from the client.

Format

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

Parameter

<n> Socket ID, ranging from 0 to 4.
<length> The length of the data received.
<data> The data received. Add 0x0d 0x0a to the end of the data. We can identify the end based on <length>.

Example

```
+TCPRECV(S): 1,10,1234567899             Socket 1 receives 10-byte data in char format from the client.
+TCPRECV(S):
0,10,30313233343536373839                Socket 0 receives 10-byte data in hexadecimal ASCII format.
```

- Additional (s) makes this command different from the receive mode of the client mode in format.

- This function might not work because carriers' networks do not support.

8.5 AT+TCPSENDS－Sending Data to the Client

To send data to the client

Ensure that a TCP connection has been set up before sending TCP data.

Format

Type	Command	Response
Execute	AT+TCPSENDS=<socket>[,<length>]<CR>	<CR><LF>> <CR><LF>OK<CR><LF> <CR><LF>+TCPSENDS: <socket>[,<length>]<CR><LF> Or <CR><LF>> <CR><LF>+TCPSENDS: <result> <CR><LF> Or <CR><LF>> <CR><LF>+TCPSENDS: <socket>, OPERATION EXPIRED<CR><LF> Or <CR><LF>+TCPSENDS: <socket> is not link<CR><LF>

Parameter

<socket>	The value of AcceptSocket, that is, the socket of the module. See the description of the AT+TCPLISTEN command.
<length>	Length of the data to be sent, ranging from 1 to 2000, unit: byte.
<result>	ERROR Buffer not enough,439

Example

```
AT+TCPSENDS=0,10
>
OK                                  10-byte data is successfully sent through socket 0.

+TCPSENDS: 0,10
AT+TCPSENDS=0,536                   Failed to send 536-byte data through socket 0 (e.g.
>                                   1234567890...) because the internal buffer is
+TCPSENDS: Buffer not enough,439    insufficient.
```

	Send 21-byte data on socket 0.
AT+TCPSENDS=0 > OK	(e.g.: 012345678901234567890 or 30313233343536373839303132333435363738393 0)
+TCPSENDS: 0,21	(When <length> is not included in the command, end the data with Ctrl+Z. The length of data sent is 2000 at most)
AT+TCPSENDS=0,1024 >	Send TCP data.
+TCPSENDS: ERROR	Congestion.
AT+TCPSENDS=0,10 +TCPSENDS: 0 is not link AT+TCPSENDS=0 +TCPSENDS: 0 is not link	No connection is set up on socket 0.
AT+TCPSENDS=0,5 > +TCPSENDS: 0,OPERATION EXPIRED	No data is input within 60 seconds after > is displayed.

9 FTP AT Commands

9.1 AT+FTPLOGIN – Logging in to the FTP Server

To log in to the FTP server.

The FTP functions cannot be used together with the internal protocol stack TCP/UDP function. Data can be read or written on the FTP server only after login.

Format

Type	Command	Response
Execute	AT+FTPLOGIN=<ip>,<port>,<user>,<pwd><CR>	<CR><LF>OK<CR><LF> <CR><LF>+FTPLOGIN: <result><CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+FTP:Server Ctrl Link Disconnect Or <CR><LF>ERROR<CR><LF>

Parameter

<ip>	IP address of the FTP server.
<port>	Port ID of the FTP server, 21
<user>	The user name to log in to the FTP server. Its length cannot exceed 100 bytes in ASCII code and comma (,) is not allowed in a name.
<pwd>	The password for a user to log in to the FTP server. Its length cannot exceed 100 bytes in ASCII code and contain comma (,) is not allowed in it.
<result>	Have Logged In: The user has logged in to the FTP server. AT Busy: The module is still executing last FTP AT command. User logged in: Logged into the FTP server successfully. 530 Not logged in: Failed to log in to the FTP server because the account or password is incorrect. GPRS DISCONNECTION: A PPP link is not set up yet.

Example

```
AT+FTPLOGIN=219.134.179.52,21,user1,pwd2009                    Log in to the server
OK
```

```
+FTPLOGIN: User logged in                          Successfully
AT+FTPLOGIN=58.60.184.213,21,neowayftp,neow
ayftp
OK

                                                   Fail to log in to the FTP server.
+FTP: Server Ctrl Link Disconnect
+FTPLOGIN: Error
```

9.2 AT+FTPLOGOUT – Logging out from the FTP Server

To log out from the FTP server.

Format

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

Parameter

N/A.

Example

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

9.3 AT+FTPSIZE – Obtaining File Size on FTP Server

To obtain the size of a file on the FTP server.

Format

Type	Command	Response
Execute	AT+FTPSIZE=<filename><CR>	<CR><LF>+FTPSIZE: <size><CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+FTPSIZE: <result> <CR><LF>

- If PPP is not activated, ERROR will be returned directly.
- If the file does not exist, ERROR is returned directly.

Parameter

<filename>	File name. The file path is relative to the FTP root directory.
<size>	File size, unit: byte.
<result>	Error Not Login: The user has not logged in to the FTP server. AT Busy: The module is still executing last FTP AT command. Error: A PPP link is not set up yet.

Example

```
AT+FTPSIZE=test.txt                    Obtain the size of test.txt in the FTP root directory.
+FTPSIZE: 1024
```

9.4 AT+FTPGET – Downloading Data from the FTP Server

To download data from the FTP server.

Format

Type	Command	Response
Execute	AT+FTPGET=<dir&filename>,<type>,<Content or Info>[,offset[,length]]<CR>	<CR><LF>+FTPGET:<length><data><CR><LF> <CR><LF>+FTP:Server Data Link Disconnect <CR><LF> <CR><LF>+FTPGET: OK.total length is <n> Or <CR><LF>+FTPGET: OK.total length is

	<m><CR><LF> Or <CR><LF>+FTPGET: Error<CR><LF>

Parameter

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

Example

```
AT+FTPGET=,1,2                                                    Obtain information
+FTPGET: 446,drw-rw-rw-   1 user    group       0 Apr 14 15:55 .  in the root
drw-rw-rw-   1 user    group         0 Apr 14 15:55 ..            directory.
-rw-rw-rw-   1 user    group    1238528 Jan 14 10:36 1M.doc
-rw-rw-rw-   1 user    group         10 Jan 15 15:01 test.txt


+FTPGET: OK.total length is 446

+FTP:Server Data Link Disconnect

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

+FTP:Server Data Link Disconnect
                                                                  Server Data Link
+FTPGET: OK.total length is 65                                    Disconnect
AT+FTPGET=test.txt,1,1                                            Upload 10-byte data.
+FTPGET: 10,0123456789

+FTP:Server Data Link Disconnect

+FTPGET: OK.total length is 10
AT+FTPGET=test.txt,1,1,2                                          Read all data after the
+FTPGET: 8,23456789                                               first byte.
```



```
+FTP:Server Data Link Disconnect

+FTPGET: OK.total length is 8
AT+FTPGET=test.txt,1,1,2,4                          Read 4-byte data after
+FTPGET: 4,2345                                     the first byte.

+FTP:Server Data Link Disconnect

+FTPGET: OK.total length is 4
AT+FTPGET=test.txt,1,3                              Obtain the length of the
+FTPGET: OK.file length is 10                       test.txt file.
AT+FTPGET=test.txt,1,2,1,2,3                        The command format is
+FTPGET: Error                                      incorrect.
```

9.5 AT+FTPPUT – Uploading Data to the FTP Server:

To upload data to the FTP server.

Format

Type	Command	Response
Execute	AT+FTPPUT=<dir&filename>,<type>,<mode>[,<size>]<CR>	<CR><LF>> <CR><LF>+FTPPUT: OK,<n><CR><LF> Or <CR><LF>+FTPPUT: <result><CR><LF>

Parameter

<dir&filename>	The name and path of the file to be uploaded The file path is relative to the FTP root directory.
<type>	File transfer mode. 1: ASCII 2: Binary
<mode>	Operation mode 1: STOR mode Create a file on the FTP server and write the data to the file. If the file exists, the original file is overwritten. 2: APPE mode Create a file on the FTP server and write the data to the file. If the file exists, the data is attached to the end of the file. 3: DELE mode Delete a file.
<size>	Data length. The data length cannot exceed 8192.
<result>	Error: The format of the AT command is incorrect or the last FTP command is not executed successfully. Error Not Login: The user has not logged in to the FTP server. AT busy: The module is still executing last FTP AT command.

SIZE Error: The value of <length> is greater than 8192.
Error TimeOut: No data input for long time.
Delete File OK: The file is deleted successfully.

Example

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

9.6 AT+FTPSTATUS – Querying FTP Connection Status

To query the FTP connection status.

Format

Type	Command	Response
Execute	AT+FTPSTATUS<CR>	<CR><LF>+FTPSTATUS: <status>,<ip>,<port><CR><LF> Or <CR><LF>+FTPSTATUS: 0<CR><LF>

Parameter

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

Example

```
                                   Query the FTP connection status.
AT+FTPSTATUS                       The module is successfully connected to the FTP
+FTPSTATUS: 1,119.139.221.66,21    server. The IP address of the FTP server is
                                   119.139.221.66 and the port is 21.
AT+FTPSTATUS
+FTPSTATUS: 0                      Not logged in.
```


10 HTTP Commands

10.1 AT+HTTPPARA – Setting HTTP Parameters

To set HTTP parameters

Format

Type	Command	Response
Set	AT+HTTPPARA=<para>,<para_value><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)

<para_value> Value of **<para>**. The value of url contains at most 2048 bytes and url supports domain name translation.

Example

```
AT+HTTPPARA =url,www.neoway.com.cn/en/index.aspx          Set the Neoway homepage as the
OK                                                       URL. The 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
OK                                                       80.
AT+HTTPPARA=port,8080                                    Set the destination port ID to
OK                                                       8080.
```


10.2 AT+HTTPSETUP – Setting Up 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> or <CR><LF>ERROR<CR><LF>

Parameter

N/A.

Example

```
AT+HTTPSETUP                                Set up an HTTP connection
OK                                          Successful
AT+HTTPSETUP                                Set up an HTTP connection
ERROR                                       DNS translation fails
```

10.3 AT+HTTPACTION – Executing HTTP Request

To execute an HTTP request

Comply with the HTTP protocol when defining packets.

Format

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

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

Parameter

<mode>	HTTP request mode, available value can be 0, 1, 2, 99 0: GET (When downloading a file by GET method, you can set the offset and size parameters to achieve a segmented download.) 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 HTTP POST

Example

```
AT+HTTPPARA=url,www.neoway.com.cn/en/index.aspx                Set the destination path.
OK                                                             Set up an HTTP connection.
AT+HTTPSETUP
OK
AT+HTTPACTION=0                                                GET request
OK

                                                               Receive the response from
+HTTPRECV:                                                     the 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
/*neoway homepage, html format, 13842 bytes*/                  the connection.
……..
/* neoway homepage*/
```

```
+HTTPCLOSED: HTTP Link Closed
AT+HTTPPARA =url,www.neoway.com.cn/en/index.aspx                Set the destination path.
OK                                                              Set up an HTTP connection
AT+HTTPSETUP                                                    HEAD request
OK
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: HTTP 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
OK                                                             as 8080.
AT+HTTPSETUP                                                   Set up an HTTP connection
OK
AT+HTTPACTION=2,23                                             POST request.
>MAC=NEOWAY&DATA=0123456                                       Send 23 bytes; enter the
OK                                                             contents to be uploaded
                                                               after > is displayed.
+HTTPRECV:
HTTP/1.1 200 OK                                                Receive the response from
Cache-Control: private, max-age=0                              the 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
X-Powered-By: ASP.NET                                          file containing the
Date: Thu, 28 Nov 2013 03:41:52 GMT                            uploaded content NEOWAY and
Connection: close                                              0123456.
Content-Length: 98

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

+HTTPRECV:
HTTP/1.1 200 OK
Cache-Control: private
Content-Length: 13842                                               Receive the response from
Content-Type: text/html; charset=utf-8                              the HTTP server.
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
Connection: close                                                   The server disconnects with
                                                                    the module after it
                                                                    finishes responding.
+HTTPCLOSED: HTTP Link Closed
```

10.4 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.

Format

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

Parameter

<result> HTTP Link Closed

Example

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

                                    Socket is closed successfully.
+HTTPCLOSE: HTTP Link Closed
```

10.5 +HTTPRECV – URC 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
URC	<CR><LF>HTTPRECV: <CR><LF><datas>

Parameter

<data> Data received through the HTTP socket

Example

```
+HTTPRECV:                                                    Report the data
HTTP/1.1 200 OK                                              received from the HTTP
Cache-Control: private                                       connection.
Content-Length: 13842
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
Connection: close


+HTTPCLOSED:
HTTP Link Closed
```

10.6 +HTTPCLOSED – HTTP Socket Closed

URC of the HTTP socket closing

Format

Type	Command
URC	<CR><LF>+HTTPCLOSE: Link Closed<CR><LF>

Parameter

N/A

Example

```
     +HTTPCLOSED: HTTP Link Closed                URC of the HTTP socket closing
```

11 HTTPS Commands

11.1 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
Set	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

<para_value> The value of <para>. The value of url contains at most 1022 bytes and url supports domain name translation. A pair of quotation marks is required for url.

Example

```
AT+HTTPSPARA=url,www.alipay.com/index.html                          Set the Alipay
OK                                                                  homepage as the
                                                                    URL. The URL
                                                                    supports domain
                                                                    name translation.
AT+HTTPSPARA=url,"132.188.73.13/prodreg/beginRegistration.action"   Set URL.
OK
AT+HTTPSPARA=port,443                                               Set the destination
OK                                                                  port ID to 443.
```


11.2 AT+HTTPSSETUP – Setting up an HTTPS Connection

To set up an HTTPS connection

The connection is set up successfully only after setting the destination address and port ID correctly.

Ensure that PPP dialing is successful before an HTTPS connection is set up.

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
AT+HTTPSSETUP                Set up an HTTPS connection.
ERROR                        Failed
```

11.3 AT+HTTPSACTION – Executing HTTPS Request

To execute an HTTPS request

Comply with the HTTPS protocol when defining packets.

Different status codes might be returned. For example, the server returns **405 Method Not Allowed** if the request methods is not supported.

Format

Type	Command	Response
Execute	AT+HTTPSACTION=<mode>[,<length><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>><post_content><CR><LF> <CR><LF>OK<CR><LF> Or <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**
<size>	Size of file to be downloaded in GET mode
<post_content>	Content sent through HTTPPOST

Example

```
AT+HTTPSPARA=url,"www.alipay.com/ index.html"                    Set the destination address. The
                                                                 default port is 443.
OK
AT+HTTPSSETUP                                                    Set up an HTTPS connection.

OK
AT+HTTPSACTION=0                                                 GET request

OK

+HTTPSRECV:                                                      Receive the response from the
HTTP/1.1 200 OK                                                  HTTPS server.
Server: spanner/1.0.6
Date: Fri, 01 Aug 2014 03:02:34 GMT
Content-Type: text/html; charset=gbk
Content-Length: 56028
Connection: close
Last-Modified: Wed, 23 Jul 2014 07:51:38 GMT
Strict-Transport-Security: max-age=31536000
Accept-Ranges: bytes
Set-Cookie:
spanner=Z761rjOVBLsAdq8c3/Rwpd9j7dWQJZjm;path=/;secure;
/*alipay homepage, html format, 56028 bytes */
……..
/* alipay homepage*/
```

```
+HTTPSCLOSED: HTTPS Link Closed                          URC that the connection is closed
                                                         after the server responds to the
                                                         request.
AT+HTTPSPARA=url,"www.alipay.com/index.html"

OK
AT+HTTPSSETUP

OK
AT+HTTPSACTION=1                                         Set the destination address. The
                                                         default port is 443.
OK                                                       Set up an HTTPS connection.

+HTTPSRECV:
HTTP/1.1 200 OK
Server: spanner/1.0.6
Date: Fri, 01 Aug 2014 03:05:41 GMT
Content-Type: text/html; charset=gbk                     HEAD request
Content-Length: 56028
Connection: close
Last-Modified: Wed, 23 Jul 2014 07:51:40 GMT
Strict-Transport-Security: max-age=31536000              HTTPS server response
Accept-Ranges: bytes
Set-Cookie:
spanner=G0TDss3KC108k1dgppqS1y6qNx1FfX2V;path=/;secure;

+HTTPSCLOSED: HTTPS Link Closed

AT+HTTPSPARA=url,"www.alipay.com/index.html"

OK
AT+HTTPSSETUP                                            Set URL.

OK
AT+HTTPSACTION=99,69                                     Use the default port 443 to set
>HEAD /index.html HTTP/1.1                               up an HTTPS connection.
HOST:www.alipay.com
connection: close                                        Use custom packet mode to send
                                                         69-byte packets.
OK

+HTTPSRECV:
HTTP/1.1 200 OK
Server: spanner/1.0.6                                    Receive HTTPS server response.
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;

+HTTPSCLOSED: HTTPS Link Closed                          The server finishes responding
                                                         and closes the connection.
```

11.4 AT+HTTPSCLOSE – Closing HTTPS Socket

To close an HTTPS socket

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

Format

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

Parameter

N/A

Example

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

+HTTPSCLOSE: HTTPS Link Closed
+HTTPSSEND: FAIL                           The module fails to send HTTPS data.
```

11.5 +HTTPSRECV – URC Notifying HTTPS Data Received

To notify the received HTTPS data.

Format

Type	Command
URC	<CR><LF>+HTTPSRECV: <datas><CR><LF>

Parameter

<datas> Data that the HTTPS socket receives

Example

```
+HTTPSRECV:
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;                    URC indicates the HTTPS data is received.
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
Connection: close

+HTTPSCLOSED: HTTPS Link Closed
```

11.6 +HTTPSCLOSED – URC Notifying HTTPS Connection Closed

To notify the HTTPS connection is closed.

Format

Type	Command	Response
URC	<CR><LF>+HTTPSCLOSED: HTTPS Link Closed <CR><LF>	

Parameter

N/A

Example

```
+HTTPSCLOSED: HTTPS Link Closed       the HTTPS socket connection is closed.
```


12 MQTT Commands

12.1 AT+MQTTCONNPARAM – User Parameter Settings

To set ID, user name, and password.

This command is invalid if an MQTT connection has been set up. The settings by this command are not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+MQTTCONNPARAM=<"clientdID">,<"username">,<"password">[,<"pubk">]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MQTTCONNPARAM?<CR>	<CR><LF>+MQTTCONNPARAM:<"clientID">,<"username">,<"password"> <CR><LF>OK<CR><LF>
Test	AT+MQTTCONNPARAM=?<CR>	<CR><LF>+MQTTCONNPARAM:<"clientID">,<"username">,<"password">[,<"pubk">] <CR><LF>OK<CR><LF>

Parameter

<"clientID"> Device ID, 256 bytes at most
<"username"> User name, 512 bytes at most
<"password"> Password, 256 bytes at most
<"pubk"> The adaptation parameter used to set up an MQTT connection.

Example

```
AT+MQTTCONNPARAM="C_201801021127","lixytest/thing0        Parameters are set successfully.
1","0lSoY/eYnlSqUeAsbAKKQ/ACmipZwEw9H7Ff0h1kOps="
OK
```

12.2 AT+MQTTWILLPARAM – Will Settings

To set will parameters.

This command is invalid if an MQTT connection has been set up. The settings by this command are not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+MQTTWILLPARAM=<retained>,<qos>,<"topicname">,<"message"><CR>	<CR><LF>+GNSSSTATE: <status><CR><LF> Or <CR><LF>OK<CR><LF>
Query	AT+MQTTWILLPARAM?<CR>	<CR><LF>+MQTTWILLPARAM:<retained>,<qos>,<"topicname">,<"message"> <CR><LF>OK<CR><LF>
Test	AT+MQTTWILLPARAM=?<CR>	<CR><LF>+MQTTWILLPARAM:<retained(0~1)>,<qos(0~1)>,<"topicname">,<"message"> <CR><LF>OK<CR><LF>

Parameter

<retained>	Retain mark, digit, 0 or 1
<qos>	Quality of service, only 0 and 1 are supported
<"topicname">	Will topic, at most 128 bytes
<"message">	Will Message, at most 1024 bytes

Example

```
AT+MQTTWILLPARAM=0,1,"lixytopic","by
by"                                    The will is set successfully.
OK
```

12.3 AT+MQTTCONN – Connection Command

To connect to the MQTT server.

The module embeds reconnection mechanism. Do not set up connection manually during reconnection until the module disconnects to the server or reports a disconnection.

Format

Type	Command	Response
Execute	AT+MQTTCONN=<"host">,<clean>,<keep_alive><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MQTTCONN?<CR>	<CR><LF>+MQTTCONN:<"ip:port">,<clean>,<keep_alive> <CR><LF>OK<CR><LF>
Test	AT+MQTTCONN=?<CR>	<CR><LF>+MQTTCONN:<"host(url:port)">,<clean(0~1)>,<keep_alive(20~180)> <CR><LF>OK<CR><LF>

- Do not repeat the operation of setting up a connection when waiting for the return value.
- If the module reports +MQTTDISCONNED: Link Closed after an MQTT connection is set up successfully, send to command to set up a connection manually.

Parameter

<"host">	Server address (URL:port)
<clean>	whether to clean session, digit type, 0-Not clean (default) 1-Clean
<keep_alive>	keepAlive time, ranging from 20 to 180, unit second

Example

```
AT+MQTTCONN="121.43.166.63:1883",0,60                Connect to the MQTT server
OK                                                   successfully.
```

12.4 AT+MQTTSUB –Subscription

To subscribe to a topic

After the modules fails to subscribe to a topic, query the status of the network and the MQTT connection and then perform next operation. When the network is in poor quality, the modules might return value late.

Format

Type	Command	Response
Execute	AT+MQTTSUB=<"topicname">,<qos><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MQTTSUB?<CR>	<CR><LF>+MQTTSUB:<"topicname">,<qos> <CR><LF>OK<CR><LF>
Test	AT+MQTTSUB=?<CR>	<CR><LF>+MQTTSUB:<"topicname">,<qos(0~1)> <CR><LF>OK<CR><LF>

The query format of this command can obtain only QoS and topic of last subscription.

Parameter

<"topicname">	Topic to subscribe to, 128 bytes at most
<qos>	Quality of service, only 0 and 1 are supported

Example

```
AT+MQTTSUB="/lixytopic",1                                    Subscribe to the topic successfully.
OK
```

12.5 AT+MQTTUNSUB – Cancelling a Subscription

To cancel a subscription.

After the modules fails to cancel a subscription, query the network status. When the network is in poor quality, the modules might return value late.

Format

Type	Command	Response
Execute	AT+MQTTUNSUB=<"topicname"><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<"topicname"> Topic name of the subscription to be canceled.

Example

```
AT+MQTTUNSUB="/lixytopic"                                    Cancel a subcription.
OK
```

12.6 AT+MQTTPUB – Topic Publish

To publish a topic.

When the network is in poor quality, the modules might return value late.

Format

Type	Command	Response
Execute	AT+MQTTPUB=<retained>,<qos>,<"topicname">,<"message">[,<formate>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+MQTTPUB=?<CR>	<CR><LF>+MQTTPUB:<retained(0~1)>,<qos(0~1)>,<"topicname">,<"message">[,format] <CR><LF>OK<CR><LF>

Parameter

<retained> Retain mark, digit type, 0 and 1
<qos> Quality of service, only 0 and 1 are supported.
<"topicname"> Topic name, 128 bytes at most
<"message"> Message, 1024 bytes at most
<formate> Message format
0: Character string
1: HEX character string

Example

```
AT+MQTTPUB=1,1,"/lixytopic","123321HELLO
"
OK
```



```
+MQTTSUB:3,"/lixytopic",11,123321HELLO
AT+MQTTPUB=1,1,"/650063","A120",0
OK                                              The topic is published successfully.
                                                The server issues the topic.
+MQTTSUB:1791,"/650063",4,A120

AT+MQTTPUB=1,1,"/650063","A120",1
OK

+MQTTSUB:1792,"/650063",2,?
```

12.7 AT+MQTTPUBIN – Topic Publish (HEX data)

To publish a topic in HEX format.

When the network is in poor quality, the modules might return value late.

Format

Type	Command	Response
Query	AT+MQTTPUBIN=<retained>,<qos>,<"topicname">,<message_len><CR>	<CR><LF>> <CR><LF>input HEX data <CR><LF>OK<CR><LF>
Test	AT+MQTTPUBIN=?<CR>	<CR><LF>+MQTTPUBIN:retained,qos,"topic",msg_len <CR><LF>OK<CR><LF>

Parameter

<retained>	Retain mark, digit type, 0 and 1
<qos>	Quality of service, only 0 and 1 are supported.
<"topicname">	Topic name, 128 bytes at most
<"msg_len">	Length of the message published in HEX format, ranging from 1 to 512 bytes.
<hex_message>	Content of the message in HEX format. Input the content after the > character.

Example

```
AT+MQTTPUBIN=1,1,"/650063",2
>
(Input the HEX data)
OK

+MQTTSUB:1793,"/650063",2,1122
```


12.8 AT+MQTTDISCONN – Disconnecting to the MQTT Server

To disconnect to the MQTT server and release resources

Format

Type	Command	Response
Execute	AT+MQTTDISCONN<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>+CME ERROR:<err>

Parameter

<err> error codes
49: Not support
50: Fail to execute the command.
51: Insufficient memory
52: Not Allowed
53: Invalid Parameter

Example

```
AT+MQTTDISCONN                          To disconnect to the MQTT server and release resources
OK
```

- After the device disconnects to the MQTT server proactively, it releases the MQTT resources and clears parameter settings.
- To publish messages after disconnecting, configure parameters and set up a connection again.
- Error will occur if the connection is closed twice.

12.9 +MQTTSUB – URC Notifying Topic Content

To notify content of the topic received from the server. When the network is in poor quality, the modules might return value late.

Format

Type	Command
URC	+MQTTSUB:<message_id>,<"topicname">,<message_len>,<message><CR>

Parameter

<message_id>	Message ID
<"topicname">	Topic name
<message_len>	The length of the data received
<message>	Data received

Example

```
+MQTTSUB:2,"/lixytopic",11,123321HELO          Receive messages published by the topic subscribed to.
```

12.10 +IMQTTDISCONN – URC Notifying MQTT Connection Closed

To notify that the MQTT connection is closed.

If the module actively disconnects from the server, no URC will be prompted.

Format

Type	Command
URC	<CR><LF>+MQTTDISCONNED: Link Closed<CR>

Parameter

N/A.

Example

```
+MQTTDISCONNED: Link Closed                    The connection is closed.
```

12.11 AT+MQTTSTATE – Querying MQTT Connection Status

To query the status of the MQTT connection. The setting by this command is not saved after the module is powered off. You need to enable the URC of connection status every time the MQTT connection is established.

Format

Type	Command	Response
Query	AT+MQTTSTATE?<CR>	<CR><LF>+MQTTSTATE: <state><CR><LF> <CR><LF>OK<CR><LF>
Set	AT+MQTTSTATE=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<mode> Switch of the URC of the connection status
0: disabled (default)
1: enabled

<state> Reconnection status
0: the MQTT connection is closed.
1: the MQTT connection is established.
2: The device is reconnecting to the MQTT server.
3: The device starts to reconnect to the MQTT server.
4: The device fails to reconnect to the MQTT server.
The 3 value of <status> cannot be queried since the status changes fast.

Example

```
AT+MQTTSTATE?

+MQTTSTATE:2                    The device is reconnecting to the MQTT server.

OK
AT+MQTTSTATE?

+MQTTSTATE:1                    The device is connected to the MQTT server.

OK
AT+MQTTSTATE?
                                The device is disconnected to the MQTT server.
+MQTTSTATE:0
```



```
OK
AT+MQTTSTATE=1                  Enable the URC of MQTT connection status.

OK

+MQTTSTATEURC: 0                0: Network disconnection

+MQTTSTATEURC: 3                3: the device starts to reconnect to the MQTT server

+MQTTSTATEURC: 2                2: the device is connecting to the MQTT server.

+MQTTSTATEURC: 4                2: the device fails to connect to the MQTT server.
```

13 MOTA Commands

13.1 AT+MOTAMODE－Setting MOTA Mode and URC

To set the download mode of MCU OAT (MOTA) upgrade package and the switch of the URC notifying the MOTA download status.

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

If automatic download is enabled, the module verifies the issue number of the MCU software after detecting a later version and then downloads the MCU software automatically.

By default, the automatic mode is enabled and the URC is disabled unless this command is issued to set the parameters during operation.

Format

Type	Command	Response
Set	AT+MOTAMODE=<auto_update>,<stat_urc><CR>	<CR><LF>OK<CR><LF>
Query	AT+MOTAMODE?<CR>	<CR><LF>+MOTAMODE: <auto_update>,<stat_urc> <CR><LF>OK<CR><LF>
Test	AT+MOTAMODE=?<CR>	<CR><LF>+MOTAMODE: (listofsupport<auto_update>s),(listofsupport<progress_urc>s),(listofsupport<stat_urc>s) <CR><LF>OK<CR><LF>

Parameter

<auto_update>	To specify whether to enable automatic download 0: Disable automatic download 1: Enable automatic download (default)
<stat_urc>	To specify whether to enable URC of status 0: Disable URC (Default) 1: Enable URC

Example

```
AT+MOTAMODE=0,1                 Disable automatic download and enable the URC.
OK
AT+MOTAMODE?
+MOTAMODE: 0,1                  Query the setting of automatic download and the URC setting.
OK
AT+MOTAMODE=?
+MOTAMODE: (0-1),(0-            Query the available value ranges of parameters.
1)
OK
```

13.2 AT+MOTADL – Downloading MOTA Package

To set the URL and port of MCU OTA server and the MCU software version for version verification and download.

- If the automatic download mode is disabled, issue the execution format of this command. The module determines whether to download the program based on the version verification and download status.
- If the automatic download mode is enabled, issue the setting format of this command to trigger version detection. After detecting a version, the module determines whether to download the program based on the version verification and download status.

If the parameter of mode is set 0, the module clears cache, ceases querying or download, and restore the status to idle.

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

Format

Type	Command	Response
Set	AT+MOTADL=<mode>[,<URL>,<PORT>[,<MCU_ITEM>][,<MCU_VER>]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT+MOTADL<CR>	<CR><LF>OK<CR><LF>
Query	AT+MOTADL?<CR>	<CR><LF>+MOTADL: <mode>,<URL>,<PORT>,<MCU_ITEM>,<MCU_VER> <CR><LF>OK<CR><LF>

Parameter

<mode> mode

	0: reset status to factory setting 1: Download file
<URL>	server address in IP format or in domain name format, character string with a pair of quotation marks. E.g. "220.181.112.244" or "www.baidu.com". Its maximum length is 128 bytes. Ensure that the address you input is correct since there is no verification.
<PORT>	server port, ranging from 1 to 65535.
<MCU_ITEM>	Name of MCU program
<MCU_VER>	version of MCU If MCU version is omitted in the command, the default setting is 0000 and the detection and download is triggered every time when this command is issued.

Example

```
AT+MOTADL=1,"220.181.112.244",8000,"0001","0001"   Set the IP address and port of the server and
OK                                                 the name and version of the MCU program.
AT+MOTADL=1,"220.181.112.244",8000                 Set the IP address and port of the server and
OK                                                 omit the name and version of the MCU program.
AT+MOTADL?
+MOTADL: "220.181.112.244",8000,"0000","0001"      Query the settings.
OK
AT+MOTADL                                          Issue this command if the automatic download
OK                                                 mode is disabled.
AT+MOTADL=0
OK                                                 Reset all status to factory settings.
```

13.3 AT+MOTASTAT – Querying Download Status of MOTA Package

To query the download status of MCU OTA package.

After the automatic download is enabled and detecting a version, the module determines whether to download the MCU package automatically based on the version verification.

This command is used to preset the default value.

Format

Type	Command	Response
Query	AT+MOTASTAT?<CR>	No later version detected <CR><LF>+MOTASTAT: <STAT>[,<ERR CODE>] <CR><LF>OK<CR><LF> Later version detected <CR><LF>+MOTASTAT:

	<STAT>,<MCU_VER_SER>,<MD5> <CR><LF>OK<CR><LF>

Parameter

<STAT>	Status of OTA package detection/download. See Appendix A.2.
<MCU_VER_SER>	Version of MCU upgrade package that the server returns
<MD5>	Checksum that the server returns
<ERR CODE>	Error code, see Appendix A.3.

Example

```
AT+MOTASTAT?
+MOTASTAT: 10                                   Idle
OK
AT+MOTASTAT?
+MOTASTAT:                                      A later version is found.
12,"0002","813970bb0ba13495c2ab114e340b3633"
OK
AT+MOTASTAT?
+MOTASTAT: 13,04                                The module fails to query later version
OK                                              because failing to set up a connection.
AT+MOTASTAT?
+MOTASTAT: 15                                   No later version
OK
AT+MOTASTAT?
+MOTASTAT: 23,05                                The module fails to download because MD5
OK                                              checksum is incorrect.
AT+MOTASTAT?
+MOTASTAT: 25                                   Download completed
OK
```

13.4 +MOTASTAT – URC Notifying MOTA Download Status

To notify the status MCU OTA download status. The URC does not include the idle state and the downloading state. Execute the query command if you need to obtain the downloading state.

Format

Type	Command
URC	No later version detected <CR><LF>+MOTASTAT: <STAT>[,<ERR CODE>]<CR><LF> Later version detected

<CR><LF>+MOTASTAT: <STAT>,<MCU_VER_SER>,<MD5><CR><LF>

Parameter

<STAT>	Status of OTA package detection/download. See Appendix A.2.
<MCU_VER_SER>	Version of MCU upgrade package that the server returns
<MD5>	Checksum that the server returns
<ERR CODE>	Error code, see Appendix.0

Example

```
AT+MOTAMODE?
+MOTAMODE: 1,1                                          Enable the URC of MOTA download status and
OK                                                      automatic download.
AT+MOTADL="220.181.112.244",8000,"0001"
OK                                                      Trigger version detection

+MOTASTAT: 11

+MOTASTAT:12,"0002","813970bb0ba13495c2ab114e340b3633"
                                                        A later version is found.
+MOTASTAT: 21                                           Start to download.

AT+MOTASTAT?
+MOTASTAT: 22                                           The status obtained by manual query is
OK                                                      downloading.

                                                        Download completed
+MOTASTAT: 25
AT+MOTAMODE?
+MOTAMODE: 0,1                                          Enable the URC and disable automatic download.
OK

AT+MOTADL="220.181.112.244",8000,"0001"
OK                                                      Trigger version detection.

+MOTASTAT: 11

+MOTASTAT:12,"0002","813970bb0ba13495c2ab114e340b3633"  A later version is found.
                                                        Trigger download manually.
AT+MOTADL
OK                                                      Start to download.

+MOTASTAT: 21                                           The status obtained by manual query is
                                                        downloading.
AT+MOTASTAT?                                            The status obtained by manual query is
+MOTASTAT: 22                                           downloading.
OK

+MOTASTAT: 23,05
```


13.5 AT+MOTADLQ – Querying MOTA Download Process

To set the URC of MOTA download progress and query the progress.

Format

Type	Command	Response
Set	AT+MOTADLQ=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+MOTADLQ?<CR>	<CR><LF>+MOTADLQ: <dl_data>,<total_data> <CR><LF>OK<CR><LF>

Parameter

<mode>	to specify whether to enable URC of download progress 0: disable (default) 1: enable
<dl_data>	length of data cached. The length of this parameter is 4 bytes and the unit is byte.
<total_data>	length of OTA package. The length of this parameter is 4 bytes and the unit is byte.

Example

```
AT+MOTADLQ?
+MOTADLQ: 00002048,00123874                    Query the settings.
OK
```

Add leading 0 if the number the of digits is less than 8.

13.6 +MOTADLQ – URC Notifying MOTA Download Progress

To notify the MOTA download progress.

After the automatic download is enabled and detecting a version, the module determines whether to download the MCU package automatically based on the version verification.

Format

Type	Command
URC	<CR><LF>+MOTADLQ: <dl_data>,<total_data><CR><LF>

Parameter

<dl_data> length of data cached. The length of this parameter is 4 bytes and the unit is byte.
<total_data> length of OTA package. The length of this parameter is 4 bytes and the unit is byte.

The report interval is 15 seconds.

Example

```
AT+MOTADLQ=1
OK

+MOTADLQ: 00002048,00123874                Enable the URC of the download progress.

+MOTADLQ: 00003072,00123874
                                           After the download starts, URCs are prompted to notify the
+MOTADLQ: 00004672,00123874                download progress till the download is completed.

……
+MOTADLQ: 00123874,00123874
```

13.7 AT+MOTATRANS – Reading Data of MOTA Package

To set reading method of MOTA package and read the data.

Format

Type	Command	Response
Execute	AT+MOTATRANS=<TRANS_MODE>[,<location >,<length>]<CR>	Transparent transmission <CR><LF>CONNECT <CR><LF><CONTENT> <CR><LF>OK<CR><LF> Read from buffer <CR><LF>+MOTATRANS: <length>,<CRC32>,<CONTENT>

	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<TRANS_MODE>	Transmission mode 0: transparent transmission 1: read from buffer
<location>	Offset, starting from 1 When offset is set to 0, the data is read from the beginning of the file.
<length>	Length of file that is read. The maximum value is 8092 and the unit is byte.
<CONTENT>	Data content
<CRC32>	CRC32 verification value

Example

```
AT+MOTATRANS=0
CONNETCT
313233                                          Transmit data transparently.
OK
AT+MOTATRANS=1,0,10
+MOTATRANS:
10,53782849,31323334353637383930                Read data from buffer.
OK
```

13.8 AT+MOTAMODE – Setting MCU OTA Upgrade Mode

To set the OTA upgrade mode of the MCU.

After setting the OTA upgrade mode, the module sends a message to the server. The setting by this command is saved after the module is powered down.

Format

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

Parameter

<result>	Upgrade result 1: MCU upgrade successfully.

Example

```
AT+MOTAUTRT=1                    Set the OTA upgrade mode of the MCU to successful upgrade.
OK
AT+MOTAUTRT?
+MOTAUTRT: 1                     Query the upgrade result.
OK
```

14 Bluetooth Data Transmission

14.1 AT+BLENAME – Setting BLE Name of Module

To set the BLE name of the module.

Execute the command before executing **BLEOPEN**. The default BT name is NWY+IMEI.

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

Format

Type	Command	Response
Execute	AT+BLENAME=<name><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+BLENAME?<CR>	<CR><LF>+BLENAME: <name>,<mac_addr> <CR><LF>OK<CR><LF>

Parameter

<name>	Name of the module Bluetooth, supports 20 bytes at most. Special characters and Chinese characters are not supported.
<mac_addr>	MAC address of the module.

Example

```
AT+BLENAME=NEOWAY
OK                                          Set the name of the module BT
AT+BLENAME?
+BLENAME: NEOWAY,35:03:00:95:2F:7A          Query the module BT name and the MAC address.
OK
```

14.2 AT+BLEAUTH – Setting BLE Authentication Code

To set the module authentication code.

Execute the command before executing BLEOPEN.

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

Format

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

Parameter

<code> Authentication code, 6-digit integer greater than 99999 and less than 1000000.

Example

```
AT+BLEAUTH=123456                          The authentication code is 123456.
OK
AT+BLEAUTH?
+BLEAUTH: 123456                           Query the authentication code.
OK
```

14.3 AT+BLEOPEN – Enabling BLE Function

To enable the BLE function of the module.

Format

Type	Command	Response
Execute	AT+BLEOPEN=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<mode> 0: non transparent transmission mode
1: transparent transmission mode

Example

```
AT+BLEOPEN=0                          Enable the BLT function in non-transparent transmission mode
OK
AT+BLEOPEN=1                          Enable the BLT function in transparent transmission mode
OK
```

- If the BLE function is enabled already, ERROR will be returned if this command is executed.
- The value of Service UUID in BLE data transmission is 0xFEE0 and the Characteristics UUID is 0xFEE1.
- Non-transparent transmission mode:

  The device reports CONNECTED if a BT connection is established.

  The device reports DISCONNECTED if a BT connection is closed.
- Transparent transmission mode:

  The device reports CONNECTED if a BT connection is established and the serial port does not display the data transmitted.

  After entering the transparent mode, size of the data to be sent depends on the device that the device connects to. It is recommended to set the data size to the MTU value.

14.4 AT+BLESTATUS – Querying BLE Status

To query the BLE status.

Format

Type	Command	Response
Query	AT+BLESTATUS?<CR>	<CR><LF>+BLESTATUS: <status> <CR><LF>OK<CR><LF>

Parameter

<status>

0: disabled
1: enabled
2: connected

Example

```
AT+BLESTATUS?
+BLESTATUS: 1                                  Query the BLE status
OK
```

14.5 AT+BLESEND – Sending BLE Data

To send data over a BLE connection.

Format

Type	Command	Response
Execute	AT+BLESEND=<length>[,<content>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>> <CR><LF>OK<CR><LF>

Parameter

<length>	Length of the data to be sent, unit: byte, ranging from 1 to 500.
<content>	Content of the data to be sent

Example

```
AT+BLESEND=3,373                    Send 373
OK
AT+BLESEND=3
>                                   Send 31 32 33 (hexadecimal format)
OK
AT+BLESEND=3
>                                   the command times out 1 minutes after > displays and no
+BLESEND: OPERATION EXPIRED         data are input.
```

14.6 AT+BLERECVMODE – Setting BLT Data Receive Mode

To set the BLT data receive mode. The setting by this command is not saved after the module is powered down.

Format

Type	Command	Response
Set	AT+BLERECVMODE=<mode><CR>	<CR><LF>OK<CR><LF>

	Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+BLERECVMODE: <mode> <CR><LF>OK<CR><LF> Or <CR><LF>+BLERECVMODE: (0,1) <CR><LF>OK<CR><LF>

Parameter

<mode>	0: ASCII mode (default) 1: HEX mode

Example

```
AT+BLERECVMODE=1                          Set the BT receive mode to HEX mode
OK
```

14.7 +BLERECV – URC Notifying BLE Data Received

To notify the received BLE data.

Format

Type	Command
URC	+BLERECV: <length>,<data><CR>

Parameter

<length>	Length of the data received, 512 bytes at most.
<data>	Data received

Example

```
+BLERECV: 5,Abcde                         The BT module receives Abcde
```

14.8 AT+BLECLOSE – Disabling BLE Function

To disable BLT function of the mdule.

If the BLT function is disabled already, ERROR will be returned after this command is executed.

Format

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

Parameter

N/A.

Example

```
AT+BLECLOSE                                          Disable BLE function.
OK
```


15 PSM&eDRX Commands

15.1 AT+CPSMS –Setting PSM Mode

To set PSM mode.

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

To use this command, send +CSCLK to set the UART clock to automatic mode and send +NVSETPM to set the power management to low or ultra-low power consumption mode.

Format

Type	Command	Response
Execute	AT+CPSMS=[<mode>[,<Requested_Periodic-RAU>[,<Requested_GPRS-READY-timer>[,<Requested_Periodic-TAU>[,<Requested_Active-Time>]]]]]<CR>	AT+CPSMS=[<mode>[,<Requested_Periodic-RAU>[,<Requested_GPRS-READY-timer>[,<Requested_Periodic-TAU>[,<Requested_Active-Time>]]]]]<CR>
Query	AT+CPSMS?<CR>	AT+CPSMS?<CR>
Test	AT+CPSMS=?<CR>	AT+CPSMS=?<CR>

Parameter

<mode>	Enable or disable PSM mode 0: disable PSM mode 1: enable PSM mode
<Requested_Periodic-RAU>	8-bit unibyte Requested periodic RAU cycle on GERAN/UTRAN network (T3312) Bit8-Bit6: unit 000 – 10 minutes 001 – 1 hour 010 – 10 hours 011 – 2 seconds 100 – 30 seconds 101 – 1 minute 110 – 320 hours

	111 - T3312 invalid Bit5-Bit1: binary-code time e.g. 00100001 indicates 1 hour
<Requested_GPRS-READY-timer>	8-bit unibyte Requested GPRS READY cycle on GERAN/UTRAN network (T3314) Bit8-Bit6: unit 000 – 2 seconds 001 – 1 minute 010 – 6 minutes 111 - T3314 invalid **Bit5-Bit1**: binary-code time e.g. 00100001 indicates 1 minute
<Requested_Periodic-TAU>	8-bit unibyte Requested periodic-TAU cycle on GERAN/UTRAN network (T3412) Bit8-Bit6: unit 000 – 10 minutes 001 – 1 hour 010 – 10 hours 011 – 2 seconds 100 – 30 seconds 101 – 1 minute 110 – 320 hours 111 - T3412 invalid Bit5-Bit1: binary-code time e.g. 00100001 indicates 1 hour
<Requested_Active-Time>:	8-bit unibyte Requested Active Time on GERAN/UTRAN network 000 – 2 seconds 001 – 1 minute 010 – 6 minutes 111 - T3324 invalid **Bit5-Bit1**: binary-code time e.g. 00100001 indicates 1 minute

- The value of <Requested_Periodic-RAU> must be greater than that of <Requested_GPRS-READY-timer>.
- The settings should be negotiated with the network. For valid values, consult your carriers.

Example

```
AT+CPSMS?
+CPSMS: 0,,,"01100000","00000000"          Query the state of PSM mode.
OK
AT+CPSMS=1
OK                                         Enable PSM mode.
```

```
AT+CPSMS=0                                          Disable PSM mode.
OK
AT+CPSMS=1,,,"01100001","00000001"                  Set PSM parameters.
OK
```

15.2 AT+CEDRXS – Setting eDRX Mode

To set eDRX mode.

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

To use this command, send **+CSCLK** to set the UART clock to automatic mode and send **+NVSETPM** to set the power management to low or ultra-low power consumption mode.

Format

Type	Command	Response
Execute	AT+CEDRXS=<mode>[,<AcT_type>[,<Requested eDRX value>]]<CR>	AT+CEDRXS=<mode>[,<AcT_type>[,<Requested eDRX value>]]<CR>
Query	AT+CEDRXS?<CR>	AT+CEDRXS?<CR>
Test	AT+CEDRXS=?<CR>	AT+CEDRXS=?<CR>

Parameter

<mode> to specify whether to enable eDRX mode
0: disable eDRX mode
1: enable eDRX mode
2: enable eDRX mode and state report
3: reset to default setting *

<AcT_type> NB-IoT networks support 5 only
0: used only for state report
1: EC-GSM-IoT (A/Gb mode)
2: GSM (A/Gb mode)
3: UTRAN (Iu mode)
4: E-UTRAN (WB-S1 mode)
5: E-UTRAN (NB-S1 mode)

<Requested_eDRX_value> Requested eDRX cycle, 4-bit character string

A/Gb mode				
4	3	2	1	GERAN eDRX cycle length duration
0	0	0	0	~1,88 seconds

0	0	0	1	~3,76 seconds
0	0	1	0	~7,53 seconds
0	0	1	1	12,24 seconds
0	1	0	0	24,48 seconds
0	1	0	1	48,96 seconds
0	1	1	0	97,92 seconds
0	1	1	1	195,84 seconds
1	0	0	0	391,68 seconds
1	0	0	1	783,36 seconds
1	0	1	0	1566,72 seconds
1	0	1	1	3133,44 seconds
Iu mode				
4	3	2	1	UTRAN eDRX cycle length duration
0	0	0	0	10,24 seconds
0	0	0	1	20,48 seconds
0	0	1	0	40,96 seconds
0	0	1	1	81,92 seconds
0	1	0	0	163,84 seconds
0	1	0	1	327,68 seconds
0	1	1	0	655,36 seconds
0	1	1	1	1310,72 seconds
1	0	0	0	1966,08 seconds
1	0	0	1	2621,44 seconds
WB-S1/NB-S1 mode				
4	3	2	1	E-UTRAN eDRX cycle length duration
0	0	0	0	5,12 seconds (WB-S1)
0	0	0	1	10,24 seconds (WB-S1)
0	0	1	0	20,48 seconds
0	0	1	1	40,96 seconds
0	1	0	0	61,44 seconds (WB-S1)/20,48 seconds (NB-S1)
0	1	0	1	81,92 seconds
0	1	1	0	102,4 seconds (WB-S1)/20,48 seconds (NB-S1)
0	1	1	1	122,88 seconds (WB-S1)/20,48 seconds (NB-S1)

1	0	0	0	143,36 seconds (WB-S1)/20,48 seconds (NB-S1)
1	0	0	1	163,84 seconds
1	0	1	0	327,68 seconds
1	0	1	1	655,36 seconds
1	1	0	0	1310,72 seconds
1	1	0	1	2621,44 seconds
1	1	1	0	5242,88 seconds (NB-S1)/ 2621,44 seconds (WB-S1)
1	1	1	1	10485,76 seconds (NB-S1)/ 2621,44 seconds (WB-S1)

The settings should be negotiated with the network. For valid values, consult your carriers.

Example

```
AT+CEDRXS=1,5,”0001”                    Set eDRX cycle to 10.24s seconds in WB-S1 mode
OK
AT+CEDRXS?
+CEDRXS: 1,5,”0001”                     Query the eDRX settings.
OK
AT+CEDRXS=0
OK                                      Disable the eDRX mode.
```

16 Other Commands

16.1 AT+CPWROFF – Powering off the Module

To power off the module

Format

Type	Command	Response
Execute	AT+CPWROFF<CR>	<CR><LF>OK<CR><LF>

Parameter

N/A

Example

```
AT+CPWROFF                                          Power off the module.
OK
```

16.2 AT+CSCLK – Setting Clock Mode

To set clock mode of serial port This command is used together with PSM and eDRX commands.

Format

Type	Command	Response
Set	AT+CSCLK=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<mode> 0: Normal mode
1: DTR controls low-frequency clock of serial port
High level: enable
Low level: disable
2: Auto mode
UART automatically exits from low-frequency clock when receiving or sending data.

Example

```
AT+CSCLK=2                                          Set to auto mode
OK
```

16.3 AT+NVSETPM – Setting PM Mode

To set power management mode.

This command is used together with PSM and eDRX commands. Settings by this command are saved after the module is powered down.

Format

Type	Command	Response
Set	AT+NVSETPM=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<mode> 0: Normal mode
1: Low power consumption mode (eDRX)
2: Ultra-low power consumption mode (eDRX and PSM)

Example

```
AT+NVSETPM=1                                        Set to low power consumption mode
OK
```

16.4 AT+PING – PING Test

PING Test.

It is recommended to execute the next PING comand after the execution of the PING command has completely finished.

Format

Type	Command	Response
Execute	AT+PING=<ip>[,<timeout>,<size>,<num>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<ip>	IP address
<timeout>	Timeout interval, 0 to 255 seconds
<size>	Size of data packet, IPv4 (36 to 1500 bytes), IPv6 (56 to 1500 bytes)
<num>	Number of ping tests, 1 to 65535

Example

```
AT+PING=58.60.184.213,255,64,4
OK
Reply from 58.60.184.213: bytes= 64 time = 764(ms), TTL = 255
Reply from 58.60.184.213: bytes= 64 time = 172(ms), TTL = 255
Reply from 58.60.184.213: bytes= 64 time = 206(ms), TTL = 255
Reply from 58.60.184.213: bytes= 64 time = 243(ms), TTL = 255

Ping statistics for 58.60.184.213
Packets: Sent = 4, Received = 4, Lose = 0 <0%>, max_delay = 764 ms, min_delay = 172 ms,
average delay = 346 ms
```

16.5 AT+NEONBIOTCFG – Enabling Extending Functions

To enable or disable extending functions, such as automatic PPP activation, time update, PSM status report, and RRC status report The settings by this command are saved after the module is powered off.

Send AT+IPR to set a fixed baud rate before using this function.

Format

Type	Command	Response
Set	AT+NEONBIOTCFG=<auto_ip>,<time_inc>,<psm_inc>,<rrc_inc><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NEONBIOTCFG?<CR>	<CR><LF>+NEONBIOTCFG: <auto_ip>,<time_inc>,<psm_inc>,<rrc_inc> <CR><LF>OK<CR><LF>

Parameter

<auto_ip>	Activate PPP automatically and report IP address after registering with network 0: Disable 1: Enable
<time_inc>	Report time after registering with network or waking up from PSM 0: Disable 1: Enable
<psm_inc>	Report PSM status (PSM ENTER, PSM WAKEUP) 0: Disable 1: Enable
<rrc_inc>	Report RRC status 0: Disable 1: Enable

Example

```
AT+NEONBIOTCFG=1,0,0,0                  Enable automatic PPP activation.
OK
AT+NEONBIOTCFG=0,1,1,0                  Enable time update and PSM report.
OK
AT+NEONBIOTCFG?
+NEONBIOTCFG: 1,1,1,1                   Query current settings.
OK
```

16.6 AT+LEDMODE – Setting LED

To enable or disable STATUS and NET indicators The settings by this command are saved after the module is powered off.

- Status indicator
  - The indicator turns on after the module is powered on.
  - The indicator keeps off if the STARUS indicator is disabled.

- NET indicator
  - Network indicator turns off if the module does not search networks and turns on if the module finds networks and does not activate PPP.
  - It blinks (on for 0.2 second and off for 1.8 second) after the module actives PPP.

Format

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

Parameter

<mode> 0: disabled (default)
1: enabled

Example

```
AT+LEDMODE=1                                    Enable STATUS and NET LED indicators.
OK
AT+LEDMODE?
+LEDMODE: 1                                     Query LED settings.
OK
```

16.7 AT+NEOFOTA – FOTA Upgrade

To control the remote upgrade of a module.

Do not power down or restart the module during upgrade.

In automatic baud rate mode, issue AT\r\n to complete detection of baud rate. Then the module returns **+PBREADY** and upgrade success.

Format

Type	Command	Response
Execute	AT+NEOFOTA=<server>,<port><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+NEOFOTA: <status><CR><LF>

Parameter

< server >	FOTA server address
<port>	FOTA server port
<status>	Upgrade status Password 0: No upgrade package is available 1: Downloaded the upgrade package successfully 2: Failed to download or query the upgrade package 3: Started to upgrade 4: Upgraded successfully 5: Failed to upgrade 6: Started to download

Example

```
AT+NEOFOTA=115.29.178.98/,80          Enable FOTA.
OK

+NEOFOTA: 1                           Download upgrade package successfully.

+NEOFOTA: 3                           Start to upgrade.

+PBREADY                              (the module restarts automatically after the upgrade is
                                      completed)
+NEOFOTA: 4
                                      Upgrade successfully.
AT+NEOFOTA=115.29.178.98/,80
OK

+NEOFOTA: 0                           No upgrade packages.
```

16.8 AT+NEOFOTAURC – FOTA Status Report

To control the status report during FOTA upgrade

The setting is not saved after the module is powered down. This setting should be configured before upgrade.

Format

Type	Command	Response
Set	AT+NEOFOTAURC=<result><CR>	<CR><LF>OK<CR><LF> <CR><LF>+NEOFOTAURC: <status> Or <CR><LF>ERROR<CR><LF>
Query	AT+NEOFOTAURC?<CR>	<CR><LF>+NEOFOTAURC: <result> <CR><LF>OK<CR><LF>

Parameter

<result> switch of status report
0: disable status report. Only upgrade result is reported.
1: enable status report (Default). All states are reported.

<status> Upgrade status
Password
0: No upgrade package is available
1: Downloaded the upgrade package successfully
2: Failed to download or query the upgrade package
3: Started to upgrade
4: Upgraded successfully
5: Failed to upgrade
6: Started to download

Example

```
AT+NEOFOTAURC=1                          Enable the status report.
OK
AT+NEOFOTA=115.29.178.98/,80
OK

+NEOFOTA: 1                              Download upgrade package successfully.

+NEOFOTA: 3                              Start to upgrade
                                         (the module restarts automatically after the
+PBREADY                                 upgrade is completed)

+NEOFOTA: 4                              Upgrade successfully
AT+NEOFOTAURC=0                          Disable the status report.
OK
AT+NEOFOTA=115.29.178.98/,80
OK                                       No upgrade packages
```

```
+NEOFOTA: 0
AT+NEOFOTAURC=0
OK                                      Disable report of process status

AT+NEOFOTA=115.29.178.98/,80
OK

                                        Upgraded successfully
+NEOFOTA: 4


AT+NEOFOTAURC=0
OK                                      Disable report of process status

AT+NEOFOTA=115.29.178.98/,80
OK
                                        Failed to upgrade
+NEOFOTA: 5
```

16.9 AT+WIFIAPSCAN – Scanning Wi-Fi Hotspot

To scan Wi-Fi hotspots around the module

Format

Type	Command	Response
Execute	AT+WIFIAPSCAN<CR>	<CR><LF>+WIFIAPSCAN: <MAC Address>,<rssi>,<channel> ..... <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<MAC Address>	MAC address
<rssi>	received signal strength indication, unit dBm
<channel>	signal channel

Example

```
AT+WIFIAPSCAN
+WIFIAPSCAN: e646daeb98b4,200,6
+WIFIAPSCAN: c4b547c9a86,189,6
+WIFIAPSCAN: 5c546d9a33a1,185,6
+WIFIAPSCAN: b89436210338,183,6
```

```
+WIFIAPSCAN: b7c7d3d44c466,179,6
OK
```

16.10 AT+UARTRSPMODE – Setting UART Output Mode

To set the maximum length of data UART outputs once and the minimum interval The setting by this command is not saved after the module is powered off.

If this command is not executed, the module outputs data at normal speed.

Format

Type	Command	Response
Set	AT+NVSETPM=<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+UARTRSPMODE?<CR>	<CR><LF>+UARTRSPMODE: <delay>,<size> <CR><LF>OK<CR><LF>
Test	AT+UARTRSPMODE=?<CR>	<CR><LF>+UARTRSPMODE: (0-1000),(20-4096) <CR><LF>OK<CR><LF>

Parameter

<delay> minimum interval of data output, 0 to 1000 ms, 0 by default
<size> maximum length of data output by UART once, 20 to 4096 bytes, 4096 by default

Example

```
AT+UARTRSPMODE=100,200                    Set the minimum interval to 100 ms and the
OK                                        maximum length of data output once to 200 bytes
```

16.11 AT+SIMSELECT – Selecting SIM Card

To select the SIM card (external SIM or embedded eSIM) to be used The setting by this command takes effect after the module is restarted and after that the setting is saved after the module is powered down.

Format

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

Parameter

<sim> 0: external SIM card (default)
1: eSIM

Example

```
AT+SIMSELECT=0                                    Enable external SIM card.
OK
```

16.12 AT+NVSETRELEASEVERSION – Setting 3GPP Release Version

To set the 3GPP release version.

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

Format

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

Parameter

<release>	0: 3GPP release version 13 (default) 1: 3GPP release version 14

Example

```
AT+NVSETRELEASEVERSION=1                                    Set 3GPP release version to R14
OK
```

A Appendix

A.1 Common Error Codes

Error Code	Meaning
1	No connection to phone
2	Phone adapter link reserved
3	Operation not allowed
4	Operation not supported
5	PH_SIM PIN required
6	PH_FSIM PIN required
7	PH_FSIM PUK required
10	SIM not inserted
11	SIM PIN required
12	SIM PUK required
13	SIM failure
14	SIM busy
15	SIM wrong
16	Incorrect password
17	SIM PIN2 required
18	SIM PUK2 required
20	Memory full
21	Invalid index
22	Not found
23	Memory failure
24	Text string too long
25	Invalid characters in text string
26	Dial string too long
27	Invalid characters in dial string
30	No network service
31	Network timeout

32	Network not allowed, emergency calls only
40	Network personalization PIN required
41	Network personalization PUK required
42	Network subset personalization PIN required
43	Network subset personalization PUK required
44	Service provider personalization PIN required
45	Service provider personalization PUK required
46	Corporate personalization PIN required
47	Corporate personalization PUK required
49	Execute not support
50	Execute fail
51	No memory
52	Option not support
53	Param invalid
58	Invalid command line

A.2 Status Codes of MCU OTA

Code	Text
10	Idle
11	Querying version
12	Found a later version
13	Failed to query later version
15	No later version
21	Start to download
22	Downloading
23	Failed to download
25	Download completed
31	Started to transmit
32	Transmission completed

A.3 Error Codes of MCU OTA Package Download Failure

Error code	Text
01	Last operation is not executed completely
02	No PPP connection
03	Connection setup times out
04	Failed to set up a connection
05	CRC32 verification failed
06	The cache is not sufficient
07	Download is interrupted