N58

AT Commands Manual

Issue 1.8 Date 2023-02-11

Notice

This document provides guide for users to use N58.

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

THIS DOCUMENT 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 with 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

1 Boot LOG Instruction 15
2 AT Syntax 16
2.1 Symbols 16
2.2 Description 16
2.3 Command Types 17
2.4 Command Response Time-Out 17
3 General Commands 21
3.1 ATI – Querying the Manufacturer Information 21
3.2 AT+GMR – Querying the Software Version 21
3.3 AT+CSQ – Querying Signal Quality 22
3.4 AT+CREG – Querying the Network Registration Status 23
3.5 AT+CEREG – Querying EPS Network Registration Status 24
3.6 AT+COPS – Selecting an Operator 25
3.7 AT+CIMI – Requesting International Mobile Subscriber Identity (IMSI) 27
3.8 AT+CGSN – Requesting Product Serial Number Identification 28
3.9 AT+GSN – Requesting Product Serial Number Identification 28
3.10 AT+CCID – Obtaining ICCID of SIM Card 29
3.11 AT+CGMM – Requesting Model Identification 29
3.12 AT+GMM – Querying Module Model 30
3.13 AT+IPR – Setting Baud Rate 31
3.14 AT+CFUN – Setting Module Functionality 31
3.15 AT+CMUX – Enabling/Disabling Multiplexing Mode 32
3.16 AT+CCLK – Clock 34
3.17 AT+CPIN – Entering PIN Code 35
3.18 AT+CLCK – Facility Lock 36
3.19 AT+CPWD – Changing the Password 37
3.20 AT+CGDCONT – Defining PDP Context 38
3.21 AT+XGAUTH – PDP Authentication 39
3.22 AT+CGATT – Setting GPRS Attach and Detach 40
3.23 ATE1/ATE0 – Enabling/Disabling the Terminal Display 42
3.24 ATD*99# – GPRS 42
3.25 AT+ENPWRSAVE – Enabling or Disabling Sleep Mode 43
3.26 AT+SIGNAL – Setting Blinking Status Signal Indicator 44
3.27 AT+CESQ – Extended Signal Quality 46
3.28 AT+NWDNS – Parsing the Domain Name 47
3.29 AT+CGREG – Querying GPRS Network Registration Status 48

4 SMS Commands 50
4.1 AT+CSMS – Selecting SMS Services 50
4.2 AT+CPMS – Setting Preferred SMS Storage 51
4.3 AT+CMGF – Setting Message Format 52
4.4 AT+CSCS – Setting the TE Character Set 53
4.5 AT+CNMI – New Message Indications to TE 54
4.6 AT+CMGR – Reading SMS Message 56
4.7 AT+CMGL – SMS Message List 58
4.8 AT+CMGS – Sending SMS Messages 60
4.9 AT+CMGW – Writing SMS Messages 61
4.10 AT+CMSS – Sending Messages from Storage 62
4.11 AT+CMGD – Deleting SMS Messages 63
4.12 AT+CSCA – Service Center Address 64
4.13 AT+CSMP – Setting Text Mode Parameters 64
4.14 AT+CSDH – Showing Text Mode Parameters 66
5 TCP/UDP Client Commands 67
5.1 AT+NETAPN – Setting Network APN 67
5.2 AT+XIIC – Setting up a PPP Link 67
5.3 AT+TCPSETUP – Setting up a TCP Connection 68
5.4 AT+TCPSEND – Sending TCP Data 70
5.5 AT+RECVMODE – Setting Receive Mode 71
5.6 +TCPRECV – URC Notifying Data Received from Server 72
5.7 A+TCPREAD – Reading TCP Data 73
5.8 AT+TCPCLOSE – Closing the TCP Connection 74
5.9 AT+UDPSETUP – Setting up a UDP Connection 74
5.10 AT+UDPSEND – Sending UDP Data 76
5.11 +UDPRECV – Receiving UDP Data 77
5.12 AT+UDPREAD – Reading UDP Data 78
5.13 AT+UDPCLOSE – Closing UDP Link 78
5.14 AT+IPSTATUS – Querying the Transparent TCP/UDP Socket Status 79
5.15 AT+TCPACK – Querying Status of Data Sent by TCP Socket 80
5.16 AT+DNSSERVER – Setting DNS Server 81
5.17 AT+PDPKEEPALIVE – Setting PDP Keepalive Heartbeat 82
5.18 AT+PDPSTATUS – Querying PDP Status 83
5.19 AT+TCPKEEPALIVE – Setting TCP Keepalive Heartbeat 83
6 TCP Server Commands 86
6.1 AT+TCPLISTEN – Setting TCP Listening for the Server 86
6.2 AT+CLOSELISTEN – Closing Listening Socket 87
6.3 AT+CLOSECLIENT – Closing Remote Socket 87
6.4 +TCPRECV(S) – Receiving Data from the Client 88
6.5 AT+TCPREADS – Reading TCP Data from the Client 89
6.6 AT+TCPSENDS – Sending Data to the Client 90

6.7 AT+CLIENTSTATUS – Querying Client Connection Status 91
6.8 AT+TCPACKS – Querying Status of Data Sent by TCP Server 91
7 TCP/UDP Transparent Commands 93
7.1 AT+TCPTRANS – Setting up a Transparent TCP Connection 93
7.2 AT+UDPTRANS – Setting up a Transparent UDP Connection 94
7.3 AT+TCPACK – Querying of Data Transmitted Status 95
7.4 AT+IPSTATUS – Querying the TCP/UDP Socket Status 96
7.5 AT+TRANSCLOSE – Closing the Transparent Connection 97
8 TCP Transparent Transmission Server Commands 99
8.1 AT+TCPSRVTRANS – Setting Listening for Transparent TCP 99
8.2 AT+CLIENTSTATUS – Querying the Status of the Transparent Client Connection 100
9 FTP Commands 102
9.1 AT+FTPSCFG – Configuring FTPS Parameter 102
9.2 AT+FTPLOGIN – Logging in to the FTP Server 103
9.3 AT+FTPLOGOUT – Logging Out from the FTP Server 105
9.4 AT+FTPGET – Downloading Data from the FTP Server 105
9.5 AT+FTPPUT – Uploading Data to the FTP Server 107
9.6 AT+FTPSIZE – Obtaining the FTP File Size 109
9.7 AT+FTPSTATUS – Querying the FTP Connection Status 109
9.8 AT+FILEFTPGET – Downloading Files to the File System 110
9.9 AT+FILEFTPPUT – Uploading Files to the FTP Server 111
9.10 AT+NWFTPRENAME – Renaming the FTP Server File or Folder 112
9.11 AT+NWFTPMKDIR – Creating an FTP Server Folder 113
9.12 AT+NWFTPRMDIR – Deleting an FTP Server Folder 114
9.13 AT+NWFTPDEL – Deleting an FTP Server File 115
10 HTTP/HTTPS Commands 117
10.1 AT+HTTPPARA – Setting HTTP Parameters 117
10.2 AT+HTTPSETUP – Setting up an HTTP Connection 118
10.3 AT+HTTPACTION – HTTP Request 118
10.4 AT+HTTPCLOSE – Closing the Socket of an HTTP Client 122
10.5 +HTTPRECV – Receiving HTTP Data 122
10.6 AT+HTTPGET – Downloading HTTP Files 124
10.7 +HTTPGETRPT – URC Notifying Downloading Results 125
10.8 AT+HTTPGETSTAT? – Querying the HTTP Downloading Result 127
10.9 +HTTPCLOSED – URC Notifying the Socket of the HTTP Client is Closed 128
10.10 AT+HTTPSCFG – Configuring SSL Parameters for HTTPS 129
10.11 AT+HTTPSPARA – Setting HTTPS Parameters 130
10.12 AT+HTTPSSETUP – Setting up an HTTPS Connection 131
10.13 AT+HTTPSACTION – Sending an HTTPS Request 132
10.14 AT+HTTPSCLOSE – Closing an HTTP Connection Proactively 134
10.15 +HTTPSRECV – URC Notifying HTTPS Data Received 135
10.16 AT+HTTPSGET – Download HTTPS Files 136

10.17 +HTTPSGETRPT – URC Notifying Downloading Result 138
10.18 AT+HTTPSGETSTAT? – Querying the Download Result 139
10.19 AT+FILEHTTPACTION – HTTP Request in File System 141
10.20 AT+FILEHTTPACTION – HTTPS Request in File System 143
10.21 +FILEHTTPSTAT – URC Notifying the HTTP(S) Uploading/Downloading Result 144
10.22 +HTTPSCLOSED – URC Notifying HTTPS Link Closed 145
11 Call Control Commands 146
11.1 ATD – Dialing Command 146
11.2 ATA – Call Answering 147
11.3 ATH – Hanging up Calls 147
11.4 AT+CLIP – Caller ID 148
11.5 ATS0 – Auto-Answer 149
11.6 AT+CLVL – Setting the Voice Volume 149
11.7 AT+CMUT – Mute Control 150
11.8 AT+CLCC – Querying Current Calls 151
11.9 AT+SETVOLTE – Setting VoLTE Switch 152
12 Wi-Fi Function 154
12.1 AT+WIFIAPSCAN – Wi-Fi Hotspot Scanning 154
12.2 AT+WIFIGSMLOC – Wi-Fi Positioning 154
13 SSL TCP Data Service 156
13.1 AT+SSLTCPCFG – Configuring SSL Parameters for TCP 156
13.2 AT+SSLTCPSETUP – Setting up a TCP Connection over SSL 157
13.3 AT+SSLTCPCLOSE – Closing TCP Connection over SSL 159
13.4 AT+SSLTCPSEND – Sending TCP Data over SSL 159
13.5 +SSLTCPRECV – URC Notifying SSLTCP Data Received 160
13.6 AT+NWCERTEENABLE–Enable Encryption of the Certificate 161
13.7 AT+CERTADD – Adding SSL Certificate 162
13.8 AT+CERTCHECK – Checking the SSL Certificate 162
13.9 AT+CERTDEL – Deleting an SSL Certificate 163
13.10 AT+SSLTCPCFGA – Configuring SSL Parameters for TCP 164
13.11 AT+SSLTCPREAD – Reading SSL TCP Data 165
13.12 AT+SSLCIPHERSET – Removing Weak Algorithm from the SSLTCP Connection 166
14 MQTT Commands 167
14.1 AT+MQTTMUX – Setting MQTT Multiplexing 167
14.2 AT+MQTTTLS – Configuring TLS Parameters 168
14.3 AT+MQTTCONNPARAM – Setting User Parameters 169
14.4 AT+MQTTWILLPARAM – Setting Will 170
14.5 AT+MQTTWILLMSG – Setting Long Will Messages 172
14.6 AT+MQTTCONN – Connection Command 172
14.7 AT+MQTTSUB – Subscribing to a Topic 174
14.8 AT+MQTTUNSUB – Canceling a Subscription 175
14.9 AT+MQTTPUB – Publishing a Topic 176

14.10 AT+MQTTPUBS－Publishing a Topic with Long Message 176
14.11 AT+MQTTDISCONN－Disconnecting to the MQTT Server 177
14.12 +MQTTSUB－Receiving Message 178
14.13 AT+MQTTSTATE－Query the MQTT Connection Status 179
15 AWS MQTT Commands 181
15.1 AT+AWSTLSCFG－Configuring AWS TLS Parameters 181
15.2 AT+AWSAUTHPARAM－Setting User Parameters 182
15.3 AT+AWSCONNPARAM－Setting the AWS Connection Parameter 182
15.4 AT+AWSCONN－Setting up the AWS MQTT Connection 183
15.5 AT+AWSSUB－Subscribing to a Topic 184
15.6 AT+AWSUNSUB－Canceling a Subscription 184
15.7 AT+AWSPUB－Publishing a Topic 185
15.8 AT+AWSDISCONN－Closing the AWS MQTT Connection 186
15.9 +AWSSUBRECV－Receiving the Topic 186
15.10 AT+AWSSTATE－Querying the MQTT Connection State 187
16 GPS Function 189
16.1 AT$MYGPSPWR－GPS Switch 189
16.2 AT$MYGPSSTATE－Querying the GPS Switch Status 189
16.3 AT$MYGPSPOS－Obtaining NMEA Data 190
16.4 AT$MYGNSSSEL－Enabling an Aid GNSS System 194
16.5 AT$MYGNSSMSG－Obtaining Positioning Data of Aid GNSS System 195
16.6 AT+SETSERVER－Setting AGPS Server 196
16.7 AT$MYGPSCFG－Setting GPS 197
17 BT/BLE General Commands 199
17.1 AT+NWBTBLEPWR－BT/BLE Power Switch 199
17.2 AT+NWBTBLENAME－Setting the BT/BLE Name 200
17.3 AT+NWBTBLEMAC－Querying the BT/BLE MAC Address 201
18 BLE Function Commands 202
18.1 AT+NWBLEROLE－Setting BLE Mode 202
18.2 AT+NWBLEADV－Setting BLE Broadcast Parameters 203
18.3 AT+NWBLEADVEN－Enabling/Disabling BLE Broadcast 204
18.4 +NWURCBLESTAT－URC Notifying BLE Status 205
19 BLE Peripherals (Client/Server) 206
19.1 AT+NWBLEPSRV－Creating Services 206
19.2 AT+NWBLEPCRT－Adding Characteristics for the Service 207
19.3 AT+NWBLEPSTR－Starting Services 208
19.4 AT+NWBLEPSEND－Sending Data 209
19.5 +NWURCBLEPRECV－URC Notifying Data Received 210
19.6 AT+NWBLEPWRITE－Writing Data 211
19.7 AT+NWBLERCVMODE–Setting the Received Data Format 212
19.8 AT+NWBLEDISCON－Querying/Disconnecting the BLE Connection 212
19.9 AT+NWBLESRVRM－Removing the Specified Service 213

19.10 AT+NWBLESRVRM－Removing the Specified Characteristic 214
19.11 AT+NWIBEACON－iBeacon Function 215
20 BLE Central Device (Client/Server) 217
20.1 AT+NWBLESCAN－Scanning the Surrounding BLE Devices 217
20.2 AT+NWBLECCON－Establishing the BLE Connection 218
20.3 AT+NWBLECDISCON－Closing the BLE Connection 219
20.4 AT+NWBLEQSRV－Finding the BLE Device Service 219
20.5 AT+NWBLEQCHAR－Finding the BLE Device Characteristic 220
20.6 AT+NWBLECSEND－Sending Data 221
20.7 +NWURCBLECRECV－Receiving Data 222
20.8 AT+NWBLECREAD－Reading Characteristic Data 223
21 DTMF Commands 225
21.1 AT+VTS－Transmitting DTMF Tones 225
22 GSM Location Command 226
22.1 AT+CIPGSMLOC－Obtaining the Location of the Module 226
23 Time Synchronization Commands 228
23.1 AT+UPDATETIME－Updating Time to Network 228
24 Network Sharing Commands 231
24.1 AT+NETSHAREMODE－Selecting Network Sharing Mode 231
24.2 AT+NETSHAREACT－Enabling Network Sharing 232
25 Statistics on Data Traffic 234
25.1 AT+FLOWCALC－Statistics on Total Data Traffic 234
25.2 AT+RATECALC－Statistics on Transient Traffic 234
26 File System Commands 236
26.1 AT+FSWF－Writing Data to File 236
26.2 AT+FSRF－Reading Data from a File 237
26.3 AT+FSRFEX－Reading File 238
26.4 AT+FSDF－Deleting a File 239
26.5 AT+FSLIST－Obtaining File List 240
26.6 AT+FSFS－Obtaining the Size of a File 241
26.7 AT+FSLS－Obtaining the Remaining Storage Size of User Disk 242
26.8 AT+FSFAT－Formating the User Disk 243
26.9 AT+FSRN－Renaming the File 244
26.10 AT+NWSPIREAD－SPI Reads the File 245
27 Audio Recording Commands 246
27.1 AT+RECMODE－Setting Recording Mode 246
27.2 AT+RECF－Starting and Stopping an Audio Recording 247
27.3 AT+RECP－Playing the Record File 248
27.4 AT+RECFG－Playing Settings of Recording File 249
27.5 AT+AUDPLAY－Playing Audio 249
27.6 AT+AUDPAUSE－Pausing Audio Playing 251

27.7 AT+AUDRESUME－Resuming Audio Playing 251
27.8 AT+AUDCFG－Settings of Audio Playing 252
27.9 AT+AUDPCMRATE－Setting the Sampling Rate of Playing the PCM Format Audio 253
27.10 AT+MICL－Setting the Microphone Gain 254
28 SIM Card Related Commands 255
28.1 AT+SIMCROSS－Switching SIM 255
29 Other Commands 257
29.1 AT$MYPOWEROFF－Powering off the Module 257
29.2 AT$MYSOCKETLED－Controlling the Indicator of Socket Status 257
29.3 AT$MYGMR－Obtaining the Firmware and Hardware Information of the Module 258
29.4 AT$MYCCID－Obtaining the CCID of the SIM Card 259
29.5 AT$MYTYPE－Querying the Type of the Module 259
29.6 AT$MYMODEM－Querying the Type of the Module 260
29.7 AT$MYBCCH－Locking BCCH Channel 261
29.8 AT$MYBAND－Locking GSM Band 263
29.9 AT$MYLACID－Querying Current LAC and CELL_ID 264
29.10 AT$MYCGED－Querying Current Channel, RX Power, and TX Power 264
29.11 AT$MYSYSINFO－Quering or Locking the Network Mode 265
29.12 AT$MYSYSINFOURC－Switch of Network Mode URC Indication 266
29.13 AT$MYURCSYSINFO－Network Mode URC Indication 267
29.14 $MYURCSYSINFO－URC Notifying Network Mode 268
29.15 AT$MYNETINFO－Setting Network Mode Choices 268
29.16 AT$MYNETAUTO－Enabling/Disabling the Default AUTO Network Modes during Startup 269
29.17 AT^SYSINFO－Setting Network Mode Choice 270
29.18 AT+XISP–Selecting Internal or External Protocol Stack 272
29.19 AT+NRSP－Querying RSRP, RSRQ, SINR of Cells in a Neighbor of a Serving Cell on the LTE Network 272
29.20 AT+NETMSG－Querying Network Registration Information 273
29.21 AT+NETDMSG－Querying Network Registration Information 275
29.22 AT+NEOFOTA－FOTA Command 276
29.23 AT+NEOFOTAURC－FOTA Status Report 277
29.24 +NWURCFOTA－URC Notifying FOTA Upgrade Status 279
29.25 AT+NWFOTA－Performing an FOTA Upgrade 279
29.26 AT+READADC－Reading ADC Value 281
29.27 AT+SIMHOTSWAP－Setting the Hotswapping Function 282
29.28 AT+MYDATAONLY－Disabling CS Services 283
29.29 AT+BANDLOCK－Locking to Band 283
29.30 AT+MYCELLINFO－Obtaining Information of Neighbor Cells 285
29.31 AT+NBANDLOCK－Locking Multiple Frequency Bands 286
29.32 AT+NFREQLOCK－Locking the Specified Frequency 288
29.33 AT+NWCUSTPIN－Setting Specified GPIO 289
29.34 AT+IPINFO－Querying Socket Connection Information 291
29.35 AT+SETTZ－Setting Clock Offset Time 292

29.36 AT+NCUSTSWITCH – Switch of Extended Functions 292
29.37 AT+FTPGETF – FTP Upgrade Command 294
29.38 AT+NWCHANNEL – Setting the Network Activation Channel 295
29.39 AT+CGACT – Activating/Deactivating PDP Context 296
29.40 AT+CUSD – Sending USSD Data 297
30 UDP Server Function 299
30.1 AT$UDPLISTEN – Creating UDP Listener as a Server 299
30.2 AT$CLOSEUDPLISTEN – Closing Listening Socket 300
30.3 AT$CLOSEUDPCLIENT – Closing UDP Remote Socket 301
30.4 $UDPRECV(S) – Notifying UDP Data Received from Clients 301
30.5 AT$UDPSEND(S) – Sending Data to Clients 302
30.6 AT$UDPCLIENTSTATUS – Querying Client Connection Status 303
30.7 AT$IPNETREAD – Reading UDP Data 304
A Error Codes 306
A.1 General Error Codes 306
A.2 FTP Error Codes 308
A.3 HTTP(S) Error Codes 308
B Reference Process of AT Command Programming 311
B.1 Content of PDU SMS Messages 311
B.2 Flowchart of Sending Text SMS Messages (Through UART) 313
B.3 Flowchart of Sending PDU SMS Messages (Through UART) 314
C Support Band List 315
C.1 GSM Bands 315
C.2 LTE Bands 315
D Result Codes 317
E FTP(S) Error Codes 319

About This Document

Scope

This document is applicable to N58.

Audience

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

Change History

Issue	Date	Change	Changed By
1.0	2019-12	Initial draft	Feng Yan
1.1	2020-06	• Added AT+CGMM command. • Added AT+CGDCONT command. • Added AT+CGREG command. • Added AT+NEOFOTA command. • Added AT+NEOFOTAURC command. • Added FTP upgrade command. • Modified the parameters of AT+TCPTRANS, AT+UDPTRANS, AT+TCPSRVTRANS commands and added settings of the optional parameters cfgt and cfgp.	Liu Feifan
1.1.2	2021-04	• Modified examples of the +TCPTRANS, +UDPTRANS and +IPSTATUS commands. • Modified examples of the +TCPSRVTRANS and +CLIENTSTATUS commands. • Modified examples of the +FTPLOGIN, +FTPGET, +FTPPUT, and +FTPSTATUS commands. • Added examples to the +UDPREAD command.	Wang Yuxin
1.2	2021-05	• Modified the description of +CSCS command. • Deleted the TTS related commands. • Modified the DTMF and recording related commands. • Added the Bluetooth related commands	Ma Bingbing Yin Jianjun Yang Yuanzheng Dong Nengxiang Yang Di

		• Modified the NEOFOTA command • Modified the TCP/UDP related commands • Modified the MQTT related commands • Modified the HTTP, file system, and GPS related commands • Modified the audio record related commands • Added the NWCHANNEL and CGACT commands • Added the Wi-Fi positioning command. • Modified the +CSCS and ATD related commands • Modified the NWCUSTPIN and ADC related commands. • Added +CESQ command • Modified the $MYBCCH, $MYLACID, $MYNETAUTO, and +MYDATAONLY commands. • Modified +CPMS test command.	Yang Mengsha Li Zhengfu Huang Weigang Yang Zhijian Zhao Liechang Liu Fang Wang Chen
1.3	2021-6	• Modified the description of the +FSWF and +FSRFEX commands • Modifed the description of +AWSAUTHPARAM command • Added the parameters for +CLOUDAUTHMODE command • Modified the examples of +FTPGET command	Zhao Liechang Wang Yuxin Li Xinahong Yang Di
1.4	2021-7	• Modified the description of +FILEHTTPACTION command.	Zhao Liechang
		• Modified the description of the +MQTTTLS command • Added the sslversion parameter.	
		• Modified +FTPSTATUS command • Modified the description for the +FILEHTTPACTION command.	Yangdi
		• Modified paramaters to +CLOUDHDAUTH command.	Zhao Liechang
		• Modified the parameter description of +MQTTTLS command. • Modified the examples of CGDCONT command.	
		• Modified the description of +CLOUDUNSUB command.	
		• Modified examples of +BANDLOCK command.	Huang Weigang
		• Added +NWDNS command.	Yang Di
1.5	2021-09	• Modified the note of +NBANDLOCK • Modified parameter description of +HTTPSCFG • Modified the note of +NWDNS • Added the +FILEFTPPUT command	Huang Weigang Yang Di

	2021-10	• Modified the parameters and examples of +SSLTCPREAD • Modified parameter description of +IPINFO • Modified description of the +TCPREADS command • Modified the parameters and examples of +FILEFTPPUT	Yang Di
	2021-12	• Modified parameter description of +CREG • Modified parameter description of +CEREG • Modified the default value of +SETVOLTE	Huang Weigang
	2021-12	• Added the +CUSD command	
	2022-01	• Modified parameter description of FSRFEX • Modified the parameters and examples of the HTTPGET/+HTTPSGET command	
1.6	2022-03	• Added the +FILEFTPGET command • Added the +NWURCFOTA and +NWFOTA commands • Added command description of +BANDLOC • Added command description of +NWFOTA	Yang Di Huang Weigang
	2022-04	• Added AT+NWCERTEENABLE • Added the <type> parameter to AT+CERTADD	Gao Zheng
	2022-05	• Modified command description of AT+IPSTATU • Modified command description of AT+FTPLOGIN	Yang Di
1.7	2022-06	• Added the UDP server AT commands (Chapter 30)	Yi Xudong
	2022-07	• Added the <pt> parameter indicating data service provider information. (section 16.6 , +SETSERVER command)	Gao Zheng
		• Added the +NWFTPRENAME command (section 9.10) • Added the +NWFTPMKDIR command (section 9.11) • Added the +NWFTPRDIR command (section 9.12) • Added the +NWFTPDEL command (section 9.12) • Added the +FTPSCFG command (section 9.1) • Added the +FTPLOGIN command description (section 9.2) • Added Appendix D "Result Code" • Added Appendix E "FTP(S) Error Codes"	Yang Di
1.7.1	2022-09	• Modified note information of the +HTTPGET command (section 10.6)	Yang Di
1.8	2022-11	• Modified the MQTT Commands and added the +MQTTMUX command (Chapter 14) • Deleted "SIM power" from the function description	Yang Di Gao He

	when <fun> = 0 in the +CFUN command (section 3.14). • Added the +CGREG command (section 3.29).

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.
ℹ	Means note or tips for readers to use the module

Related Documents

Neoway_N58_Datasheet

Neoway_N58_Product_Specifications

Neoway_N58_HW_User_Guide

Neoway_N58_EVK_User_Guide

1 Boot LOG Instruction

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

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

Network indicator status

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

2 AT Syntax

2.1 Symbols

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

2.2 Description

Prefix

AT or at

Command Line

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

Extended commands, defined by Neoway

Joint Mark

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

Termination Character

<CR>, i.e. 0x0D

Response Syntax

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

Response can be one or multiple messages.

Result Syntax

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

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

For the error codes, see Appendix A.

2.3 Command Types

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

2.4 Command Response Time-Out

Every command issued to the Neoway module returns a result response and the time needed to process the given command and return the response varies, depending on the command type.

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

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

No.	Command	Estimated maximum time to get response (Seconds)
1	AT+COPS	180
2	AT+CLCK	15
3	ATD*99#	30
4	AT+CMGR	30
5	AT+CMGL	30
6	AT+CMGS	30
7	AT+XIIC	60
8	AT+TCPSETUP	60
9	AT+TCPSEND	30
10	AT+TCPCLOSE	5
11	AT+UDPSETUP	30
12	AT+UDPSEND	30
13	AT+TCPSENDS	30
14	AT+TCPTRANS	60
15	AT+FTPLOGIN	30
16	AT+FTPLOGOUT	30
17	AT+FTPGET	30
18	AT+FTPPUT	30
19	AT+FTPSIZE	30
20	AT+FILEFTPGET	60
21	AT+NWFTPRENAME	30
22	AT+NWFTPMKDIR	30
23	AT+NWFTPRMDIR	30

24	AT+NWFTPDEL	30
25	AT+HTTPSETUP	60
26	AT+HTTPACTION	60
27	AT+HTTPCLOSE	60
28	AT+HTTPGET	60
29	AT+HTTPSSETUP	60
30	AT+HTTPSACTION	60
31	AT+HTTPSCLOSE	60
32	AT+HTTPSGET	60
33	AT+FILEHTTPACTION	60
34	AT+FILEHTTPSACTION	60
35	ATD	60
36	AT+WIFIAPSCAN	30
37	AT+WIFIGSMLOC	30
38	AT+CLOUDSUB	30
39	AT+CLOUDPUB	30
40	AT+CLOUDUNSUB	30
41	AT+CLOUDDISCONN	30s+100ms
42	AT+MQTTWILLMSG	30
43	AT+MQTTSUB	10
44	AT+MQTTUNSUB	10
45	AT+MQTTPUB	30
46	AT+MQTTPUBS	30
47	AT+MQTTDISCONN	10
48	AT+AWSCONN	20
49	AT+NWBTBLEPWR	10
50	AT+NWBLESCAN	Customizing
51	AT+CIPGSMLOC	10
52	AT+UPDATETIME	60
53	AT+FSWF	Customizing, 0 - 240
54	AT$CLOSEUDPLISTEN	5
55	AT$CLOSEUDPCLIENT	5
56	AT$UDPSEND(S)	30

57	AT+MYCELLINFO	300
58	AT+CGACT	60
59	AT+NWFOTA	90

3 General Commands

3.1 ATI – Querying the Manufacturer Information

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

Format

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

Parameter

<manufacturer>	Module manufacturer, module name, software version.
<module_version>	Module model
<soft_version>	software version

Example

```
ATI
NEOWAY                                           Manufacturer
N58                                              Module model
V001                                             Version
OK
```

3.2 AT+GMR – Querying the Software Version

To query the software version

Format

Type	Command	Response

Execute	AT+GMR<CR>	<CR><LF>+GMR: <reversion> <CR><LF>OK<CR><LF>

Parameter

<reversion>   Software version of the module.

Example

```
AT+GMR                                                  Query the software version
+GMR: N58-R04-STD-BZ-03
OK
```

3.3 AT+CSQ – Querying Signal Quality

To query the receiving signal strength indication (RSSI).

Format

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

Parameter

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

	signal	rssi
0	<4 or 99	<-107 dBm or unknown
1	<10	< -93 dBm
2	<16	< -111 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                                                  Query the signal quality.
+CSQ: 19,2
OK
```

3.4 AT+CREG – Querying the Network Registration Status

To query the network registration status of the module.

Format

Type	Command	Response
Execute	AT+CREG=[<n>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CREG:<stat>[,<lac>,<ci>[,<Act>]]<CR><LF>
Query	AT+CREG?	<CR><LF>+CREG: <n>,<stat> <CR><LF>OK<CR><LF>
Test	AT+CREG=?	<CR><LF>+CREG: range of supported<n> <CR><LF>OK<CR><LF>

Parameter

<n>
0: Disable network registration unsolicited result code (default setting).
1: Enable network registration unsolicited result code +CREG: <stat>.
2: Enable network registration and location information (Cell ID, Local ID) unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>]]

<stat>
0: Not registered, the module is not currently searching an operator to register to
1: Registered the home network
2: Not registered, but the module is currently trying to attach or searching an operator to register to
3: Registration denied
4. Unknown code
5: Registered, roaming
6: ltesms only home
7: ltesms only roaming
8: EMER SVCE ONLY
9: CSFB NOT PREFER HOME
10: CSFB NOT PREFER ROAMING

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

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

<Act> 0: GSM
1: GSM compact
2: UTRAN
3: GSM w/EGPRS
4: UTRAN w/HSDPA
5: UTRAN w/HSUPA
6: UTRAN w/HSDPA AND w/HSUPA
7: E-UTRAN
8: UTRAN w/HSPA+

Example

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

3.5 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>
Query	AT+CEREG?<CR>	<CR><LF>+CEREG: <stat>[,[<tac>],[<ci>],[<AcT>][,,[,[<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>

Parameter

<n>
0: Disable network registration unsolicited result code (default setting).
1: Enable network registration unsolicited result code +CREG: <stat>.
2: Enable network registration and location information (Cell ID, Local ID) unsolicited result code +CREG: <stat>[,[<lac>],[<ci>],[<AcT>]]
4: Enable network registration unsolicited result codes containing Active-Time and Periodic-TAU

<stat>
0: Not registered, the module is not currently searching an operator to register to
1: Registered the home network
2: Not registered, but the module is currently trying to attach or searching an operator to register to
3: Registration denied
4. Unknown code
5: Registered, roaming

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

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

<Act>
0: GSM
1: GSM compact
2: UTRAN
3: GSM w/EGPRS
4: UTRAN w/HSDPA
5: UTRAN w/HSUPA
6: UTRAN w/HSDPA and HSUPA
7: E-UTRAN

Example

```
AT+CEREG?                Query the network registration status of the module.
+CEREG: 0,1              Registered the home network.
OK
AT+CEREG=1               Enable unsolicited codes of network registration.
OK
AT+CEREG=?               Query the value range of the network registration status parameter.
+CEREG: (0-2,4)
OK
```

3.6 AT+COPS – Selecting an Operator

To select an operator.

Format

Type	Command	Response
Execute	AT+COPS=[<mode>[,<format>[,<oper>>[,<AcT>]]]]<CR>	<CR><LF>OK<CR><LF>
Query	AT+COPS?<CR>	<CR><LF>+COPS:<mode>[,<format>,<oper>[,<AcT>]]
Test	AT+COPS=?<CR>	<CR><LF>+COPS: [list of supported (<stat>,long alphanumeric <oper>,short alphanumeric <oper>,numeric <oper>[,<AcT>])s][,,(list of supported<mode>s),(list of supported <format>s)]<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode>	To set automatic network selection or manual selection. 0: Automatic selection (ignore the parameter <oper>) 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). 1: Short format alphanumeric <oper> 2: Numeric <oper>
<oper>	It is given in <format>. This field may be in 16-character long alphanumeric format, 8-characters short alphanumeric format, or 5-character numeric format (MCC/MNC).
<AcT>	indicates the radio access technology and its value can be 0, 1, and 2. 0: GSM 1: GSM compact 3: GSM w/EGPRS 7: E-UTRAN

Example

```
AT+COPS=0,0                         Automatic network selection is enabled. Long alphanumeric
OK                                  mode.
AT+COPS=0,2                         Set to digital mode
OK
AT+COPS?                            China Mobile
+COPS: 0,0,"CHINAMOBILE",7
OK
AT+COPS?                            If it is set to digital mode, get the number 46000.
```

```
+COPS: 0,2,"46000",7
OK
AT+COPS?                            China Unicom
+COPS: 0,0,"CHINA UNICOM",7
OK
AT+COPS?                            If it is set to digital mode, then get the number 46001.
+COPS: 0,2,"46001",7
OK
AT+COPS?                            China Telecom
+COPS: 0,0,"CHINA TELECOM",7
OK
AT+COPS?                            If it is set to digital mode, get the number 46011.
+COPS: 0,2,46011,7
OK
AT+COPS=2                           Deregister the network.
OK
```

3.7 AT+CIMI – Requesting International Mobile Subscriber Identity (IMSI)

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

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 Identity.
IMSI is a character string of 15 digits and starts with 3-bits of MCC and 2-bits of MNC. It is used to authenticate the SIM card.

Example

```
AT+CIMI                                          Query the IMSI.
+CIMI: 460020188385503                           IMSI: 460022201575463
OK
AT+CIMI                                          Query the IMSI.
ERROR                                            No SIM card is installed.
```

3.8 AT+CGSN – Requesting Product Serial Number Identification

To obtain the product serial number, identified as the IMEI of the module.

Format

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

Parameter

<IMEI>   Product serial number identification of the module.

Example

```
AT+CGSN                                              Read command
+CGSN: 355897043139120
OK
```

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

3.9 AT+GSN – Requesting Product Serial Number Identification

To obtain the product serial number, identified as the IMEI of the module.

Format

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

Parameter

<IMEI>   Product serial number identification of the module, a character string of 15 digitals.

Example

```
AT+GSN                                                  Query the IMEI number.
+GSN: 355897043139120
OK
```

3.10 AT+CCID – Obtaining ICCID of 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>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

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

Example

```
AT+CCID                                                 Read command
+CCID: 89860002190810001367
OK
AT+CCID                                                 The SIM card is not inserted.
ERROR
```

3.11 AT+CGMM – Requesting Model Identification

To request model identification which consists of a single line containing the name of the module.

Format

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

Parameter

<model>	Module model

Example

```
AT+CGMM                                           Request model identification of the module.
+CGMM: N58
OK
```

3.12 AT+GMM – Querying Module Model

To request model identification which consists of a single line containing the name of the module.

Format

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

Parameter

<model>	Module model

Example

```
AT+GMM                                            Request model identification which consists of
+GMM: N58                                         a single line containing the name of the
OK                                                module.
```

3.13 AT+IPR – Setting Baud Rate

To set the baud rate of the module.

If 0 is returned after querying, the baud rate of the module is self-adaptive. The UART communication is baud rate self-adaptive by default.

Format

Type	Command	Response
Execute	AT+IPR=<baud rate><CR>	<CR><LF>OK<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>	Baud rate (0, 2400, 4800, 9600, 14400, 19200, 28800, 33600, 38400, 57600, 115200, 230400, 460800, 921600, 2166666)

Example

```
AT+IPR=115200                                  Set the baud rate to 115200 bps.
OK
AT+IPR?                                        Query the current baud rate.
+IPR: 115200
OK
AT+IPR=?                                       Query the available baud rate range.
+IPR:0,2400,4800,9600,14400,19200,28800,3360
0,38400,57600,115200,2166666

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

3.14 AT+CFUN – Setting Module Functionality

To select the level of functionality of the module by setting <fun>. <fun>: only certain values are supported.

The setting is not saved after the module is powered off.

Format

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

Parameter

<fun>	0: minimum functionality 1: normal operation mode (default) 4: turn off the TX and RX RF circuits (Flight mode)
<rst>	0: do not reset the MT before setting it to <fun> power level 1: reset the MT before setting it to <fun> power level

Example

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

3.15 AT+CMUX – Enabling/Disabling Multiplexing Mode

This command is used to manage (enable or disable) the multiplexing protocol control channel.

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 network and the rest are used for AT command sending and receiving.

Executing AT+CMUX=0 to enable the multiplexing protocol control channel is recommended.

Format

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

Parameter

<mode> The mode of MUX that is enabled, integer type.
0: Basic option (default value)
1: Advanced option (not supported)

<subset> Subset of frame format, integer type
0: UIH frames used only (default value).
1: UI frames used only (not supported currently)

<port_speed> UART port rate, integer type
1: 9600 bit/s
2: 19200 bit/s
3: 38400 bit/s
4: 57600 bit/s
5: 115200 bit/s (default)
6: 230400 bit/s

<N1> Maximum frame size. Integer type, ranging from 1 to 2048.
The default value is 31.
For Advanced option, the default value is 64.

<T1> Acknowledgment timer in unite of ten milliseconds, integer type, ranging from 1 to 255, where 10 is default (equal to 100 ms).

<N2> Maximum number of re-transmissions, integer type, ranging from 0 to 5. The default value is 3.

<T2> Response timer for the multiplexer control channel in units of ten milliseconds, integer type. Ranging from 2 to 255. The default value is 30 (300 ms).

<T3> Wake up response timer in seconds, integer type. Ranging from 1 to 255. 1 indicates 1 second. The default value is 10s.
(Not supported)

<k>	Window size, integer type. (Not supported)

<T2> must be greater than <T1>.

Example

```
AT+CMUX=0                                           Basic option
OK
AT+CMUX=2                                           ERROR is returned because the set value
ERROR                                               exceeds the parameter range.
AT+CMUX=0,0,,512,254,5,255                          Basic option.
OK
AT+CMUX=1,0,,512,254,5,255                          Advanced option.
OK
AT+CMUX=?
+CMUX:  (0,1),(0),(1-6),(1-2048),(1-255),(0-        Query the available range of parameters.
100),(2-255),(1-255),(1-7)
OK
AT+CMUX?                                            The command format is incorrect.
ERROR
```

3.16 AT+CCLK – Clock

To set and query the real-time clock.

The settings are not saved after the module is powered off. The default clock is GMT+0.

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>OK<CR><LF>

Parameter

<time>	Character string in format of yy/MM/dd,hh:mm:ss[TZ].

TZ A 2-digit number indicates the time difference between local time and GMT. This value is displayed only when the network is supported.

Example

```
AT+CCLK="18/07/01,14:54:01"
                                          Set the real-time clock.
OK
AT+CCLK?
+CCLK: "18/07/01,14:54:10+32"             Query the current clock.
OK
AT+CCLK=14/07/02,10:48:50
                                          The parameter format of <time> is incorrect.
ERROR
```

3.17 AT+CPIN – Entering PIN Code

to enter a password or query whether or not the module requires a password which is necessary before it can be operated. The password may be SIM PIN, SIM PUK, PH-SIM PIN, etc.

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

Format

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

Parameter

<pin>, <newpin> String type

<code> READY: ME is not pending for any password.
SIM PIN: ME is waiting SIM PIN to be given.
SIM PUK: ME is waiting SIM PUK to be given.
SIM PIN2: ME is waiting SIM PIN2 to be given.
SIM PUK2: ME is waiting SIM PUK2 to be given

Example

```
AT+CPIN?
+CPIN: READY                               Query whether PIN code is required.
OK                                         You don't need to enter any password.
AT+CPIN?                                   Query whether PIN code is required.
+CPIN: SIM PIN                             PIN code is required.
OK
AT+CPIN="1234"
OK                                         Correct PIN code is entered.

+PBREADY                                   The SIM card is unlocked.
AT+CPIN?                                   If PIN code is input incorrectly for three times,
+CPIN: SIM PUK                             PUK is required to unlock.
OK
AT+CPIN="12345678","4321"
OK                                         Enter PUK code and then enter the new PIN code.

+PBREADY                                   The SIM card is unlocked.
```

3.18 AT+CLCK – Facility Lock

To lock, unlock or interrogate a MT or a network facility <fac>. The setting by this command valid after the module is restarted.

Format

Type	Command	Response
Set	AT+CLCK=<fac>,<mode>[,<passwd>[,<class>]]<CR>	● When <mode>=2: <CR><LF>+CLCK: <status>[,<class1> [<CR><LF>+CLCK: <status>,<class2>[...]] <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> ● When <mode>≠2: <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+CLCK=?<CR>	<CR><LF>+CLCK: (list of supported <fac> values) <CR><LF>OK<CR><LF>

Parameter

<fac>   Value with double quotes.
"OI": Outgoing international calls

"SC": SIM card
"AO": All outgoing calls
"OX": All outgoing international calls except to the home country
"FD": Fixed dialing of the SIM card

<mode> 0: unlock
1: lock
2: registration

<status> 0: not active
1: active

<passwd> Password or operation code; string type, string with double quotes.

<classx> 1: voice service
2: data service
4: fax service
8: SMS
16: data circuit sync
32: data circuit async
64: dedicated packet access
128: dedicated PAD access

Example

```
AT+CLCK="SC",2
+CLCK: 0
OK
AT+CLCK=?
+CLCK:(“SC","FD","AO","OX","OI")  Query the network information related to the module.
OK
AT+CLCK="SC",1,"1234"             Lock the current SIM card. "1234" is PIN code of current SIM
OK                                card.
AT+CLCK="SC",0,"1234"             Unlock the current SIM card. "1234" is PIN code of current
OK                                SIM card.
AT+CLCK="SC",1,"2222"
                                  Incorrect PIN code is inputted.
ERROR
```

3.19 AT+CPWD – Changing the Password

To set a new password for the facility lock function defined by command **Facility Lock**.

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

Format

Type	Command	Response
Set	AT+CPWD=<fac>,<oldpwd>,<newpwd><CR>	<CR><LF>OK<CR><LF>

		Or <CR><LF>ERROR<CR><LF>
Test	AT+CPWD=?<CR>	<CR><LF>+CPWD:(list of supported (<fac>,<pwdlength>)s) <CR><LF>OK<CR><LF>

Parameter

<fac>	Value with double quotes. "P2": SIM PIN2 "SC": SIM card
<oldpwd>	string with double quotes; the old password or operation code.
<newpwd>	string with double quotes; the new password or operation code.

Example

```
AT+CPWD=?
+CPWD: ("SC",8),("P2",8)           Query the parameters range.
OK
AT+CPWD="SC","1234","0000"         Modify the PIN code of the current SIM card. "1234" is the old
OK                                 PIN code and "0000" is the new PIN code.
AT+CPWD=SC,1234,0000               The command format is incorrect; a pair of quotation marks is
ERROR                              a must for the value.
```

3.20 AT+CGDCONT – Defining PDP Context

To specify GPRS PDP context parameter values for a PDP context.

Format

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

	(<cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>)] <CR><LF>OK<CR><LF>

Parameter

<cid> integer type; specifies a particular PDP context definition. The parameter is local to the TE-MT interface and is used in other PDP context-related commands. (The minimum value is 1).

<PDP_type> string type; specifies the type of packet data protocol.
"IP" Internet Protocol (IETFSTD 5)

<APN> Access Point Name, string type; a logical name that is used to select the GGSN or the external packet data network.

<PDP_address> String type, identifies the MT in the address space applicable to the PDP.
TE will provide a value for this parameter after PDP starts if it is null or omitted. If TE fails to provide, a dynamic address is requested, and even if the address is assigned during the PDP startup process, it is returned empty when queried with this command.

<d_comp> Integer type, controls PDP data compression (applicable for SNDCP only)
0 - off (if omitted, this parameter is the default value).

<h_comp> integer type; controls PDP header compression 0 - off (default value)

<pd1>, … <pdN> String type, their definitions are corresponding to <PDP_type>

Example

```
AT+CGDCONT=1,"IP","CMNET"                  Set the PDP type to IP and set the APN name
OK                                         to CMNET.
AT+CGDCONT?
+CGDCONT: 1,"IP","CMNET"," IPV4:0.0.0.0",0,0      Query the current PDP format.
OK
AT+CGDCONT=?
+CGDCONT: (1-7),(IP,IPV6,IPV4V6,PPP,Non-IP),(0-   Query the value range of the PDP format, the
3),(0-4)                                          number of parameters.
OK
```

3.21 AT+XGAUTH – PDP Authentication

PDP authentication.

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.

The default user name and password of the China Unicom SIM card is "card" and "card".

This <cid> parameter corresponds to the <cid> parameter of the +CGDCONT command.

Format

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

Parameter

<cid> (PDP Context Identifier) a numeric parameter that specifies a particular PDP context definition.
This <cid> parameter corresponds to the <cid> parameter of the +CGDCONT command.

<auth> Authentication type
0: NONE
1: PAP (default)
2: CHAP
When the PDP authentication typeb is NONE, the command contains the <name> and <pwd> parameters.

<name> User name

<pwd> Password

Example

```
AT+XGAUTH=1,1,”gsm”,”1234”                  Set the first PDP authentication.
OK
AT+XGAUTH=?
+XGAUTH: (1-7),(0-2),32,32                  Query the value range of the parameters.
OK
```

3.22 AT+CGATT – Setting GPRS Attach and Detach

To attach the MT to, or detach the MT from, the Packet Domain service. The setting by this command is not saved after the module is powered off.

If the initial PDP context is supported, the context with <cid>=0 is automatically defined at startup.

Ensure that the GPRS attach is set before the PPP connection is set up.

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

Format

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

Parameter

<state>	0: detached 1: attached

Example

```
AT+CGATT=1
OK                                          GPRS is attached successfully.
AT+CGATT=0
OK                                          GPRS is detached successfully.
AT+CGATT=0
GPRS DISCONNECTION                          Send the return value of this command after
                                            AT+XIIC=1 is executed to establish a PPP connection.
OK
AT+CGATT=0                                  The SERROR is returned because no SIM card is
ERROR                                       inserted.
AT+CGATT?                                   Query the current GPRS status.
+CGATT: 0
OK
AT+CGATT=?                                  Query the valid parameter values for the command.
+CGATT: (0-1)
OK
```

3.23 ATE1/ATE0 – Enabling/Disabling the Terminal Display

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

The terminal display function is enabled by default.

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

Format

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

Parameter

<value>   Whether to enable the terminal display function of AT commands
0: disabled (default)
1: enable

ATE = ATE0

Example

```
                                                Turn on the terminal display function.
ATE1
OK                                              Send “AT”
AT
OK                                              Serial tool displays “AT” and “OK”

ATE0                                            Turn off the terminal display function.
OK
                                                Send “AT”
OK                                              Serial tools displays only “OK”.
```

3.24 ATD*99# – GPRS

To establish a GPRS connection using external protocol stacks.

Ensure that the module has registered the network (through CREG) and set APN before dialing any number.

Format

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

Parameter

N/A.

Example

```
ATD*99#                                                     Start a dial-up connection.
CONNECT                                                     Successful
```

3.25 AT+ENPWRSAVE – Enabling or Disabling Sleep Mode

To enable or disable sleep mode. The settings by this command are not saved after the module is powered off.

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

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

Format

Type	Command	Response
Set	AT+ENPWRSAVE=<n>[,<usb>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+ENPWRSAVE?<CR>	<CR><LF>+ENPWRSAVE: <n><usb> <CR><LF>OK<CR><LF>

Parameter

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

<usb> 0: Forbid USB remote wake-up (default)
1: allow USB remote wakeup (the module enters sleep mode only after the USB host suspends the USB. After the USB host resumes the USB bus or there are network DL events (data, SMS, call), the module wakes up the USB host through the USB bus.)

Example

```
AT+ENPWRSAVE=1,1
OK                                  Set the module to allow sleep mode. Allow USB remote wakeup.
AT+ENPWRSAVE?
+ENPWRSAVE: 1,0                     Query current sleep mode status.
OK
```

3.26 AT+SIGNAL – Setting Blinking Status Signal Indicator

To set the different blinking status of the signal indicator.

The default status setting is 7.

If the status is set to 0 to 6, the indicator will be always on when there is an incoming call or SMS in sleep mode.

The settings by this command are saved after the module is powered off. Setting of 8, 9, or 10 are not saved.

Format

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

Parameter

<value> Integer type, ranging from 0 to 11.
0: Blink once every second in normal situation. Being off or on if any abnormality

occurs.
1: Blink once every second after the module is connected to the GPRS network. Being off in any other situations.
2: Flash and blink. Flash every 250 ms for the GPRS data service and blink every second in other normal situations.
3: Be on after the GPRS network is connected and blink every second in other situations.
4: Be on after the GPRS network is connected and being off in other situations.
5: Be off if the SIM card cannot be detected after the module is started, blink every second if the SIM card is detected,
and be on after the GPRS network is connected.
6: Four indicator states:
If no SIM card is installed or the SIM card does not register network, the indicator blinks every one second and is on for 0.1 second.
If the SIM card registered network, the indicator blinks every three second and is on for 0.1 second.
If the GPRS network is connected, the indicator blinks every 250 ms and is on for 0.1 second.
The indicator is always on during a call.
7: Four indicator states:
Off: No SIM card, not registered
On: registered network
On for 0.2 second and off for 1.8 second: Obtained the IP address
On for 1.8 second and off for 0.2 second: Connected to the server
8: Be off all the time
9: Be on all the time
10: custom on/off interval, which is defined in <low_interval>/<high_interval>.
Indicator off time, ranging from 10 to 65535, unit: ms
On time, ranging from 10 to 65535, unit: ms
11: Be off all the time: no register network
On for 100 ms and off for 100 ms: searching the network
On for 100 ms and off for 3000 ms: register network
On for 100 ms and off for 300 ms: GPRS network is connected.

Example

```
AT+SIGNAL?                The current signal indicator status is 2.
+SIGNAL: 2
OK
AT+SIGNAL=3               Set current signal indicator status to 3.
OK
AT+SIGNAL=100             Incorrect parameter settings, the parameter value is out of range.
ERROR
AT+SIGNAL=?               The available value of the signal indicator status ranges from 0 to
+SIGNAL: (0-11)           7.
OK
```

3.27 AT+CESQ – Extended Signal Quality

To query the extended signal quality.

- If the current serving cell is not a GERAN cell, <rxlev> and <ber> are set to value 99.

If the current serving cell is not a UTRA FDD or UTRA TDD cell, <rscp> and <ecno> are set to 255.

If the current serving cell is not an E-UTRA cell, <rsrq> and <rsrp> are set to 255.

- If the current serving cell is not an NR cell, <ss_rsrq>, <ss_rsrp> and <ss_sinr> are not displayed.

For the detailed rule, see the 3GPP TS 27.007 8.69.

Format

Type	Command	Response
Execute	AT+CESQ<CR>	<CR><LF>+CESQ: <rxlev>,<ber>,<rscp>,<ecno>, <rsrq>,<rsrp>,<ss_rsrq>,<ss_rsrp>,<ss_sinr> <CR><LF>OK<CR><LF>
Test	AT+CESQ=?<CR>	<CR><LF>+CESQ: (list of supported <rxlev>s),(list of supported <ber>s),(list of supported <rscp>s),(list of supported <ecno>s),(list of supported <rsrq>s),(list of supported <rsrp>s),(list of supported <ss_rsrq>s),(list of supported <ss_rsrp>s),(list of supported <ss_sinr>s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<rxlev>	integer type, received signal strength level
<ber>	Bit Error Rate
<rscp>	received signal strength indicator
<ecno>	downlink carrier interference ratio
<rsrq>	reference signal quality
<rsrp>	reference signal received power
<ss_rsrq>	reference signal command (based on synchronization signal)
<ss_rsrp>	reference signal received power (based on synchronization signal)

<ss_sinr> signal-to-noise and interference ratio (based on synchronization signal)

Example

```
AT+CESQ                                              Query the signal quality.
+CESQ: 99,99,255,255,16,47
OK
AT+CESQ=?                                            Signal display range
+CESQ: (0-62,99),(0-7,99),(255),(255),(0-
34,255),(0-97,255)
OK
```

3.28 AT+NWDNS – Parsing the Domain Name

To query the DNS parsing result after the module establishes a dial-up connection using the internal protocol stack successfully.

Before executing this command, ensure that the dial-up connection is established through the AT+XIIC command successfully.

Ensure that the entered content is correct since its correctness is not verified.

Currently the query command only retrieves the IPv4 address.

Format

Type	Command	Response
Execute	AT+NWDNS=<hostname><CR>	<CR><LF>+NWDNS: <Sign>,<IP><CR><LF> <CR><LF>+NWDNS: <Sign>,<IP><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NWDNS?<CR>	<CR><LF>+NWDNS: <Sign>,<IP><CR><LF> <CR><LF>+NWDNS: <Sign>,<IP><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<hostname> Hostname, character type, 128 bytes in length at most; double quotation marks are required.

<IP> IP address, string type

<Sign> IPv4 or IPv4, string type

Example

```
AT+NWDNS=”WWW.BAIDU.COM”          An empty string is returned because there is no IPv6 address.
+NWDNS: IPV4,”220.181.112.244”

+NWDNS: IPV6,””

OK
AT+NWDNS=”www.google.com”
                                  Querying of the google domain name times out.
ERROR
AT+NWDNS=”www.google.com”         No dialing; the returned value indicates that PDP is not
ERROR                             activated.
AT+NWDNS?
+NWDNS: IPV4,”220.181.112.244”
                                  Query the obtained IP address.
+NWDNS: IPV6,””

OK
```

3.29 AT+CGREG – Querying GPRS Network Registration Status

To query the GPRS network registration status.

Format

Type	Command	Response
Execute	AT+CGREG=[<n>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CGREG?<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CGREG:<stat>[,<lac>,<ci>[,<Act>]]<CR><LF>
Test	AT+CGREG=?<CR>	<CR><LF>+CGREG: <n>,<stat> <CR><LF>OK<CR><LF>

Parameter

<n>	Integer type Result code presentation mode

0: Disable network registration unsolicited result code (default).
1: Enable network registration unsolicited result code
2: Enable network registration unsolicited result code (CELL ID, LOCAL ID)

<stat> Registration status, integer type.
0: Not registered, the module is not currently searching an operator to register to
1: Registered the home network
2: Not registered, but the module is currently trying to attach or searching an operator to register to
3: Registration denied
4. Unknown code
5: Registered, roaming
6: ltesms only home
7: ltesms only roaming
8: EMER SVCE ONLY
9: CSFB NOT PREFER HOME
10: CSFB NOT PREFER ROAMING

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

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

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

Example

```
AT+CGREG=1                Enable network registration unsolicited result code.
OK                        After this set command is executed, OK is not displayed.

AT+CGREG?
+CGREG: 0,1               Query the GPRS network registration status.
OK
AT+CGREG=?
+CGREG: (0-2)             Query the range of the command parameter.
OK
```

4 SMS Commands

4.1 AT+CSMS – Selecting SMS Services

To check the message types supported by the module. There are three message types: mobile-originated SMS messages (SMS-MO), mobile-terminated SMS messages (SMS-MT) and cell broadcast messages (SMS-CB).

Format

Type	Command	Response
Execute	AT+CSMS=<service><CR>	<CR><LF>+CSMS: <mt>,<mo>,<bm><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CSMS?<CR>	<CR><LF>+CSMS: <service>,<mt>,<mo>,<bm><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+CSMS=?<CR>	<CR><LF>+CSMS: (list of supported <service>s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<service>	0: GSM 03.40 and 03.41 (the syntax of SMS AT commands is compatible with GSM 07.05 Phase 2 1: GSM 03.40 and 03.41 (the syntax of SMS AT commands is compatible with GSM 07.05 Phase 2+
<mt>,<mo>,<bm>	0: Not support 1: support

The default value is 0,1,1,1.

Example

```
AT+CSMS=1
+CSMS: 1,1,1                         Set SMS service to 1.
OK
AT+CSMS?
+CSMS: 1,1,1,1                       Query the current parameter values.
OK
AT+CSMS=?
+CSMS: (0-1)                         Query the value range of SMS service.
OK
```

4.2 AT+CPMS – Setting Preferred SMS Storage

To set preferred SMS storage.

Format

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

Parameter

<mem1> Memory from which SMS messages are read and deleted, string type
<mem2> Memory to which writing and sending operations are made, string type
<mem3> memory to which received SMS are preferred to be stored (unless forwarded
"SM": directly to TE; refer command New Message Indications +CNMI)
"ME": SIM message storage
"MT": ME message storage
any of the storages associated with ME
<mem1> String type, for example, “SM”, “ME”

"SM": SIM only
"ME": ME only

<used>	number of messages currently in memory.
<total>	total number of message locations in memory.

The received SMS message is stored in SM by default.

Example

```
AT+CPMS="SM"                                                 Set the SMS storage to "SM",
+CPMS: 0,50,0,50,0,50                                        that is, store SMS messages in
OK                                                           SIM card.
AT+CPMS?                                                     Query the capacity of current
+CPMS: "SM", 0, 50, "SM", 0, 50,"SM", 0, 50                  SMS storage.
OK
AT+CPMS=?                                                    Query the memory that can be
+CPMS: ("ME","SM","MT"),("ME","SM","MT"),("ME","SM","MT")    set.
OK
```

4.3 AT+CMGF – Setting Message Format

Set command tells the TA, which input and output format of messages to use.

Format

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

Parameter

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

Example

```
AT+CMGF=1                                   Set SMS to text mode.
OK
AT+CMGF?
+CMGF: 1                                    Query the current mode of SMS message input.
OK
AT+CMGF=?
+CMGF: (0-1)                                Query the value range of <mode>.
OK
```

4.4 AT+CSCS – Setting the TE Character Set

To set “TE” character set.

Format

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

Parameter

<chset> "GSM": default GSM alphabet (GSM03.38.6.2.1).
"IRA": international reference alphabet)(ITU-T T.50).
"UCS2":16-bit universal multiple-octet coded character set (USO/IEC10646). UCS2 character strings are
converted to hexadecimal numbers from 0x0000 to 0xFFFF. UCS2 encoding is used only in some character string of the statement; the rest of the commands and responses are still in the IRA alphabet format.
"PCCP936": the same as the GBK encoding format.
"IRA": international reference alphabet) (ITU-T T.50).

The default character set is "PCCP936".

Example

```
AT+CSCS="GSM"                              Set “TRA” character.
OK
AT+CSCS?
+CSCS: "PCCP936"                           Query the format of current character set.
OK
AT+CSCS=?                                  Query the character set formats that the module
+CSCS: ("GSM","HEX","PCCP936","UCS2")      supports.
OK                                         A list of character set formats are returned.
```

4.5 AT+CNMI – New Message Indications to TE

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

Format

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

Parameter

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

<mt>   Set the format of the new SMS instruction codes. The default value is 0.
0: SMS instruction codes will not be sent to the end device.

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

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

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

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

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

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

<ds> report status of SMS message sending. The default value is 1.

0: no status report of SMS message sending.

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

<bfr> The default value is 0.

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

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

- The default value is 0, 0, 0, 0, 0;
- The recommended setting is +CNMI: 2,1,0,0,0 (new messages are stored on SIM card rather than displayed directly) or +CNMI: 2,2,0,0,0 (new messages are displayed rather than stored on SIM card).
- SMS message types:
- Class 0: Displayed not stored
- Class 1: Stored in ME
- Class 2: Stored in SIM

Class 3: Sent to TE

Example

```
AT+CNMI=1,1,0,0,0                          Set the SMS message indication mode.
OK
AT+CNMI=?
+CNMI: (0-3),(0-3),(0,2),(0-1),(0-1)       Query the range of command parameters that can be
OK                                         set.
```

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

4.6 AT+CMGR – Reading SMS Message

To read SMS messages stored in current memory (use the AT+CPMS command to specify the current memory)

If the received message is unread, its status in the storage changes to received read after executing this command.

Format

Type	Command	Response
Execute	AT+CMGR=<index><CR>	TEXT mode (+CMGF=1) • The command is executed successfully and the command belongs to SMS-DELIVER: <CR><LF>+CMGR: <stat>,<oa>,[<alpha>],<scts>[,<tooa>,<fo>,<pid>,<dcs>,<sca>,<tosca>,<length>]<CR><LF><data> • The command is executed successfully and the command belongs to SMS-SUBMIT: <CR><LF>+CMGR: <stat>,<da>,[<alpha>][,<toda>,<fo>,<pid>,<dcs>,[<vp>],<sca>,<tosca>,<length>]<CR><LF><data> • The command is executed successfully and the command belongs to SMS-STATUS REPORT: <CR><LF>+CMGR: <stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>, <st> • The command is executed successfully and the command belongs to SMS-COMMAND: <CR><LF>+CMGR: <stat>,<fo>,<ct>[,<pid>,[<mn>],[<da>],[<toda>],<length><CR><LF><cdata>] • The command is executed successfully and the command belongs to CBM-STORAGE: <CR><LF>+CMGR: <stat>,<sn>,<mid>,<dcs>,<page>,<pages><CR><LF><data> When PDU mode (+CMGF=0) and the command is

	executed successfully: <CR><LF>+CMGR: <stat>,[<alpha>],<length><CR><LF><pdu> ● Failed to execute the command: <CR><LF>ERROR<CR><LF>

Parameter

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

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

<alpha> String type alphanumeric representation of <da> or <oa> corresponding to the entry found in MT phonebook.

<length> The number of octets of the given TP-level data unit (octets that do not contain the service center address)

<pdu> PDU data

Example

```
AT+CMGR=1                                                                  Read the message
+CMGR: "REC READ","66421",,"11/09/13,16: 37: 59+32"                        indexed as 1.
050003140401E27778592EA7E7EBE9373C3C279BCF68F59AADC7FED62779BA596D7EBA
EB5B91EBD16A5D46C35F98406A744E311A95C32594DA75688B50EADACA6D689150EADF
1B2BC5E579AD575E5B5582D5EABD5624C36A3D56C375C0E1693CD6835DB0D9783A15C9
1D2E06BDAA558AC1F60C52B937CADCD2B747AA9021BDEC627E8E9441BD42655DEF446
OK

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

+CMTI: "SM",39
AT+CMGR=39                                                                 Incoming SMS.
+CMGR: 0,,23                                                               Read the SMS message.
0891683110501905F0240BA18177377949F50000413062312503230468341A0D
OK
AT+CMGF=1                                                                  Set the text mode.
OK
```

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

4.7 AT+CMGL – SMS Message List

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

Format

Type	Command	Response
Execute	AT+CMGL[=<stat>]<CR>	TEXT mode (+CMGF=1) • The command is executed successfully SMS-SUBMITs or SMS-DELIVERs: <CR><LF>+CMGL: <index>,<stat>,<oa/da>,[<alpha>],[<scts>][,<tooa/toda>,<length>]<CR><LF><data>[<CR><LF>+CMGL: <index>,<stat>,<da/oa>,[<alpha>],[<scts>][,<tooa/toda>,<length>]<CR><LF><data>[...]] • The command is executed successfully and the command belongs to SMS-STATUS-REPORTs: <CR><LF>+CMGL: <index>,<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st>[<CR><LF>+CMGL: <index>,<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st>[...]] • The command is executed successfully and the command belongs to SMS-COMMANDs: <CR><LF>+CMGL: <index>,<stat>,<fo>,<ct>[<CR><LF>+CMGL: index>,<stat>,<fo>,<ct>[...]] • The command is executed successfully and the command belongs to CBM storage: <CR><LF>+CMGL: <index>,<stat>,<sn>,<mid>,<page>,<pages><CR><LF><data>[<CR><LF>+CMGL: <index>,<stat>,<sn>,<mid>,<page>,<pages><CR><LF><data>[...]]

		When PDU mode (+CMGF=0) and the command is executed successfully: <CR><LF>+CMGL: <index>,<stat>,[<alpha>],<length><CR><LF><pdu>[<CR><LF>+CMGL: <index>,<stat>,[<alpha>],<length><CR><LF><pdu>[...]] Failed to execute the command: <CR><LF>ERROR<CR><LF>
Test	AT+CMGL=?<CR>	<CR><LF>+CMGL: (list of supported <stat>s)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<stat>

String type or numeric type.
When AT+CMGF=1:
"REC UNREAD": received unread
"REC UNREAD": received read
"STO UNSENT": stored unsent
"STO SENT": stored sent
"ALL": all SMS messages
When AT+CMGF=0:
0: received unread
1: received read
2: Stored unsent SMS messages
3: Stored sent SMS messages
4: All SMS messages

Example

```
AT+CMGL="ALL"
+CMGL: 1,"REC READ","66421","","2011/09/13 16: 37: 59+32"
050003140401E27778592EA7E7EBE9373C3C279BCF68F59AADC78FED62779BA596D7EBAEB5B91EBD16A5D46C35F9840
6A744E311A95C32594DA75688B50EADACA6D689150EADF1B2BC5E579AD575E5B5582D5EABD5624C36A3D56C375C0E16
93CD6835DB0D9783A15C91D2E06BDAA558AC1F60C52B937CADCD2B747AA9021BDEC627E8E9441BD42655DEF446
+CMGL: 14,"STO SENT","66045","",
050003010401E27778592EA7E7EBE9373C3C279BCF68F59AADC78FED62779BA596D7EBAEB5B91EBD16A5D46C35F9840
6A744E311A95C32594DA75688B50EADACA6D689150EADF1B2BC5E579AD575E5B5582D5EABD5624C36A3D56C375C0E16
93CD6835DB0D9783A15C91D2E06BDAA558AC1F60C52B937CADCD2B747AA9021BDEC627E8E9441BD42655DEF446
+CMGL: 44,"REC UNREAD","8615719556937","","2011/09/30 03: 00: 55+32"
5E7F4E1C79FB52A863D0919260A8003A4E2D536B75286237003100350037003100390035003500360039003300377ED
960A86765753500326B21002C6700540E4E006B21572800320039002F00300039002000320030003A00340038002C60
A853EF6309901A8BDD952E621690099879952E76F463A556DE62E8
OK
AT+CMGL=?                                                   Query in text format (AT+CMGF=1).
+CMGL: ("REC UNREAD", "REC READ", "STO UNSENT", "STO
SENT", "ALL")
```

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

4.8 AT+CMGS – Sending SMS Messages

To send an SMS message from the module to the network. The network will return reference value <mr> to the module after the SMS message is sent successfully.

Format

Type	Command	Response
Execute	Text mode: AT+CMGS=<da>[,<toda>]<CR>text is entered<Ctrl+Z/ESC> PDU mode: AT+CMGS=<length><CR>PDU is given<Ctrl+Z/ESC>	Sent successfully in Text mode: <CR><LF>+CMGS: <mr>[,<scts>]<CR><LF> <CR><LF>OK<CR><LF> Sent successfully in PDU mode: <CR><LF>+CMGS: <mr>[,<ackpdu>]<CR><LF> <CR><LF>OK<CR><LF> Failed to execute the command: <CR><LF>ERROR<CR><LF>

Parameter

<da>	The destination number of the SMS messages.
<text>	SMS message content in text mode.
<length>	the byte length of the SMS message content in PDU mode.
<mr>	Message reference number
<CR>	end character.
<Ctrl+Z>	indicates the end of the input message, → in the example.
<ESC>	indicates giving up the input message.
<scts>	Service center time stamp
<ackpdu>	GPP 23.040 RP-User-Data element of RP-ACK PDU

Example

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

AT+CMGS="15889758493"
> This is the text➔                                       AT+CMGF=1 might not be executed.
ERROR
AT+CMGS=33                                                PDU mode(+CMGF=0)
>0891683108705505F001000B815118784271F20008146DF157335E025
B9D5B89533A59276D6A80545EFA➔
+CMGS: 119
OK
```

4.9 AT+CMGW – Writing SMS Messages

To write an SMS message into the memory. The location information <index> will be returned after the message is saved correctly.

Format

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

Parameter

<da> The destination number of the SMS messages.

<text> SMS message content in text mode.

<length> the byte length of the SMS message content in PDU mode.

<index> Location information

<CR> end character.

<Ctrl+Z> indicates the end of the input message,
➔ in the example.

<ESC> indicates giving up the input message.

If you use the serial-port commissioning tool to send PDU SMS, please press the carriage return character manually after the AT+CMGS command, or send <CR> in hexadecimal.

Example

```
AT+CMGW="091137880"                                               TEXT mode (+CMGF=1).
>This is the text<Ctrl+Z>
+CMGW: 15
OK
                                                                  the number in text mode must
AT+CMGW=091137880                                                 be enclosed with double
ERROR                                                             quotation marks.

AT+CMGW=31
>                                                                 PDU mode (+CMGF=0)
0891683108705505F001000B813124248536F300081200400026002A535A53
D153A653C1532052C7<Ctrl+Z>
+CMGW: 1
OK
```

4.10 AT+CMSS – Sending Messages from Storage

To send an SMS message specified by <index> in the memory (SMS-SUBMIT).
The network returns reference value <mr> to the end device after the SMS message is sent successfully.

Format

Type	Command	Response
Execute	AT+CMSS=<index>[,<da>[,<toda>]] <CR>	Sent successfully in Text mode: <CR><LF>+CMSS: <mr>[,<scts>]<CR><LF> <CR><LF>OK<CR><LF> Sent successfully in PDU mode: <CR><LF>+CMSS: <mr>[,<ackpdu>]<CR><LF> <CR><LF>OK<CR><LF> Failed to execute the command: <CR><LF>ERROR<CR><LF>

Parameter

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

Example

```
AT+CMSS=2
+CMSS: <mr>        Send the SMS messages stored in memory 2.
OK
AT+CMSS=2          No SMS message is stored in memory 2 or the SMS message number in memory 2 is
ERROR              incorrect.
```

4.11 AT+CMGD – Deleting SMS Messages

To delete SMS messages from the current memory.

Format

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

Parameter

<index> The recording number of the stored SMS messages.

<delflag> Integer type

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

1: delete all read SMS messages.

2: delete all read and sent SMS messages.

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

4: delete all messages.

If the <delflag> parameter has been set, the <index> parameter will be ignored.

Example

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

```
AT+CMGD=?                     Query the value ranges of parameters.
+CMGD:(1-40),(0-4)
OK
AT+CMGD=5                     The 5th message does not exist.
ERROR
```

4.12 AT+CSCA – Service Center Address

To set the SMSC address. N58 does not supports this command while in CDMA mode.

Format

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

Parameter

<sca> SC address, string with double quotes.
<tosca> The format of the SMS center number. 129 indicates national number. 145 indicates international number.

A pair of quotation marks is not a must for the SMSC address.

Example

```
AT+CSCA="+8613800755500",145             Set an international SMSC address.
OK
AT+CSCA?
+CSCA: "+8613800755500",145              Query the SMSC address.
OK
```

4.13 AT+CSMP – Setting Text Mode Parameters

To select values for additional parameters needed when SM is sent to the network or placed in a storage when text format message mode is selected. It is possible to set the validity period starting from when the SM is received by the SMSC (<vp> is in range 0... 255) or define the absolute time of the validity period termination (<vp> is a string).

Format

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

Parameter

<fo> depending on the command or result code: first octet of GSM 03.40 SMS-DELIVER SMSSUBMIT (default 17), SMS-STATUS-REPORT, or SMS-COMMAND (default 2) in integer format.

<vp>

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

<pid> TP-Protocol-Identifier in integer format (default 0)

<dcs> Cell Broadcast Data Coding Scheme in integer format (default 0).

The default value is 17,167,0,0.

Example

```
                             Set the text mode parameters to 17,167,0,0.
AT+CSMP=17,167,0,0           No status report; the validity period of the information is 24
OK                           hours; Only messages in text format can be sent.

AT+CSMP?
+CSMP: 17,167,0,0            Query the current settings of the text mode.
OK
```

4.14 AT+CSDH – Showing Text Mode Parameters

To control whether detailed header information is shown in text mode result codes. This command is valid in text mode, which can be set by AT+CMGF=1.

Format

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

Parameter

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

Example

```
AT+CSDH=0
OK
AT+CMGR=0                                                  Set the header information to not
+CMGR: "RECREAD","13510895077",,"15/07/23,20: 58: 28       display.
+32"                                                       Read the 0th message.
abc
OK
AT+CSDH=1
OK
AT+CMGR=0                                                  Set the detailed header information to
+CMGR: "RECREAD","13510895077",,"15/07/23,20: 58: 28       display.
+32",161,36,0,0,"+8613010888500",145,3                     Read the 0th message.
abc
OK
AT+CSDH?
+CSDH: 0                                                   Query the current parameter setting of
OK                                                         the command.
AT+CSDH=?
+CSDH: (0-1)                                               Query the value range of parameter in
OK                                                         the command.
```

5 TCP/UDP Client Commands

5.1 AT+NETAPN – Setting Network APN

To set the network APN.

Format

Type	Command	Response
Set	AT+NETAPN="APN","username","password"<CR>	<CR><LF>OK<CR><LF>
Query	AT+NETAPN?<CR>	<CR><LF>+NETAPN: "APN","username","password" <CR><LF>OK<CR><LF>

Parameter

APN	GPRS network access point
username	GPRS user name
password	GPRS password

Example

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

5.2 AT+XIIC – Setting up a PPP Link

To set up a PPP link.

Format

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

Parameter

<n>	0: Disconnect the PPP link 1: Activate the PPP link.
<state>	0: PPP connection is closed. 1: PPP connection is activated.
<ip>	IP address

Example

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

- Before executing this command, use the AT+CGDCONT command to set the parameters including <APN>.
- Ensure that the module registers the network before using the AT+XIIC=1 command to set up PPP link. Use AT+GREG? to check whether the module registers the network or not. If +CREG: 0,1 or +CREG: 0,5 is returned, the module did not register to the network.

5.3 AT+TCPSETUP – Setting up a TCP Connection

To set up a TCP connection.

Format

Type	Command	Response
Execute	AT+TCPSETUP=<n>,<ip>,<port><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	+TCPSETUP: 0,<result>	

Parameter

<n>	Socket ID, ranging from 0 to 5.
<ip>	destination IP address, in form of xx.xx.xx.xx or www.xxxxxx.com (domain name with or without quotation marks).
<port>	Destination port ID in decimal ASCII code.
<result>	Result code OK FAIL

Example

```
AT+TCPSETUP=0,220.199.66.56,6800
OK                                         Set up a connection to 220.199.66.56,6800 on socket 0.
                                           Successful
+TCPSETUP: 0,OK
AT+TCPSETUP=0,neowayjsr.oicp.net,60010     Set up a connection to neowayjsr.oicp.net,60010 on
OK                                         socket 0.
                                           Set up a connection to neowayjsr.oicp.net, 60010 on
+TCPSETUP: 0,OK                            socket 0.
+TCPCLOSE: 0,Link Closed                   The socket is closed.
AT+TCPSETUP=1,192.168.20.6,7000            Failed to set up a connection to 192.168.20.6,7000 on
OK                                         socket 1. The server may be not started; the IP
                                           address or the port number may be incorrect; the SIM
+TCPSETUP: 1,FAIL                          card fee may be overdue.
AT+TCPSETUP=0,neowayjsr.oicp.net,60010     A TCP/UDP connection has been set up on socket 0.
+TCPSETUP: 0, ERROR1
AT+TCPSETUP=6,192.168.20.6,7000            Parameters are set incorrectly.
+TCPSETUP: ERROR
AT+TCPSETUP=0.58.60.184.213.10012          Parameters are set incorrectly.
+TCPSETUP: ERROR
AT+TCPSET=0,58.60.184.213,10012            The AT command is not complete.
ERROR
```

5.4 AT+TCPSEND – Sending TCP Data

To send TCP data.

This command support data sending in command mode and buffer mode as well as in ASCII and HEX format. The module will return > after this command is sent. Send UDP data 50 ms to 100 ms later.

The mode setting is not saved. Set it when sending data.

- Ensure that a TCP connection has been set up before sending TCP data.
- Run AT+IPSTATUS to check the buffer size before sending data.
- When ASCII data in command mode is required to be sent, length of the <content> parameter must be less than or equal to 1024 bytes.
- To send data containing more than 15 commas, use buffer mode.

Format

Type	Command	Response
Execute	AT+TCPSEND=<n>,<length>[[,<content>][,mode]]<CR>	<CR><LF>><content> <CR><LF>OK<CR><LF> Or <CR><LF>OK<CR><LF> Or <CR><LF>+TCPSEND: ERROR<CR><LF> Or <CR><LF>+TCPSEND: <n>, OPERATION EXPIRED<CR><LF> Or <CR><LF>+TCPSEND: SOCKET ID OPEN FAILED<CR><LF> Or <CR><LF>+TCPSEND: DATA LENGTH ERROR<CR><LF> Or <CR><LF>ERROR<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, unit: byte. 1 to 4096 for ASCII data sent in buffer mode.

1 to 2018 for HEX data sent in buffer mode.
1 to 512 for data sent in command mode (HEX).
1 to 512 for data sent in command mode (ASCII).

<content> data sent in command mode with a length ranging from 0 to 1024.

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

Example

```
AT+TCPSEND=0,1
>
                                     Send 1-byte data on socket 0.
OK
                                     Successful

+TCPSEND: 0,1
AT+TCPSEND=0,1024,,1
                                     Send 1024-byte data in hexadecimal format in buffer mode.
>
OK

                                     Successful
+TCPSEND: 0,1024
AT+TCPSEND=0,6,"123459"
OK                                   Command mode (Only plain text can be sent, not special
                                     symbols.)
+TCPSEND: 0,6
AT+TCPSEND=0,3,”313233”,1
                                     Send data in hexadecimal format in command mode.
OK

                                     Successfully
+TCPSEND: 0,3
AT+TCPSEND=0,10
>                                    No data is input within 30 seconds after > is displayed.
+TCPSEND: 0,OPERATION EXPIRED
AT+TCPSEND=0,1                       One-byte data fails to be sent on socket 0 because the
+TCPSEND: SOCKET ID OPEN FAILED      socket is not opened.
AT+TCPSEND=0,4097                    4097-byte data fails to be sent on socket 0 because data
+TCPSEND: DATA LENGTH ERROR          length exceeds the limit.
```

5.5 AT+RECVMODE – Setting Receive Mode

To set the receive mode of TCP and UDP data. The setting by this command is not saved after the module is powered off.

Do not send this command during communication because it will clear the buffer. This command also works for UDP data.

Format

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

Parameter

<n>	receive mode 0: buffer the TCP or UDP data received and sending command to read the data by MCU is required. 1: print the TCP or UDP data received to UART directly (default).
<mode>	report format 0: ASCII report (default). 1: Hexadecimal

Example

```
AT+RECVMODE=0                                   Set the receive mode to 0.
OK
AT+RECVMODE=1,1                                 Print data and report data in HEX format.
OK
AT+RECVMODE=?                                   Query the value range can be set.
+RECVMODE: (0-1),(0-1)
OK
```

5.6 +TCPRECV – URC Notifying Data Received from Server

To notify TCP data received from the TCP server.

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, end with 0x0d 0x0a; users can determine the end according to the <message_len>.

Example

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

5.7 A+TCPREAD–Reading TCP Data

To read TCP data.

Format

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

Parameter

<n>	Socket ID, ranging from 0 to 5.

<length>	maximum length of data allowed to read, ranging from 1 to 2048.

Example

```
+TCPRECV: 0                        RECVMODE=0
AT+TCPREAD=0,100                   Data received on socket 0.
+TCPREAD: 0,10,1234567890          Read data.
OK                                 The data read is 1234567890.
```

The receive mode is set by the +RECVMODE command.

5.8 AT+TCPCLOSE – Closing the TCP Connection

To close the TCP connection.

Format

Type	Command	Response
Execute	AT+TCPCLOSE=<n><CR>	<CR><LF>+TCPCLOSE: <n>,OK<CR><LF> Or <CR><LF>+TCPCLOSE: ERROR<CR><LF>
URC	+TCPCLOSE:0,Link Closed	

Parameter

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

Example

```
AT+TCPCLOSE=1                    Close the TCP connection on socket 1.
+TCPCLOSE: 1,OK                  Successfully.
AT+TCPCLOSE=2                    Socket number error
+TCPCLOSE: ERROR
+TCPCLOSE: 0,Link Closed         The TCP connection is closed.
                                 The server sends closing command or the network encounters
                                 abnormality or weak signals.
```

5.9 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 running this command.

Format

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

- If the parameter is incorrect, +UDPSETUP: ERROR is prompted.
- If the socket ID is used, for example, socket 0, +UDPSETUP: 0,ERROR1 is prompted.

Parameter

<n>	Socket ID, ranging from 0 to 5.
<ip>	destination IP address, in form of xx.xx.xx.xx or www.xxxxxx.com (domain name)
<port>	Destination port ID in decimal ASCII code.
<result>	Result codes OK FAIL ERROR1: a connection is already set up on the socket.

Example

```
AT+UDPSETUP=1,220.199.66.56,7000
OK                                       Set up a connection to 220.199.66.56,7000
                                         Successfully
+UDPSETUP: 1,OK
AT+UDPSETUP=0,neowayjsr.oicp.net,60010
                                         Set up a connection to neowayjsr.oicp.net,60010 on
OK                                       socket 0
                                         Successfully
+UDPSETUP: 0,OK
AT+UDPSETUP=0,58.60.184.213,11008
                                         A TCP/UDP connection is set up on socket 0.
+UDPSETUP: 0, ERROR1
AT+UDPSETUP=1,192.168.20.6,7000
OK
                                         Failed to set up a connection to 192.168.20.6,7000

+UDPSETUP: 1,FAIL

AT+UDPSETUP=6,192.168.20.6,6800
                                         Socket number error
+UDPSETUP: ERROR
AT+UDPSETUP=0.58.60.184.213.10012
                                         The punctuations in the command are incorrect.
+UDPSETUP: ERROR
```

```
AT+UDPSET=0,58.60.184.213,10012
ERROR                                    The AT command is not complete.
```

5.10 AT+UDPSEND – Sending UDP Data

To send UDP data.

Ensure that the UDP link is set up before sending UDP data. Set it when sending data. In Buffer mode, the module will return > after this command is sent. Send UDP data 50 ms to 100 ms later.

- In buffer mode, at most 2048 bytes can be sent in HEX format and at most 4096 bytes can be sent in ASCII format.
- To decrease the packet loss rate, do not send data more than 1472 bytes each time.
- For how to send quotation marks and backslash in command mode, see the Example.
- The mode parameter can be omitted. Data in ASCII format supports escape mode by default.
- When ASCII data in command mode is required to be sent, length of the <content> parameter must be less than or equal to 102 bytes.

Format

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

Parameter

<n>	Socket ID, ranging from 0 to 5. A UDP connection is established on the socket.
<length>	Length of the data to be sent, unit: byte. 1 to 2018 for HEX data sent in buffer mode. 1 to 4096 for ASCII data sent in buffer mode. 1 to 512 for data sent in command mode (HEX). 1 to 512 for data sent in command mode (ASCII).
<content>	data sent in command mode, ranging from 0 to 1024.

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

<mode> data format
0: ASCII
1: HEX

Example

```
AT+UDPSEND =0,1024,,1
>                                          Send 1024-byte data in hexadecimal format in buffer
OK                                         mode.

+UDPSEND: 0,1024                           Successfully
AT+UDPSEND=0,10,"DEGHHRFRRD",0
                                           In command mode, send data in ASCII mode.
OK
                                           Successfully
+UDPSEND: 0,10
AT+UDPSEND=0,4097                          4097-byte data fails to be sent on socket 0 because
+UDPSEND: DATA LENGTH ERROR                data length exceeds the limit.
AT+UDPSEND=1,6,”313233343536”,1
OK                                         Send data in hexadecimal format in command mode.

+UDPSEND: 0,6                              Successfully
AT+UDPSEND=0,10                            After the data sending command is input and > is
>                                          returned, no more data is entered in 30 seconds. Then
+UDPSEND: 0,OPERATION EXPIRED              the expiration information is displayed.
```

5.11 +UDPRECV – Receiving UDP Data

To indicate that UDP data has been received.

Format

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

Parameter

<n> Socket ID, ranging from 0 to 5.

<length> Length of the data received.

<data> data received. Add 0x0d 0x0a to the end of the data. You can identify the end based on <length>.

Example

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

5.12 AT+UDPREAD – Reading UDP Data

To read UDP data. Executing the +RECVMODE command to select the receive mode is required.

Format

Type	Command	Response
Execute	AT+UDPREAD=<n>[,<length>]<CR>	<CR><LF>+UDPREAD: <n>,<length>,<data> <CR><LF>OK<CR><LF> Or <CR><LF>+UDPREAD:SOCKET ID OPEN FAILED<CR><LF> Or <CR><LF>+UDPREAD: ERROR<CR><LF>

Parameter

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

Example

+UDPRECV: 0	Data received on socket 0.
AT+UDPREAD=0,100	Read data.
+UDPREAD: 0,10,1234567890 OK	The data read is 1234567890.
AT+UDPREAD=1,100 +UDPREAD: SOCKET ID OPEN FAILED	No connection is set up on socket 1.
AT+UDPREAD=0,0 +UDPREAD: ERROR	Parameters are set incorrectly.

5.13 AT+UDPCLOSE – Closing UDP Link

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 5.

Example

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

5.14 AT+IPSTATUS – Querying the Transparent TCP/UDP Socket Status

To query the transparent TCP/UDP socket status.

Due to the characteristic of UDP, this command only queries whether a link has been established by the command, and does not represent the true status of the link.

Format

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

Parameter

<STATUS>	Socket type, value: CONNECT or DISCONNECT.

<CONNECT or DISCONNECT>   Socket type, value: CONNECT or DISCONNECT; CONNECTING or DISCONNECTING

<TCP or UDP>   socket type, value: 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 is set up on socket 0 and the buffer size
+IPSTATUS: 0,CONNECT,TCP,4096        is 4096 bytes.
AT+IPSTATUS=0                        Currently the UDP connection is only established on socket
+IPSTATUS: 0,CONNECT,UDP,0           0.
AT+IPSTATUS=1                        No TCP or UDP connection is set up on socket 1.
+IPSTATUS: 1,DISCONNECT
AT+IPSTATU                           The AT command is not complete.
ERROR
AT+IPSTATUS=6                        The socket number in the command is incorrect.
ERROR
```

Querying of the <send-buffer-size> parameter is not supported on the UDP link.

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

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

Parameter

<n>	Socket ID, ranging from 0 to 5.

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

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

Example

```
AT+TCPACK=0                                20-byte data is transmitted from socket 0 and the
+TCPACK: 0,20,20                           receiver acknowledges 20-byte data.
AT+TCPACK=0                                128-byte data is transmitted from socket 0 and the
+TCPACK: 0,128,120                         receiver 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
```

5.16 AT+DNSSERVER – Setting DNS Server

To set primary and secondary DNS servers.

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

Format

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

Parameter

<n>	DNS server number
1: primary DNS server
2: secondary DNS server

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

Example

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

5.17 AT+PDPKEEPALIVE – Setting PDP Keepalive Heartbeat

To set PDP keepalive heartbeat.

Set up a PPP connection before setting the domain name parameters.

Activate PDP before sending this command.

Format

Type	Command	Response
Set	AT+PDPKEEPALIVE=<onoff>,<inerval><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+PDPKEEPALIVE?<CR>	<CR><LF>+CMEE: <n> <CR><LF>OK<CR><LF>

Parameter

<onoff>	Heartbeat switch 0: disable (default) 1: enable
<inerval>	Heartbeat interval, unit:s, ranging from 1 to 65535.

Example

```
AT+PDPKEEPALIVE?                          Query the heartbeat setting.
+PDPKEEPALIVE: 1,5
OK
AT+PDPKEEPALIVE=1,60                      Enable the heartbeat function; set its interval to
OK                                        60s.
```

5.18 AT+PDPSTATUS – Querying PDP Status

To query the status of PDP.

The status is returned immediately if PDP keepalive heartbeat is enabled.

If PDP keepalive heartbeat is disabled, the status is returned after a delay (200 ms to 10000 ms, depending on network conditions).

Format

Type	Command	Response
Execute	AT+PDPSTATUS<CR>	<CR><LF>+PDPSTATUS: <status><CR><LF>

Parameter

<status> CONNET
DISCONNECT
PSEUDO_CONNECT

Example

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

5.19 AT+TCPKEEPALIVE – Setting TCP Keepalive Heartbeat

To set the TCP keepalive heartbeat.

The settings by this command are not saved after the module is powered off. Execute this command before setting up a TCP connection. It is valid for all connections. DO NOT send it after establishing a TCP connection.

This function consumes data traffic.

Format

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

Parameter

<mode> 0: disable
1: enabled

<time> Interval for which the TCP is idle before the module sends the KEEPALIVE packet to the remote server.
A value between 30 seconds to 7200 seconds is recommended and the default value is 120 seconds.

<interval> Interval for the module to resend the KEEPALIVE packet since it sends last time and
does not receive response. The value ranges from 1s to 1800s, and the default value is 75s.

<keepcount> Count of retransmissions, ranging from 1 to 15, and the default value is 9.

- The <time> for sending heartbeat packets varies with the network environment. Set <time> according to the network environment. If the value of < time> is too large, the terminal may have a false connection, and the <interval> time exceeds <time> will not be resent; if the values of <time> and <interval> are too short, the terminal may disconnect due to the hibernation mechanism of the module air port. If the interval is too short and multiple heartbeat packets are sent during the hibernation period, the heartbeat packets will be sent out together after waking up. The receiving side thinks that the sticky packet data is invalid and does not reply to the acknowledgement message, and if the terminal does not receive the acknowledgement message several times, it thinks that the connection is invalid and disconnects actively.
- Recommended ranges:

	<time>: 120 - 300s <interval>: 40 - 100s

Example

```
AT+TCPKEEPALIVE=1                                Enable the KEEPALIVE function.
OK

AT+TCPKEEPALIVE=1,120,75,9                       Enable and set the KEEPALIVE parameters.
OK
AT+TCPKEEPALIVE=0                                Disable the KEEPALIVE function.
OK
AT+TCPKEEPALIVE?                                 Query the setting of the KEEPALIVE parameters.
+TCPKEEPALIVE: 1,120,75,9
OK
AT+TCPKEEPALIVE=?
+TCPKEEPALIVE: (0-1),(30-7200),(1-1800),(0-15)
OK
```

6 TCP Server Commands

6.1 AT+TCPLISTEN – Setting TCP Listening for 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: <status><CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+TCPLISTEN?<CR>	<CR><LF>+TCPLISTEN: <status><CR><LF>

Parameter

<port>	Port ID
<socket>	Socket ID
<status>	Listening… bind error not listening listening status

Example

```
AT+TCPLISTEN=6800                       Listening port ID: 6800
+TCPLISTEN: 0,OK                        The server starts to listen.
AT+TCPLISTEN=6800                       Listening port ID: 6800
+TCPLISTEN: bind error                  The server fails to bind.
AT+TCPLISTEN=6800
                                        Transparent listening is set already.
+TCPLISTEN: Listening...
AT+TCPLISTEN?                           Query the listening status. Here the server is in the
+TCPLISTEN: listening status            listening status.
AT+TCPLISTEN?                           Query the listening status. Here the server is not in
+TCPLISTEN: not listening               the listening status.
```

```
Connect                                    Receive the connection request from the client.
AcceptSocket=1,ClientAddr=119.123.77.133   AcceptSocket indicates the socket ID on the module,
,ClientPort=8000                           and 119.123.77.133 is the IP address of the client.
```

6.2 AT+CLOSELISTEN – Closing Listening Socket

To close the socket connection.

Format

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

Parameter

<socket_id>   Socket ID

Example

```
+CLOSELISTEN: 0,local link closed    The host closes the socket or network abnormalities occur.
AT+CLOSELISTEN                       The connections to client are closed.
+CLOSELISTEN: 0,local link closed
AT+CLOSELISTEN
+CLOSECLIENT: All remote link
closed                               The connections to client are closed.

+CLOSELISTEN: 0,local link closed
AT+TCPSRVTRANS?
+TCPSRVTRANS: not listening          Failed to close the connection since there is no a
AT+CLOSELISTEN                       listening socket.
ERROR
```

6.3 AT+CLOSECLIENT – Closing Remote Socket

To close remote sockets.

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 ID

Example

```
AT+CLOSECLIENT
+CLOSECLIENT: 1,remote link closed          There is no parameter in this command. Remote
                                            sockets are closed successfully.

+CLOSECLIENT: 2,remote link closed
AT+CLOSECLIENT=1                            The command contains parameters. One remote socket
+CLOSECLIENT: 1,remote link closed          is closed successfully.
AT+CLOSECLIENT=1                            No client connected to socket 1.
ERROR
AT+CLOSECLIENT                              All clients are closed.
+CLOSECLIENT: All remote link closed
```

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

To indicate data received from the client.

Format

Type	Command
URC	+TCPRECV(S): <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(S): 1,10,1234567899      Socket 1 receives 10-byte data in character format from the
                                  client.
```

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

Note that the parameters are different from that of the client mode.

6.5 AT+TCPREADS – Reading TCP Data from the Client

To read TCP data from the client.

Format

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

Parameter

<n> Socket ID, ranging from 0 to 5.
<length> maximum length of data allowed to read, ranging from 1 to 2048.
<content> data read

Example

```
+TCPRECV(S): 1                   RECVMODE=0
AT+TCPREADS=1,100                Socket 1 receives 10-byte data in character format from the
+TCPREADS: 1,10,1234567890       client.
OK
```

6.6 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: Buffer not enough,439<CR><LF> Or <CR><LF>+TCPSENDS: <socket> is not link<CR><LF> Or <CR><LF>+TCPSENDS: <socket>, OPERATION EXPIRED<CR><LF>

Parameter

<socket> 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 4096, unit: byte.

Example

```
AT+TCPSENDS=0,10
>                            Send 10-byte data on socket 0.
OK                           (E.g.: 1234567890).

+TCPSENDS: 0,10
AT+TCPSENDS=0                Send 21-byte data on socket 0.
>                            (E.g.: 012345678901234567890).
OK                           The command ends with Ctrl+Z if no data length is contained. The
                             data length should not exceed 4096 bytes.
+TCPSENDS: 0,21
AT+TCPSENDS=0,5
>                            No data is input within 30 seconds after > is displayed.
+TCPSENDS: 0,OPERATION
EXPIRED
```

6.7 AT+CLIENTSTATUS – Querying Client Connection Status

To query the connection status of the client.

Format

Type	Command	Response
Execute	AT+CLIENTSTATUS=<cid><CR>	<CR><LF>+CLIENTSTATUS: <socket>,<CONNECTor DISCONNECT>,<TCP orINVALID>, <send-buffer-size><CR><LF>

Parameter

<channel> Value of AcceptSocket, that is, the socket of the module. See the description of the AT+TCPLISTEN command.

<CONNECT or DISCONNECT> Socket type, value: CONNECT or DISCONNECT.

<TCP or INVALID> Socket type, value: TCP or INVALID.

<send-buffer-size> The size of the available send buffer on the module, in decimal ASCII mode. Unit: byte.

If the socket is invalid, it may be the listen socket of TCP/UDP client or server.

Example

```
AT+CLIENTSTATUS=0                         A TCP connection to the client has been set up on the
+CLIENTSTATUS: 0,CONNECT,TCP,61440        socket 0 and the buffer size is 61440 bytes.
AT+CLIENTSTATUS=4                         No connection is set up on socket 4.
+CLIENTSTATUS: 4,DISCONNECT
AT+CLIENTSTATUS=1                         Type of the connection on socket 1 is invalid. It is
+CLIENTSTATUS: 1,CONNECT,INVALID          not a TCP connection.
```

6.8 AT+TCPACKS – Querying Status of Data Sent by TCP Server

To query the size of data successfully sent and received over the TCP connection.

Format

Type	Command	Response
Execute	AT+TCPACKS=<socket><CR>	<CR><LF>+TCPACKS: <socket>,<data_sent>,<acked_recv> Or <CR><LF>+TCPACKS: <socket>,<DISCONNECT> Or <CR><LF>ERROR<CR><LF>

Parameter

<socket>	Value of AcceptSocket detected, that is, the socket used by the server to establish a connection with the module. It ranges from 0 to 5.
<data_sent>	Size of data successfully sent to the client.
<acked_recv>	Size of data acknowledged by the client.

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

Example

```
AT+TCPACKS=0                 The module sends 20-byte data to client through socket 0 and the
+TCPACKS: 0,20,20            client acknowledges 20-byte data.
AT+TCPACKS=0                 The module sends 128-byte data to client through socket 0 and the
+TCPACKS:0,128,120           client acknowledges 120-byte data.
AT+TCPACKS=1                 No connection is set up on socket 1.
+TCPACKS: 1,DISCONNECT
AT+TCPACKS=6                 The socket ID is incorrect.
ERROR
```

7 TCP/UDP Transparent Commands

7.1 AT+TCPTRANS – Setting up a Transparent TCP Connection

To set up a transparent TCP connection.

- 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 the command mode and ATO to switch it to the data mode.
- Do not establish non-transparent data services when using the transparent command, since the transparent command conflicts with other non-transparent data services.
- The module exits from the transparent connection if a call or message is incoming.
- At most 2048-byte data can be sent or received per packet in transparent mode.
- To make the settings take effect, <cfgt> and <cfgp> are required to be set simultaneously.
- TCP data can be transparently transmitted after the TCP connection is set up successfully and +TCPTRANS:OK is returned.

Format

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

Parameter

<ip>	destination IP address, in form of xx.xx.xx.xx or www.xxxxxx.com (domain name with or without quotation marks).
<port>	Destination port ID in decimal ASCII code.
<cfgt>	Time to wait per packet sent, ranging from 1 to 65535, 100 by default, unit:ms.

<cfgp>   Threshold value of data packet to be transmitted, ranging from 1 to 2048.
The default value is 2048.

<result>   Result codes
OK
FAIL

Example

```
AT+TCPTRANS=neowayjsr.oicp.net,60010
OK                                         Set up a connection to neowayjsr.oicp.net, 60010.
                                           Successfully.
+TCPTRANS:OK
AT+TCPTRANS=220.199.66.56,6800
OK                                         Set up a connection to neowayjsr.oicp.net, 6800.
                                           Failed, because the port number 6800 is out of
+TCPTRANS:FAIL                             range.
                                           ERROR is returned after the command is executed
AT+TCPTRANS=220.199.66.56,6800             because a transparent (TCP, UDP, TCP server)
ERROR                                      connection is already set up.
AT+TCPTRANS=220.199.66.56,                 The command format is incorrect.
+TCPTRANS:ERROR
+TCPTRANS: Link Closed                     The connection is closed passively.
```

7.2 AT+UDPTRANS – Setting up a Transparent UDP Connection

To set up a transparent UDP connection.

- 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.
- Do not establish non-transparent data services when using the transparent command, since the transparent command conflicts with other non-transparent data services.
- The module exits from the transparent connection if a call or message is incoming.
- At most 2048-byte data can be sent or received per packet in transparent mode.
- To make the settings take effect, <cfgt> and <cfgp> are required to be set simultaneously.
- UDP data can be transparently transmitted after the UDP connection is set up successfully and +UDPTRANS:OK is returned.

Format

Type	Command	Response
Set	AT+UDPTRANS=<ip>,<port>[,<cfgt>,cfgp]<CR>	<CR><LF>OK<CR><LF> <CR><LF>+UDPTRANS:<result><CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<ip> destination IP address, in form of xx.xx.xx.xx or www.xxxxxx.com (domain name with or without quotation marks).

<port> Destination port ID in decimal ASCII code.

<cfgt> Time to wait per packet sent, ranging from 1 to 65535, 100 by default, unit:ms.

<cfgp> Threshold value of data packet to be transmitted, ranging from 1 to 2048. The default value is 2048.

<ip> destination IP address, in form of xx.xx.xx.xx or www.xxxxxx.com (domain name with or without quotation marks).

<result> Result codes
OK
FAIL

Example

```
AT+UDPTRANS=220.199.66.56,6800
OK                                        Set up a transparent UDP connection.

+UDPTRANS: OK                             Successfully.
AT+UDPTRANS=neowayjsr.oicp.net,60010
OK                                        Set up a transparent UDP connection by using domain
                                          name
+UDPTRANS:OK                              Successfully.
AT+UDPTRANS=220.199.66.56,
ERROR                                     The command format is incorrect.

AT+UDPTRANS=220.199.66.56,6800            ERROR is returned after the command is executed
+UDPTRANS: ERROR                          because a transparent (TCP, UDP, TCP server)
                                          connection is already set up.
```

7.3 AT+TCPACK – Querying of Data Transmitted Status

To query the size of data successfully sent and received over the transparent TCP connection.

Format

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

Parameter

<data_send> Size of data successfully sent transparently through this socket, unsigned 64-bit integer in decimal ASCII. Unit: byte

<acked_recv> Size of data successfully acknowledged transparently by the receiver, unsigned 64-bit integer in decimal ASCII. Unit: byte

Example

```
AT+TCPACK                      1024-byte data is transmitted from socket 0 and the receiver
+TCPACK: 1024,1024             acknowledges 1024-byte data.
AT+TCPACK                      No transparent connection is set up.
+TCPACK: DISCONNECT
AT+TCPACK                      Only a transparent UDP connection is established.
+TCPACK: NO TCP LINK
```

7.4 AT+IPSTATUS – Querying the TCP/UDP Socket Status

To query the transparent TCP/UDP socket status.

Format

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

Parameter

<state>	Connection status CONNECT DISCONNECT CONNECTING DISCONNECTING
<type>	Connection type (optional) TCP UDP
<send-buffer-size>	The size of the available send buffer on the module, in decimal ASCII mode. Unit: byte. (Optional)

Example

```
AT+IPSTATU
ERROR                                The AT command is not complete.
AT+IPSTATUS                          The transparent TCP connection is set up, and the
+IPSTATUS:CONNECT,TCP,61440          available buffer size is 61440 bytes.
AT+IPSTATUS                          A transparent UDP connection is established, and the
+IPSTATUS:CONNECT,UDP,61440          available buffer size is 61440 bytes.
AT+IPSTATUS
+IPSTATUS:DISCONNECT                 No transparent connection is set up.
```

7.5 AT+TRANSCLOSE – Closing the Transparent Connection

To close the transparent transmission connection.

Format

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

Parameter

<n> 0: actively close the transparent TCP connection.
1: proactively close the transparent UDP connection.

Example

```
AT+TRANSCLOSE
                                  Actively close the transparent TCP connection.
+TRANSCLOSE: 0,OK
AT+TRANSCLOSE
                                  Fail to set up a transparent TCP/UDP connection.
ERROR

AT+TRANSCLOSE
                                  Proactively close the transparent UDP connection.
+TRANSCLOSE: 0,OK
                                  Successful
+TCPTRANS:Link Closed             The transparent TCP connection is closed passively.
+UDPTRANS:Link Closed             The transparent UDP connection is closed passively.
```

8 TCP Transparent Transmission Server Commands

8.1 AT+TCPSRVTRANS – Setting Listening for Transparent TCP

To set transparent listening for the TCP server.

A connection must be set up between the server and the client through a socket before the server transparently transmits TCP data.

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

Only the SIM cards with fixed IP addresses can be used as servers.

Do not establish non-transparent data services when using the transparent command, since the transparent command conflicts with other non-transparent data services.

The server set up in transparent mode can be connected to only one TCP client (transparent mode or non-transparent mode).

To make the settings take effect, <cfgt> and <cfgp> are required to be set simultaneously.

The module will automatically disconnect from the client if a call or message is incoming.

Format

Type	Command	Response
Set	AT+TCPSRVTRANS=<port>[[,<cfgt>][,cfgp]]<CR>	<CR><LF>+TCPSRVTRANS:<status><CR><LF>
Query	AT+TCPSRVTRANS?<CR>	<CR><LF>+TCPSRVTRANS:<status><CR><LF>

Parameter

<port>	Port ID
<cfgt>	Time to wait per packet sent, ranging from 1 to 65535, 500 by default, unit:ms.
<cfgp>	Threshold value of data packet to be transmitted, ranging from 1 to 2048. The default value is 2048.

<status>	Listening… bind error not listening listening status OK GPRS DISCONNECTION

Example

AT+TCPSRVTRANS=6800 +TCPSRVTRANS: OK	Listening port ID: 6800 The transparent listening of the TCP server is started.
AT+TCPSRVTRANS=6800 +TCPSRVTRANS:bind error	The server fails to bind.
AT+TCPSRVTRANS=6800 +TCPSRVTRANS:Listening...	Transparent listening is set already.
AT+TCPSRVTRANS? +TCPSRVTRANS:listening status	Query the listening status. Here the server is in the listening status.
AT+TCPSRVTRANS? +TCPSRVTRANS:not listening	Query the listening status. Here the server is not in the listening status.
AT+TCPSRVTRANS=5000 +TCPSRVTRANS:GPRS DISCONNECTION	PDP is not activated
Connect AcceptSocket=1,ClientAddr=119.123.77.133,ClientPort=8000 Receive the connection request from the client. AcceptSocket indicates the socket ID on the module, and 119.123.77.133 is the IP address of the client.	

8.2 AT+CLIENTSTATUS – Querying the Status of the Transparent Client Connection

To query the status of the transparent connection to the client.

Format

Type	Command	Response
Execute	AT+CLIENTSTATUS<CR>	<CR><LF>+CLIENTSTATUS: <state>,<type>, <send-buffer-size><CR><LF>

Parameter

<state>	Connection status CONNECT DISCONNECT
<type>	Connection type TCP
<send-buffer-size>	The size of the available send buffer on the module, in decimal ASCII mode. Unit: byte.

Example

AT+CLIENTSTATUS +CLIENTSTATUS: CONNECT,TCP,61440	The transparent TCP connection is set up, and the available buffer size is 61440 bytes.
AT+CLIENTSTATUS +CLIENTSTATUS:DISCONNECT,TCP,61440	The transparent TCP connection is not set up, and the available buffer size is 61440 bytes.

9 FTP Commands

9.1 AT+FTPSCFG – Configuring FTPS Parameter

To configure SSL encryption options

Format

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

Parameter

<type> SSL options
sslversion: SSL version
authmode: authentication mode
ciphersuite: Cipher suite
cacert: CA certificate
clientcert: Client certificate
clientkey: Client key

<type_name> setting of SSL
sslversion:
0: SSL3.0
1: TLS1.0
2: TLS1.1
3: TLS1.2
Authmode:
0: No authentication
1: Manage server authentication

2: Manage server and client authentication if requested by the remote server
Cacert: string type, CA certificate
Clientcert: string type, client certificate
Clientkey: string type, client key

If **authmode** is set to **0**, you do not have to set **cacert**, **clientcert**, or **clientkey**.

Exemple

```
AT+FTPSCFG=”sslversion”,0                  Set the SSL version to ssl3.0.
OK
AT+FTPSCFG=”authmode”,0                    Set no authentication.
OK
AT+FTPSCFG=”cacert”,ca.pem                 Set the name of the CA certificate (the
OK                                         certificate needs to be added in advance.)
AT+FTPSCFG?
+FTPSCFG: 0,1,ca.pem,cc.pem,ck.pem         Query current SSL settings.
OK
AT+FTPSCFG =?
+FTPSCFG: <type>,<type_name>               Query the value range of the parameter.
OK
```

9.2 AT+FTPLOGIN – Logging in to the FTP Server

To log in to the FTP server.

- The FTP functions can be used together with the internal protocol stack TCP/UDP function.
- Before performing the FTP operation, you need to log in to the FTP server.
- The FTP function is in passive mode by default.

Format

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

		Disconnect<CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	+FTPLOGIN:<result>	

Parameter

<ip> IP address of the FTP server.

<port> Port number of the FTP server; generally it is 21.

<user> User name used to log in to the FTP server. Its length cannot be larger than 100 ASCII codes and it cannot contains any comma.

<pwd> Password used to log in to the FTP server. Its length cannot be larger than 100 ASCII codes and it cannot contains a comma.

<ftpmode> FTP mode:
0: PASV (passive mode, default)
1: PORT (active mode)

<result> Result codes
Have Logged In: The user has logged in to the FTP server.
AT Busy: Last FTP AT command has not been executed completely.
User logged in: The user logs in to the FTP server successfully.
530 Not logged in: The user fails to log in to the FTP server because the user account or password is incorrect.
GPRS DISCONNECTION: The user logged in to the FTP server before a PPP link is set up.

Example

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

+FTPLOGIN: User logged in
AT+FTPLOGIN=183.239.240.40,12150,pp,123
OK                                                Fail to log in to the server; the user
                                                  account or password is incorrect.

+FTPLOGIN: 530 Not logged in
AT+FTPLOGIN=58.60.184.213,21,neoway,neoway
OK
                                                  Failed to log in to the FTP server.

+FTPLOGIN: FAIL
+FTP:Server Control Link Disconnect               The FTP control link closes.
+FTP: Server Data Link Disconnect                 The FTP data link closes.
```

9.3 AT+FTPLOGOUT – Logging Out from the FTP Server

To log out from the FTP server.

Format

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

Example

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

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[,lenth]] <CR>	<CR><LF>+FTPGET: Error Not Login<CR><LF> Or <CR><LF>+FTPGET: Error TimeOut<CR><LF> Or <CR><LF>+FTPGET: <length>,<data><CR><LF> Or <CR><LF>+FTPGET: OK.total length is <n><CR><LF> Or <CR><LF>ERROR<CR><LF>

URC	+FTPSTATE: <result>

Parameter

<dir&filename>	Path and name of the file to be read. The file path is relative to the FTP root path.
<type>	File transmission 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
<offset>	Specifies offset of file content.
<lenth>	Length of file downloaded from the start point, ranging from 1 to 8192
<length>	Data length
<data>	Data content
<n>	The module reads data successfully and the data length is n.

Example

```
AT+FTPGET=,1,2
OK

+FTPGET:446,drw-rw-rw-   1 user    group
0 Apr 14 15:55 .
drw-rw-rw-   1 user    group         0
Apr 14 15:55 ..                             Obtain information under the root 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
OK

+FTPGET:65,-rw-rw-rw-   1 user    group
10 Jan 15 15:01 test.txt                    Obtain information of test.txt.
+FTPGET:OK.total length is 65

+FTP:Server Data Link Disconnect
AT+FTPGET=123.txt,1,1
+FTPGET: File Not Found                     The file is inexistent.
AT+FTPPUT=test.txt,1,2,10
>                                           10-byte data is successfully uploaded.
+FTPPUT: OK,10
AT+FTPGET=test.txt,1,1
+FTPGET:10,0123456789                       Read all data.
+FTPGET: OK.total length is 10
```

```
+FTP:Server Data Link Disconnect
AT+FTPGET=test.txt,1,1,2
+FTPGET:8,23456789
+FTPGET:OK.total length is 8          Offset 2 bytes, read all data after the third byte.

+FTP:Server Data Link Disconnect
AT+FTPGET=test.txt,1,1,2,4
+FTPGET:4,2345                        Offset 2 bytes, read 4-byte data after the third
+FTPGET:OK.total length is 4          byte.
```

9.5 AT+FTPPUT – Uploading Data to the FTP Server

To upload data to the FTP Server.

Format

Type	Command	Response
Execute	AT+FTPPUT=<filename>,<type>,<mode>[,<size>]<CR>	In non-transparent mode <CR><LF>+FTPPUT: OK,<size><CR><LF> In transparent mode: <CR><LF>CONNECT <CR><LF>+FTPPUT: OK,<size><CR><LF> Or <CR><LF>+FTPPUT:Error Not Login<CR><LF> Or <CR><LF>+FTPPUT:AT Busy<CR><LF> Or <CR><LF>+FTPPUT:SIZE Error (non-transparent mode) Or <CR><LF>+FTPPUT:OK,<n><CR><LF> Or <CR><LF>+FTPPUT:Delete File OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

- In transparent mode, after data is transmitted completely, you can execute +++ (excluding <CR><LF>) to end the upload process.
- Executing +++ (excluding <CR><LF>) can end the upload process anytime.

Parameter

<filename> Name of the file to be uploaded.
The file path is relative to the FTP root path.

<type> File transmission mode
1: ASCII
2: Binary

<mode> Operating mode
1: STOR mode Create a file in the server and write data into the file; if the file exists already, overwriting the original file.
2: APPE mode Create a file in the server and write data into the file; if the file exists already, write the data at the end of the file.
3: DELE mode Delete a file.

<size> Data length, ranging from 1 to 8192. Transparent mode is used if this parameter is omitted.

<n> Length of the file sent.

- **+FTPPUT: AT Busy**: Last FTP-related AT command has not been executed completely.
- Executing +++ (excluding <CR><LF>) to exit from transparent mode and end the uploading.
- If the file you upload over a connection in transparent mode is large, the port will be occupied all the time that affects the sending and receiving of other commands. So, it is recommended to transmit files in buffer mode; when you need to send large files, use APPE mode to send them in segment.

Example

```
                               Upload the 10-byte test.txt file in ASCII mode and the operation
AT+FTPPUT=test.txt,1,1,10      is
> 1234567890                   in STOR mode.
+FTPPUT:OK,10

                               Upload the 10-byte test.txt file in ASCII mode and the operation
AT+FTPPUT=test.txt,1,2,10      is
> 1234567890                   in APPE mode.
+FTPPUT:OK,10

AT+FTPPUT=test.txt,1,3,0       Delete the test.txt file.
+FTPPUT:Delete File OK
AT+FTPPUT=test.txt,1,1
CONNECT                        Transparent mode, upload the 10-byte test.txt file in ASCII mode
1234567890                     and the operation is in STOR mode.
+FTPPUT:OK,10
AT+FTPPUT=test.txt,1,2
CONNECT                        Transparent mode, upload the 10-byte test.txt file in ASCII mode
1234567890                     and the operation is in APPE mode.
+FTPPUT:OK,10
AT+FTPPUT=test.txt,1,3
                               Transparent mode; delete the test.txt file.
+FTPPUT:Delete File OK
```

9.6 AT+FTPSIZE – Obtaining the FTP File Size

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

Format

Type	Command	Response
Execute	AT+FTPSIZE=<filename><CR>	<CR><LF>+FTPSIZE: <size> <CR><LF>OK<CR><LF> Or <CR><LF>+FTPSIZE: File Not Found<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<filename>	Name of the file to be uploaded.
<size>	The actual file size

Example

```
AT+TPSIZE=test_500k.txt
+FTPSIZE: 512000                          The file size is 512000 bytes.
OK
AT+FTPSIZE=test.txt
+FTPSIZE: File Not Found                  The queried file is inexistent.
AT+FTPSIZE=test_500.txt,100
ERROR                                     The format of the AT command is incorrect
```

9.7 AT+FTPSTATUS – Querying the 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>

Parameter

<status>	0: FTP connection is not set up. 1: FTP connection is set up.
<ip>	IP address of the FTP server.
<port>	Port number of the FTP server.

Example

```
AT+FTPSTATUS                          Query the FTP link status.
+FTPSTATUS:1,119.139.221.66,21        Set up an FTP link, and display the IP address and
                                      port ID of the server.
AT+FTPSTATUS                          Query the FTP link status.
+FTPSTATUS:0,0.0.0.0,21               The FTP connection is not set up yet.
```

9.8 AT+FILEFTPGET–Downloading Files to the File System

To download files to the file system. Offset download is supported.

In the download process, ensure that the file system has sufficient free space. You can run the AT+FSLS command to query the size of the free space.

Before download, run the +FTPLOGIN command to set up an FTP link.

After download, run the +FTPLOGOUT command to close the FTP link.

Format

Type	Command	Response
Execute	AT+FILEFTPGET=<dir&filename>[,<offset>[,<length>]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+FILEFTPGET?<CR>	<CR><LF>OK<CR><LF>
Test	AT+FILEFTPGET=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<dir&filename>	Path and name of the file to be read(Note: The file path is relative to the FTP root path.)
<offset>	Offset of the file content, ranging from 0 to 2097152

<lenth>   Length in bytes of the file content that is read. The value ranges from 1 to 20480. If <offset> and <lenth> are not specified, the entire file is downloaded.

Example

```
AT+FILEFTPGET=text.txt           Download the text.txt file. The command is executed
OK                               successfully, and download starts.

+FILEFTPSTAT:1,1024              Download is completed. The length of the file content is 1024.
AT+FILEFTPGET=text.txt,1,2       Download the text.txt file. The command is executed
OK                               successfully, and download starts.

+FILEFTPSTAT:1,2                 Download is completed. The length of the file content is 2.
AT+FILEFTPGET=text.txt           Download the text.txt file. The command is executed
OK                               successfully, and download starts.
```

9.9 AT+FILEFTPPUT – Uploading Files to the FTP Server

To upload a file to the file system.

Before uploading a file, ensure that the file exists.

You can query the local file list by the AT+FSLIST? command.

"OK" or "ERROR" is prompted within 300 ms after a command execution. But the process of getting upload result is asynchronous and its response time depends on the size of the uploaded file and the network condition.

Format

Type	Command	Response
Execute	AT+FILEFTPPUT=<filename>[,<offset>[,<length>]]<CR>	<CR><CF>OK<CR><CF> <CR><LF>+FILEFTPSTAT:<result>,<len><CR><LF> Or <CR><CF>ERROR<CR><CF>

Parameter

<filename>   Name of the file to be uploaded
<offset>   File offset, ranging from 0 to 2097152.
<length>   Length of the file to be uploaded, ranging from 1 to 8192.

<result>   Upload result
            0: failed
            1: successful
<len>   Length of the uploaded file.

Example

```
AT+FILEFTPPUT="test.txt"
OK                                Upload  the test.txt file to the FTP server.

+FILEFTPSTAT: 1,51000             The file has been uploaded successfully.
AT+FILEFTPPUT="test.bin"
OK                                Failed to upload the test.bin file. 1024-byte data has been
                                  uploaded.
+FILEFTPSTAT: 0,1024
                                  Failed to upload the file since the file does not exist, the
AT+FILEFTPPUT="1111"              parameters are incorrect, or the module does not log in to
ERROR                             the FTP server.

AT+FILEFTPPUT=test.txt,100,100
OK                                The file offset is 100.

+FILEFTPSTAT: 1,100
```

9.10 AT+NWFTPRENAME – Renaming the FTP Server File or Folder

To rename the FTP server file or file folder.

Format

Type	Command	Response
Execute	AT+NWFTPRENAME=<old_name>,<new_name><CR>	<CR><LF>OK<CR><LF> <CR><LF>+NWFTPRENAME: <err>,<protocol_error><CR><LF> Or <CR><LF>ERROR<CR><LF> <CR><LF>+NWFTPRENAME: <err>,<protocol_error><CR><LF>

Parameter

<old_name>   String character type.
              Old name of the file or file folder in the FTP(S) server (directory

supported). 255 bytes at most.

<new_name> New name of the file or file folder of the FTP(S) folder (directory supported). 255 bytes at most.

<err> 0 indicates successful execution
Another value: failed execution
For details, see Appendix D .

<protocol_error> Integer type
Original error codes of the FTP(S) server.
These error codes are defined in FTP(S) protocol; they are only for reference. For details, see Appendix E . 0 indicates invalid.

Example

```
AT+NWFTPRENAME="old_name.txt","new_name.txt"            Modify the FTP file name.
OK

+NWFTPRENAME: 0,200
AT+NWFTPRENAME="test/old_test.txt","test/new_test.txt"
OK                                                       Modify the name of the file under
                                                         the test directory in the FTP
                                                         server.
+NWFTPRENAME: 0,200
AT+NWFTPRENAME="test_old_dir","test_new_dir"            Modify the FTP folder name.
OK

+NWFTPRENAME: 0,200
```

9.11 AT+NWFTPMKDIR – Creating an FTP Server Folder

To create a folder in the FTP server

Format

Type	Command	Response
Execute	AT+NWFTPMKDIR=<folder_name><CR>	<CR><LF>OK<CR><LF> <CR><LF>+NWFTPMKDIR: <err>,<protocol_error><CR><LF> Or <CR><LF>ERROR<CR><LF> <CR><LF>+NWFTPMKDIR: <err>,<protocol_error><CR><LF>

Parameter

<folder_name>	Character string type, name of the folder in the FTP server. 255 bytes at most.

<err>	0 indicates successful execution
Another value: failed execution
For details, see Appendix D .

<protocol_error>	Integer type
Original error codes of the FTP(S) server.
These error codes are defined in FTP(S) protocol; they are only for reference. For details, see Appendix E . 0 indicates invalid.

Example

```
AT+NWFTPMKDIR="test_dir"                    Create a folder in the FTP server
OK

+NWFTPMKDIR: 0,200
```

9.12 AT+NWFTPRMDIR – Deleting an FTP Server Folder

To delete a folder in the FTP server.

Format

Type	Command	Response
Execute	AT+NWFTPRMDIR=<folder_name><CR>	<CR><LF>OK<CR><LF> <CR><LF>+NWFTPRMDIR: <err>,<protocol_error><CR><LF> Or <CR><LF>ERROR<CR><LF> <CR><LF>+NWFTPRMDIR: <err>,<protocol_error><CR><LF>

Parameter

<folder_name>	Character string type, name of the folder in the FTP server. 255 bytes at most.

<err>	0 indicates successful execution
Another value: failed execution
For details, see Appendix D .

<protocol_error>	Integer type
Original error codes of the FTP(S) server.

	These error codes are defined in FTP(S) protocol; they are only for reference. For details, see Appendix E . 0 indicates invalid.

Exemple

```
AT+NWFTPRMDIR="test_dir"                     Delete a folder in the FTP server
OK

+NWFTPRMDIR: 0,200
```

9.13 AT+NWFTPDEL – Deleting an FTP Server File

To delete a file in the FTP server.

Format

Type	Command	Response
Execute	AT+NWFTPDEL=<file_name><CR>	<CR><LF>OK<CR><LF> <CR><LF>+NWFTPDEL: <err>,<protocol_error><CR><LF> Or <CR><LF>ERROR<CR><LF> <CR><LF>+NWFTPDEL: <err>,<protocol_error><CR><LF>

Parameter

<file_name>	Character string type, name of the file in the FTP server. 255 bytes at most.
<err>	0 indicates successful execution Another value: failed execution For details, see Appendix D .
<protocol_error>	Integer type Original error codes of the FTP(S) server. These error codes are defined in FTP(S) protocol; they are only for reference. For details, see Appendix E . 0 indicates invalid.

Exemple

```
AT+NWFTPDEL="test.txt"        Delete an FTP server in the FTP server
OK

+NWFTPDEL: 0,200
```

10 HTTP/HTTPS 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 parameters:
url: destination address
port: destination port ID (no default value)
keepalive: used to set the HTTP long connection.
When para_value=0 (default), the long connection is disabled.
When para_value=1, the long connection is enabled.
recvmode: Receive mode:
para_value=0 indicates receive mode by default; one HTTP response only contains one +HTTPPRECV: header identification; para_value=1 indicates data is displayed in the +HTTPRECV: <length>,<data> format.

<para_value>	The value of the <para>, 2048 bytes at most for url; 443 by default for port.

Example

```
AT+HTTPPARA=url,”www.neoway.com.cn/en/index.aspx”      Setthe Neoway homepage as the URL. The
OK                                                     URL supports domain name translation.

AT+HTTPPARA=url,”121.15.200.97/Service1.asmx/GetNote”  Set URL.
OK
AT+HTTPPARA=url,                                       The AT command format is incorrect.
ERROR
AT+HTTPPARA=port,80                                    Set the destination port to 80.
OK
AT+HTTPPARA=port,8080                                  Set the destination port to 8080.
```

```
OK
```

10.2 AT+HTTPSETUP – Setting up an HTTP Connection

To set up an HTTP connection.

- An HTTP connection is set up successfully only after the destination address and port ID are set correctly.
- Before setting up an HTTP Connection, ensure that a dial-up connection (AT+XIIC=1) is set up successfully.

Format

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

Parameter

N/A.

Example

```
AT+HTTPSETUP                      Set up an HTTP connection.
OK                                Successful
AT+HTTPSETUP                      Set up an HTTP connection.
+HTTPSETUP: FAIL                  Failed
```

10.3 AT+HTTPACTION – HTTP Request

To execute the HTTP request.

Format

Type	Command	Response
Execute	• AT+HTTPACTION=<mode>[	<CR><LF>OK<CR><LF>

	,<length>[,<type>]]<CR> ● AT+HTTPACTION=<mode>[,<offset>,<size>]<CR>	Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+HTTPACTION:SOCKET ID OPEN FAILED<CR><LF> Or <CR><LF>+HTTPSEND: ERROR<CR><LF>

Parameter

<mode> HTTP request mode
0: GET
1: HEAD
2: POST
99: OPEN_MODE, custom packet mode

<length> The length of the POST content or user-defined packet length, maximum value 2048.

<type> POST request data type.
0: x-www-form-urlencoded
1: text
2: json
3: xml
4: html

<offset> Offset in GET mode. Specify the starting location of the download.

<size> Size of file to be downloaded in GET mode.

- Comply with the HTTP protocol when defining packets.
- Add a carriage return to the end of the packets if the HTTP request is set to custom packet mode.

Example

```
                                                            Set the destination address. The
AT+HTTPPARA=url,”www.neoway.com.cn/en/index.aspx”           default port is 80.
OK
AT+HTTPSETUP                                                Set up an HTTP connection.
OK
AT+HTTPACTION=0                                             GET request
OK

+HTTPRECV:                                                  Receive the response from the HTTP
HTTP/1.1 200 OK                                             server.
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: Thu, 28 Nov 2013 03: 06: 57 GMT
Connection: close
Content-Length: 13842

/*neoway homepage, html format, 13842 bytes*/
…….
/* neoway homepage*/

+HTTPCLOSED: HTTP Link Closed

                                                            The server finishes the response
                                                            and disconnects the connection.
                                                            Set the destination address. The
AT+HTTPPARA=url,"www.neoway.com.cn/en/index.aspx"           default port is 80.
OK
AT+HTTPSETUP                                                Set up an HTTP connection.
OK
AT+HTTPACTION=1                                             HEAD request
OK

+HTTPRECV:                                                  The HTTP server responds.
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 URL.
OK
AT+HTTPPARA=port,8080                                       Set the destination port to 8080.
OK
AT+HTTPSETUP                                                Set up an HTTP connection.
OK
AT+HTTPACTION=2,25                                          POST request. Send 23 bytes; enter
>MAC=NEOWAY&DATA=0123456                                    the contents to be uploaded after >
                                                            is displayed.
OK

+HTTPRECV:
HTTP/1.1 200 OK                                             Receive the response from the HTTP
Cache-Control: private, max-age=0                           server.
Content-Type: text/xml; charset=utf-8
Server: Microsoft-IIS/7.5
```

```
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
Date: Thu, 28 Nov 2013 03: 41: 52 GMT
Connection: close
Content-Length: 98

<?xml version="1.0" encoding="utf-8"?>
<string xmlns="http: //wsliu.cn/">NEOWAY+0123456
</string>
                                                            The server replies an XML file
+HTTPCLOSED: HTTP Link Closed                               containing the uploaded content
                                                            NEOWAY and 0123456.

                                                            The server disconnected with the
                                                            module after it finished
                                                            responding.
                                                            Set URL.

                                                            Use the default port 80 to set up
AT+HTTPPARA=url,"www.neoway.com.cn/en/index.aspx"           an HTTP connection.
OK
AT+HTTPSETUP
OK                                                          Use custom packet mode to send 76-
AT+HTTPACTION=99,76                                         byte packets.
>HEAD /en/index.aspx HTTP/1.1
connection: close
HOST: www.neoway.com.cn

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

                                                            The server finishes responding and
                                                            close the connection.
AT+HTTPACTION=0                                             PPP is not enabled or SOC
+HTTPACTION:SOCKET ID OPEN FAILED                           connection encounters an error.
AT+HTTPACTION=0
+HTTPSEND: ERROR                                            Failed to send data.
AT+HTTPACTION=2,adasd
ERROR                                                       Other errors
```

10.4 AT+HTTPCLOSE – Closing the Socket of an HTTP Client

To close the socket of an HTTP client.

Format

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

Parameter

<result>	HTTP Link Closed: HTTP link is closed

- After the +HTTPCLOSE command is sent, the HTTP socket is closed and the setting of +HTTPPARA is cleared.
- Only OK is returned after running this command if the HTTP socket is not connecting.

Example

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

+HTTPCLOSE: HTTP Link Closed
AT+HTTPCLOSE
OK                                                     OK is returned.
```

10.5 +HTTPRECV – Receiving HTTP Data

To indicate data received from the HTTP connection.

Format

Type	Command
URC	<CR><LF>+HTTPRECV: <datas> <CR><LF>+HTTPRECV: <length>,<datas>

Parameter

<length>	Data length.
<datas>	Data received from the HTTP/HTTPS connection.

Example

```
+HTTPRECV: HTTP/1.1 200 OK
Cache-Control: private
Content-Length: 13842
Content-Type: text/html; charset=utf-8
Server: Microsoft-IIS/7.5
Set-Cookie: ASP.NET_SessionId=pvlaai3fizxg44eyvyqsyenk; path=/;
HttpOnly                                                            Receive data from the HTTP
X-AspNet-Version: 2.0.50727                                         connection.
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
+HTTPRECV: 803,HTTP/1.1 206 Partial Content
Cache-Control: no-cache
Connection: Keep-Alive
Content-Length: 10
Content-Range: bytes 0-9/14615
Content-Type: text/html
Date: Tue, 10 Jul 2018 00:55:30 GMT
Etag: "5b3c3650-3917"
Last-Modified: Wed, 04 Jul 2018 02:52:00 GMT
P3p: CP=" OTI DSP COR IVA OUR IND COM "
Pragma: no-cache
Server: BWS/1.1                                                     Format of the received data
Set-Cookie: BAIDUID=F18E6894A34321D8CF9AAF28C14FACC9:FG=1;          when RECVMODE is set to 1.
expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/;
domain=.baidu.com
Set-Cookie: BIDUPSID=F18E6894A34321D8CF9AAF28C14FACC9;
expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/;
domain=.baidu.com
Set-Cookie: PSTM=1531184130; expires=Thu, 31-Dec-37 23:55:55 GMT;
max-age=2147483647; path=/; domain=.baidu.com
Vary: Accept-Encoding
X-Ua-Compatible: IE=Edge,chrome=1

<!DOCTYPE
```

10.6 AT+HTTPGET – Downloading HTTP Files

To download HTTP files.

- This is an asynchronous command. OK is returned immediately after the command is executed. The downloading, decompressing, and verification processes are done in background.
- When <check_type> and <check_value> are ignored, no verification will be performed after the download is complete.
- When setting <dir_mode>, the <check_type> and <check_value> parameters can be omitted. The external flash should be initialized by +NWYSPIFLASH first.
- The downloading, decompressing, and verification results are reported through +HTTPGETSTAT. For details, see its definitions.
- This command is dedicated for a specified customer. To save the module's memory space, executing this command will delete the file downloaded in Flash last time by this command.

Format

Type	Command	Response
Execute	AT+HTTPGET=<type>[,<check_type>,<check_value>[,<dir_mode>]]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<type>	Decompression type 0: unzip (default) 1: zip (unsupported) 2 to 99: reserved.
<check_type>	Type of checking the integrity of compressed passage. 0: MD5 (default) 1 - 99: reserved
<check_value>	Check code, used with <check_type>.
<dir_mode>	Select the storage location 0: local (default) 1: external flash

Example

```
AT+HTTPPARA=url,120.86.64.161/0.2M.txt                  Download the file.
OK
AT+HTTPPARA=port,10141
OK
AT+HTTPSETUP
OK
AT+HTTPGET=0
OK

+HTTPGETRPT: 10

+HTTPCLOSED: HTTP Link Closed
AT+HTTPGET=0,0,eaf84487e190bc79af55c972bbc63e3f
OK                                                      The file is downloaded successfully.

+HTTPGETRPT: 30,303
AT+HTTPGET=0,0,eaf84487e190bc79af55c972bbc63e3f
OK

+APHTTPGETRPT: 31
AT+HTTPGET=0,,,1                                        Verification fails.
OK                                                      Download the file to the external flash,
                                                        no verification.
+HTTPGETRPT: 10
```

10.7 +HTTPGETRPT – URC Notifying Downloading Results

To notify downloading results.

- This command indicates the execution result of AT+HTTPGET.
- Different result codes are reported according different results during downloading, verification, and decompression processes.

Format

Type	Command
URC	<CR><LF>+HTTPGETRPT: <state_type>[,<err_code>]<CR><LF>

Parameter

<state_type> Status type
10: successful download
11: failed download
20: successful decompression
21: failed decompression
30: successful verification
31: failed verification

<err_code> Error response encountered during HTTP GET.
301 Moved Permanently
302 Found
303 See Other
304 Not Modified
305 Use Proxy
307 Temporary Redirect
400 Bad Request
401 Unauthorized
402 Payment Required
403 Forbidden
404 Not Found
405 Method Not Allowed
406 Not Acceptable
407 Proxy Authentication Required
408 Request Timeout
409 Conflict
410 Gone
411 Length Required
412 Precondition Failed
413 Payload Too Large
414 URI Too Long
415 Unsupported Media Type
416 Requested Range Not Satisfiable 417 Expectation Failed
500 Internal Server Error
501 Not Implemented
502 Bad Gateway
503 Service Unavailable
504 Gateway Timeout
505 HTTP Version Not Supported

Example

```
AT+HTTPPARA=url, mybank.icbc.com.cn/icbc/perbank/index.jsp          Download the file.

AT+HTTPGET=0
OK
```

```
+HTTPGETRPT: 10
AT+HTTPGET=1
OK

+HTTPGETRPT: 20

AT+HTTPGET=1,0,eaf84487e190bc79af55c972bbc63e3f
OK
+HTTPGETRPT: 30,303

AT+APHTTPGET=1,0,eaf84487e190bc79af55c972bbc63e3f
OK

+APHTTPGETRPT: 31
```

10.8 AT+HTTPGETSTAT? – Querying the HTTP Downloading Result

To query the HTTP GET process and the downloading result.

Format

Type	Command	Response
Query	AT+HTTPGETSTAT?<CR>	<CR><LF>+HTTPGETSTAT: <state_type>[,<err_code>]<CR><LF>

Parameter

<state_type>   Status type

0: unknown result
10: successful download
11: failed download
20: successful decompression
21: failed decompression
30: successful verification
31: failed verification

<err_code>   Error response encountered during HTTP GET.

301 Moved Permanently
302 Found
303 See Other
304 Not Modified
305 Use Proxy
307 Temporary Redirect

400 Bad Request
401 Unauthorized
402 Payment Required
403 Forbidden
404 Not Found
405 Method Not Allowed
406 Not Acceptable
407 Proxy Authentication Required
408 Request Timeout
409 Conflict
410 Gone
411 Length Required
412 Precondition Failed
413 Payload Too Large
414 URI Too Long
415 Unsupported Media Type
416 Requested Range Not Satisfiable 417 Expectation Failed
500 Internal Server Error
501 Not Implemented
502 Bad Gateway
503 Service Unavailable
504 Gateway Timeout
505 HTTP Version Not Supported

Example

```
AT+HTTPGET=0
OK
AT+HTTPGETSTAT?
+HTTPGETSTAT: 10
OK
AT+APHTTPGET=1
+HTTPGETSTAT: 30
OK
```

10.9 +HTTPCLOSED – URC Notifying the Socket of the HTTP Client is Closed

To notify the socket of an HTTP client is closed.

Format

Type	Command
URC	<CR><LF>+HTTPCLOSED: HTTP Link Closed<CR><LF>

Parameter

N/A.

Example

```
+HTTPCLOSED: HTTP Link Closed                                    The HTTP connection closes.
```

10.10 AT+HTTPSCFG – Configuring SSL Parameters for HTTPS

To configure SSL parameters for HTTPS,

Format

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

Parameter

<type>   SSL parameter options
sslversion: SSL protocol version
authmode: authentication mode
cacert: CA certificate.
clientcert: Client certificate
clientkey: Client key

sni: extended option of TLS Server Name Indication

<type_name> Settings for SSL parameter, the relations between the <type> and <type_name> values are as follows:,
sslversion
0: SSL3.0
1: TLS1.0
2: TLS1.1
3: TLS1.2
authmode
0: No authentication
1: Manage server authentication
2: Manage server and client authentication if requested by the remote server
cacert: string type, CA certificate
clientcert: string type, client certificate
clientkey: string type, client key
sni
0: disable
1: enabled

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

Example

```
AT+HTTPSCFG=”sslversion”,0              Set the SSL version to SSL3.0.
OK
AT+HTTPSCFG =”authmode”,0               Set the authentication mode to no authentication.
OK
AT+HTTPSCFG=”cacert”,ca.pem             Set the name of the CA certificate (adding the
OK                                      certificate in advance is required).
AT+HTTPSCFG?                            Query the current SSL settings.
+HTTPSCFG: 0,1,ca.pem,cc.pem,ck.pem
OK
AT+HTTPSCFG=?                           Query the available parameter value ranges.
+HTTPSCFG: <type>,<type_name>
OK
```

10.11 AT+HTTPSPARA – Setting HTTPS Parameters

To set HTTPS parameters.

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 address
port: destination port ID
keepalive: used to set the long-time connection.
recvmode: Receive mode:
para_value=0 indicates receive mode by default; one HTTP response only contains one +HTTPPRECV: header identification; para_value=1 indicates data is displayed in the +HTTPRECV: <length>,<data> format.

<para_value> Corresponding to the value of <para>, 512 bytes at most for URL; 443 by default for port.URLsupport domain name resolution.

- To send new HTTPS request, set new HTTPS parameters.
- After the +HTTPSCLOSE command is sent, the connection is closed and parameter settings will not be cleared.

Example

```
                                                                  Set the ICBC homepage as the
AT+HTTPSPARA=url,mybank.icbc.com.cn/icbc/perbank/index.jsp        URL. The URL supports domain
OK                                                                name resolution.
AT+HTTPSPARA=url,132.188.73.13/prodreg/beginRegistration.action   Set the destination address
OK                                                                to 132.188.73.13.
AT+HTTPSPARA=port,443                                             Set the destination port to
OK                                                                443.
```

10.12 AT+HTTPSSETUP – Setting up an HTTPS Connection

To set up an HTTPS connection.

- An HTTP connection is set up successfully only after the destination address and port ID are set correctly.
- Before setting up an HTTP Connection, ensure that a dial-up connection (AT+XIIC=1) is set up successfully.

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
+HTTPSSETUP: OK                              The FTP connection is established.
AT+HTTPSSETUP                                Set up an HTTPS connection.
ERROR                                        Successful
```

10.13 AT+HTTPSACTION – Sending an HTTPS Request

To send an HTTPS request.

Format

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

Parameter

<mode> HTTPS request mode
0: GET
1: HEAD
2: POST
99: OPEN_MODE, custom packet mode

<length> The length of the POST content or user-defined packet length, maximum value 2048. This parameter cannot be omitted when <mode> is set to POST or OPEN_MODE.

<type> POST request data type.
0: x-www-form-urlencoded
1: text
2: json
3: xml
4: html

<offset> Offset in GET mode. Specify the starting location of the download.

<size> Size of file to be downloaded in GET mode.

- Comply with the HTTP protocol when defining packets.
- When using user-defined packet, pay attention to whether the tool you are using comes with carriage return and line feed characters.

Example

```
AT+HTTPSPARA=url,support.cdmatech.com/login/                Set the destination path.
OK
AT+HTTPSPARA=port,443                                       Set the destination port to 443.
OK
AT+HTTPSSETUP                                               To set up an HTTPS connection.
OK
AT+HTTPSACTION=0
OK
                                                            GET request
+HTTPSRECV:
HTTP/1.1 200 OK                                             Receive the response from the HTTPS
Server: QUALCOMM                                            server.
X-Powered-By: Servlet/2.5 JSP/2.1
Content-Type: text/html; charset=ISO-8859-1
Date: Sat, 15 Feb 2014 05:58:54 GMT
Content-Length: 7630
Connection: close
Set-Cookie:
JSESSIONID=8V1dS1CpzlPcyNl2LzJZLQgDxWclpMJzP3FHZhVhpGb83G
VM02sn!1955538012; path=/; HttpOnly
/*home page, html format*/
……..                                                      The server finishes the response
/*Homepage content*/                                        and disconnects the connection.
```

```
+HTTPSCLOSED: HTTPS Link Closed
AT+HTTPSPARA=url,support.cdmatech.com/login/
OK
AT+HTTPSPARA=port,443
OK
AT+HTTPSSETUP
OK
AT+HTTPSACTION=1
OK

+HTTPSRECV:
HTTP/1.1 200 OK
Server: QUALCOMM
X-Powered-By: Servlet/2.5 JSP/2.1
Content-Type: text/html; charset=ISO-8859-1
Date: Sat, 15 Feb 2014 6:05:39 AM GMT
Content-Length: 0
Connection: close
Set-Cookie:
JSESSIONID=qyNVS1DSmnjS9cvh72yW1xz1jtjBBRj0yv0zTmMy2LVyBG7HK02b!1955538012; path=/; HttpOnly

+HTTPSCLOSED: HTTPS Link Closed
AT+HTTPSPARA=url,mybank.icbc.com.cn/icbc/perbank/index.jsp
OK
AT+HTTPSPARA=port,443
OK
AT+HTTPSSETUP
OK
AT+HTTPSACTION=99,500
>POST /icbc/perbank/index.jsp HTTP/1.1<CRLF> /*custom
header information*/
Connection: close<CRLF>   /*custom header information*/
Host: mybank.icbc.com.cn<CRLF>   /*custom header
information*/
Content-Length: 10<CRLF>    /*custom header information*/
Content-Type: application/x-www-form-urlencoded<CRLF>
/*custom header information*/
<CRLF><CRLF>
 /*Content to be sent*/
……

+HTTPSRECV:
/*home page, html format*/
…….. 
/*Homepage content*/
+HTTPSCLOSED: HTTPS Link Closed
```

Set the destination path.

Set the destination port to 443.

To set up an HTTPS connection.

HEAD request

HTTPS server responses.

OPEN_MODE, user-defined packet mode. Note that the packet length contains the user-defined header information.

10.14 AT+HTTPSCLOSE – Closing an HTTP Connection Proactively

To actively close an HTTPS connection.

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

Format

Type	Command	Response
Execute	AT+HTTPSCLOSE<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	+HTTPSCLOSE: <state>	

Parameter

<state>	HTTPS Link Closed	Close the HTTPS connection.

Example

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

+HTTPSCLOSE: HTTPS Link Closed
```

10.15 +HTTPSRECV – URC Notifying HTTPS Data Received

To notify data received over the HTTPS connection.

Format

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

Parameter

<datas>	Data received from the HTTP/HTTPS connection.

<length>	Length of the data received.

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;
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
```

Data received over the HTTPS connection.

```
+HTTPSRECV: 832,HTTP/1.1 206 Partial Content
Server: Tengine/2.1.0
Date: Tue, 10 Jul 2018 1:09:25 AM GMT
Content-Type: text/html; charset=utf-8
Content-Length: 10
Connection: keep-alive
x-server-id: 40-5005
request-id: 0bea4b2215311849654971530e6674
Accept-Ranges: bytes
set-cookie: ctoken=MBHI38pHhdL6q0ltGFqjkviz; path=/;
domain=.alipay.com; secure
set-cookie:
ALIPAYJSESSIONID=jMi6e4Q2JmIN8HRk68wm53KXisfnB5H0homeproxy;
path=/; domain=.alipay.com
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
x-content-type-options: nosniff
x-download-options: noopen
strict-transport-security: max-age=31536000
Content-Range: bytes 0-9/21651
x-readtime: 2
Set-Cookie: ssl_upgrade=0;path=/;secure;
Set-Cookie:
spanner=aGuTtGMbvBcOy1dCyZ/e4JI97JSiPcR1Xt2T4qEYgj0=;path=/;
secure;
Via: spanner-internet-g2-35.em14[206]
```

Format of the received data when RECVMODE is set to 1.

10.16 AT+HTTPSGET – Download HTTPS Files

To download HTTPS files.

Format

Type	Command	Response
Execute	AT+HTTPSGET=<type>[,<check_type>,<check_value>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<type>	Decompression type 0: unzip (default) 1: zip (unsupported) 2 - 99: reserved
<check_type>	Type of checking the integrity of compressed passage. 0: MD5 (default) 1 - 99: reserved
<check_value>	Check code, used with <check_type>.

- This is an asynchronous command. OK is returned immediately after the command is executed. The downloading, decompressing, and verification processes are done in background.
- When <check_type> and <check_value> are ignored, no verification will be performed after the download is complete.
- When setting <dir_mode>, the <check_type> and <check_value> parameters can be omitted. The external flash should be initialized by +NWYSPIFLASH first.
- The downloading, decompressing, and verification results are reported through +HTTPGETSTAT. For details, see its definitions.

Example

```
AT+HTTPSPARA=url, mybank.icbc.com.cn/icbc/perbank/index.jsp        Download the file.
OK
AT+HTTPSGET=0
OK
+HTTPSGETRPT: 10
AT+HTTPSGET=0,0,eaf84487e190bc79af55c972bbc63e3f
OK
+HTTPSGETRPT: 30,303

AT+HTTPSGET=0,0,eaf84487e190bc79af55c972bbc63e3f
OK
+APHTTPSGETRPT: 31
AT+HTTPSGET=0,,,1                                                  Download the file to the
OK                                                                 external flash. No
                                                                   verification.
+HTTPSGETRPT: 10
```

10.17 +HTTPSGETRPT – URC Notifying Downloading Result

To notify HTTPS downloading results in GET method.

- This command indicates the execution result of AT+HTTPGET.
- Different result codes are reported according different results during downloading, verification, and decompression processes.

Format

Type	Command
URC	<CR><LF>+HTTPSGETRPT: <state_type>[,<err_code>]<CR><LF>

Parameter

<state_type> Status type
10: successful download
11: failed download
20: successful decompression
21: failed decompression
30: successful verification
31: failed verification

<err_code> Error response encountered during HTTP GET.
301 Moved Permanently
302 Found
303 See Other
304 Not Modified
305 Use Proxy
307 Temporary Redirect
400 Bad Request
401 Unauthorized
402 Payment Required
403 Forbidden
404 Not Found
405 Method Not Allowed
406 Not Acceptable

407 Proxy Authentication Required
408 Request Timeout
409 Conflict
410 Gone
411 Length Required
412 Precondition Failed
413 Payload Too Large
414 URI Too Long
415 Unsupported Media Type
416 Requested Range Not Satisfiable
417 Expectation Failed
500 Internal Server Error
501 Not Implemented
502 Bad Gateway
503 Service Unavailable
504 Gateway Timeout
505 HTTP Version Not Supported

Example

```
AT+HTTPSPARA=url, mybank.icbc.com.cn/icbc/perbank/index.jsp        Download the file.
OK

AT+HTTPSGET=0
OK

+APHTTPSGETRPT: 10

AT+HTTPSGET=1
OK
+HTTPSGETRPT: 20

AT+HTTPSGET=1,0,eaf84487e190bc79af55c972bbc63e3f
OK
+HTTPSGETRPT: 30,303

AT+HTTPSGET=1,0,eaf84487e190bc79af55c972bbc63e3f
OK
+HTTPSGETRPT: 31
```

10.18 AT+HTTPSGETSTAT? – Querying the Download Result

To query the HTTPS GET process and the download result.

Format

Type	Command	Response
Execute	AT+HTTPSGETSTAT?<CR>	<CR><LF>+HTTPSGETSTAT: <state_type>[,<err_code>]<CR><LF>

Parameter

<state_type>	Status type 0: unknown result 10: successful download 11: failed download 20: successful decompression 21: failed decompression 30: successful verification 31: failed verification
<err_code>	Error response encountered during HTTP GET. 301 Moved Permanently 302 Found 303 See Other 304 Not Modified 305 Use Proxy 307 Temporary Redirect 400 Bad Request 401 Unauthorized 402 Payment Required 403 Forbidden 404 Not Found 405 Method Not Allowed 406 Not Acceptable 407 Proxy Authentication Required 408 Request Timeout 409 Conflict 410 Gone 411 Length Required 412 Precondition Failed 413 Payload Too Large 414 URI Too Long 415 Unsupported Media Type 416 Requested Range Not Satisfiable 417 Expectation Failed 500 Internal Server Error 501 Not Implemented 502 Bad Gateway 503 Service Unavailable 504 Gateway Timeout

505 HTTP Version Not Supported

Example

```
AT+HTTPSGET=0
OK

AT+HTTPSGETSTAT?
+HTTPSGETSTAT: 10,303
OK

AT+APHTTPSGET=1
OK

+HTTPSGETSTAT: 30,303
OK
```

10.19 AT+FILEHTTPACTION – HTTP Request in File System

HTTP Request in File System

- Establish an HTTP connection before executing this command.
- Before downloading files through HTTP GET, ensure that there are enough remaining space in the file system.

Format

Type	Command	Response
Execute	AT+FILEHTTPACTION=<mode>,<length>,<type>,<dir&filename><CR> AT+FILEHTTPACTION=<mode>,<offset>,<size>[,dir&filename]<CR>	<CR><LF>OK<CR><LF> Or AT+CMEE=0 <CR><LF>ERROR<CR><LF> Or AT+CMEE=1 <CR><LF>+CME ERROR:<errcode><CR><LF> Or <CR><LF>+CME ERROR:<errtext><CR><LF>

Parameter

<mode> HTTP request mode
0: GET
1: POST

<length> Length of the POST content, supports 524288 bytes at most.

<type> POST request data type.
0: x-www-form-urlencoded
1: text
2: json
3: xml
4: html

<offset> Offset in GET mode. Specify the starting location of the download.

<size> Length of file downloaded in GET method. Generally, the maximum value of <size> is 524288. When an external flash is connected, the value of <size> supports 2166720 bytes at most.

<dir&filename> Path and name of the file needed.
When mode=0, you can specify the name of the file saved locally.

<errcode> The corresponding values of A and B are as follows:
49 -- The Execute Command Not Support
51 -- No Memory
53 -- Parameters are Invalid
66 -- File too Large
300 -- Netif is Error
301 -- HTTPACTION is Needed First
303 -- HTTPPARA CID Invalid
303 -- HTTPPARA CID Invalid
1001 -- PDP Not Active

<errtext> See the description in <errcode>.

Example

```
AT+FILEHTTPACTION=0,0,524288           Read 512 KB data starting from the first byte.
OK
                                       The 512 KB data is downloaded successfully.
+FILEHTTPSTAT: 0,1,524288
AT+FILEHTTPACTION=1,524288,0,text.txt  The file with 524288 data length
OK
                                       is posted successfully.
+FILEHTTPSTAT: 1,1,524288
AT+FILEHTTPACTION=0,0,524288           Failed command execution. 1001 indicates PDP not
+CME ERROR: 1001                       active.
```

10.20 AT+FILEHTTPACTION – HTTPS Request in File System

HTTPS Request in File System

- Establish an HTTP connection before executing this command.
- Before downloading files through HTTP GET, ensure that there are enough remaining space in the file system.

Format

Type	Command	Response
Execute	AT+FILEHTTPSACTION=<mode>,<length>,<type>,<dir&filename><CR> AT+FILEHTTPSACTION=<mode>,<offset>,<size><CR>	<CR><LF>OK<CR><LF> Or AT+CMEE=0 <CR><LF>ERROR<CR><LF> Or AT+CMEE=1 <CR><LF>+CMEERROR:<errcode><CR><LF> Or AT+CMEE=2 <CR><LF>+CME ERROR:<errtext><CR><LF>

Parameter

<mode> HTTP request mode
0: GET
1: POST

<length> Length of the POST content, supports 524288 bytes at most.

<type> POST request data type.
0: x-www-form-urlencoded
1: text
2: json
3: xml
4: html

<offset> Offset in GET mode. Specify the starting location of the download.

<size> Size of file to be downloaded in GET mode. The value supports 524288 bytes at most.

<dir&filename> Path and name of the file needed. The file path is relative to the root path of the
<errcode> file system.

The corresponding values of A and B are as follows:
49 -- The Execute Command Not Support
51 -- No Memory
53 -- Parameters are Invalid
66 -- File too Large
300 -- Netif is Error
301 -- HTTPACTION is Needed First
303 -- HTTPPARA CID Invalid
1001 -- PDP Not Active

<errtext> See the description in <errcode>.

Example

```
AT+FILEHTTPSACTION=0,0,524288              Read 512 KB data starting from the first byte.
OK
                                           The 512 KB data is downloaded successfully.
+FILEHTTPSTAT: 0,1,524288
AT+FILEHTTPSACTION=1,524288,0,text.txt     The file with 524288 data length
OK
                                           is posted successfully.
+FILEHTTPSTAT: 1,1,524288
AT+FILEHTTPSACTION=0,0,524288              Failed command execution. 1001 indicates PDP not
+CME ERROR: 1001                           active.
```

10.21 +FILEHTTPSTAT – URC Notifying the HTTP(S) Uploading/Downloading Result

To notify the HTTP(S) uploading/downloading result.

Format

Type	Command
URC	<CR><LF>+FILEHTTPSTAT: <mode>,<stat>[,<length>]<CR><LF> <CR><LF>+FILEHTTPSTAT: <stat>,<errcode><CR><LF>

Parameter

<mode> HTTP(S) request type
0: HTTP(S) GET
1: HTTP(S) POST

<stat> Downloading/uploading result code
0: Downloading/uploading failed

	1: Downloading/uploading successful
<length>	Length of file downloaded/uploaded, unit: byte.

Example

```
AT+FILEHTTPSACTION=0,0,524288          Read 512 KB data starting from the first byte.
OK
                                       The 512 KB data is downloaded successfully.
+FILEHTTPSTAT: 0,1,524288


AT+FILEHTTPSACTION=1,524288,0,text.txt The file with 524288 data length
OK
                                       is posted successfully.
+FILEHTTPSTAT: 1,1,524288
```

10.22 +HTTPSCLOSED – URC Notifying HTTPS Link Closed

To notify the HTTPS link is closed.

Format

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

Parameter

N/A.

Example

```
+HTTPSCLOSED: HTTPS Link Closed                 The HTTPS connection is disconnected.
```

11 Call Control Commands

11.1 ATD – Dialing Command

To initialize a voice call; the dialing string consists of numbers and modifiers and must end with a semicolon.

voice call: ATD<dial string>;

Format

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

Parameter

<dial string>	Phone number
<n>	Phonebook entry index number
name	name in current phonebook.
<result>	CONNECT: the callee answers the call. NO CARRIER: the callee hangs up the call or the call times out.

Example

```
ATD10010;                       Make a call.
OK

CONNECT                         The callee answers.
ATD>4;                          Use the Phonebook entry index number to make a call.
OK

CONNECT
ATH
OK
ATD>"Comneon";                  Use the name in the phonebook to make a call. The callee answers
OK                              and then hangs up the call.

CONNECT
```

```
NO CARRIER
```

11.2 ATA – Call Answering

To answer the call and establish a call connection.

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

Format

Type	Command	Response
Execute	ATA<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>BUSY<CR><LF>

Parameter

N/A.

Example

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

11.3 ATH – Hanging up Calls

To hang up all calls.

Format

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

Parameter

N/A.

Example

```
ATH                                              End the call connection.
OK
ATH                                              Reject the incoming call (hang up the call).
OK
```

11.4 AT+CLIP – Caller ID

To enable or disable caller ID.

Format

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

Parameter

<n>	0: disable caller ID (default value)
	1: enable caller ID
<m>	0: CLIP not provisioned
	1: CLIP provisioned
	2: unknown (e.g. no network, etc.)
<tosca>	The format of the SMS center number.

Example

```
AT+CLIP=1                                        Enable the caller ID function.
OK
RING                                             An incoming call from 136********.

+CLIP: "136********",161,"",0,"",0
AT+CLIP?                                         Query the setting of the caller ID.
+CLIP: 1,1                                        (default)
OK
```

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

11.5 ATS0 – Auto-Answer

To control the auto-answer mode of the module.

If **ATS0=0**, the auto-answer function is not enabled; otherwise, the module will automatically answer the call after ringing for certain times.

Format

Type	Command	Response
Set	ATS0=<value><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	ATS0?<CR>	<CR><LF><value> <CR><LF>OK<CR><LF>

Parameter

<value>	Integer type, ranging from 0 to 255. The default value is 000.

Example

```
ATS0=1                                    Set the auto-answer for one ring.
OK
ATS0?                                     Query the status of the auto-answer function.
001
OK
```

11.6 AT+CLVL – Setting the Voice Volume

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

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

Format

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

Parameter

<level> Integer type, ranging from 0 to 100; the smaller the parameter value, the smaller the level. The default value is 60.

Example

```
AT+CLVL=4                           Set the level of the voice volume to 4.
OK
AT+CLVL?                            Query the level of voice volume of the module.
+CLVL: 4
OK
AT+CLVL=?                           Query the valid voice volume level for the module.
+CLVL: (0-100)
OK
```

11.7 AT+CMUT – Mute Control

To set mute control of the voice calls.

The setting is only valid during a call.

Format

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

Test	AT+CMUT=?<CR>	<CR><LF>+CMUT: (value range of <n>),(value range of <m>) <CR><LF>OK<CR><LF>

Parameter

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

Example

```
AT+CMUT=0,0                                 Disable the mute mode.
OK
AT+CMUT=1                                   Enable mute control before a call.
OK
AT+CMUT?                                    Query whether the mute mode is enabled.
+CMUT: 0,0
OK
AT+CMUT=?                                   Query the value range of mute mode function.
+CMUT: (0-1)
OK
```

11.8 AT+CLCC – Querying Current Calls

To query current calls and their status.

Format

Type	Command	Response
Execute	AT+CLCC<CR>	<CR><LF>[+CLCC: <idx>,<dir>,<stat>,<mode>,<mpty>,[<number>,<type>,[<alpha>]]] <CR><LF>[+CLCC: <idx>,<dir>,<stat>,<mode>,<mpty>,[<number>,<type>,[<alpha>]]] <CR><LF>[...]]] <CR><LF>OK<CR><LF>

Parameter

<idx> Caller ID descripted in section 4.5.5.1 of 3GPP TS 22.030 [19], integer, this digit can be used in the +CHLD command.

<dir>	0: mobile originated calls 1: mobile terminated calls
<stat>	Call status 0: active 1: held 2: dialing 3: alerting 4: incoming 5: waiting
<mode>	Call types 0: voice 1: data 2: fax
<mpty>	Multiparty calls 0: Non-multiparty calls 1: one of the multiparty calls
<number>	phone number
<type>	Number type 145 international numbering schemes (contains the character "+") 129 national numbering schemes
<alpha>	Phonebook number entry; the character format is set based on the setting in +CSCS.

Example

```
AT+CLCC                                           Incoming call
+CLCC: 1,1,4,0,0,"13596722590",129
OK
AT+CLCC                                           Initiate a call
+CLCC: 1,0,2,0,0,"13596722590",129
OK
```

11.9 AT+SETVOLTE – Setting VoLTE Switch

To set the VoLTE switch.

Format

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

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

Parameter

<onoff>	0: Disable VoLTE 1: Enable VoLTE (default)

Example

```
AT+SETVOLTE=0                                   Disable VoLTE.
OK
AT+SETVOLTE=1                                   Enable VoLTE.
OK
AT+SETVOLTE?                                    Query whether VoLTE is enabled.
+SETVOLTE: 1
OK
```

12 Wi-Fi Function

12.1 AT+WIFIAPSCAN – Wi-Fi Hotspot Scanning

To scan for 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>	Physical address
<rssi>	signal strength.
<channel>	Channel number

Example

```
AT+WIFIAPSCAN                                  Start to scan hotspot and output all the scanning
+WIFIAPSCAN: ec6c9f4be889,-93,1                results.
+WIFIAPSCAN: ec6c9f4be880,-99,1
+WIFIAPSCAN: ec6c9f4be87a,-96,1
OK
```

12.2 AT+WIFIGSMLOC – Wi-Fi Positioning

To perform Wi-Fi positioning function.

Before positioning, you need to scan Wi-Fi hot spot and then make a dial-up connection.

Format

Type	Command	Response
Execute	AT+WIFIGSMLOC =<n><CR>	<CR><LF>FIGSMLOC: <fail_string><CR><LF> Or <CR><LF>+WIFIGSMLOC: {<result_string>} <CR><LF>+WIFIGSMLOC: OK<CR><LF> Or <CR><LF><code> <CR><LF>+WIFIGSMLOC: FAIL<CR><LF> Or <CR><LF>OK <CR><LF>+WIFIGSMLOC: TIMEOUT<CR><LF>

Parameter

<fail_string>	GPRS DISCONNECTION ERROR LINK NOT FREE
<result_string>	String containing latitude and longitude
<code>	401: Unauthorized 400: Bad Request 404: Not Found 408: Request Timeout 500: Server Error

Example

```
AT+WIFIGSMLOC=1+WIFIGSMLOC:                                   Positioning successfully.
{"location":{"lat":34.2060764,"lng":108.8360664},"accuracy":50.0}
+WIFIGSMLOC: OK
```

13 SSL TCP Data Service

13.1 AT+SSLTCPCFG – Configuring SSL Parameters for TCP

To configure SSL encryption options.

Format

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

Parameter

<type> SSL parameter options
sslversion: SSL protocol version
authmode: authentication mode
ciphersuite: Cupher cuite
cacert: CA certificate
clientcert: Client certificate
clientkey: Client key

<type_name> Settings for SSL parameter, the relations between the <type> and <type_name> values are as follows:,

- sslversion
  0: SSL3.0
  1: TLS1.0
  2: TLS1.1
  3: TLS1.2

- authmode
  0: no authentication, indicating
  +
   you do not have to set other parameters, such as cacert, clientcert, and clientkey.
  1: Manage server authentication
  2: Manage server and client authentication if requested by the remote server
- ciphersuite: reserved.
- cacert: string, CA certificate
- clientcert: string, client certificate
- clientkeyClientkey: string, client key

Example

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

13.2 AT+SSLTCPSETUP – Setting up a TCP Connection over SSL

To set up a TCP connection over SSL.

Do not establish non-transparent data services when using the transparent command, since the transparent command conflicts with other non-transparent data services.

Format

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

		<CR><LF>+SSLTCPSETUP: GPRS DISCONNECTION
Query	AT+SSLTCPSETUP?	<CR><LF>+SSLTCPSETUP: <socket_id>,<ip>,<port>,<mode> [<CR><LF>+SSLTCPSETUP: <socket_id>,<ip>,<port>,<mode>]… <CR><LF>OK<CR><LF>
Test	AT+SSLTCPSETUP=?	<CR><LF>+SSLTCPSETUP:<socket_id>,<ip>,<port>,<mode><CR><LF>

Parameter

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

Example

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

13.3 AT+SSLTCPCLOSE – Closing TCP Connection over SSL

To close a TCP connection over SSL.

Format

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

Parameter

<n>	socket ID, ranging from 0 to 5.
<result>	OK ERROR Link Closed

Example

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

13.4 AT+SSLTCPSEND – Sending TCP Data over SSL

To send TCP data over SSL.

Format

Type	Command	Response
Execute	AT+SSLTCPSEND=<socket_id>,<data_length>	<CR><LF>> <CR><LF>+SSLTCPSEND: <socket_id>,<result>

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

Parameter

<socket>	ranging from 0 to 5, used to identify the connection to the server.
<data_length>	data length, ranging from 1 to 4096.
<result>	OK FAIL

Example

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

13.5 +SSLTCPRECV – URC Notifying SSLTCP Data Received

To notify SSLTCP data is received.

Format

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

Parameter

<socket_id> ranging from 0 to 5, used to identify the connection to the server. This value shall be the same with the socket value set in the SSLTCPSETUP command.

<data_length> Length of the data received.
<data> data received.

Example

```
+SSLTCPRECV: 1,20,12345678901234567890     Receive 20-byte data over socket 0.
```

13.6 AT+NWCERTEENABLE–Enable Encryption of the Certificate

To enable encryption of the certificate. Only after the AWS certificate encryption is enabled, can the certificate be successfully encrypted and added to the module by executing the AT+CERTADD command (type=1, current only AWS MQTT certificate is supported).

Format

Type	Command	Response
Execute	AT+NWCERTEENABLE=<type>,<enable><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<type> 1 AWS certificate
  2 TCPS certificate
  3 HTTPS certificate
<enable> 0 unencrypted
  1 encrypted

Example

```
AT+NWCERTEENABLE=1,1
                                                Enable AWS certificate encryption.
OK
```

13.7 AT+CERTADD – Adding SSL Certificate

To add an SSL certificate to the module.

The writing process can be interrupted by +++.

Currently, only the AWS MQTT certificate can be encrypted. That is, to add an SSL certificate you must execute AT+NWCERTEENABLE=1,1 first and then execute the AT+CERTADD (<type> =1) command.

Format

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

Parameter

<file_name>	Name of the certificate written to the module
<length>	Certificate file name
<type>	1: AWS certificate 2: TCPS certificate 3: HTTPS certificate

Example

```
AT+CERTADD=ca_cert.pem,1428
CONNECT                            Write the 1428-byte ca_cert.pem certificate to the module.
+CERTADD: 1428,OK
AT+CERTADD=client_cert.pem,1938
CONNECT                            Write a 1938-byte client_cert.pem certificate to the module.
+CERTADD: 1938,OK
AT+CERTADD=client_key.pem,1097
CONNECT                            Write a 1097-byte client_cert.pem certificate to the module.
+CERTADD: 1097,OK
```

13.8 AT+CERTCHECK – Checking the SSL Certificate

To check the SSL certificate.

Format

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

Parameter

<file_name>   Certificate file name to be confirmed.

Example

```
AT+CERTCHECK=ca_cert.pem                  Check the ca_cert.pem certificate.
+CERTCHECK: ca_cert.pem,OK
AT+CERTCHECK=client_cert.pem              Check the client_cert.pem certificate.
+CERTCHECK: client_cert.pem,OK
AT+CERTCHECK=client_key.pem               The client_key.pem certificate does not exist.
+CERTCHECK: ERROR
AT+CERTCHECKT?
cacert.pem                                Query the added file.
keycert.pem
OK
```

13.9 AT+CERTDEL – Deleting an SSL Certificate

To delete an SSL certificate.

Format

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

Parameter

<file_name>   Certificate file name to be deleted.

Example

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

13.10 AT+SSLTCPCFGA – Configuring SSL Parameters for TCP

To configure SSL encryption options.

Before configuring the SSL encryption options, you need to import the certificate in advance. Use the AT+CERTADD command to import the certificate. The certificate can be set to null.

Format

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

Parameter

<sslversion> SSL protocol version
0: SSL3.0
1: TLS1.0
2: TLS1.1
3: TLS1.2

<authmode> authentication mode
0: No authentication

1: Require authentication server
2: Two-way authentication

<cacert> CA certificate.
<clientcert> Client certificate
<clientkey> Client key

Example

```
AT+SSLTCPCFGA=3,1,"ca.pem","",""                 Set TLS1.2.
OK                                               Verifying the server is required.
                                                 Set the CA certificate to ca.pem.
                                                 Other certificates are null.

AT+SSLTCPCFGA?                                   Query the current SSL settings.
+SSLTCPCFGA: 0,1,ca.pem,cc.pem,ck.pem
OK
```

13.11 AT+SSLTCPREAD – Reading SSL TCP Data

To read SSL TCP data.

Format

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

Parameter

<n> Socket ID, ranging from 0 to 5.
<length> Length of data read, ranging from 1 to 2048.
<len> The length of data read.
<data> data read

Example

```
AT+SSLTCPSETUP=0,58.60.184.213,12004,0
OK
+SSLTCPSETUP: 0,OK                         Data received on socket 0.
AT+SSLTCPSEND=0,10                         Read data.
>                                          The data read is 1234567890.
+SSLTCPSEND: 0,OK

+SSLTCPRECV: 0
```

```
AT+SSLTCPREAD=0,2048
+SSLTCPREAD: 0,10,1111111111
OK
```

13.12 AT+SSLCIPHERSET – Removing Weak Algorithm from the SSLTCP Connection

To remove the weak algorithm from the SSL TCP connection.

Format

Type	Command	Response
Set	AT+SSLCIPHERSET=<enable><CR><LF>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+SSLCIPHERSET?<CR>	<CR><LF>+SSLTCPCFGA: <enable><CR><LF>OK<CR><LF>
Test	AT+SSLCIPHERSET=?<CR>	<CR><LF>OK<CR><LF>

Parameter

<enable> Whether to remove the weak algorithm. Integer type, ranging from 0 to 1.
0: disable (default)
1: enable

Example

```
AT+SSLCIPHERSET=1
                            Remove the weak algorithm from the SSL TCP connection successfully.
OK
AT+SSLCIPHERSET?
+SSLCIPHERSET: 1            Query the current setting.
OK
```

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

14 MQTT Commands

14.1 AT+MQTTMUX－Setting MQTT Multiplexing

To set MQTT multiplexing.

Format

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

Parameter

<n>	0: Single-channel MQTT (default)
	1: MQTT multiplexing

Example

```
AT+MQTTMUX=1                                                  Set MQTT multiplexing
OK
```

- After MQTT multiplexing is set, other MQTT commands should add <sid> the first parameter. <sid> ranges from 0 to 5.

  For example:

- Set single-channel MQTT (AT+MQTTMUX=0)

  AT+MQTTCONNPARAM=”1”,"neoway","password" (not sid paranmeter)

- Set MQTT multiplexing (AT+MQTTMUX=1)

  AT+MQTTCONNPARAM=0,"1","neoway","password"（contains an sid parameter).

14.2 AT+MQTTTLS – Configuring TLS Parameters

To configure MQTT TLS parameters.

Add the certificate through the AT+CERTADD command.

Format

Type	Command	Response
Set	• MQTTMUX=0: AT+MQTTTLS=<type>,<type_name><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
	• MQTTMUX=1 AT+MQTTTLS=<sid><type>,<type_name><CR>	
Query	AT+MQTTTLS?<CR>	• MQTTMUX=0: <CR><LF>+MQTTTLS:<sslmode>,<authmode>,<rootca_name>,<clientcert_name>,<clientkey_name> <CR><LF>OK<CR><LF>
		• MQTTMUX=1: <CR><LF>+MQTTTLS:<sid>,<sslmode>,<authmode>,<rootca_name>,<clientcert_name>,<clientkey_name>
Test	AT+MQTTTLS=?<CR>	• MQTTMUX=0: <CR><LF>+MQTTTLS: <type>,<value> <CR><LF>OK<CR><LF>
		• MQTTMUX=1: <CR><LF>+MQTTTLS: <sid>,<type>,<value> <CR><LF>OK<CR><LF>

Parameter

<sid> Socket ID, ranging from 0 to 5.

<type> Parameter type.
sslmode: Whether to enable authentication mode.
authmode: authentication mode
rootca: CA certificate.
clientcert: Client certificate
clientkey: client key
sslversion: SSL protocol version

<type_name> Corresponding parameter value

The relations between the <type> and <type_name> values are as follows:,

- sslmode
  0: not authentication
  1: authentication
- authmode
  0: verify optional
  1: verify required
  Note: this parameter is valid only when sslmode=1.
- rootca: string: CA certificate.
- clientcert: string, name of the client certificate
- clientkey: clientkey: string, name of the client key
- sslversion: The default value is 3.
  0: SSL3.0
  1: TLS1.0
  2: TLS1.1
  3: TLS1.2

Example

```
AT+MQTTTLS=authmode,1                                 Set the authentication mode to verify
OK                                                    required.
AT+MQTTTLS?                                           Query the current SSL settings.
+MQTTTLS: 1,1,ca.pem,cc.pem,ck.pem,3
OK
AT+MQTTTLS=?                                          Query the available parameter value ranges.
+MQTTTLS: <type>,<type_name>
OK
```

14.3 AT+MQTTCONNPARAM – Setting User Parameters

To set the ID, user name, password parameters.

The user name and password can be null.

Format

Type	Command	Response
Set	• MQTTMUX=0 AT+MQTTCONNPARAM=<”clientdID”>,<”username”>,<”password”><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
	• MQTTMUX=1 AT+MQTTCONNPARAM=<sid>,<”	

	clientdID">,<"username">,<"password"><CR>	
Query	AT+MQTTCONNPARAM?<CR>	- MQTTMUX=0 <CR><LF>+MQTTCONNPARAM:<"clientID">,<"username">,<"password"><CR><LF> <CR><LF>OK<CR><LF> - MQTTMUX=1 <CR><LF>+MQTTCONNPARAM:<sid><"clientID">,<"username">,<"password"><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+MQTTCONNPARAM=?<CR>	- MQTTMUX=0 <CR><LF>+MQTTCONNPARAM:<cliendid>,<username>,<password> <CR><LF>OK<CR><LF> - MQTTMUX=1 <CR><LF>+MQTTCONNPARAM:<sid>,<cliendid>,<username>,<password> <CR><LF>OK<CR><LF>

Parameter

<sid>	Socket ID, ranging from 0 to 5
<clientID>	Client ID, 256 bytes at most.
<username>	User name, 512 bytes at most.
<password>	Password, 256 bytes at most.

Example

```
AT+MQTTCONNPARAM="C_201801021127","lixytest/thing01","0   Parameters are set successfully.
lSoY/eYnlSqUeAsbAKKQ/ACmipZwEw9H7Ff0h1kOps="
OK
```

14.4 AT+MQTTWILLPARAM – Setting Will

To set will parameters.

These parameters cannot be set when the MQTT connection is established. The parameters set by this command is not saved, you need to reset them after the module is powered off.

Format

Type	Command	Response
Set	• MQTTMUX=0 AT+MQTTWILLPARAM=<retained>,<qos>,<”topicname”>,<”message”><CR> • MQTTMUX=1 AT+MQTTWILLPARAM=<sid>,<retained>,<qos>,<”topicname”>,<”message”><CR>	<CR><LF>OK<CR><LF>
Query	AT+MQTTWILLPARAM?<CR>	• MQTTMUX=0 <CR><LF>+MQTTWILLPARAM:<retained>,<qos>,<"topicname">,<"message"> <CR><LF>OK<CR><LF>
		• MQTTMUX=1 <CR><LF>+MQTTWILLPARAM:<sid>,<retained>,<qos>,<"topicname">,<"message"> <CR><LF>OK<CR><LF>
Test	AT+MQTTWILLPARAM=?<CR>	• MQTTMUX=0 <CR><LF>+MQTTWILLPARAM:<retained>,<qos>,<topicname>,<message> <CR><LF>OK<CR><LF> • MQTTMUX=1 <CR><LF>+MQTTWILLPARAM:<sid>,<retained>,<qos>,<topicname>,<message> <CR><LF>OK<CR><LF>

Parameter

<sid>	Socket ID, ranging from 0 to 5
<retained>	Retain mark, digit type.
<qos>	quality of service, 0 to 1.
<”topicname”>	Will topic name, 128 at most.
<”message”>	Will Message, 1024 at most.

Example

```
AT+MQTTWILLPARAM=0,1,"neoway02",”byby”
                                             The will is set successfully.
OK
```

14.5 AT+MQTTWILLMSG – Setting Long Will Messages

To set long will messages or will messages of non-character string by specifying retained, qos, topic, and message length.

Format

Type	Command	Response
Set	- MQTTMUX=0 AT+MQTTWILLMSG=<retained>,<qos>,<"topic">,<msg_length><CR>	<CR><LF>> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
	- MQTTMUX=1 AT+MQTTWILLMSG=<sid>,<retained>,<qos>,<"topic">,<msg_length><CR>	

Parameter

<sid>	Socket ID, ranging from 0 to 5
<retained>	Retain mark, digit type, 0 and 1.
<qos>	QoS of the published message.
<"topic">	Topic that is published.
<willmsg_length>	Length of the message body, 10240 bytes at most. Enter the message content of the length specified by after>.

Example

```
AT+MQTTWILLMSG =1,1,"neoway02",10                          Set will message.
>                                                          Successful

OK
AT+MQTTWILLMSG=1,1,"neoway02",10                           Set will message.
>                                                          Failed

+MQTTWILLMSG: Timeout!
```

14.6 AT+MQTTCONN – Connection Command

To connect to the MQTT server.

Wait for the return value during the module is connecting to the server. You cannot perform the connection operation again if no value is return.

After the connection is set up successfully, if the module reports **+MQTTDISCONNED:Link Close**d while you have not send the connection close command set up the connection manually.

Format

Type	Command	Response
Set	- MQTTMUX=0 AT+MQTTCONN=<"host">,<clean>,<keep_alive><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
	- MQTTMUX=1 AT+MQTTCONN=<sid>,<"host">,<clean>,<keep_alive><CR>	
Query	AT+MQTTCONN?<CR>	- MQTTMUX=0 <CR><LF>+MQTTCONN:<"ip:port">,<clean>,<keep_alive> <CR><LF>OK<CR><LF>
		- MQTTMUX=1 <CR><LF>+MQTTCONN:<sid>,<"ip:port">,<clean>,<keep_alive> <CR><LF>OK<CR><LF>
Test	AT+MQTTCONN=?<CR>	- MQTTMUX=0 <CR><LF>+MQTTCONN:<host>,<clean>,<keep_alive> <CR><LF>OK<CR><LF>
		- MQTTMUX=1 <CR><LF>+MQTTCONN:<sid>,<"ip:port">,<clean>,<keep_alive> <CR><LF>OK<CR><LF>

Parameter

<sid>	Socket ID, ranging from 0 to 5
<"host">	Server address (URL:port).
<clean>	Whether to clean session, digit type. 0: Not clean (default) 1: Clean
<keep_alive>	timeout period, ranging from 20 to 180, unit: s

Example

```
AT+MQTTCONN="121.43.166.63:1883",0,60
                                                    The connection is set up successfully.
OK
```

14.7 AT+MQTTSUB - Subscribing to a Topic

To subscribe to a topic.

If the subscription is failed perform the operation again after querying the MQTT connection and network connection status. When the network is poor, the return value is slow.

The query command is valid only when the connection is set up. You can only query the latest subscribed QoS and topic.

Format

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

Parameter

<sid>   Socket ID, ranging from 0 to 5

<"topicname"> Topic name, 128 at most.
<qos> Quality of service, 0,1 to 2

Example

```
AT+MQTTSUB="neoway02",1
OK                                                     Subscribe to the topic successfully. The
                                                       server issues the topic retained last
                                                       time.
+MQTTSUB:9,"neoway02",11,neoway mqtt
AT+MQTTSUB= neoway02,1
OK                                                     Subscribed to the topic successfully.
```

14.8 AT+MQTTUNSUB – Canceling a Subscription

To cancel a subscription of the specified topic.

When you fail to cancel the subscription, query the network status. When the network is poor, the return value is slow.

Format

Type	Command	Response
Execute	• MQTTMUX=0 AT+MQTTUNSUB=<"topicname"><CR> • MQTTMUX=1 AT+MQTTUNSUB=<sid>,<"topicname"><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<sid> Socket ID, ranging from 0 to 5
<"topicname"> Name of the topic to be unsubscribed to, 128 at most.

Example

```
AT+MQTTUNSUB="neoway02"
OK                                                          Cancel a subscription.
```

14.9 AT+MQTTPUB – Publishing a Topic

To publish a topic.

When the network is poor, the return value is slow.

It is recommended to publish topic containing the "\" character using the AT+MQTTPUBS command since "\" is an escape character.

Format

Type	Command	Response
Execute	• MQTTMUX=0 AT+MQTTPUB=<retained>,<qos>,<"topicname">,<"message"><CR> • MQTTMUX=1 AT+MQTTPUB=<sid>,<retained>,<qos>,<"topicname">,<"message"><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<sid>	Socket ID, ranging from 0 to 5
<retained>	Retain mark, digit type, 0 and 1.
<qos>	quality of service, 0, 1, 2.
<"topicname">	Name of the topic that is published, 128 at most.
<"message">	Message that is published, 1024 at most.

Example

```
AT+MQTTPUB=1,1,"neoway02",”neoway mqtt”
OK                                                The topic is published successfully.
AT+MQTTPUB=1,1,"neoway02",”neoway mqtt”
OK                                                The topic is published successfully. The server
                                                  issues the topic.
+MQTTSUB:5,"neoway02",11, neowaymqtt
```

14.10 AT+MQTTPUBS – Publishing a Topic with Long Message

To publishing a topic with long message.

Format

Type	Command	Response
Set	• MQTTMUX=0 AT+MQTTPUBS=<retained>,<qos>,<"topic">,<msg_length><CR> • MQTTMUX=1 AT+MQTTPUBS=<cid>,<retained>,<qos>,<"topic">,<msg_length><CR>	<CR><LF>> <CR><LF>OK<CR><LF> Or <CR><LF>><CR><LF> <CR><LF>+MQTTPUBS: Timeout!<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<sid>	Socket ID, ranging from 0 to 5
<retained>	Retain mark, digit type, 0 and 1.
<qos>	QoS of the published message.
<"topic">	Topic that is published.
<msg_length>	Length of the message body, 10240 bytes at most. Enter the message content of the length specified by after >.

Example

```
AT+MQTTPUBS=1,1,"lixytopic",10     The message is published successfully.
>
OK
AT+MQTTPUBS=0,1,"lixytopic",12     Failed to publish the message, the writing operation times
>                                  out.
+MQTTPUBS: Timeout!
```

14.11 AT+MQTTDISCONN – Disconnecting to the MQTT Server

To close the MQTT connection.

The device disconnects to the MQTT server proactively and releases the MQTT resources. Then the MQTT resources are released. If you need to publish messages, you need to reset the MQTT connection parameters and set up the connection again.

Format

Type	Command	Response
Execute	● MQTTMUX=0 AT+MQTTDISCONN<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Set	● MQTTMUX=1 AT+MQTTDISCONN=<sid><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<sid>	Socket ID, ranging from 0 to 5

Example

```
AT+MQTTDISCONN                                          To close the MQTT connection.
OK
```

14.12 +MQTTSUB – Receiving Message

To receive the topic. When the network is poor, the return value is slow.

Format

Type	Command
URC	● MQTTMUX=0 +MQTTSUB:<message_id>,<"topicname">,<message_len>,<message><CR> ● MQTTMUX=1 +MQTTSUB:<sid>,<message_id>,<"topicname">,<message_len>,<message><CR>

Parameter

<sid>	Socket ID, ranging from 0 to 5
<message_id>	Message ID
<"topicname">	Will topic, value with double quotes.
<message_len>	Length of the data received.
<message>	data received.

Example

```
+MQTTSUB:":1,"neoway02",5,12345                                  Receive the topic.
```

14.13 AT+MQTTSTATE – Query the MQTT Connection Status

To query the MQTT connection status.

The setting by this command is not saved after the module is powered off.
Every tine you set up an MQTT connection, enable the URC of MQTT connection status.

Format

Type	Command	Response
Query	AT+MQTTSTATE?<CR>	• MQTTMUX=0 <CR><LF>+MQTTSTATE: <state><CR><LF> <CR><LF>OK<CR><LF> • MQTTMUX=1 <CR><LF>+MQTTSTATE: <sid>,<state><CR><LF> <CR><LF>OK<CR><LF>
Set	• MQTTMUX=0 AT+MQTTSTATE<CR> • MQTTMUX=1 AT+MQTTSTATE=<sid><CR>	• MQTTMUX=0 <CR><LF>+MQTTSTATE: <state><CR><LF> <CR><LF>OK<CR><LF> • MQTTMUX=1 <CR><LF>+MQTTSTATE: <sid>,<state><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<sid>	Socket ID, ranging from 0 to 5
<state>	Reconnection status 0: the connection has been closed 1: the connection is established.

Example

```
AT+MQTTSTATE?          Query the MQTT connection state.
+MQTTSTATE: 1          1 indicates the MQTT connection is established successfully.
```

```
OK
AT+MQTTSTATE?         Query the MQTT connection state.
+MQTTSTATE: 0         0 indicates the MQTT connection is closed.
OK
AT+MQTTSTATE=0        Query the MQTT connection status.
+MQTTSTATE: 0         0 indicates that the MQTT connection is closed.
OK
AT+MQTTMUX=1
OK
AT+MQTTSTATE=2        Query the MQTT connection status.
+MQTTSTATE: 2,0       0 indicates that MQTT connection on socket 2 is closed.
OK
```

When MQTTMUX=0, <sid> of the AT+MQTTMUX command must be set to 0.

When MQTTMUX=1, only the status of the MQTT connection on socket 0 can be obtained when querying using AT+MQTTMUX?.

15 AWS MQTT Commands

15.1 AT+AWSTLSCFG – Configuring AWS TLS Parameters

To configure AWS TLS parameters.

Add the certificate through the AT+CERTADD command.

Format

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

Parameter

<type>	Parameter type.
<value>	Corresponding parameter value
<authmode>	authentication mode 0: verify optional 1: verify required
<rootca>	string, CA certificate
<clientcert>	string, client certificate
<clientkey>	string, client key

Example

```
AT+AWSTLSCFG=authmode,1                    Set the authentication mode to verify
OK                                         required.
```

```
AT+AWSTLSCFG?                                       Query the current SSL settings.
+AWSTLSCFG: 1,ca.pem,cc.pem,ck.pem
OK
AT+AWSTLSCFG=?                                      Query the available parameter value ranges.
+AWSTLSCFG: <type>,<value>
OK
```

15.2 AT+AWSAUTHPARAM – Setting User Parameters

To set the ID, user name, password parameters.

The current version 2.3.0 does not require the username and password parameters. They are optional.

Format

Type	Command	Response
Set	AT+AWSAUTHPARAM=<cliendID>,<username>,<password><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<cliendID>	Client ID, 128 bytes at most.
<username>	User name, 512 bytes at most.
<password>	Password, 256 bytes at most.

Example

```
AT+AWSAUTHPARAM=1234567890,test,test           Parameters are set successfully.
OK
```

15.3 AT+AWSCONNPARAM – Setting the AWS Connection Parameter

To set the AWS connection parameter.

Format

Type	Command	Response
Set	AT+AWSCONNPARAM=<host>,<enable_reconnect><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<host> Server address (URL:port).

<enable_reconnect> Whether to enable reconnecting after a disconnection.
0: disable
1: enable

Example

```
AT+AWSCONNPARAM=a1epg1vh6w7hlk.iot.us-east-2.amazonaws.com:443,1     The connection parameters
OK                                                                    are set successfully.
```

15.4 AT+AWSCONN – Setting up the AWS MQTT Connection

To set up the AWS MQTT connection.

The current SDK version is 2.3.0 and only clean =1 and version=4 are supported.

Format

Type	Command	Response
Execute	AT+AWSCONN=<keepAlive>,<clean>,<version><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<keepAlive> Keepalive interval, ranging from 30 to 1200s; 60s by default

<clean> whether to clean session, numeric type,

0: Not clean
1: Clean
<version> MQTT version 4 = 3.1 1

Example

```
AT+AWSCONN=60,1,4                                       The connection is set up successfully.

OK
```

15.5 AT+AWSSUB – Subscribing to a Topic

To subscribe to a topic.

Format

Type	Command	Response
Execute	AT+AWSSUB=<topicname>,<qos><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<topicname> Topic name, 128 at most.
<qos> quality of service, 0 to 1.

Example

```
AT+AWSSUB=nwy_test/01,1                                 Subscribed to the topic successfully.
OK
```

15.6 AT+AWSUNSUB – Canceling a Subscription

To cancel a subscription of the specified topic.

Format

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

Parameter

<topicname> Name of the topic to be unsubscribed to, 128 at most.

Example

```
AT+AWSUNSUB=nwy_test/01                            Cancel a subscription.
OK
```

15.7 AT+AWSPUB – Publishing a Topic

To publish a topic.

Format

Type	Command	Response
Execute	AT+AWSPUB=<retained>,<qos>,<topicname>,<length><CR>	<CR><LF>> <CR><LF>OK<CR><LF> Or <CR><LF>> <CR><LF>OK <CR><LF>+AWSPUB: OK<CR><LF> Or <CR><LF>+AWSPUB: ERROR<CR><LF>

Parameter

<retained> Retain mark, digit type, 0 and 1.
The current SDK version is 2.3.0 and only retained=0 is supported.

<qos> quality of service, 0 to 1.

<topicname> Name of the topic that is published, 128 at most.

<length> Length of the message body, 10240 bytes at most. Enter the message content of the length specified by after >.

Example

```
AT+AWSPUB=1,1," nwy_test/01",11            The topic is published successfully.
>
OK
AT+AWSPUB=1,1," nwy_test/01",11            The topic is published successfully. The server
>                                          issues the topic.
OK
+AWSPUB: OK

+AWSSUBRECV:5,"nwy_test/01",11,12332ELO
```

15.8 AT+AWSDISCONN – Closing the AWS MQTT Connection

To close the AWS MQTT connection.

Format

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

Parameter

N/A.

Example

```
AT+AWSDISCONN                                           Disconnect to the MQTT server.
OK
```

15.9 +AWSSUBRECV – Receiving the Topic

To receive the topic.

Format

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

Parameter

<message_id>	Message ID
<topicname>	Topic name
<message_len>	Length of the data received.
<message>	data received

Example

```
+AWSSUBRECV: 5,"nwy_test/01",5,12345                    Receive the topic.
```

15.10 AT+AWSSTATE – Querying the MQTT Connection State

To query the MQTT connection status.

Format

Type	Command	Response
Execute	AT+AWSSTATE?<CR>	<CR><LF>+AWSSTATE: <n><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<n>   Connection status
0: the connection is closed
1: the connection is set up

Example

```
AT+AWSSTATE?          Query the MQTT connection state.
+AWSSTATE: 1          1 indicates the MQTT connection is established successfully.
OK
```

```
AT+AWSSTATE?           Query the MQTT connection state.
+AWSSTATE: 0           0 indicates the MQTT connection is closed.
OK
```

16 GPS Function

16.1 AT$MYGPSPWR – GPS Switch

GPS switch

Format

Type	Command	Response
Set	AT$MYGPSPWR=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n> Select request type
0: stop the GPS service
1: start the GPS service

- Disable the GPS service if you do not require the GPS service since the GPS service will take up more resources of the module.
- After the successful request, the module will take time to start GPS positioning; the specific time depends on environmental factors, pay attention to the antenna matching.

Example

```
AT$MYGPSPWR=1                                                       Stop the GPS service.
OK
AT$MYGPSPWR=0                                                       Close the GPS service.
OK
```

16.2 AT$MYGPSSTATE – Querying the GPS Switch Status

To query the GPS Switch Status.

Format

Type	Command	Response
Execute	AT$MYGPSSTATE<CR>	<CR><LF>$MYGPSSTATE: gps closed<CR><LF> Or <CR><LF>$MYGPSSTATE: gps opened<CR><LF>

Parameter

N/A.

Example

```
AT$MYGPSPWR=1
OK                                                                Enable GPS.
AT$MYGPSSTATE                                                     Query GPS status.
$MYGPSSTATE: gps opened
OK
```

16.3 AT$MYGPSPOS – Obtaining NMEA Data

To obtain NMEA data.

- The obtained data is GPS coordinates.
- It takes time to fix the position for the first time after GPS is enabled.
- V003 and later versions support query and mode parameter setting for all formats.

Format

Type	Command	Response
Execute	AT$MYGPSPOS=<TYPE>,<Mode><CR>	0: NMEA$GPGGA format <CR><LF>$MYGPSPOS: $GPGGA,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>,<12>,<13>,<14>*<hh> <CR><LF>OK<CR><LF> 1: NMEA$GPGSA format <CR><LF>$MYGPSPOS: $GPGSA,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>,<12>,<13>,<14>,<15>,<16>,<17>,*<hh><CR><LF>OK<CR><LF> 2: NMEA$GPGSV format

	<CR><LF>$MYGPSPOS: $GPGSV, <1>,<2>,<3>,<4>,<5>,<6>,<7>[,<4>,<5>,<6>,<7>...]*<hh> <CR><LF>$GPGSV, <1>,<2>,<3>,<4>,<5>,<6>,<7>[,<4>,<5>,<6>,<7>...]*<hh> <CR><LF>... <CR><LF>OK<CR><LF> 3: NMEA$GPRMC format <CR><LF>$MYGPSPOS: $GPRMC,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>,<12>*<hh> <CR><LF>OK<CR><LF> 4: NMEA$GPVTG format <CR><LF>$MYGPSPOS: $GPVTG,<1>,T,<2>,M,<3>,N,<4>,K,<5>*<hh> <CR><LF>OK<CR><LF> 5: NMEA$GPGLL format <CR><LF>$MYGPSPOS: $GPGLL,<1>,<2>,<3>,<4>,<5>,<6>*<hh> <CR><LF>OK<CR><LF>

Parameter

<TYPE> GPS data type, integer type
0: NMEA$GPGGA format
1: NMEA$GPGSA format
2: NMEA$GPGSV format
3: NMEA$GPRMC format
4: NMEA$GPVTG format
5: NMEA$GPGLL format
6: output all GPS information

<Mode> Output mode
0: output once (default)
1: output periodically; the output information is specified by the type parameter.
2: disable periodic output

- 0: NMEA$GPGGA format

<1> UTC time of position fix, hh mm ss

<2> latitude, dd mm mmmm

<3> Latitude direction
N: north latitude
S: south latitude

<4> longitude, ddd mm mmmm

<5> longitude direction
E: east longitude
W: west longitude

<6> Indicates the GPS status

	0: no fix
	1: GPS fix
	2: differential GPS fix
	3: not valid
	6: estimated
<7>	Number of satellites in use
<8>	Horizontal Dilution of Precision (HDOP)
<9>	altitude above mean sea level (geoid)
<10>	units of altitude
	M: meter
<11>	geoidal height
<12>	unit of geoidal height
<13>	time since last DGPS update
<14>	DGPS reference station ID
<hh>	checksum

- 1: NMEA$GPGSA format

<1>	Mode
	A: Automatic
	M: Manual
<2>	fix mode
	1: no fix
	2: 2D fix
	3: 3D fix
<3>	PRN number of satellite used for fix
<4>	PRN number in second channel
<5>	PRN number in third channel
<6>	PRN number in forth channel
<7>	PRN number in fifth channel
<8>	PRN number in sixth channel
<9>	PRN number in seventh channel
<10>	PRN number in eighth channel
<11>	PRN number in ninth channel
<12>	PRN number in tenth channel
<13>	PRN number in eleventh channel
<14>	PRN number in twelfth channel
<15>	position (3D) dilution of precision (PDOP) (0.5 - 99.9)
<16>	HDOP (0.5 - 99.9)
<17>	VDOP (0.5 - 99.9)
<hh>	checksum

- 2: NMEA$GPGSV format

<1>	The total number of GSV sentences
<2>	number of the GSV sentence
<3>	Total number of satellites in view, ranging from 00 to 12.
<4>	SV PRN number, ranging from 01 to 32.
<5>	Evaluation in degrees, ranging from 00 to 90 degree.

<6> Azimuth, degrees from true north, 000 to 359 actual value.

<7> SNR (C/No), 00-99 dB (null when not tracking)

<hh> checksum

8-11: Information about second SV, same as field 4-7

12-15: Information about third SV, same as field 4-7

16-19: Information about fourth SV, same as field 4-7 These fields are null if <hh> is not set.

The number of $GPGSV sentences is same as the value of <1>.

- 3: NMEA$GPRMC format

<1> UTC time of position fix, hhmmss

<2> status of position fix

A: valid

V: invalid

<3> latitude of fix, ddmm.mmmm (the preceding 0 is also transmitted)

<4> Latitude direction

N: north latitude

S: south latitude

<5> Longitude of fix, ddmm.mmmm (the preceding 0 is also transmitted)

<6> longitude direction

E: east longitude

W: west longitude

<7> Speed over ground, 0 to 999 kilometers/hour

<8> Course over ground, 0 to 359 degrees, true (the preceding 0 is also transmitted.)

<9> UTC date, format: ddmmyy

<10> Magnetic deviation, degrees (000.0 - 180.0; the preceding 0 is also transmitted.)

<11> Magnetic Deviation direction

E: East

W: West

<12> Differential mode

A: automatic

D: differential exclusively

E: estimated

N: invalid data

<hh> checksum

- 4: NMEA$GPVTG format

<1> True course over ground

<2> Magnetic course over ground

<3> Speed over ground

<4> Speed over ground

<5> Differential mode

A: automatic

D: differential exclusively

E: estimated

N: invalid data

<hh> checksum

- 5: NMEA$GPGLL format

<1>　Latitude ddmm.mmmm

<2>　Latitude direction
N: north latitude
S: south latitude

<3>　Longitude dddmm.mmmm

<4>　longitude direction
E: east longitude
W: west longitude

<5>　UTC time of position fix, hhmmss.sss

<6>　Status
A: fix
V: no fix

<hh>　checksum

Example

```
AT$MYGPSPOS=0                                                      Obtain position information
$MYGPSPOS:                                                         of the module.
$GPGGA,060239.00,2241.170914,N,11359.187225,E,2,16,2.5,116.6,M,,,
,*39
OK
AT$MYGPSPOS=1                                                      Obtain the GPGSA data
$MYGPSPOS: $GPGSA,A,2,01,11,16,22,,,,,,,,,2.4,2.2,1.0*32           format.
OK
AT$MYGPSPOS=2                                                      Obtain the GPGSV data
$MYGPSPOS:                                                         format.
$GPGSV,5,1,19,01,57,160,50,11,84,151,31,16,09,090,41,22,10,140,49
*74
$GPGSV,5,2,19,04,40,184,,07,57,314,,08,87,050,,10,03,164,*7B
$GPGSV,5,3,19,15,15,270,,19,,,,20,,,,21,50,071,*78
$GPGSV,5,4,19,24,13,196,,26,05,309,,28,,,,32,,,*7A
$GPGSV,5,5,19,41,,,41,42,,,43,50,,,43*72
OK
AT$MYGPSPOS=3                                                      Obtain the GPRMC data
$MYGPSPOS:                                                         format.
$GPRMC,074855.00,A,2241.207019,N,11359.188919,E,0.0,78.5,050517,2
.3,W,A*16
OK
AT$MYGPSPOS=4                                                      Obtain the GPVTG data
$MYGPSPOS: $GPVTG,78.5,T,80.8,M,0.0,N,0.0,K,A*29                   format.
OK
AT$MYGPSPOS=5                                                      Obtain the GPGLL data
$MYGPSPOS: $GPGLL,2241.207179,N,11359.188345,E,074856.00,A*0F      format.
OK
```

16.4 AT$MYGNSSSEL – Enabling an Aid GNSS System

To enabling an aid GNSS system.

- Before selecting an aid GNSS system, enable GPS.
- Positioning data of the aid GNSS system can be queried by $MYGNSSMSG.
- V003 and the later version support this command.

Format

Type	Command	Response
Execute	AT$MYGNSSSEL=<n><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	GNSS type 0: BD 1: GLONASS 3: BD and GALILEO

Example

```
AT$MYGNSSSEL=1                          Enable GLONASS aid positioning.
OK
AT$MYGNSSSEL=0                          Enable BDS aid positioning.
OK
```

16.5 AT$MYGNSSMSG – Obtaining Positioning Data of Aid GNSS System

To obtain positioning data of the aid GNSS system.

V003 and the later version support this command.

Format

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

Parameter

N/A.

Example

```
AT$MYGNSSMSG                                                        GLONASS is enabled as for
$GLGSV,3,1,09,,,,37,70,08,234,20,82,42,033,18,73,28,337,30*5D       aid positioning.
$GLGSV,3,2,09,80,47,040,22,79,14,102,39,84,19,196,35,83,73,158,43
*68
$GLGSV,3,3,09,71,08,282,16*5D
$GNGNS,015559.80,2241.185287,N,11359.175639,E,AAA,10,0.8,111.3,-
1.0,,*0F
$GNGSA,A,2,73,79,83,84,,,,,,,,,1.1,0.8,0.8,2*3E
OK
AT$MYGNSSMSG                                                        Only BD is enabled for aid
$GNGNS,015559.80,2241.185287,N,11359.175639,E,AAA,10,0.8,111.3,-    positioning.
1.0,,*0F
$GNGSA,A,2,201,203,204,206,209,,,,,,,,1.4,1.1,0.8,4*03
OK
```

16.6 AT+SETSERVER – Setting AGPS Server

To set the server of AGPS.

- Ensure that GPS is enabled before executing this command.
- The AGPS server needs to be deployed by the operator or provided by a third party, the example supl.qxwz.com is for testing purposes only.
- If a private network card or an IoT card is used, ensure that it can access to the set URL.

Format

Type	Command	Response
Set	AT+SETSERVER=<n>,<url>,<port> [<usr>,<pwd>,<pt>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<n>	Network mode type 1: 3GPP2 3: 3GPP (only supported under the GSM or LTE network.)
<url>	Server adress, support IPv4, IPv6,and URL.
<port>	Server port
<pwd>	password
<pt>	Provider type 0: Techtotop 1: Zhongke Microelectronics 2: Unicorecomm

Example

```
AT+SETSERVER=3,agnss.neoway.com,80,,,2          Set the IP address for the AGPS server.
OK
```

16.7 AT$MYGPSCFG – Setting GPS

To set GPS parameters.

Format

Type	Command	Response
Set	AT$MYGPSCFG=<type>,<value><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYGPSCFG?<CR>	<CR><LF>$MYGPSCFG: <type>,<value> <CR><LF>OK<CR><LF>

Parameter

<type> Set type

0: set GPS update frequency.

1: set NMEA data report

2: set format of NMEA data report

Others: reserved

<value> set based on <type>.

type=0, update interval, ranging from 1000 to 60000 ms.

type=1, specify whether to report. 0: disable report. 1: enable report (default).

type=2, decimal number. The following list masks of different NMEA formats. All enabled by default.

GPGGA	0x00000001
GPRMC	0x00000002
GPGSV	0x00000004
GPGSA	0x00000008
GPVTG	0x00000010
GLGSV	0x00000080
GNGSA	0x00000100
GNGGA	0x00000200
GNGLL	0x00000400
BDGSV	0x00000800
GNRMC	0x00001000
GNVTG	0x00002000
GNZDA	0x00004000
GPZDA	0x00008000
GPTXT	0x00010000
GPGLL	0x00020000

Example

```
AT$MYGPSCFG=0,1000                     Set GPS update frequency to 1 second. NMEA data is
OK                                     reported every one second.
AT$MYGPSCFG?                           Query parameter settings.
$MYGPSCFG: 0,1000

$MYGPSCFG: 1,1

$MYGPSCFG: 2,4294967295

OK
AT$MYGPSCFG=2,1                        Set to report GPGGA data only.
OK
AT$MYGPSCFG=1,0                        Disable NMEA report.
OK
```

17 BT/BLE General Commands

17.1 AT+NWBTBLEPWR – BT/BLE Power Switch

To enable/disable the BT/BLE power, to enable/disable the BT protocol stack and to complete the Initialization function for data transmission.

The BT/BLE power is disabled by default; after it is enabled, the module can be found and connected.

Format

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

Parameter

<status> Integer type, ranging from 0 to 1.
0: disable
1: enable

Example

```
AT+NWBTBLEPWR=1               Enable the BT/BLE module.
OK
AT+NWBTBLEPWR?                Read the current BT/BLT status
+NWBTBLEPWR:1                 Enabled
OK
```

```
AT+NWBTBLEPWR=?                Read the parameter list.
+NWBTBLEPWR: (0-1)
OK
```

17.2 AT+NWBTBLENAME–Setting the BT/BLE Name

To set/query the name of the current BT/BLE device.

The BT name can be changed only when the BT function is disabled.

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

Format

Type	Command	Response
Set	AT+NWBTBLENAME=<name><CR>	<CR><LF>OK<CR><LF>
Test	AT+NWBTBLENAME=?<CR>	<CR><LF>+NWBTBLENAME: <name> <CR><LF>OK<CR><LF>

Parameter

<name> Name of the current Bluetooth, character type (Chinese is not supported), ranging from 1 to 24.

Example

```
AT+NWBTBLENAME=”Neoway”                                  Set the BT/BLE name to Neoway.
OK

AT+NWBTBLENAME?                                          READ the name of the current BT/BLE module.
+NWBTBLENAME:Neoway
OK

AT+NWBTBLENAME=?                                         Read the parameter list.
+NWBTBLENAME: <name>
OK
```

17.3 AT+NWBTBLEMAC – Querying the BT/BLE MAC Address

To query the MAC address of the current BT/BLE device.

Format

Type	Command	Response
READ	AT+NWBTBLEMAC?<CR>	<CR><LF>+NWBTBLEMAC: <mac><CR><LF> <CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT+NWBTBLEMAC?
+NWBTBLEMAC:12:7B:59:96:96:1B                READDBT/BLE the MAC address is
OK                                           12:7B:59:96:96:1B.
```

18 BLE Function Commands

⚠ Currently, peripherals that support BLE function support the functions such as creating services, adding property features, and start services; please read the device's GATT service before use to determine whether the module supports dynamically added services.

18.1 AT+NWBLEROLE – Setting BLE Mode

To set the BLE mode (switch between master mode and slave mode); set the mode when the BLE function is disabled.

Format

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

Parameter

<ble_role>	BLE molde 0: slave mode (default) 1: master mode

Example

```
AT+NWBLEROLE=1                Set to master mode.
OK

AT+NWBLEROLE?                 Query what the BLE mode is set.
+NWBLEROLE: 1
```

```
OK
```

18.2 AT+NWBLEADV – Setting BLE Broadcast Parameters

To set BLE broadcast parameters.

- The module broadcasts packets through the fixed channels 37, 38, and 39.
- Before executing this command, ensure that the broadcast function is disabled.

Format

Type	Command	Response
Set	AT+NWBLEADV=<min>,<max><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NWBLEADV?<CR>	<CR><LF>+NWBLEADV: <min>,<max><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+NWBLEADV=?<CR>	<CR><LF>+NWBLEADV: range of supported(<min>-<max>),range of supported (<min>-<max>)<CR><LF> <CR><LF>OK<CR><LF>

Parameter

<min> Minimum interval of broadcast, integer type, ranging from 32 to 16384.
<max> Maximum interval of broadcast, integer type, ranging from 32 to 16384.

Actual broadcast interval: (min/max) * 0.625ms; actual broadcast interval ranges from 20ms to 10.28s.

Example

```
AT+NWBLEADV=100,5000                               Set the broadcast parameter.
OK

AT+NWBLEADV?
+NWBLEADV: 100,5000                                Read the current broadcast parameter.
```

```
OK

AT+NWBLEADV=?                                  Read the parameter list.
+NWBLEADV:(32-16384),(32-16384)
OK
```

18.3 AT+NWBLEADVEN – Enabling/Disabling BLE Broadcast

To enable/disable the BLE broadcast function.

Format

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

Parameter

<enable>   Integer type
            0: disable
            1: enable

Example

```
AT+NWBLEADVEN=1                                Enable the BLE broadcast function.
OK
AT+NWBLEADVEN?
+NWBLEADVEN: 1                                 Read the current broadcast Status.
OK

AT+NWBLEADVEN=?                                Read the parameter list.
+NWBLEADVEN:(0-1)
OK
```

- After the BLE device is set to slave mode, the broadcast function is enabled by default and it is disabled after a BLE connection is set up.
- When the BLE device is in master mode its broadcast function cannot be switched.

18.4 +NWURCBLESTAT – URC Notifying BLE Status

To notify the current BLE status.

Format

Type	Response
URC	<CR><LF>+NWURCBLESTAT: <status>[,<mac>]<CR><LF>

Parameter

<status>　0: connection close
　1: successful

<mac>　MAC address of the connected device.

Example

```
+NWURCBLESTAT: 1,2A:02:0A:72:C8:53                  The BLE device is connected successfully.
+NWURCBLESTAT: 0                                    The BLE connection is closed successfully.
```

19 BLE Peripherals (Client/Server)

19.1 AT+NWBLEPSRV – Creating Services

To create services.

Disable the BT/BLE function before creating services.

2 services can be created at most; 5 characteristics can be added under each service.

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

Format

Type	Command	Response
Set	AT+NWBLEPSRV=<app_id >,<uuid>,<num>,<p><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWBLEPSRV=?<CR>	<CR><LF>+NWBLEPSRV:<app>,<uuid>,range of supported<num>,range of supported<p> <CR><LF>OK<CR><LF>
URC	<CR><LF>+NWBLEPSRV: <srv_id><CR><LF>	

Parameter

<app_id> app number; specifies which app requires adding services; it shall be a created app (you can execute NWBLEREG? to query the status.)
<uuid> Set <app_id> to 0 since app registration function is not supported currently.
<num> The created UUID number.
<p> Characteristics number to be added, integer type, ranging from 1 to 5.
Whether it is a main service (be sure to create a main service under each app), integer type, ranging from 0 to 1.

Example

```
AT+NWBLEPSRV=0,”0x1808”,2,1            Create the 0x1808 (glucose) service that is a main service
OK                                     two characteristics under app0.
+NWBLEPSRV: 1
```

```
AT+NWBLEPSRV=?                         Read the parameter list.
+NWBLEPSRV: <0>,<uuid>,<1-5>,<0-1>
OK
```

19.2 AT+NWBLEPCRT–Adding Characteristics for the Service

To add characteristics.

Disable the BT/BLE function before add characteristics.

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

Format

Type	Command	Response
Set	AT+NWBLEPCRT=<app_id>,<srv_id>,<uuid>,<slt>,<per>,<cp><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWBLEPCRT=?<CR>	<CR><LF>+NWBLEPCRT:<app>,range of supported<srv>,<uuid>,range of supported<slt>,range of supported<per>,range of supported<cp> <CR><LF>OK<CR><LF>
URC	<CR><LF>+NWBLEPCRT:<crt_id><CR><LF>	

Parameter

<app_id> app ID, specifies which app requires adding services; prompted by +NWBLEREG.
Set <app_id> to 0 since app registration function is not supported currently.
<srv_id> Service ID, specifies which service requires adding characteristics;
<uuid> UUID of this characteristic.
<slt> Whether to select the description.
0: No select
1: select
Read and write permissions, integer type
<per> 0: read-only
1: write-only
2: read and write
Characteristic feature, integer type.
<cp> 0: write

1: read
2: inform
3: display
4: all of the above are supported

Example

```
AT+NWBLEPCRT=0,1,”0x9999”,0,2,4                   Add the 0x9999 characteristic to the 0x1808
OK                                                service under the app0. The description is not
+NWBLEPCRT: 0                                     selected. The permissions are set to
                                                  write|read|inform|display.
AT+NWBLEPCRT=?                                    Read the parameter list.
+NWBLEPCRT: <0>,<1-2>,<uuid>,<0-1>,<0-2>,<0-4>
OK
```

19.3 AT+NWBLEPSTR – Starting Services

To start a service.

Disable the BT/BLE function before starting a service.

Format

Type	Command	Response
Set	AT+NWBLEPSTR=<app_id>,<srv_id><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWBLEPSTR=?<CR>	<CR><LF>+NWBLEPSTR: <app_id>,range of supported<srv_id> <CR><LF>OK<CR><LF>

Parameter

<app_id> app ID, specifies which app requires adding services; prompted by +NWBLEPREG?.

<srv_id> Set <app_id> to 0 since app registration function is not supported currently.
Specify which service to start, the service UUID is reported by +NWBLEPSRV.

Example

```
AT+NWBLEPSTR=0,1                            Start the 0x1808 (glucose) under app 0.
OK
```

```
AT+NWBLEPSTR=?                              Read the parameter list.
+NWBLEPSTR: <0>,<1-2>
OK
```

19.4 AT+NWBLEPSEND – Sending Data

To send data to the central device.

Format

Type	Command	Response
Set	AT+NWBLEPSEND=<srv_id>,<crt_id>,<op>,<mode>, len<CR> Input data after the “>” character	<CR><LF>><CR><LF> <input data> If the connection is established and the sending buffer is not full: <CR><LF>OK<CR><LF> If the connection has not been established, closed abnormally, or the parameters are incorrect, the send buffer is full: <CR><LF>ERROR<CR><LF>
Test	AT+NWBLEPSEND=?<CR>	<CR><LF>+NWBLEPSEND:<srv_id>,<crt_id>,<op>,<mode>,<len><CR><LF> <CR><LF>OK<CR><LF>
URC	+NWBLEPSEND	<CR><LF>+NWBLEPSEND:<state><CR><LF>

Parameter

<srv_id> Service number, specifies which service to send data to; it must be a created
<crt_id> service; reported by +NWBLEPSRV.
<op> Characteristic number, specifies which characteristic to send data to; reported by +NWBLEPCRT.
Integer type
<mode> 0: send a notification
1: send an indication
Integer type
<len> Send mode
<state> 0: send data in HEX mode
1: send data in ASCII mode
data length, integer type, ranging from 1 to 1024.
0: Failed to send

1: sent successfully

Example

```
AT+NWBLEPSEND=0,0,0,1,5                         Send hello to the characteristic 0 under the app 0.
>hello                                          Successful
OK
+NWBLEPSEND:1
AT+NWBLEPSEND=?                                 Read the parameter list.
+NWBLEPSEND: <0-2>,<0-4>,<0-1>,<0-1>,<1-
1024>
OK
```

- The module comes with a service (srv_id=0) and its UUID is 0xFEE0; The crt_id of the characteristic (UUID=0xFEE1) is 0.
- Whether data can be sent successfully depends on whether the characteristic has the corresponding read and write permissions. Informing /indicating permissions
- The supported data length ranges from 1 to 1024.

19.5 +NWURCBLEPRECV – URC Notifying Data Received

To notify data received. The URC is presented on the AT channel.

Format

Type	Command
URC	<CR><LF>+NWURCBLEPRECV:<srv_id>,<crt_id>,<len>,<data><CR><LF>

Parameter

<data>	Data content
<len>	Data length
<srv_id>	Service ID
<crt_id>	Characteristic ID

Example

```
                                                     The received data is 123; it is in ASCII
+NWURCBLEPRECV: 0,0,3,123                            format.
+NWURCBLEPRECV:0,0,5,3132333435                      Displayed in HEX format.
```

19.6 AT+NWBLEPWRITE – Writing Data

To write data to the characteristic.

Format

Type	Command	Response
Set	AT+NWBLEPWRITE=<srv_id>,<crt_id>,<mode>, <data><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWBLEPWRITE=?<CR>	<CR><LF>+NWBLEPWRITE:range of supported<srv_id>,range of supported<crt_id>,range of supported<mode>,<data> <CR><LF>OK<CR><LF>

Parameter

<srv_id>	Service ID, integer type, ranges from 0 to 2.
<crt_id>	Characteristic ID, integer type, ranges from 0 to 4.
<mode>	Data mode, integer type, ranging from 0 to 1. 0: HEX 1: ASCII
<data>	Data content

Example

```
AT+NWBLEPWRITE=1,0,0,33363636                Write hexadecimal data 33363636 to the characteristic 0
OK                                           under the service 1.
AT+NWBLEPWRITE=?                             Read the parameter list.
+NWBLEPWRITE: <0-2>,<0-4>,<0-1>,<data>
OK
```

19.7 AT+NWBLERCVMODE–Setting the Received Data Format

To set the received data format.

Format

Type	Command	Response
Set	AT+NWBLERCVMODE=<mode><CR>	<CR><LF>OK<CR><LF>
Query	AT+NWBLERCVMODE?	<CR><LF>NWBLERCVMODE:<mode><CR><LF> <CR><LF>OK<CR><LF>
Test	AT+NWBLERCVMODE=?<CR>	<CR><LF>+NWBLERCVMODE:range<mode><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<mode>   Integer type, ranging from 0 to 1.
0: HEX mode
1: ASCII mode (default)

Example

```
AT+NWBLERCVMODE=1                                  Set to receive mode.
OK
AT+NWBLERCVMODE=?                                  Read the parameter list.
+NWBLERCVMODE: <0-1>
OK
```

19.8 AT+NWBLEDISCON – Querying/Disconnecting the BLE Connection

To query the current connection information; to disconnected from the remote BLE device.

Format

Type	Command	Response

Execute	AT+NWBLEDISCON<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NWBLEDISCON?	<CR><LF>+NWBLEDISCON:<CR><LF> <CR><LF><index>,<name>,<mac><CR><LF> <CR><LF><index>,<name>,<mac><CR><LF> ... <CR><LF><index>,<name>,<mac><CR><LF> ... <CR><LF>OK<CR><LF>

Parameter

<index>	Index of the remote device
<name>	Name of the remote device
<mac>	MAC address of the remote device.

Example

```
AT+NWBLEDISCON                                    Close the connection to the device.
OK
AT+NWBLEDISCON?                                   Query the current connected device list.
+NWBLEDISCON:
1,65:7F:49:B2:21:6D
OK
```

19.9 AT+NWBLESRVRM – Removing the Specified Service

To remove the specified service.

Close the BLE connection before removing a service.

Format

Type	Command	Response
Set	AT+NWBLESRVRM=<app_id>,<srv_id><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWBLESRVRM=?<CR>	<CR><LF>+NWBLESRVRM:

		<app_id>,range of supported<srv_id> <CR><LF>OK<CR><LF>

Parameter

<app_id> app ID (it is returned once the app is created successfully.)
Set <app_id> to 0 since app registration function is not supported currently.

<srv_id> Service ID (it is returned once the characteristic is created successfully.)

Example

```
AT+NWBLESRVRM=0,1                                   Remove service 1 under app 0.
OK
                                                    Read the parameter list.
AT+NWBLESRVRM=?
+NWBLESRVRM: <0>,<1-2>
OK
```

19.10 AT+NWBLESRVRM – Removing the Specified Characteristic

To remove the specified characteristic.

Close the BLE connection before removing a characteristic.

Format

Type	Command	Response
Set	AT+NWBLECRTRM=<app_id>,<srv_id>,<crt_id><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWBLECRTRM=?<CR>	<CR><LF>+NWBLECRTRM: <app_id>,value range of <srv_id>,value range of<crt_id><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<app_id> app ID (it is returned once the app is created successfully.)
Set <app_id> to 0 since app registration function is not supported currently.

<srv_id> Service ID, ranging from 1 to 2 (it is returned once the characteristic is created

successfully.)

<crt_id> Characteristic ID, ranging from 0 to 4 (it is returned once the characteristic is created successfully.)

Example

```
AT+NWBLECRTRM=0,0,0                                 Remove characteristic 0 of service 0 under app
OK                                                  0.
                                                    Read the parameter list.
AT+NWBLESRVRM=?
+NWBLESRVRM:<0>,<1-2>,<0-4>
OK
```

19.11 AT+NWIBEACON – iBeacon Function

iBeacon function.

Close the broadcast function when setting the iBeacon parameters.

iBeacon function is disabled after BT is restarted.

Format

Type	Command	Response
Set	AT+NWIBEACON=<uuid>,<major>,<minor><CR>	<CR><LF>OK<CR><LF>
Test	AT+NWIBEACON=?<CR>	<CR><LF>+NWIBEACON: <uuid>,<major>,<minor><CR><LF> <CR><LF>OK<CR><LF>
Query	AT+NWIBEACON?<CR>	<CR><LF>+NWIBEACON: <uuid>,<major>,<minor><CR><LF> <CR><LF>OK<CR><LF>

Parameter

<uuid> UUID value, 16-byte hex data.
<major> Major value, 2-byte hex data.
<minor> Minor value, 2-byte hex data.

Example

```
AT+NWIBEACON=B9007F30F5F8466EAFF925556B57FE55,1   Set the iBeacon parameters.
234,0001
OK
                                                  Read the parameter list.
AT+NWIBEACON=?
+NWIBEACON: <uuid>,<major>,<minor>
OK
AT+NWIBEACON?                                     Query the iBeacon parameters.
+NWIBEACON:
B9007F30F5F8466EAFF925556B57FE55,1234,0001
OK
```

20 BLE Central Device (Client/Server)

20.1 AT+NWBLESCAN – Scanning the Surrounding BLE Devices

To scan the surrounding BLE devices.

When the scanned BLE device is Chinese name, it may display garbled code.

The number of scans is related to the signal strength of the current BLE device.

Format

Type	Command	Response
Set	AT+NWBLESCAN=<scan _time ><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	+NWURCBLESCAN	<CR><LF>+NWURCBLESCAN:<CR><LF> <CR><LF><name>,<mac>,<addr_type><CR><LF> ...

Parameter

<scan_time> Scanning time, integer type, ranging form 3 to 60s.
<name> Name of the remote BLE device
<mac> MAC address of the remote BLE device.
<addr_type> Address type
0: public address
1: random address
2: RPA public address
3: RPA random address

Example

```
AT+NWBLESCAN=50                        Set 50s to scan peripheral devices, timeout stops
OK                                     scanning.
+NWURCBLESCAN:
```

```
Honor V10,38:37:8b:71:28:c6,0          The scanning result is prompted.
Shitou,5c:c3:07:16:dc:ce,0
```

20.2 AT+NWBLECCON – Establishing the BLE Connection

To establish a BLE connection between the module and the BLE device.

Format

Type	Command	Response
Set	AT+NWBLECCON=<addr_type>,<mac><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NWBLECCON?<CR>	<CR><LF>+NWBLECCON:<CR><LF> <CR><LF><index>,<mac><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<addr_type> Address type, integer type. Parameter range is obtained through +NWURCBLESCAN.

<mac> MAC address of the BLE device, string type, the parameter range is indicated by +NWURCBLESCAN.

<index> Index of the remote device

Example

```
AT+NWBLECCON=0,”58:86:ed:28:bb:5a”          Set up a connection with the specified BLE device.
OK
+NWURCBLESTAT: 1,58:86:ED:28:BB:5A          If the connection is set up successfully
                                             +NWURCBLESTAT: <status>[,<mac>] is prompted.
AT+NWBLECCON?                                Query the index number and MAC address of the remote
+NWBLECCON:                                  BLE connection.
1,58:86:ED:28:BB:5A                          If no connection is established, <index> and <mac>
OK                                           are not displayed.
```

20.3 AT+NWBLECDISCON – Closing the BLE Connection

To close the BLE connection.

Format

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

Parameter

N/A.

Example

```
AT+NWBLECDISCON                           To close the connection to the specified BLE device.
OK
```

20.4 AT+NWBLEQSRV – Finding the BLE Device Service

To find the BLE device service.

Format

Type	Command	Response
Execute	AT+NWBLEQSRV<CR>	<CR><LF>+NWBLEQSRV:<CR><LF> <CR><LF><srv_id>,<srv_uuid>,<srv_type><CR><LF> … <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<srv_id>	Service ID, integer type, increasing from 0.

<srv_uuid>   Service UUID
<srv_type>   Service type
                    0: secondary service
                    1: primary service

Example

```
AT+NWBLEQSRV          Start to find services
+NWBLEQSRV:           The found services
0,0x1808,1            Primary service (service ID is 0, UUID is 0x1808, and property type is 1)
1,0x1811,0            Secondary service (service ID is 1, UUID is 0x1811, and property type is 0)
...
OK
```

20.5 AT+NWBLEQCHAR – Finding the BLE Device Characteristic

To find the BLE device characteristic.

Format

Type	Command	Response
Set	AT+NWBLEQCHAR=<srv_id><CR>	<CR><LF>+NWBLEQCHAR:<CR><LF> <CR><LF><srv_id>,<char_id>,<char_uuid>,<char_prop><CR><LF> … <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT+NWBLEQCHAR<CR>	<CR><LF>+NWBLEQCHAR:<CR><LF> <CR><LF><srv_id>,<char_id>,<char_uuid>,<char_prop><CR><LF> … <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<srv_id>   Service ID, integer type, the parameter range is indicated by the AT+NWBLEQSRV or AT+NWBLEQSRV command.

<char_id>	Characteristic ID, integer type, increasing from 0.
<char_uuid>	characteristic UUID of the service
<char_prop>	Characteristic property of the service: Broadcast 0x01 Read 0x02 Write without response 0x04 Write 0x08 Notify 0x10 Indicate 0x20 Authenticated Signed Write 0x40 Extended Properties 0x80

Example

```
AT+NWBLEQCHAR=0          Obtain the information of characteristic 0 under the specified service.
+NWBLEQCHAR:             The obtained information of the specified characteristic 0.
0,0,0x2906,0xff          The service ID is 0, the characteristic ID is 0, the characteristic UUID
0,1,0x2907,0xef          is 0x2906, and the service characteristic property is 0xff (all
...                      properties are supported)
                         The service ID is 0, the characteristic ID is 1, the characteristic UUID
OK                       is 0x2906, and the service characteristic property is 0xef (only
AT+NWBLEQCHAR            Extended Properties is not supported)
+NWBLEQCHAR:             ...
0,0,0x2906,0xff
0,1,0x2907,0xef          Obtain all characteristic information of the service.
1,0,0x2906,0xff          The obtained information of all the characteristics.
1,1,0x2907,0xef          The service ID is 0, the characteristic ID is 0, the characteristic UUID
...                      is 0x2906, and the service characteristic property is 0xff.

OK
                         The service ID is 1, the characteristic ID is 0, the characteristic UUID
                         is 0x2906, and the service characteristic property is 0xff.
```

20.6 AT+NWBLECSEND – Sending Data

To send data to a slave BLE device as a master BLE device.

Format

Type	Command	Response
Set	AT+NWBLECSEND=<srv_id>,<char_id>,<mode>,<len><CR>	<CR><LF>><CR><LF> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
URC	<CR><LF>+NWBLECSEND: <state><CR><LF>	

Parameter

<srv_id> Service ID, integer type, the parameter range is indicated by the AT+NWBLEQSRV or AT+NWBLEQSRV command.

<char_id> Characteristic ID, integer type, the parameter range is indicated by AT+NWBLEQCHAR.

<mode> Send mode, integer type.
0: HEX
1: ASCII mode (default)

<len> Data length, parameter type, ranges from 1 to 1024.

<state> 0: Failed to send
1: sent successfully

Example

```
AT+NWBLECSEND=1,1,1,5         Set the parameters for data sending
>hello                        Input data to be sent after the “>” character is prompted.
OK                            OK is returned because the connection is established and the
Or                            sending buffer is not full.
ERROR
+NWBLECSEND:1                 ERROR is returned because the connection has not been established,
                              closed abnormally, or the parameters are incorrect, or the send
                              buffer is full.
                              The URC is prompted to indicate that data is sent successfully.
```

20.7 +NWURCBLECRECV – Receiving Data

To indicate that data is received by the BLE device in master mode.

Format

Type	Command
URC	<CR><LF>+NWURCBLECRECV: <srv_id>,<crt_id>,<len>,<data><CR><LF>

Parameter

<data> Data content
<len> Data length
<srv_id> Service ID
<char_id> Characteristic ID

Example

```
+NWURCBLECRECV:3,0,3,123      The BLE device (service ID is 3, characteristic ID is 0) receives 3-
                              byte data 123 in ASCII format.
```

The receiving data is in ASCII format; you can use AT+NWBLERCVMODE to modify the format if required.

20.8 AT+NWBLECREAD – Reading Characteristic Data

To read the characteristic information from the slave BLE device.

Format

Type	Command	Response
Set	AT+NWBLECREAD=<srv_id>,<char_id>,<mode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+NWBLECREAD=?<CR>	<CR><LF>+NWBLECREAD:range of supported<srv_id>,range of supported<crt_id>,range of supported<mode> <CR><LF>OK<CR><LF>
URC	+NWURCBLECREAD:<data_len>,<data>	

Parameter

<srv_id>	Service ID, integer type, ranges from 0 to 9.
<char_id>	Characteristic ID, integer type, ranges from 0 to 9.
<mode>	Data mode, integer type, ranging from 0 to 1. 0: HEX 1: ASCII
<data_len>	Data length
<data>	Data content

Example

```
AT+NWBLECREAD=4,0,0                          Read the information in HEX format of the
OK                                           characteristic of which the service is 4 and the
+NWURCBLECREAD:                              characteristic ID is 0.
245,33363636000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00000000000000000000000000000000000000000
00
AT+NWBLECREAD=?                              Read the parameter list.
+NWBLECREAD: <0-9>,<0-9>,<0-1>
OK
```

21 DTMF Commands

21.1 AT+VTS – Transmitting DTMF Tones

To transmitting BTMF tones.

This command is valid during a call.

Format

Type	Command	Response
Execute	AT+VTS=<DTMF><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Test	AT+VTS=?<CR>	<CR><LF>+VTS: (value range of <DTMF>) <CR><LF>OK<CR><LF>

Parameter

<DTMF>	A single ASCII character in the set 0-9, #, *, A-D.
<TONE>	Ranges from 1 to 10, unit tone*100 ms.

Example

```
AT+VTS=?                                                Query the DTMF signal range of the
+VTS: (0-9,*,#,A,B,C,D),(1-255)                         module.
OK
AT+VTS=1                                                Set during a call.
OK
AT+VTS=1                                                Set not during a call.
ERROR
```

22 GSM Location Command

22.1 AT+CIPGSMLOC – Obtaining the Location of the Module

Obtain the base station location information of the module.

The obtained data is GPS coordinates.

The current coordinates of latitude and longitude are valid and precision is reserved (0.0 by default).

Format

Type	Command	Response
Execute	AT+CIPGSMLOC[=<n>]<CR>	<CR><LF>+CIPGSMLOC: <fail_string><CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: {<result_string>} <CR><LF>+CIPGSMLOC: OK<CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: <code><CR><LF> <CR><LF>+CIPGSMLOC: FAIL<CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>+CIPGSMLOC: TIMEOUT<CR><LF>

Parameter

<n> Request selection
0: close positioning request
1: enable positioning through multiple base stations

<fail_string> Prompt strings for request failure
CONTACT FAIL
LINK FAIL
LINK NOT FREE

<result_string> String containing latitude and longitude

<code> Respond code of the server after the request is successful and no latitude and longitude data is returned.
401: Unauthorized

400: Bad Request
404: Not Found
408: Request Timeout
500: Server Error

Example

```
AT+CIPGSMLOC                                                The command is executed
OK                                                          successfully.

+CIPGSMLOC:{"location":{"lat":22.69083,"lng":113.985228},
"accuracy":0.0}                                             Report the location information of
+CIPGSMLOC: OK                                              the module.
AT+CIPGSMLOC                                                Failed to translate the server DNS
+CIPGSMLOC: CONTACT FAIL                                    name.
AT+CIPGSMLOC
                                                            Failed to connect to the server.
+CIPGSMLOC: LINK FAIL
AT+CIPGSMLOC
                                                            Location request succeeded.
OK

                                                            Since the base station queried is
+CIPGSMLOC: 404
                                                            not included, the result cannot be
                                                            calculated.
+CIPGSMLOC: FAIL
                                                            Request multiple base stations
AT+CIPGSMLOC=1                                              positioning.
OK                                                          The last request has not been
AT+CIPGSMLOC=1                                              responded to, and the link has not
+CIPGSMLOC: LINK NOT FREE                                   been released. Request again,
                                                            prompting that the link is occupied.
AT+CIPGSMLOC=0                                              Close the request.
OK                                                          The link is released.
AT+CIPGSMLOC=1
                                                            positioning through multiple base
OK
                                                            stations is requested successfully.

+CIPGSMLOC: {"location": {"lat": 22.689646628671216,
                                                            The module reports the position
"lng":113.98586121790129},"accuracy":0.0}
                                                            coordinates.
+CIPGSMLOC: OK
```

23 Time Synchronization Commands

23.1 AT+UPDATETIME – Updating Time to Network

To update the module time to the network time.

Set up a PPP link (AT+XIIC=1) before sending this command. Send AT+CCLK? to query whether RTC is synchronized to the current network time after this command is sent successfully.

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

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

Format

Type	Command	Response
Execute	AT+UPDATETIME=<mode>[,<serv_ip>,<time>[[,<TZ>][,<DST>]]]<CR>	<mode>=0 <CR><LF>+UPDATETIME: Last Update Time yyyy-mm-dd,hh:mm:ss <CR><LF>OK<CR><LF> <mode>=1 <CR><LF>OK<CR><LF> <CR><LF>+UPDATETIME:<result code><CR><LF> Or <CR><LF>OK<CR><LF> <CR><LF>Time Updating,Please Wait...<CR><LF> <CR><LF>+UPDATETIME: Update To yyyy-mm-dd,hh:mm:ss<CR><LF>
Query	AT+UPDATETIME?<CR>	<CR><LF>+UPDATETIME: <serv_ip>,<time>,<TZ>,<DST> <CR><LF>OK<CR><LF>
Test	AT+UPDATETIME=?<CR>	<CR><LF>+UPDATETIME: (value range of <mode>),,(value range of <time>),,(value range of <DST>) <CR><LF>OK<CR><LF>

Parameter

<mode>	0: Query mode. Query when the time was updated to the network time last time. 1: Setting mode. Synchronize the time to the network time.
<serv_ip>	The IP address of the time server, in form of xx.xx.xx.xx or domain name.
<time>	the timeout period, ranging from 1 to 60, unit: second.
<TZ>	Time zone, in format of E/W+digits; E8 by default. E: east time zone, 0 to 13 W: west time zone, 0 to 12 0: Zero time zone
<DST>	Daylight Saving Time 1: Select DST auto-adjustment 0: Not select (by default)
<result code>	No PPP Link Time Out Time Data Is Null Send Request Fail Domain Name Invalid Socket Error

Example

```
AT+UPDATETIME=0                                    Query when the time was updated last time.
+UPDATETIME:Last Update Time 2014-03-31,11:10:26
OK                                                 The updated time: 2014-03-31,11:10:26
AT+UPDATETIME=0                                    Query when the time was updated last time.
+UPDATETIME: Last Update Time 0000-00-00,00:00:00
OK                                                 The time was not updated.
AT+UPDATETIME=1,210.72.145.44,10
                                                   No PPP connection is set up.
+UPDATETIME: No PPP Link

                                                   Synchronize with the network time of
AT+UPDATETIME=1,210.72.145.44,10                   210.72.145.44.
OK                                                 Time out: 10s.
                                                   The default time zone is East 8.
Time Updating,Please Wait...                       Daylight saving time is not selected to
                                                   prompt.
+UPDATETIME: Time Out                              Synchronization times out because the
                                                   network is busy.
                                                   Update the time to that of the server
AT+UPDATETIME=1,128.138.141.172,10,"E8",0
                                                   128.138.141.172.
OK
                                                   Time out: 10s.
                                                   The time zone is set to East 8.
Time Updating,Please Wait...
                                                   Daylight saving time is not selected to
                                                   prompt.
+UPDATETIME: Update To 2014-03-31,11:32:55
                                                   Time is updated successfully.
AT+UPDATETIME=1,time.windows.com,10,"W12",1        Synchronize with the network time of
OK                                                 time.windows.com.
                                                   Time out: 10s.
Time Updating,Please Wait...                       The time zone is set to West 12.
                                                   Select daylight saving time.
+UPDATETIME: Update To 2014-04-12,15:17:48         Time is updated successfully.
```

AT+UPDATETIME=1,128.138.141.172,10,"W12",1 OK +UPDATETIME: Send Request Fail	Time update request sending fails. The reason probably is bad network connection or inability to support time update.
AT+UPDATETIME=1,time.windows.com,10,"W12",1 +UPDATETIME: Domain Name Invalid	The domain name is invalid. The possible reason is the SIM (USIM) card is out of credit.
AT+UPDATETIME=1,time.windows.com,10,"W12",1 OK +UPDATETIME: Socket Error	Socket error. The possible reason might be network congestion.
AT+UPDATETIME? +UPDATETIME: 128.138.141.172,10,"E8",0 OK	Query the IP address of the server to which the time is updated and the timeout period, time zone, and DTS.
AT+UPDATETIME=? +UPDATETIME: (0-1),,(1-30),,(0-1) OK	Query available parameter value ranges.
AT+UPDATETIME=0 +UPDATETIME:Last Update Time 2014-03-31,11:10:26 OK	Query when the time was updated last time. Update to 2014-03-31,11:10:26
AT+UPDATETIME=0 +UPDATETIME: Last Update Time 0000-00-00,00:00:00 OK	Query when the time was updated last time. The time was not updated.
AT+UPDATETIME=1,210.72.145.44,10 +UPDATETIME: No PPP Link	No PPP connection is set up.
AT+UPDATETIME=1,210.72.145.44,10 OK Time Updating,Please Wait... +UPDATETIME: Time Out	Synchronize with the network time of 210.72.145.44. Time out: 10s. The default time zone is East 8. Daylight saving time is not selected to prompt. Synchronization times out because the network is busy.

24 Network Sharing Commands

24.1 AT+NETSHAREMODE – Selecting Network Sharing Mode

To select the network sharing mode.

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

Format

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

Parameter

<share_mode>	0: RNDIS 1: ECM

Example

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

24.2 AT+NETSHAREACT – Enabling Network Sharing

To enable network sharing.

Format

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

Parameter

<cid> PDP context, ranging from 2 to 7.

<action> Action
0: stop
1: start

<auto> Specifies whether to enable network sharing automatically after the module is powered on.
0: manual
1: auto

<APN> Access Point Name, ranging from 0 to 99.

<user_name> User name, ranging from 0 to 64.

<passwd> Character string type, password, ranging from 0 to 64.

<auth_type> Authentication type
0: NONE
1: PAP (default)
2: CHAP
3: PAP and CHAP

<ip_family> IP family
4: IPv4 (default)

	6: IPv6
	10: IPv4_6
<stat>	0: disconnect
	1: connect
<atuo>	0: manual
	1: auto
<err_code>	not supported currently
<wx_stat>	not supported currently
<PDP_type>	IPv4
<share_mode>	RNDIS/ECM

Example

```
AT+NETSHAREACT=2,1,0,ctnet,card,card,1                  Enable USB network sharing. The
OK                                                      autostart function is not set.
AT+NETSHAREACT=?                                        Query the ranges of the parameters.
+NETSHAREACT: (2-7),(0-1),(0-
1),"apn","user","passwd",(0-3),(4,6,10)
OK
AT+NETSHAREACT?                                         Query network sharing status.
+NETSHAREACT: 0,0,,,"IPV4",RNDIS
OK
```

25 Statistics on Data Traffic

25.1 AT+FLOWCALC – Statistics on Total Data Traffic

To collect the statistics on total data bytes that the module transmits and receives.

This command is used to count the total amount of traffic currently passing through the module in byte.

This command supports only statistics on data that the module transmitted and received using external protocols.

V003 and the later version support this command.

Format

Type	Command	Response
Query	AT+FLOWCALC?<CR>	<CR><LF>+FLOWCALC: <rx_count>,<tx_count> <CR><LF>OK<CR><LF>

Parameter

<rx_count>	Total data the module received, unit: byte.
<tx_count>	Total data the module transmitted, unit byte.

Example

```
AT+FLOWCALC?                          Query the total data that the module transmitted and
+FLOWCALC: 1355,1260                  received.
OK
AT+FLOWCALC?                          Query the total data that the module transmitted and
+FLOWCALC: 0,0                        received.
OK
```

25.2 AT+RATECALC – Statistics on Transient Traffic

To collect statistics on transient data bytes that the module transmits and receives within 100 ms.

This command supports only statistics on data that the module transmitted and received using external protocols.

V003 and the later version support this command.

Format

Type	Command	Response
Execute	AT+RATECALC?<CR>	<CR><LF>+RATECALC: <rx_count>,<tx_count> <CR><LF>OK<CR><LF>

Parameter

<rx_count>	Total data the module received, unit: byte.
<tx_count>	Total data the module transmitted, unit byte.

Example

```
AT+RATECALC?                          Query the total data that the module
+RATECALC: 1355,1260                  transmitted and received within 100ms.
OK
AT+RATECALC?                          Query the total data that the module
+RATECALC: 0,0                        transmitted and received within 100ms.
OK
```

26 File System Commands

26.1 AT+FSWF – Writing Data to File

To write data to a file.

Format

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

Parameter

<file_name>	File name, at most 50 characters
<mode>	0: If the file already exists, new data is written to the start of the file and then it overwrites original data. 1: If the file already exists, new data is written to the end of the file.
<size>	Data size, ranging from 0 to 1024*1024, unit: byte. The data to be written should not exceed <size>.
<time>	timeout period, ranging from 0 to 240000, unit: ms

- The data to be written should not exceed <size>.
- The size of the user disk is 1M, i.e. the total size of the files that can be stored is 1M.
- If SD card is supported, the command with the value of <file_name> containing /sdcard can be used to operate SD card files.
- If external flash is supported, include **/dataflash** () in file name to operate files in the SD card.

Example

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

```
OK
AT+FSWF="test.txt",1,1024,10000          The command times out after 10 seconds.
>
+FSWF: Timeout!
AT+FSWF="test.txt",1,1024,60001          ERROR is returned because the set value exceeds the
ERROR                                    parameter range.
AT+FSWF=”/sdcard0/test.txt”,1,10,60000   Write data to the test.txt file in SD card.
>
OK
AT+FSWF=/dataflash/test.txt,1,10,120000  Write data to the 10.txt file on the external flash.
>
OK
```

26.2 AT+FSRF – Reading Data from a File

To read data from a file

Format

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

Parameter

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

- the specified data size cannot exceed the total size of the file.
- <size> and <position> is determined by the file size.
- If SD card is supported, the command with the value of <file_name> containing /sdcard can be used to operate SD card files.
- If external flash is supported, include **/dataflash** () in file name to operate files in the SD card.

Example

```
AT+FSRF="test.txt",0,10             Read 10-byte data from the beginning of the test.txt
+FSRF: 10, start01234               file.
OK
AT+FSRF="test.txt",0,0              Read 0-byte data from the beginning of the test.txt file.
+FSRF: 0,
OK
AT+FSRF="test.txt",0,1025           ERROR is returned because <size> exceeds the file size.
ERROR
AT+FSRF="test.txt",1,20,2           Read 20-byte data from the second byte of the test.txt
+FSRF: 20, tart0123456789012345     file.
OK                                  The data is read successfully.
AT+FSRF="test.txt",1,0,2            Read 0-byte data from the second byte of the test.txt
+FSRF: 0,                           file.
OK
AT+FSRF="test.txt",1,10,1025        <Position> exceeds the file size.
ERROR
AT+FSRF="/sdcard0/test.txt",0,10    Read content of the test.txt file from the SD card.
+FSRF: 10, start01234
OK
AT+FSRF=/dataflash/test.txt,0,10    Read content of the test.txt file from the external
+FSRF: 10,aaaaaaaaaa                flash.
OK
```

26.3 AT+FSRFEX – Reading File

To read a file.

Format

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

Parameter

<file_name>	File name, the file length does not exceed 50 characters.
<checksum_mode>	Verification type 1: XOR check method 2 - 99: reserved
<mode>	0: read data from the beginning of the file. 1: to read data from the position specified in <position>.

<size>　　data size, not exceed the size of the file, it cannot be set to 0.

<position>　　the position in the file, where data to be read starts, valid when <mode> is set to 1, 0 is invalid. Currently 0x40000000 is supported at most.

<lenth>　　Length of the file to be read.

<content>　　Content of the file to be read.

<checksum>　　Content of the file to be read.

- <size> cannot exceed the total size of the file.
- <size> and <position> is determined by the file size.
- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

```
AT+FSRFEX="test.txt",1,0,10          Read 10-byte data from the beginning of the test.txt
+FSRFEX: 10,XXXX,start01234          file.
OK
AT+FSRFEX=/sdcard0/123.txt,1,0,10    Read 10-byte data from the beginning of the 123.txt
+FSRFEX: 10,XXXX,1234567890          file from the SD card.
OK
AT+FSRFEX=/dataflash/test.txt,1,0,10 Read 10-byte data from the beginning of the test.txt
+FSRFEX: 10,6425,aaaaaaaaaa          file from the external flash.
OK
```

26.4 AT+FSDF – Deleting a File

To delete a file.

Format

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

Parameter

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

- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

```
AT+FSDF="test.txt"                    Delete the text.txt file.
OK
AT+FSDF="123.txt"                     Delete the 123.txt file. ERROR is returned because the
ERROR                                 file does not exist.
AT+FSDF=/sdcard0/1.txt                Delete 1.txt from SD card
OK
AT+FSDF=/dataflash/test.txt           Delete the test.txt file from the external flash.
OK
```

26.5 AT+FSLIST – Obtaining File List

To obtain the list of files in the file system.

Format

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

Parameter

<directory>   Specify the directory to use when querying the file list in the SD card or external flash.

- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

```
AT+FSLIST?                              File name before comma
i.amr,6181                              File size after comma
file.txt,6000
OK
AT+FSLIST?                              No file in the file system
OK
AT+FSLIST=/sdcard0                      Query file list in /sdcard0
OK
AT+FSLIST=/dataflash                    Query the file list from the external flash.
+FSLIST: /dataflash/test.txt,10
OK
```

26.6 AT+FSFS – Obtaining the Size of a File

To obtain the size of a file.

Format

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

Parameter

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

- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

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

```
AT+FSFS=/sdcard0/1.txt                 Obtain the size of the 1.txt file in SD card.
+FSFS: 10
OK
AT+FSFS=/dataflash/test.txt            Query size of the test.txt file in the external flash.
+FSFS: 10
OK
```

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

To obtain the remaining storage size of user disk.

Format

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

Parameter

<directory> Specify the directory to use when querying the remaining size of the SD card or external flash.

<size> Remaining storage size of user disk.

- The block size of the file system is 500 byte, based on the actual file occupied block size, SD card or external flash different models block size may be different.
- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

```
AT+FSLS?                The remaining storage size of user disk is 64500 bytes.
+FSLS: 64500            64500=500*129, that is, there are 129 data blocks (500 bytes/block).
OK
AT+FSLS=/sdcard0        Query the remaining size of the SD card.
+FSLS: 1535345345
OK
AT+FSLS=/dataflash      Query the remaining size of the external flash.
+FSLS: 15878544
OK
```

26.8 AT+FSFAT – Formating the User Disk

To format the user disk.

Format

Type	Command	Response
Execute	AT+FSFAT[=<directory>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<directory> Specify the directory to use when formatting the SD card or external flash.

- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

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

```
ERROR

AT+FSFAT=/sdcard0          Format the SD card.
OK
AT+FSFAT=/dataflash        Format the external flash.
OK
```

26.9 AT+FSRN – Renaming the File

To rename the file.

If the file renamed exists, it will overwrite the original file.

If the file renamed does not exist, errors will be returned.

Format

Type	Command	Response
Execute	AT+FSRN=<src_file_name>,<dst_file_name><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<src_file_name>	File name to be renamed, at most 50 characters
<dst_file_name>	File name that after renaming, at most 50 characters

- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

```
AT+FSRN="test.txt","dst.txt"                             The file is renamed successfully.
OK
AT+FSRN="test1.txt","dst.txt"                            Failed to rename the file.
ERROR
AT+FSRN=/dataflash/test.txt,/dataflash/123.txt           Rename the file in the external
OK                                                       flash.
```

26.10 AT+NWSPIREAD – SPI Reads the File

SPI reads the file.

Format

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

Parameter

<spi_id> SPI number, integer type
1: SPI1
2: SPI2

<file_name> File name, its length is not greater than 50 characters.

<mode> 0: read data from the beginning position of the file.
1: read file from the position specified in <position>.

<size> Data size. <size> cannot be larger than the size of data read from the file. Setting to 0 is allowed.

<position> Specify the position of the file from which data starts to be read. <position> is available when <mode>=1. <size> and <position> are determined by the file size.

- <size> cannot be greater than the total size of the file.
- <size> and <position> are determined by the file size.
- If SD card is supported, the command with the value of <file_name> containing **/sdcard0** can be used to operate SD card files.
- If external flash is supported, execute the +NWYSPIFLASH command to perform initialization and then include **/dataflash** in file name to operate files in the SD card.

Example

```
AT+NWSPIREAD=1,"test",0,200            Read 200-byte data from the beginning of test.txt.
OK
```

27 Audio Recording Commands

27.1 AT+RECMODE – Setting Recording Mode

To set the buffer mode of record data.

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

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

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

Format

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

Parameter

<mode>	0: record local audio (default) 1: record downlink audio of voice (not supported) 2: record uplink audio of voice (not supported) 3: record mixed audio of voice
<file_name>	saved file name for audio record. Record files are saved from 001.pcm.
<format>	file format 0: pcm (default) 1: wav 2: mp3 (not supported) 3: amr

Example

```
AT+RECMODE=3,voice.pcm,0                                Set record mixed audio of voice.
OK
AT+RECMODE=0,audio.pcm,0                                Set record local audio.
OK
AT+RECMODE?                                             Query the settings for audio record.
+RECMODE: 0
OK
AT+RECMODE=?                                            Query the value range of the parameters.
+RECMODE: (0-3),<0:pcm,1:wav,2:mp3,3:war>,(0,1,2,3)
OK
```

27.2 AT+RECF – Starting and Stopping an Audio Recording

To start of stop an audio recording.

Set the record mode before starting an audio recording. If the recording file name is not set, the file is saved starting from 001.pcm.

The later recording file with the same name with the existing file will overwrite it.

The recording time supports 1 minute at most. If the recording is not stopped after 1 minute by executing AT+RECF=0, +RECF:FINISH will be prompted to indicate that the recording is stopped.

Format

Type	Command	Response
Execute	AT+RECF=<val>[,<quailty>]<CR>	<CR><LF>OK<CR><LF>

Parameter

<val>	1: stop an audio recording
	1: start an audio recording
<quality>	0: low
	1: medium
	2: high
	3: best (default)

The quality parameter is valid only when the format is set to AMR by the recmode command, and the others are invalid.

Example

```
AT+RECF=1                                                Start an audio recording.
OK
AT+RECF=0                                                To stop an audio recording.
OK
AT+RECF=1                                                The recording times out.
OK

+RECF: FINISH
```

27.3 AT+RECP – Playing the Record File

To play the record file.

The record file can be played in spare time. +RECP:FINISH is prompted after the playback of the record file completes.

Format

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

Parameter

<sta> Audio play switch
0: stop playing
1: start to play

<file_name> audio record file saved in the module, supports playing 16 wav files at most.

Example

```
AT+RECP=1,001.wav                                        Play the record file 001.wav.
OK

+RECP:FINISH
AT+RECP=0                                                Stop playing the audio record.
OK
```

27.4 AT+RECFG – Playing Settings of Recording File

Playing settings of recording file.

Format

Type	Command	Response
Set	AT+RECFG=<type>,<value><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+RECFG?<CR>	<CR><LF>+RECFG: <type>,<value> <CR><LF>OK<CR><LF>

Parameter

<type>   Parameter type
0: unterval of multi-file playing
1 - 99: reserved

<value>   When type=0, the corresponding value is interval, ranging from 0 to 10, unit: s. It is 0 by default.

Example

```
AT+RECFG=0, 5                                       Set the file playing interval to 5s.
OK
AT+RECFG?                                           Query the value of the playing setting.
+RECFG: 0,5
OK
```

27.5 AT+AUDPLAY – Playing Audio

To play an audio file.

Format

Type	Command	Response
Execute	AT+AUDPLAY=<type>,<file_name0>[,<file_name1>…,<file_nameX>],<mode><CR>	<CR><LF>OK<CR><LF> <CR><LF> +AUDPLAY:PLAY SUCCESSED<CR><LF>

	Or <CR><LF>OK<CR><LF> <CR><LF>+AUDPLAY: PLAY STOP Or <CR><LF>ERROR<CR><LF> <CR><LF>+AUDPLAY: PLAY FAILED<CR><LF> Or <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<type> Playing type
0: stop playing
1: play WAV
2: play PCM (the corresponding sampling rate is set using the AUDPCMRATE command.)
3: play AMR (TBD)

<file_name0~X> File name, at most 50 characters. Multi-file file (16 at most) playing is supported.

<mode> Playing mode
0: Play out of a call. You need to stop the audio playing before answering an incoming call.
1: play during a call and the other side can hear the audio played (not supported).

Example

```
AT+FSWF="test.amr",1,1024,10000   Write the test.amr file to user disk. The data content comply
>                                 with the AMR filr format. For example, #!AMR...

OK                                Play the test.arm file.
AT+AUDPLAY=3,"test.amr",0         +AUDPLAY: PLAY SUCCESSED is prompted after the playing
OK                                completes.

+AUDPLAY: PLAY SUCCESSED
AT+AUDPLAY=2,"test1.amr",0        The test1.amr file does not exist, file content is incorrect.
ERROR

AT+AUDPLAY=0                      Stop playing.
OK

+AUDPLAY: PLAY STOP
AT+AUDPLAY=0                      ERROR is returned because no audio is played.
ERROR
```

27.6 AT+AUDPAUSE – Pausing Audio Playing

To pause the audio playing.

Format

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

Parameter

N/A.

Example

```
AT+AUDPLAY=2,"i.pcm",0
OK

AT+AUDPAUSE                           Execute this command to pausing playing when playing a video.
OK

AT+AUDRESUME
OK

+AUDPLAY: PLAY SUCCESSED
```

27.7 AT+AUDRESUME – Resuming Audio Playing

To resume the audio playing.

Format

Type	Command	Response
Execute	AT+AUDRESUME<CR>	<CR><LF>OK<CR><LF> <CR><LF>+AUDPLAY: PLAY SUCCESSED Or <CR><LF>ERROR<CR><LF>

Parameter

N/A.

Example

```
AT+AUDPLAY=2,"i.pcm",0
OK

AT+AUDPAUSE
OK

AT+AUDRESUME                       Execute this command to resume playing after a pausing
OK                                 command.

+AUDPLAY: PLAY SUCCESSED
```

27.8 AT+AUDCFG – Settings of Audio Playing

Settings of the audio playing.

Format

Type	Command	Response
Set	AT+AUDCFG=<type>,<value><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+AUDCFG?<CR>	<CR><LF>+AUDCFG: <type>,<value> <CR><LF>OK<CR><LF>

Parameter

<type>	Parameter type 0: unterval of multi-file playing 1 - 99: reserved
<value>	When type =0, the corresponding value is interval, ranging from 0 to 10, unit: s. It is 0 by default.

Example

```
AT+AUDCFG=0, 5                                         Set the file playing interval to 5s.
OK
```

```
AT+AUDCFG?                                              Query the value of the playing setting.
+AUDCFG: 0,5
OK
```

27.9 AT+AUDPCMRATE – Setting the Sampling Rate of Playing the PCM Format Audio

To set the sampling rate of playing the PCM format audio

Format

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

Parameter

<rate>	0: the sampling rate is 8 kHz, 16-bit data 1: the sampling rate is 16 kHz, 16-bit data
<value>	When type=0, the corresponding value is interval, ranging from 0 to 10, unit: s. It is 0 by default. When type =0, the corresponding value is interval, ranging from 0 to 10, unit: s. It is 0 by default.

Example

```
AT+AUDPCMRATE=0                     Set the sampling rate to 8 kHz, 16-bit data
OK
AT+AUDPCMRATE?                      Query the sampling rate.
+AUDPCMRATE: 0
OK
```

27.10 AT+MICL – Setting the Microphone Gain

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

Format

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

Parameter

<level>	Microphone gain level, ranging from 0 to 9.

Example

```
AT+MICL=3                                         Set the MIC gain level to 3.
OK

AT+MICL?                                          Query the current MIC gain level.
+MICL: 3
OK

AT+MICL=?                                         Query the range of the MIC gain level.
+MICL: (0-9)
OK
```

28 SIM Card Related Commands

28.1 AT+SIMCROSS－Switching SIM

To switch SIM. Card slot 1 is used by default at the first startup.

Currently N58 only supports dual cards single standby. If only one SIM card is used (ensure that it is valid) and the issue of failing to register network occurs, it is recommended to run the +SIMCROSS? command to query whether the correct SIM card slot is selected. Try to use this executed command to switch the SIM card slot if it an incorrect card slot is selected.

When only one SIM card is used, it is recommended to run the NWDSIMCFG command to query which of the card slot is valid before inserting the SIM card. Otherwise, the module may fail to register to the network due to wrong location of the SIM card.

When switching from the currently used card slot to another card slot, the command setting takes effect after the module restarts.

Format

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

Parameter

<sim_id>	SIM card identification 1: SIM 1 (default setting upon first boot) 2: SIM 2

Example

```
AT+SIMCROSS=1                               Switch to SIM 1. This setting is valid after
OK                                          restart.
AT+SIMCROSS=?                               Query the range of SIM card selection.
+SIMCROSS: (1-2)
OK
AT+SIMCROSS?                                Query the current selected SIM card.
+SIMCROSS: 1
OK
```

29 Other Commands

29.1 AT$MYPOWEROFF – Powering off the Module

To power off the module.

Format

Type	Command	Response
Execute	AT$MYPOWEROFF<CR>	<CR><LF>OK<CR><LF>

Parameter

N/A.

Example

```
AT$MYPOWEROFF                                        Power off the module.
OK
```

29.2 AT$MYSOCKETLED – Controlling the Indicator of Socket Status

To control the indicator of socket status.

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

Format

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

Parameter

<ONOFF>	Integer type, phone activity status 0: the LED status when the socket is not connected. 1: the LED status when the socket is connected.

Example

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

29.3 AT$MYGMR – Obtaining the Firmware and Hardware Information of the Module

To obtain the firmware and hardware information of the module.

Format

Type	Command	Response
Execute	AT$MYGMR<CR>	<CR><LF><module_manufacture> <CR><LF><module_model> <CR><LF><firmware_version> <CR><LF><firmware_release_date> <CR><LF><module_hardware_version> <CR><LF><module_hardware_release_date> <CR><LF>OK<CR><LF>

Parameter

<module_manufacture>	Manufacture code, 4 bytes in ASCII format.
<module_model>	Module mode, 8 bytes in ASCII format.
<firmware_version>	Firmware version, 4 bytes in ASCII format.
<firmware_release_date>	Firmware release date, DDMMYY, 6 bytes in ASCII format.
<module_hardware_version>	Hardware version, 4 bytes in ASCII format.

<module_hardware_release_date>   Hardware release date

Example

```
AT$MYGMR
NEO6
N58
V001
121019
V1.0
020919
OK
```

29.4 AT$MYCCID – Obtaining the CCID of the SIM Card

To obtain the circuit card identifier (CCID) of the SIM card.

Format

Type	Command	Response
Execute	AT$MYCCID<CR>	<CR><LF>$MYCCID: <SIM_CCID> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<SIM_CCID>   String type, CCID of the SIM card.

Example

```
AT$MYCCID
$MYCCID: "89860112965403839541"
OK
AT$MYCCID                                    Error is returned because no SIM card is inserted.
ERROR
```

29.5 AT$MYTYPE – Querying the Type of the Module

To query the type of the module.

Format

Type	Command	Response
Query	AT$MYTYPE?<CR>	<CR><LF>$MYTYPE: <mode>,<network_type>, <extended_feature> <CR><LF>OK<CR><LF>

Parameter

<mode> Work mode. Extensible bytes are indicated by bit.
1 at bit 0: supporting transparent mode
1 at bit 1: supporting non-transparent mode

<network_type> Network type. Extensible bytes are indicated by bit.
1 at bit 0: GSM network supports GPRS
1 at bit 1: WCDMA network
1 at bit 2: TD-SCDMA network
1 at bit 3: CDMA 2000
1 at bit 4: CDMA EVDO
1 at bit 5: LTE
1 at bit 6: PSTN
1 at bit 7: extending one byte

<extended_feature> Extensible bytes are indicated by bit.
1 at bit 0: supporting GPS positioning function
1 at bit 1: supporting BeiDou positioning function
1 at bit 7: extending one byte

Example

```
AT$MYTYPE?
$MYTYPE: 03,3F,00
OK
```

29.6 AT$MYMODEM – Querying the Type of the Module

To query the type of the module.

Format

Type	Command	Response
Query	AT$MYMODEM?<CR>	<CR><LF>$MYMODEM: <mode>,<network_type>

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

Parameter

<mode>	Work mode. Extensible bytes are indicated by bit. 1 at bit 0: supporting transparent mode 1 at bit 1: supporting non-transparent mode
<network_type>	Extensible bytes are indicated by bit, 16-bit. 1 at bit 0: GPRS 1 at bit 1: CDMA network 1 at bit 2: TD-SCDMA network 1 at bit 3: WCDMA 1 at bit 4: CDMA EVDO 1 at bit 5: TDD-LTE 1 at bit 6: FDD-LTE Bits 8 to 15: Reserved

Example

```
AT$MYMODEM?
$MYMODEM: 03,7F00
OK
```

29.7 AT$MYBCCH – Locking BCCH Channel

To lock BCCH channel.

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

Querying of the channel information by AT$MYBCCH? is not supported currently.

Format

Type	Command	Response
Execute	AT$MYBCCH=<mode>[,<bcch1>],[<bcch2>],[<bcch3>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>$MYBCCH: +BA(num): <CR><LF><bcch1>,<mcc1>,<mnc1>,<lac1>,<cell-id1> <CR><LF><bcch2>,<mcc2>,<mnc2>,<lac2>,<cell-id2>

		<CR><LF><bcch3>,<mcc3>,<mnc3>,<lac3>,<cell-id3> ... <CR><LF>OK<CR><LF> Or <CR><LF>$MYBCCH: UNLOCKED <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT$MYBCCH?<CR>	<CR><LF>$MYBCCH: <bcch1>,<mcc1>,<mnc1>,<lac1>,<cell-id1> <CR><LF>OK<CR><LF>
Test	AT$MYBCCH=?<CR>	<CR><LF>$MYBCCH: <mode list>,<bcch1>,…<bcch3> <CR><LF>OK<CR><LF>

Parameter

<mode> Integer. The settings of locked channel can be saved after the module is powered off only by running AT$MYBCCH=1,XX. If the BCCH locked does not exist or its signal is weak, the module cannot register network.
0: unlock
1: lock the BCCH cell
2: list IDs of seven BCCH channels that have the strongest signals at current place.

<bcch>	Channel ID
<num>	The number of BCCH channels in the BA list (7 at most)
<mcc>	Mobile country code
<mnc>	Mobile network number
<lac>	Location ID, four-byte characters in hexadecimal format
<cell-id>	Cell ID, hexadecimal

Example

```
AT$MYBCCH=2
$MYBCCH: +BA(7):
120, 460, 1, 2543, A85D
734, 460, 1, 2543, AB12
118, 460, 1, 2543, A8AD
115, 460, 1, 2543, A9A7
124, 460, 1, 2543, A85F
731, 460, 1, 2543, B046
123, 460, 1, 2543, A8A5
OK
AT$MYBCCH=1,120
OK
AT$MYBCCH=?
```

```
$MYBCCH: (0,1,2),120,0,0
OK
AT$MYBCCH?
$MYBCCH: 734,460,1,2543,AB12
OK

AT$MYBCCH=0
OK
```

29.8 AT$MYBAND – Locking GSM Band

To lock to the specified GSM frequency.

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

Format

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

Parameter

<band> Band to be locked. The setting of this parameter will be saved after the module is powered off.
0: AUTO
1: GSM_EGSM_900
2: GSM_DCS_1800

Example

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

29.9 AT$MYLACID – Querying Current LAC and CELL_ID

To query current LAC and CELL_ID.

Format

Type	Command	Response
Execute	AT$MYLACID<CR>	<CR><LF>ERROR<CR><LF> Or <CR><LF>$MYLACID: <LAC>,<CELL_ID> <CR><LF>OK<CR><LF>

Parameter

<LAC>	Location Area Code, hexadecimal
<CELL_ID>	CELL_ID, hexadecimal

Example

```
AT$MYLACID                                               Query the current location of the user.
$MYLACID: 2543, a85d
OK
```

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

To query the current channel, RX power and TX power.

Format

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

Parameter

<Channel>	Channel
<Rx power>	RX power, unit dBm
<Tx power>	TX power, unit dBm

Example

```
AT$MYCGED                The current channel is 48, RX power is -75 dBm, and TX power is 25
$MYCGED: 48,-75,25       dBm.
OK
AT$MYCGED                The current channel is 48, RX power is -44 dBm, and no TX power.
$MYCGED: 48,-44,-1
OK
```

29.11 AT$MYSYSINFO – Quering or Locking the Network Mode

To query or lock the network mode.

When the network mode is set to any non-AUTO mode, the module automatically checks the network status.

If the module registers a network, it will determine whether the network is the mode set by the command. The module will switch to the set network if the registered network is not the mode set by the command.

If the module fails to register any network within interval set by $MYNETAUTO, the module will automatically switch to the AUTO mode.

Format

Type	Command	Response
Set	AT$MYSYSINFO=<SysMode><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Execute	AT$MYSYSINFO<CR	<CR><LF>$MYSYSINFO: <Sys_Mode>,<mnc> <CR><LF>OK<CR><LF>
Test	AT$MYSYSINFO=?<CR>	<CR><LF>$MYSYSINFO: value range of <SysMode> <CR><LF>OK<CR><LF>

Parameter

<SysMode> Network mode
1: AUTO
2: 2G (including GSM and EDGE)
4: 4G (including FDD-LTE and TDD-LTE)

<Sys_Mode> Network mode
0: No service
2: 2G (including GSM and EDGE)
4: 4G (including FDD-LTE, TDD-LTE)

<mnc> Network carrier code
00: Fail to register
01: China Mobile
02: China Unicom
03: China Telecom
04: unknown

Example

```
AT$MYSYSINFO                     The module registered to the 4G network of China Mobile.
$MYSYSINFO: 4,01
OK
AT$MYSYSINFO=1                   Set network mode to auto.
OK

AT$MYSYSINFO=?                   Query the value range of the parameter.
$MYSYSINFO: 1,2,4
OK
```

29.12 AT$MYSYSINFOURC – Switch of Network Mode URC Indication

To control network mode URC indication.

Format

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

Parameter

<ONOFF>   Integer type
0: disable network mode URC (default)
1: enable network mode URC

Example

```
AT$MYSYSINFOURC=1
OK
```

29.13 AT$MYURCSYSINFO – Network Mode URC Indication

To control network mode URC indication.

Format

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

Parameter

<ONOFF>   Integer type
0: disable network mode URC (default)
1: enable network mode URC

Example

```
AT$MYURCSYSINFO=1
OK
```

29.14 $MYURCSYSINFO – URC Notifying Network Mode

To notify the network mode.

Format

Type	Command
URC	<CR><LF>$MYURCSYSINFO: <SysMode>,<mnc><CR><LF>

Parameter

<SysMode>   Network mode
0: No service
2: 2G (including GSM and EDGE)
4: 4G (including FDD-LTE, TDD-LTE)

<mnc>   Network carrier code
00: Fail to register
01: China Mobile
02: China Unicom
03: China Telecom

Example

```
AT$MYSYSINFOURC=1                The module registered the 4G network of China Mobile.
OK

$MYURCSYSINFO: 4,01
```

29.15 AT$MYNETINFO – Setting Network Mode Choices

To set network mode choices.

When the network is set to any non-AUTO mode, the module will automatically check the network status.

- If the module registers a network, it will determine whether the network is the mode set by the command. The module will switch to the set network if the registered network is not the mode set by the command.
- If the module fails to register any network within interval set by $MYNETAUTO, the module will automatically switch to the AUTO mode.

Format

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

Parameter

<mode> Network mode
1: AUTO
2: 2G (including GSM)
4: 4G (including FDD-LTE, TDD-LTE)

Example

```
AT$MYNETINFO=1
OK
AT$MYNETINFO?
$MYNETINFO: 1
OK
AT$MYNETINFO=?
$MYNETINFO: 1,2,4
OK
```

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

To control that the network mode is configured to AUTO automatically once the module is powered on.

V003 and the later version support the setting for CYCLE.

Format

Type	Command	Response
Set	AT$MYNETAUTO=<ONOFF>[,<CYCLE>]<CR	<CR><LF>OK<CR><LF>

	>	Or <CR><LF>ERROR<CR><LF>
Query	AT$MYNETAUTO?<CR>	<CR><LF>+GTUMODE: <ONOFF> <CR><LF>OK<CR><LF>

Parameter

<ONOFF> 0: Disable the default AUTO network modes during startup
1: Enable the default AUTO network modes during startup

<CYCLE> the period that the module restores to auto mode after the network mode is locked and the module failed to register with the network. This value ranges 3 to 1440 minutes, 3 minutes by default.

Example

```
AT$MYNETAUTO=0
OK
AT$MYNETAUTO?
$MYNETAUTO: 0,3
OK
```

29.17 AT^SYSINFO – Setting Network Mode Choice

To set network mode choice.

Format

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

Parameter

<srv_status> 0: no service
1: there are limited services
2: there are services
3: there are area-limited services
4: power saving status

<srv_domain>	0: no service 1: CS service 2: PS service 3: PS and CS services 4: EPS service
<roam_status>	0: non roaming status 1: roaming status
<sys_mode>	0: No service 1: AMPS mode 2: CDMA mode 3: GSM mode 4: EVDO mode or TDS mode 5: WCDMA mode 6: GPS mode 7: GSM and WCDMA mode 8: CDMA Hybridmode 9: LTE mode 10: GSM, WCDMA, and LTE mode
<sim_state>	0: invalid SIM card status 1: valid SIM card status 255: the SIM card is inexistent or PIN code is required.
<reserve>	Reserved field
<sys_submode>	System submode 0: no service 1: GSM mode 2: GPRS mode 3: EDGE mode 4: WCDMA mode 5: HSDPA mode 6: HSUPA mode 7: HSUPA and HSDPA modes 8: TD-SCDMA mode 9: TDD_SUBACT 10: FDD_SUBACT

Example

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

29.18 AT+XISP–Selecting Internal or External Protocol Stack

To select internal or external protocol stack.

To use external protocol stack, send AT+XISP=1 to set before using.

Format

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

Parameter

<n>	0: Select internal TCP/IP protocol stack (default) 1: Select external protocol stack

Example

```
AT+XISP=0                  Specify internal protocol stack.
OK
AT+XISP?                   Query which protocol stack is used. Internal protocol stack is used.
+XISP: 0
OK
```

29.19 AT+NRSP – Querying RSRP, RSRQ, SINR of Cells in a Neighbor of a Serving Cell on the LTE Network

To query RSRP, RSRQ, SINR in a neighbor of the LTE cell.

This command is valid only on LTE networks.

Format

Type	Command	Response
Execute	AT+NRSP<CR>	<CR><LF>+NRSP: <rsrp1>,<rsrq1>,<rsrp2>,<rsrq2>,…,<sinr> <CR><LF>OK<CR><LF>

Parameter

<rsrpN>　Reference Signal Received Quality, the unit is 0.1 dBm. It is valid only on LTE networks. N is the number of cells in a neighbor of the serving cell: <rsrp1>,<rsrq1>,<rsrp2>,<rsrq2>,…, <rsrpN>,<rsrqN>.

<sinr>　Signal-to-Interference-Plus-Noise Ratio, unit 0.1 dB, valid on an LTE network/

Example

```
AT+NRSP                                                               Read command
+NRSP: -920,-75,-930,-65,-870,-115,-780,-65,-880,-175,-990,-135,107
OK
```

29.20 AT+NETMSG – Querying Network Registration Information

To query network registration information

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

A predefined null message is returned when the module is not successfully registered to the network. If the network exception is that the Internet is not registered during the switching of network mode, the information returned at this time is also completely empty.

<LAC>,<BSIC> are compound query items, displaying the network location information of the current module.

Format

Type	Command	Response
Execute	AT+NETMSG<CR>	<CR><LF>+NETMSG: <MCC+MNC>, [<LAC>/<TAC>], [<CELL_ID>], [<BSIC><Phy_cellid>], <BAND>, <ARFCN>, <RX_dBm>, <TX_dBm>, <net_mode> <CR><LF>OK<CR><LF>

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

Parameter

<MCC+MNC>	MCC: Mobile Country Code, decimal MNC: Mobile Network Code, decimal
[<LAC>/<TAC>]	Location Area Code, hexadecimal SID on CDMA1X networks TAC on LTE networks
[<CELL_ID>]	Cell ID, on other network modes, hexadecimal
[<BSIC>/<Phy_cellid>]	Decimal BSIC on GSM network, hexadecimal Phy_cellid on LTE networks, 0 on other networks
<BAND>	operating band GSM 900 DCS1800 PCS 1900 LTE BAND 1 … LTE BAND 43
<ARFCN>	Absolute radio-frequency channel number
<RX dBm>	RX power, unit: dBm (199 indicates invalid)
<TX dBm>	TX power, unit: dBm (199 indicates invalid)
<net_mode>	Mode of network registered with NONE GSM GPRS TDD LTE FDD LTE

Example

```
AT+NETMSG                                            Query the details used to register the
+NETMSG: "460+00", 286F, 00000088, 95, LTE BAND 40,  network.
38950, -46, 199, "TDD LTE"
OK
AT+NETMSG                                            Query the details used to register the
+NETMSG: "460+00", 286F, 00000FCB, 26, GSM 900, 20,  network.
32, -46, "GPRS"
OK
AT+NETMSG                                            The module has not been registered with
+NETMSG: "0", 0, 0, 0, 0, 0, 0, 0, "NONE"            any network or the network encountered
OK                                                   abnormalities.
                                                     No SIM card is inserted.
AT+NETMSG
ERROR
```

29.21 AT+NETDMSG – Querying Network Registration Information

To query the current network registration information.

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

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

RX power, RSRQ, and SINR are valid only on LTE networks.

Format

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

Parameter

<MCC+MNC>	MCC: Mobile Country Code, decimal MNC: Mobile Network Code, decimal
[<LAC>/<TAC>]	LAC on CDMA1X networks TAC on LTE networks Location Area Code on other network modes, hexadecimal
[<CELL_ID>]	NID on CDMA1X networks Cell ID, on other network modes, hexadecimal
[<BSIC>/<Phy_cellid>]	Base Station Identity Code/Physical Cell ID of LTE networks Net_mode: NONE GSM GPRS TDD LTE FDD LTE
<net_mode>	Network mode NONE GSM GPRS

	TDD LTE FDD LTE
<BAND>	operating band GSM 900 GSM 1800 GSM 1900 LTE BAND 1 … LTE BAND 43
<ARFCN>	Network Bands
<RX dBm>	RX power, unit: dBm (199 indicates invalid)
<TX dBm>	TX power, unit: dBm (199 indicates invalid)
<SID>	System Identity Number on a CDMA1X network
<NID>	Network Identity Number on a CDMA1X network
<BID>	BID on a CDMA1X network
<RSRP>	Reference Signal Received Power, unit 0.1 dBm, valid on an LTE network
<RSRQ>	Reference Signal Received Quality, unit 0.1 dB, valid on an LTE network
<SINR>	Signal-to-Interference-Plus-Noise Ratio, unit 0.1 dB, valid on an LTE network

Example

AT+NETDMSG +NETDMSG: "460+00", 0x286F, 0x00000088, 95, "TDD LTE", LTE BAND 40, 38950, -49, -8, 0x0, 0x0, 0x0, -730, -55, 108 OK	Query the network registration information on an LTE network.
AT+NETDMSG +NETDMSG: "460+00", 0x286F, 0x00000FCB, 26, "GPRS", GSM 900, 20, 26, -46, 0x0, 0x0, 0x0, 0, 0, 0 OK	Query the network registration information on a non-LTE 3GPP network.
AT+NETDMSG +NETDMSG: "0", 0, 0, 0, "NONE", 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 OK	The module has not been registered with any network or the network encountered abnormalities.

29.22 AT+NEOFOTA – FOTA Command

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

Do not power down or restart the module during upgrade.

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

If PDP context is not activated, +NWFOTA: GPRS DISCONNECTION is prompted.

Format

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

Parameter

<status>   Upgrade status
0: no valid OTA packages
1: download the upgrade package successfully (download to RAM)
2: abnormal download network
3: start to perform a local upgrade.
4: the local upgrade is performed successfully.
5: fail to perform the local upgrade.
6: insufficient download space
7: fails to download upgrade package
8: fails to verify upgrade package

Example

```
AT+NEOFOTA=115.29.212.25/,80
                                 Start to upgrade.
OK

                                 The upgrade package is downloaded successfully.
+NEOFOTA: 1
                                 Start to upgrade.
+NEOFOTA: 3
                                 After the upgrade is successful, the module restarts
OK
                                 automatically.

+NEOFOTA: 4
                                 Upgrade successfully.
                                 AT+NEOFOTA=115.29.212.25/,80
AT+NEOFOTA=115.29.212.25/,80
                                 OK
OK

+NEOFOTA: 0
                                 +NEOFOTA: 0
```

29.23 AT+NEOFOTAURC－FOTA Status Report

To control the status report during an FOTA upgrade.

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

Format

Type	Command	Response
Execute	AT+NEOFOTAURC=<result><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<result> Switch of status report

0: disable status report. Only upgrade result is reported. The upgrade results including the following:

Upgrade successfully

Failed to upgrade

No upgrade package

Failed to download

1: enable status report (default). All states are reported.

Example

```
AT+NEOFOTAURC=1                 All states are reported.
OK

AT+NEOFOTA=115.29.212.25/,80
OK

+NEOFOTA: 1                     The upgrade package is downloaded successfully.
+NEOFOTA: 3                     Start to upgrade.
OK                              After the upgrade is successful, the module restarts
                                automatically.
+NEOFOTA: 4
                                Upgrade successfully.
AT+NEOFOTAURC=0                 Status reports during the disable process.
OK

AT+NEOFOTA=115.29.212.25/,80
OK

+NEOFOTA: 4                     The module is upgraded successfully.
```

29.24 +NWURCFOTA – URC Notifying FOTA Upgrade Status

To notify the FOTA upgrade status.

Format

Type	Response
Unsolicited result code	<CR><LF>+NWURCFOTA: <status><CR><LF>

Parameter

<status> Upgrade status
0: no valid OTA packages
1: start to download the OTA package
2: the OTA package is downloaded successfully.
3: failed to download the OTA package.
4: start to perform a local upgrade.
5: the local upgrade is performed successfully.
6: fail to perform the local upgrade.

Example

```
+NWURCFOTA: 1                             Start to download the OTA package.

+NWURCFOTA: 2                             The upgrade package is downloaded successfully.
                                          Start to perform the local upgrade.
+NWURCFOTA: 4                             After the upgrade is successful, the module restarts
                                          automatically.
+NWURCFOTA: 5
                                          no upgrade packages.
+NWURCFOTA: 0
```

29.25 AT+NWFOTA – Performing an FOTA Upgrade

To upgrade the module firmware over the air.

- Do not power down or restart the module during the upgrade process.
- After the module upgrade process is completed the module will restart automatically. After the

restart, issue AT\r\n to the module for baud rate self-adaptation; only after the baud rate self-adaptation is performed, can **+NWFOTA: 5** notifying a successful upgrade be prompted.

- If PPP is not activated, +NWFOTA: NET DISCONNECTION is prompted.

Format

Type	Command	Response
Execute	AT+NWFOTA=<server>,<port>[,type]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<server> FOTA address of the HTTP server, string type, maximum length is 1024 bytes.

<port> Port number of the FOTA server, integer type.

〈type〉 Server protocol type, integer type.
0: HTTP protocol (default).
1: HTTPS protocol. (not supported currently)

Example

```
AT+NWFOTA=”fota.neoway.com”,80
OK

+NWURCFOTA: 1                                        Start to download the OTA package.

+NWURCFOTA: 2                                        The upgrade package is downloaded
                                                     successfully.
+NWURCFOTA: 4                                        Start to perform the local upgrade.

+NWURCFOTA: 5                                        After the upgrade is successful, the
                                                     module restarts automatically.
```

```
AT+NWFOTA=”fota.3rdparty.com/http/test.bin”,80,0     Perform an upgrade via the third-party
OK                                                   HTTP FOTA server

+NWURCFOTA: 1                                        Start to download an upgrade package

+NWURCFOTA: 2                                        The upgrade package is downloaded
                                                     successfully.
+NWURCFOTA: 4
                                                     Start to perform a local upgrade.
                                                     (after the upgrade process is completed,
                                                     the module will restarted automatically.
+NWURCFOTA: 5                                        And the following URC is prompted after
                                                     the module is self-adapted).
                                                     The module is upgraded successfully.
```

29.26 AT+READADC – Reading ADC Value

To read the value from pins corresponding to the three ADC channels.

Refer to the pin definitions listed in *Neoway_N58_Hardware_User_Guide*.

Format

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

Parameter

<channel>	1 2-pin88 2 4-VBAT
<value>	The corresponding ADC value.

Example

```
AT+READADC=1      To read the value from the pin corresponding to ADC 1 is unsupported.
ERROR

AT+READADC=2      Read the value from the pin corresponding to ADC 2.
+READADC:2,1810
OK
AT+READADC=3      Read ADC value of the VBAT channel
+READADC:3,3840
OK
```

29.27 AT+SIMHOTSWAP – Setting the Hotswapping Function

Enable the hot-swapping function for the SIM card.

This command function is valid only when the hardware supports hot-swapping. The setting by this command takes effect immediately after the module is powered off.

Format

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

Parameter

<onoff>	0: disable 1: enable

Example

```
AT+SIMHOTSWAP=0                                   Disable the hot-swapping function.
OK
AT+SIMHOTSWAP=1                                   Enable the hot-swapping function.
OK
```

```
AT+SIMHOTSWAP?                                  Query the hot-swapping status.
+SIMHOTSWAP: 1
OK
AT+SIMHOTSWAP=?                                 Query the value range of the parameters.
+SIMHOTSWAP: (0-1)
OK
```

29.28 AT+MYDATAONLY – Disabling CS Services

To disable voice call service.

The setting of this command takes effect after the module is restarted. Since N58 supports VoLTE by default, if you need to disable voice services completely, execute AT+SETVOLTE=0 and AT&W.

Format

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

Parameter

<onoff>	0: ON 1: OFF

Example

```
AT+MYDATAONLY=0                                     Enable voice call service.
OK
AT+MYDATAONLY=1                                     Disable voice call service.
OK
AT+MYDATAONLY?                                      Query the status of voice call service.
+MYDATAONLY: 1
OK
```

29.29 AT+BANDLOCK – Locking to Band

To lock to a frequency band.

The network mode will change as a frequency band is locked. E.g. the network mode will change to LTE ONLY after locking to LTE_B1.

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

If the frequency band currently locked is invalid, registration failure will occur.

Format

Type	Command	Response
Execute	AT+BANDLOCK=<band_string><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+BANDLOCK?<CR><CR>	<CR><LF>+BANDLOCK: <band_string> <CR><LF>OK<CR><LF> Or <CR><LF>+BANDLOCK:NONE <CR><LF>OK<CR><LF>
Test	AT+BANDLOCK=?<CR>	<CR><LF>+BANDLOCK: <band_string_list> <CR><LF>OK<CR><LF>

Parameter

<band_string>	Band in string type The value can be any band that the hardware supports. Issue a query command before locking to a frequency band to check the band supported.
<band_string_list>	List of frequency band supported, which is for reference only. Refer to the product specifications if required.

Example

```
AT+BANDLOCK=?                                                     Query the frequency bands
+BANDLOCK:                                                        supported.
GSM_900,GSM_850,GSM_1800,GSM_1900,LTE_B1,LTE_B3,LTE_B5,LTE_B7,
LTE_B8,LTE_B20,LTE_B28.BLTE_B38,LTE_B39,LTE_B40,LTE_B41,AUTO
OK
AT+BANDLOCK=LTE_B1                                                Lock to LTE_B1.
OK
AT+BANDLOCK?                                                      Query the frequency band locked
+BANDLOCK: LTE_B1                                                 to.
OK
AT+BANDLOCK=AUTO                                                  Restore to auto frequency band
OK                                                                mode.
```

29.30 AT+MYCELLINFO – Obtaining Information of Neighbor Cells

To obtain the information of neighbor cells available for the module.

If the module is installed an SIM card, it might disconnect to the network since the module performs full bands scanning after executing this command.

Only LTE cell scanning is supported.

Format

Type	Command	Response
Execute	AT+MYCELLINFO<CR>	+MYCELLINFO: {<NcellNum>,(<NcellIndex>,<NcellMode>,<Nmcc>,<Nmnc>,<Nlac>,<Narfcn>,<Npci>,<NLTE_RSRP>,<NLTE_RSRQ>,<NrxLevl>)} <CR><LF>OK<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF>

Parameter

<NcellNum>	The number of neighbor cells
<NcellIndex>	The index of neighbor cells
<Smcc/Nmcc>	Mobile country code CDMA –SID, system ID
<Nmnc>	Mobile network code
<Nlac>	location area code LTE - tac, track area code
<Narfcn>	Absolute radio frequency channel code
<Npci>	CDMA/HDR - pn LTE – pci, physic cell identify
<NLTE_RSRP>	Optional parameter. 10 times the actual RSRP value in LTE mode, unit: dBm. The actual RSRP value ranges from -44 to -140.
<NLTE_RSRQ>	Optional parameter. 10 times the actual RSRP value in LTE mode, unit: dBm. The actual RSRP value ranges from -20.0 to -3.0.
<NrxLevl>	signal strength.
<LTE>	Signal strength, unit: dB

Example

AT+MYCELLINFO +MYCELLINFO: {5, (0, 460, 01, 9547, 3765, 342, -760, -130, -91), (1, 460, 01, 9547, 1650, 344, -860, -110, -86), (2, 460, 00, 10351, 38544, 90, -870, -90, -79), (3, 460, 00, 10351, 37900, 29, -920, -110, -79), (4, 460, 00, 10351, 40936, 29, -940, -80, -75)} OK	Obtain the information of all LTE cells.

Description of the return value format:
Since there is too much information about all cells, the cell information results are output in a JSON-like format.

Command operation caution
This command is asynchronous so no code will be returned if the next command is sent when execution of the preceding command has not finish.
Do not perform other commands before the module returns any value to this command. Otherwise, the module might encounter network issues.
Restart the module if the module encounters any abnormality due to incorrect operations.

Command data accuracy caution
This command is only valid for stationary targets or slow-moving targets and the accuracy of the data is valid when the target has no major position changes.

29.31 AT+NBANDLOCK – Locking Multiple Frequency Bands

Lock one and more frequency bands by band mask.

After frequency bands are locked through this AT command, the network mode changes. For example, the network mode becomes LTE ONLY if you lock the frequency band to LTE B1. LTE and GSM cannot be locked simultaneously.

The setting by this command takes effect immediately and it is saved after the module is powered off.

If the frequency band currently locked is invalid, registration failure will occur.

This command can lock 5 frequencies at most at the same time.

For the band settings, see Appendix C Support Band List.

Format

Type	Command	Response
Set	AT+NBANDLOCK=<mode>[,<LTE_Bands>][,<GSM_Bands>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>+NBANDLOCK: <mode>,<SupportBandlist> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+NBANDLOCK: <err><CR><LF>
Query	AT+NBANDLOCK?<CR>	<CR><LF>+NBANDLOCK: <mode>,<Bandlist> <CR><LF>+NBANDLOCK: <mode>,<Bandlist> <CR><LF><CR><LF>OK<CR><LF> Or <CR><LF>+NBANDLOCK: NONE <CR><LF><CR><LF>OK<CR><LF>
Test	AT+NBANDLOCK=?<CR>	<CR><LF>+NBANDLOCK: (listofsupport<mode>s),(list of support <LTE_BAND>s),(list of support <GSM_BAND>s) <CR><LF><CR><LF>OK<CR><LF>

Parameter

<mode> Network mode

0: unlock all

1: lock all band combinations under the LTE network

2: lock all band combinations under the GSM network

<LTE_Bands> frequency bands supported by LTE protocol, displayed and input in hexadecimal format.

<GSM_Bands> frequency bands supported by GSM protocol, displayed and input in hexadecimal format.

Each bit indicates a band according to the setting of <mode>. 0 indicates disabled and 1 indicates enabled. For details about the match between bit and band, see the Appendix.

Example

```
AT+NBANDLOCK=1,1              Lock LTE BAND1.
OK
AT+NBANDLOCK=2,1              Lock GSM BAND1.
OK
```

```
AT+NBANDLOCK?                Query the status of band locked. LTE band1 is locked.
+NBANDLOCK: 1,0x1            GSM bands are not locked.
+NBANDLOCK: 2,0x0

OK
AT+NBANDLOCK=?               Query frequency band locked. Convert 0x1e0080800d5 to
+NBANDLOCK: 1,0x1e0080800d5  11100000000001000000010000000000011010101; it indicates that the
+NBANDLOCK: 2,0xec           module supports LTE_B1, LTE_B3, LTE_B5, LTE_B7, LTE_B8, LTE_B20,
                             LTE_B28, BLTE_B38, LTE_B39, LTE_B40, and LTE_B41.
OK                           Convert 0xec into 11101100; it indicates that the module supports
                             GSM_900P,GSM_900E,GSM_850,GSM_1800,GSM1900.
```

29.32 AT+NFREQLOCK – Locking the Specified Frequency

To lock a combination of frequencies.

When no band is locked, this command can lock any available frequency. When any band is locked, this command is only used to lock the available frequencies of the locked band. As the actual network environments are different, the correction and verification for the input frequency information is unsupported.

When locking a band, ERROR will be returned if you use this command to lock the frequencies that do not belong to the locked band.

When no band is locked, depending on the setting of the frequency, the network will be registered to the band corresponding to the best frequency.

The command supports to lock 9 frequencies at most. The setting is valid immediately after setting, and is saved after the module is powered off.

Format

Type	Command	Response
Execute	AT+NFREQLOCK=<mode>[,<frequency1>,<frequency2>,<frequency3>,<frequency4>,<frequency5>,<frequency6>,<frequency7>,<frequency8>,<frequency9>]<CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF>
Query	AT+NFREQLOCK?<CR>	<CR><LF>+NFREQLOCK: <mode>,<frequency1>,<frequency2>... … <CR><LF>OK<CR><LF>

	Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+CME ERROR: <err><CR><LF>

Parameter

<mode>	Network mode 0: unlock all 1: LTE (including FDD\TDD)
<frequency>	at most 9 frequencies can be input 1 - 65535: frequency range

Example

```
AT+NFREQLOCK=1,38950,38950      Lock the LTE frequency 38950.
OK
AT+NFREQLOCK?                   Query the frequency locking status.
+NFREQLOCK: 1,38950
OK
AT+NFREQLOCK=0                  Unlock all frequencies locked.
OK

AT+NFREQLOCK?
+NFREQLOCK: 1,0
OK
AT+NFREQLOCK=1, 38950,38400     Lock the specific frequencies 38400 and 38950 of the same band
OK                              in LTE network.
AT+NFREQLOCK?
+NFREQLOCK: 1,38400,38950
OK
```

29.33 AT+NWCUSTPIN – Setting Specified GPIO

To set the specified GPIO.

When <type> is set to 1 or 2, the command is valid immediately after setting. The setting by this command is saved after the module is powered off and it is valid after restart.

When <type> is set to 3, the command is valid immediately. The setting by this command is not saved after the module is powered off.

Format

Type	Command	Response
Set	AT+NWCUSTPIN =<type><CR>	<CR><LF>+NWCUSTPIN:<type><pin><dir><value> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+NWCUSTPIN?<CR>	<CR><LF>+NWCUSTPIN:<type><CR> <CR><LF>OK<CR><LF>
Test	AT+NWCUSTPIN=?<CR>	<CR><LF>+NWCUSTPIN: (range of supported <type>) <CR><LF>OK<CR><LF>

Parameter

<type>	1: pin78-RING, pin80-NWY_NET_LIGHT 2: pin13-RING, pin83-NWY_NET_LIGHT (default)) 3: to set the level of the specified pin. 4: to query the level of the specified pin. 5: to disable GPIO of the specified pin.
<pin>	13, 50, and 83 are supported.
<dir>	Level direction 0: INPUT 1: OUTPUT -1: unconfigured
<value>	Voltage level 0: low level 1: high level -1: unconfigured

Example

```
AT+NWCUSTPIN =1                                  Supports V10 GPIO.
OK
AT+NWCUSTPIN?                                    Query the set values.
+NWCUSTPIN: 1
OK
AT+NWCUSTPIN=3,13,1,1                            Set PIN13 to output high level.
OK
AT+NWCUSTPIN=3,13,1,1                            Set PIN13 to output low level.
OK
AT+NWCUSTPIN=4,13                                Query the level of pin 13.
+NWCUSTPIN: 13,1,1
OK
AT+NWCUSTPIN=5,13                                Disable GPIO of PIN13.
OK
```

```
AT+NWCUSTPIN =?                                 Query values range of this command.
+NWCUSTPIN: (1-5),(13-50-83),(-1-0-1),(-1-0-1)
OK
```

29.34 AT+IPINFO – Querying Socket Connection Information

To query the socket connection information.

Format

Type	Command	Response
Execute	AT+IPINFO<CR>	<CR><LD>+IPINFO: <SocketID>,<LocalIP>,<local_port>,<gate>,<DNS1>,<DNS2>,<type>,<dest_ip>,<dest_port> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>

Parameter

<LocalIP>	Local IP address, character type
<local_port>	Local port (varying with channels)
<gate>	Gateway
<DNS1>	Primary DNS server
<DNS2>	Standby DNS server
<type>	TCP Client/TCP Server/TCP Accept/UDP 0: TCP Client 1: UDP 2: TCP Server 3: UDP Accept
<dest_ip>	Destination IP address
<dest_port>	Destination port number

Example

```
AT+IPINFO                                             Query the connection status on socket 0.
+IPINFO:
0,10.13.70.121,29492,0,202.96.134.33,202.96.128.166,0
,58.60.184.213,12005
OK
AT+IPINFO                                             No socket connection information.
```

```
OK
```

29.35 AT+SETTZ – Setting Clock Offset Time

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

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

Format

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

Parameter

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

Example

```
AT+SETTZ?                                        Query the current offset time.
+SETTZ: +32
OK
AT+SETTZ=+32                                     Query current status.
OK
AT+CCLK?
+CCLK: "80/01/06,00:56:50+32"
OK
```

29.36 AT+NCUSTSWITCH – Switch of Extended Functions

Switch of extended functions. The setting by this command is not saved after the module is powered off.

Execute this command before the TCP, HTTP, or MQTT (if standard MQTT data wake-ups are supported) connection is established.

- The general N58 firmware supports standard TCP and standard HTTP data wake-ups.
- The N58 firmware developed on 1.4 baseline (the firmware version containing "R08") supports standard TCP, standard HTTP, and standard MQTT data wake-ups.

  Executing the AT+GMR command can query the module firmware version.

Format

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

Parameter

<typeX> Extended functions
3: read TCP packets in segment format (read without receiving the segment)
4: configure the wake-up source
Currently, only type 3 and type 4 are supported.

<ModeX> 0: disable TCP Nagle algorithm
1: enable TCP Nagle algorithm (default)

<ModeX> 0: default, read the TCP packet in segment format
1: automatic packet merging

<ModeX>

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

0X0001 only SMS status indication is enabled
0X0002 only CALL status indication is enabled
0X0003 a combination of CALL+SMS status indication is enabled
0X0004 only data status indication is enabled (data transmitted in the internal protocol stack services)
0X0005 a combination of DATA+SMS status indication is enabled

0X0006 a combination of DATA+CALL status indication is enabled
0X0007 a combination of DATA+CALL+SMS status indication is enabled (default)

<TypeX> reserved
<ModeX> reserved

Example

```
AT+NCUSTSWITCH=3,1                                  Set automatic packet merging.
OK
AT+NCUSTSWITCH?
+NCUSTSWITCH: 3,1                                   Query the parameters setting.
OK
AT+NCUSTSWITCH=?
+NCUSTSWITCH: (1-4),(0-7)                           Query the range of the parameters.
OK
AT+NCUSTSWITCH=4,1                                  Set the ringing status; enable the URC of SMS
OK                                                  status.
AT+NCUSTSWITCH?
+NCUSTSWITCH: 3,0                                   Query the settings.
+NCUSTSWITCH: 4,7                                   The URC of SMS status is enabled.
OK
```

29.37 AT+FTPGETF – FTP Upgrade Command

To upgrade the module over the air.

Format

Type	Command	Response
Set	AT+FTPGETF=<ip>,<port>,<mode>,<filename>,<user>,<pwd><CR>	<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>+FTPGETF:<result code><CR><LF>

Parameter

<ip> IP address or domain name of the FTP server.
<port> Port number of the FTP server.
<mode> Mode, fixed to 0.
<filename> File name of the upgrade package on the FTP server.
<user> User name used to log in to the FTP server. Its length cannot be larger than 100 ASCII codes and it cannot contains a comma.

<pwd> Password used to log in to the FTP server. Its length cannot be larger than 100 ASCII codes and it cannot contains a comma.

<result code> LOGIN OK: the module logs in to the FTP server successfully.
FILE END: the file is downloaded successfully.
ERROR PPP: PPP is not activated.
ERROR SOCKET: the module fails to obtain a SOCKET.
ERROR EVENT: the module fails to set SOCKET properties.
ERROR ADDR: the module fails to connect to the FTP IP address.
ERROR DOMAIN: the module fails to connect to the FTP domain name.
ERROR CONNECT: the module fails to connect to the FTP socket.
ERROR LOGIN: the module fails to log into the FTP server.
ERROR FSIZE: the module fails to obtain the file size. ERROR PASV: the module fails to establish a data connection
ERROR PASV: the module fails to establish a data connection through FTP.
ERROR HEADER: the download file fails in the header verification.
ERROR LENGTH: the module fails to check the length of the downloaded file.
ERROR DISCONNECT: the link is disconnected abnormally.
ERROR TIMEOUT: timeout

Example

```
AT+FTPGETF=58.60.184.213,12006,0,111.h,test,test     Start to perform the upgrade.
OK

+FTPGETF: LOGIN OK                                   Log in to the FTP server successfully.
+FTPGETF: RATE 1400/13090716                         URC of the download progress
…                                                    Continuously report the download progress.
+FTPGETF: RATE 13090716/13090716                     URC of the download progress
+FTPGETF: FILE END                                   Downloaded successfully.
AT+FTPGETF =58.60.184.213,12006,0,111.h,test,test
+FTPGETF: ERROR PPP                                  No dial-up connection
AT+FTPGETF =58.60.184.213,12006,0,111.h,test,test    Start to perform the upgrade.
OK

+FTPGETF: LOGIN OK                                   Log in to the FTP server successfully.
+FTPGETF: ERROR CONNECT                              Report the error information.
```

29.38 AT+NWCHANNEL – Setting the Network Activation Channel

To set the channel number.

Format

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

Parameter

<cid> A numeric value, 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 (the minimum value is 1; the specific value range is defined depends on the platform.) is returned by the test form of the command.

Example

```
AT+NWCHANNEL=1                          Set the activation channel to 1.
OK
AT+NWCHANNEL?
+NWCHANNEL: 1                           Query the value set.
OK
AT+NWCHANNEL=?                          Query the value range of the parameters.
+NWCHANNEL: (1-5)
OK
```

- This command is used to specify CID and the corresponding APN and local IP address before network connections at various transport and application layers are established.
- The relations among CID, APN, and the local IP address can be queried through AT+CGDCONT.

29.39 AT+CGACT－Activating/Deactivating PDP Context

To activate or deactivate the specified PDP context(s).

Before executing this command, use the AT+CGDCONT command to set the parameters including <APN> and ensure that the module registers to the network successfully.

Format

Type	Command	Response
Set	AT+CGACT=<state>[,<cid>[,cid]….]<CR>	<CR><LF>+CGACT: <cid>,<state> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CGACT?<CR>	<CR><LF>+CGACT:<cid>,<state> <CR><LF>OK<CR><LF>
Test	AT+CGACT=?<CR>	<CR><LF>+CGACT: (list of supported <state>s) <CR><LF>OK<CR><LF>

Parameter

<state> Integer type
0: deactivated
1: activated

<cid> PDP activation channel, ranging from 1 to 7. A maximum of 5 channels can be activated simultaneously.

Example

```
AT+CGACT=1,1                          Activate the PDP context of the first channel.
+CGACT: 1,1
OK
AT+CGACT?
+CGACT: 1,1                           Query the PDP channel activated.
OK
AT+CGACT=?                            Query the value range of <state>.
+CGACT: (0,1)
OK
```

29.40 AT+CUSD – Sending USSD Data

To send Unstructured Supplementary Service Data (USSD)

Format

Type	Command	Response
Execute	AT+CUSD=[<n>[,<str>[,<dcs>]]]<CR>	<CR><LF>+CUSD: <m>[,<str>,<dcs>]

		<CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF>
Query	AT+CUSD?<CR>	<CR><LF>+CUSD: <m> <CR><LF>OK<CR><LF>
Test	AT+CUSD=?<CR>	<CR><LF>+CUSD: (value range of <n>) <CR><LF>OK<CR><LF>

Parameter

<n> specifies whether to display return codes
0: do not display return codes
1: display return codes
2: Cancel the request (not available for read command)

<str> USSD string. If this parameter is not set, the module does not access network.

<dcs> 3GPP TS 23.038 [25] Cell Broadcast Data Coding Scheme in integer format (default 0).

<m> 0: no further user action required
1: further user action required
2: USSD terminated by network
3: other local client has responded
4: operation not supported
5: network response times out

Example

```
AT+CUSD=1,"*100#",15                                           The operator supports this
+CUSD: 2,                                                      data service.
"04110430043B0430043D0441003A003300350031002C0035003104400020",
72
OK
```

30 UDP Server Function

30.1 AT$UDPLISTEN – Creating UDP Listener as a Server

To create a UDP listener as a server.

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

Format

Type	Command	Response
Set	AT$UDPLISTEN=<port>[,<recv_mode>]<CR>	<CR><LF>$UDPLISTEN: <socket>,OK<CR><LF> <CR><LF>$UDPLISTEN: listening status…<CR><LF> Or <CR><LF>$UDPLISTEN: not listening<CR><LF>
Query	AT$UDPLISTEN?<CR>	<CR><LF>$UDPLISTEN: listening status <CR><LF>OK<CR><LF> Or <CR><LF>$UDPLISTEN: not listening <CR><LF>OK<CR><LF>
Test	AT$UDPLISTEN=?<CR>	<CR><LF>$UDPLISTEN: (range of supported <port>),(range of supported <recv_mode>) <CR><LF>OK<CR><LF>

Parameter

<port>	port number, ranging from 1 to 65535
<recv_mode>	Data receiving mode 0: output the data after receiving it (default) 1: save the data to the buffer after receiving it and send **$IPNETREAD** to read it later

Example

```
AT$UDPLISTEN=6000                          Set a listening socket on port 6000.
$UDPLISTEN: 0,OK                           The server starts to listen.
Or
```

```
$UDPLISTEN: bind error                   Fails to bind
AT$UDPLISTEN=6000                        Set a listening socket on channel 0.
ERROR                                    No PPP connection is set up yet.
AT$UDPLISTEN=6000
                                         A listening socket is set up already.
Listening…

AT$UDPLISTEN=?                           Query the value range of the parameters.
$UDPLISTEN: (1-65535),(0-1)
OK
AT$UDPLISTEN?                            Query the listening status. Here the server is in the
$UDPLISTEN: listening status             listening status.
OK

AT$UDPLISTEN?                            Query the listening status. Here the server is not in
$UDPLISTEN: not listening                the listening status.
OK
```

30.2 AT$CLOSEUDPLISTEN – Closing Listening Socket

To close listening sockets.

Format

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

Parameter

<socket>	Socket ID

Example

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

$CLOSEUDPLISTEN: 0,local link closed
                                              The connections to the server are closed.
$CLOSEUDPCLIENT: 1,remote link closed
```

30.3 AT$CLOSEUDPCLIENT – Closing UDP Remote Socket

To clear a specified UDP client address or all UDP client addresses

Format

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

Parameter

<Socket>	Socket ID

Example

```
AT$CLOSEUDPCLIENT                                   Clear all UDP client addresses.

$CLOSEUDPCLIENT: 1,remote link closed

$CLOSEUDPCLIENT: 2,remote link closed
AT$CLOSEUDPCLIENT=1                                 Clear the address of the client on socket .

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

30.4 $UDPRECV(S) – Notifying UDP Data Received from Clients

To notify UDP data received from clients.

Format

Type	Response
URC	$UDPRECV(S): <socket>,<length>,<data>

Parameter

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

Example

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

30.5 AT$UDPSEND(S) – Sending Data to Clients

To send data to the client

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

For the first time, the server can send data to a client only after receiving the data from the client.

Format

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

Parameter

<socket>	value of **AcceptSocket**, that is, the socket that the client sets up with the module.

<length>   length of the data to be sent, value ranges from **1** to **1024**, unit: byte.
<err>   ERROR
SOCKET ID NOT ACTIVE
DATA LENGTH ERROR

Example

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

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

$UDPSENDS: SOCKET ID NOT ACTIVE
AT$UDPSENDS=0,10                              No UDP connection is set up on socket 0 through
$UDPSEND: ERROR                               listening
AT$UDPSENDS=0,10
>
$UDPSENDS: Error!TimeOut                      Data input times out
AT$UDPSENDS=0,0,5120                          Incorrect dayta length
$UDPSENDS: DATA LENGTH ERROR
```

30.6 AT$UDPCLIENTSTATUS – Querying Client Connection Status

To query the client connection status

Format

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

Parameter

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

Example

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

30.7 AT$IPNETREAD – Reading UDP Data

To read UDP data that the server receives and saves in the buffer

A connection that is set to automatic data reception do not support reading data with this command.

If the <length> is greater than the actual length of the data, the module read the data of the actual length. The buffer for each socket is 10 KB.

Format

Type	Command	Response
Execute	AT$IPNETREAD=<n>,[<len>]<CR>	<CR><LF>$IPURCREAD:<channel>,<len>,<data> <CR><LF>OK<CR><LF> Or <CR><LF>ERROR<CR><LF> Or <CR><LF>$IPURCREAD: ERROR Or <CR><LF>$IPURCREAD: SOCKET ID OPEN FAILED

Parameter

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

Example

```
$IPURCREAD: 0                 Socket 0 receives data.
AT$IPNETREAD=0,5              Read 5-byte data.
```

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

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

A Error Codes

A.1 General Error Codes

Code(AT+CMEE=1)	Text(AT+CMEE=2)
0	PHONE FAILURE
1	NO CONNECTION TO PHONE
2	PHONE-ADAPTOR LINK RESERVED
3	OPERATION NOT ALLOWED
4	OPERATION NOT SUPPORTED
5	PH-SIM PIN REQUIRED
6	PH-FSIM PIN REQUIRED
7	PH-FSIM PUK REQUIRED
10	SIM NOT INSERTED (SEE NOTE 1)
11	SIM PIN REQUIRED
12	SIM PUK REQUIRED
13	SIM FAILURE (SEE NOTE 1)
14	SIM BUSY (SEE NOTE 1)
15	SIM WRONG (SEE NOTE 1)
16	INCORRECT PASSWORD
17	SIM PIN2 REQUIRED
18	SIM PUK2 REQUIRED
20	MEMORY FULL
21	INVALID INDEX
22	NOT FOUND
23	MEMORY FAILURE
24	TEXT STRING TOO LONG
25	INVALID CHARACTERS IN TEXT STRING
26	DIAL STRING TOO LONG
27	INVALID CHARACTERS IN DIAL STRING

30	NO NETWORK SERVICE
31	NETWORK TIMEOUT
32	NETWORK NOT ALLOWED - EMERGENCY CALLS ONLY
40	NETWORK PERSONALIZATION PIN REQUIRED
41	NETWORK PERSONALIZATION PUK REQUIRED
42	NETWORK SUBSET PERSONALIZATION PIN REQUIRED
43	NETWORK SUBSET PERSONALIZATION PUK REQUIRED
44	SERVICE PROVIDER PERSONALIZATION PIN REQUIRED
45	SERVICE PROVIDER PERSONALIZATION PUK REQUIRED
46	CORPORATE PERSONALIZATION PIN REQUIRED
47	CORPORATE PERSONALIZATION PUK REQUIRED
48	HIDDEN KEY REQUIRED (SEE NOTE 2)
49	EAP METHOD NOT SUPPORTED
50	INCORRECT PARAMETERS
51	COMMAND IMPLEMENTED BUT CURRENTLY DISABLED
52	COMMAND ABORTED BY USER
53	NOT ATTACHED TO NETWORK DUE TO MT FUNCTIONALITY RESTRICTIONS
54	MODEM NOT ALLOWED - MT RESTRICTED TO EMERGENCY CALLS ONLY
55	OPERATION NOT ALLOWED BECAUSE OF MT FUNCTIONALITY RESTRICTIONS
56	FIXED DIAL NUMBER ONLY ALLOWED - CALLED NUMBER IS NOT A FIXED DIAL NUMBER (REFER 3GPP TS 22.101 [147])
57	TEMPORARILY OUT OF SERVICE DUE TO OTHER MT USAGE
58	LANGUAGE/ALPHABET NOT SUPPORTED
59	UNEXPECTED DATA VALUE
60	SYSTEM FAILURE
61	DATA MISSING
62	CALL BARRED
63	MESSAGE WAITING INDICATION SUBSCRIPTION FAILURE
100	UNKNOWN

A.2 FTP Error Codes

Code(AT+CMEE=1)	Text(AT+CMEE=2)
421	SERVICES CANNOT BE PROVIDED. CLOSE THE CONTROL CONNECTION.
425	FAIL TO OPEN THE DATA CONNECTION.
426	CLOSE THE CONNECTION AND TERMINATE THE TRANSMISSION.
450	THE REQUESTED FILE OPERATION IS NOT EXECUTED.
451	TERMINATE THE REQUESTED OPERATION: A LOCAL ERROR OCCURS.
452	THE REQUESTED OPERATION IS NOT EXECUTED: INSUFFICIENT SYSTEM STORAGE SPACE.
500	THE COMMAND IS UNABLE TO IDENTIFY BECAUSE OF INCORRECT FORMAT.
501	SYNTAX ERROR.
502	COMMAND NOT EXECUTED.
503	COMMAND SEQUENCE ERROR.
504	THE COMMAND SPECIFYING THE PARAMETER IS NOT EXECUTED.
530	USERS DO NOT LOG IN.
532	ACCOUNT INFORMATION IS REQUIRED FOR FILE STORAGE.
550	THE REQUESTED OPERATION IS NOT EXECUTED.
551	THE REQUESTED OPERATION IS TERMINATED: UNKNOWN PAGE TYPE.
552	THE REQUESTED FILE OPERATION IS TERMINATED: STORAGE SPACE EXCEEDED.
553	THE REQUESTED OPERATION IS NOT EXECUTED: INVALID FILE NAME.
600	UNKNOWN ERROR.

A.3 HTTP(S) Error Codes

Code(AT+CMEE=1)	Text(AT+CMEE=2)
300	Multiple Choice
301	Moved Permanently

302	Found
303	See Other
304	Not Modified
305	Use Proxy
306	unused
307	Temporary Redirect
308	Permanent Redirect
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Payload Too Large
414	URI Too Long
415	Unsupported Media Type
416	Requested Range Not Satisfiable
417	Expectation Failed
418	I'm a teapot
421	Misdirected Request
422	Unprocessable Entity (WebDAV)
423	Locked (WebDAV)
424	Failed Dependency (WebDAV)
425	Too Early
426	Upgrade Required

428	Precondition Required
429	Too Many Requests
431	Request Header Fields Too Large
451	Unavailable For Legal Reasons
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version Not Supported
506	Variant Also Negotiates
507	Insufficient Storage
508	Loop Detected (WebDAV)
510	Not Extended
511	Network Authentication Required

B Reference Process of AT Command Programming

B.1 Content of PDU SMS Messages

<PDU> SMS message sending format:

1>: 0891

08: indicates the length of the SMSC address information

91: indicates the format of the SMSC address

2>: Inversion of every two bits (add F if the bits are not sufficient) in SMSC number, fixed. For example, China Unicom 8613010888500 should be 683108705505F0 here.

3>: 0100

01: Indicates basic parameters

00: indicates message baseline value

4>: Convert the receiving number into hexadecimal. For example, the number length is 11 bits and then the hexadecimal length should be 0B.

5>: 81 (Receiving mode) there are multiple receiving modes. 81 indicates that the receiving mode is unknown.

6>: Inversion of every two bits (add F if the bits are not sufficient) in the recipient number. For example, 13421839693 should be 3124819396F3 after conversion.

7>: 0008

8>: The hexadecimal length of the SMS message content. For example, the UCS2 code of hello is 00080A00680065006C006C006F, that is 10 bits and the hexadecimal length is 0A.

9>: Message content, for example, the USC2 code of hello is 00080A00680065006C006C006F.

One PDU message contains the above 9 parts and the parameter values are determined by the actual situation.

If the SMSC address length is 0, replace 08 with 00 and the SMSC type and address fields must be omitted.

The following is an example of the PDU message whose SMSC address length is not 0:

0891683110808805F001000B813124819396F300080A00680065006C006C006F

Wherein,

0891

683108705505F0: SMSC number of China Unicom

0100

0B: the length of the recipient number

81: Receiving mode

3124819396F3: The number of recipient

0008

0A: The length of the content

00680065006C006C006F: SMS message content

Message content: hello

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

The following is an example of the PDU message whose SMSC address length is **0**:

0001000B813124819396F300080A00680065006C006C006F

Wherein,

00: SMSC address information length

SMSC number is not needed.

0100

0B: the length of the recipient number

81: Receiving mode

3124819396F3: The number of recipient

0008

0A: The length of the content

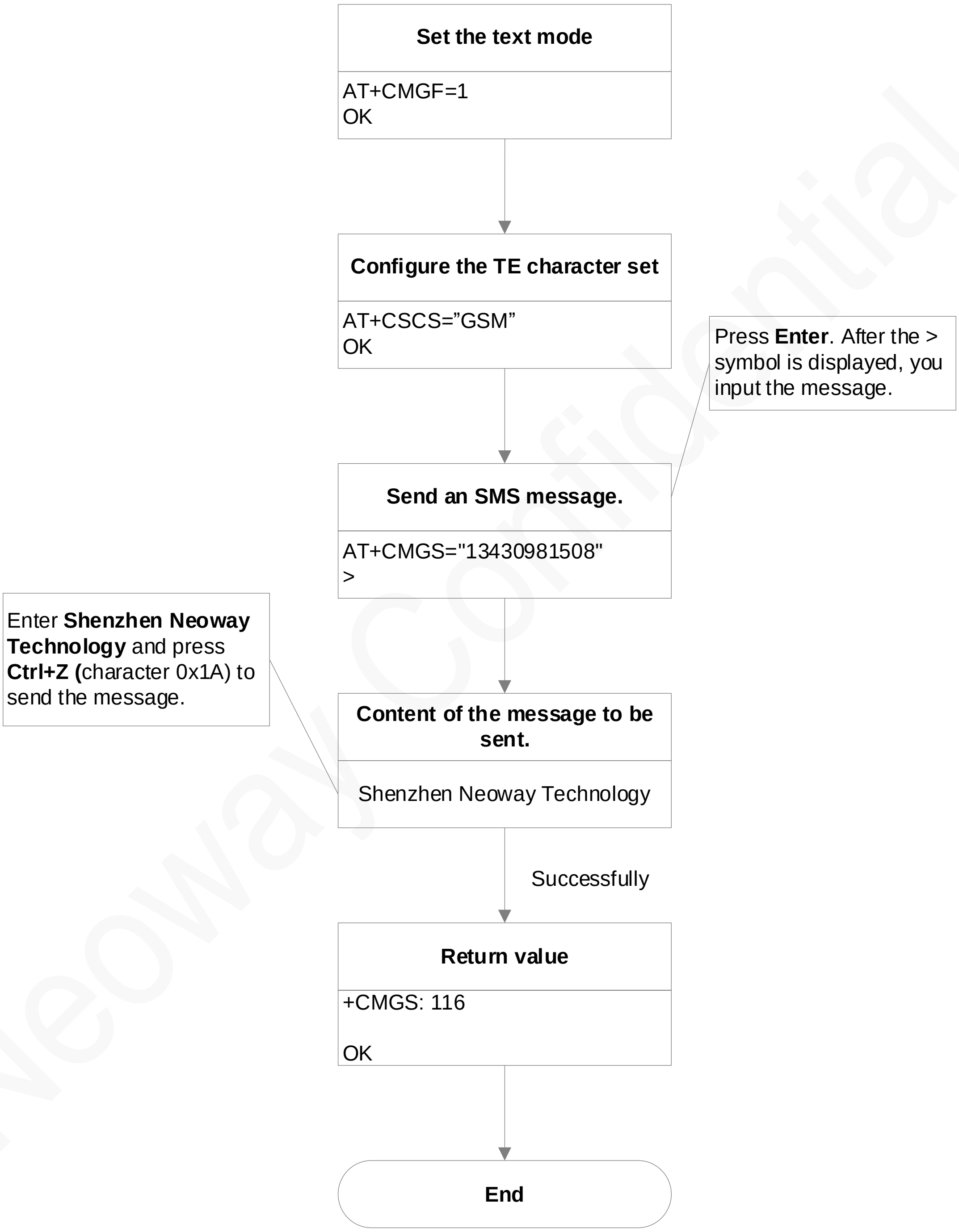
00680065006C006C006F: SMS message content

SMS message content: hello

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

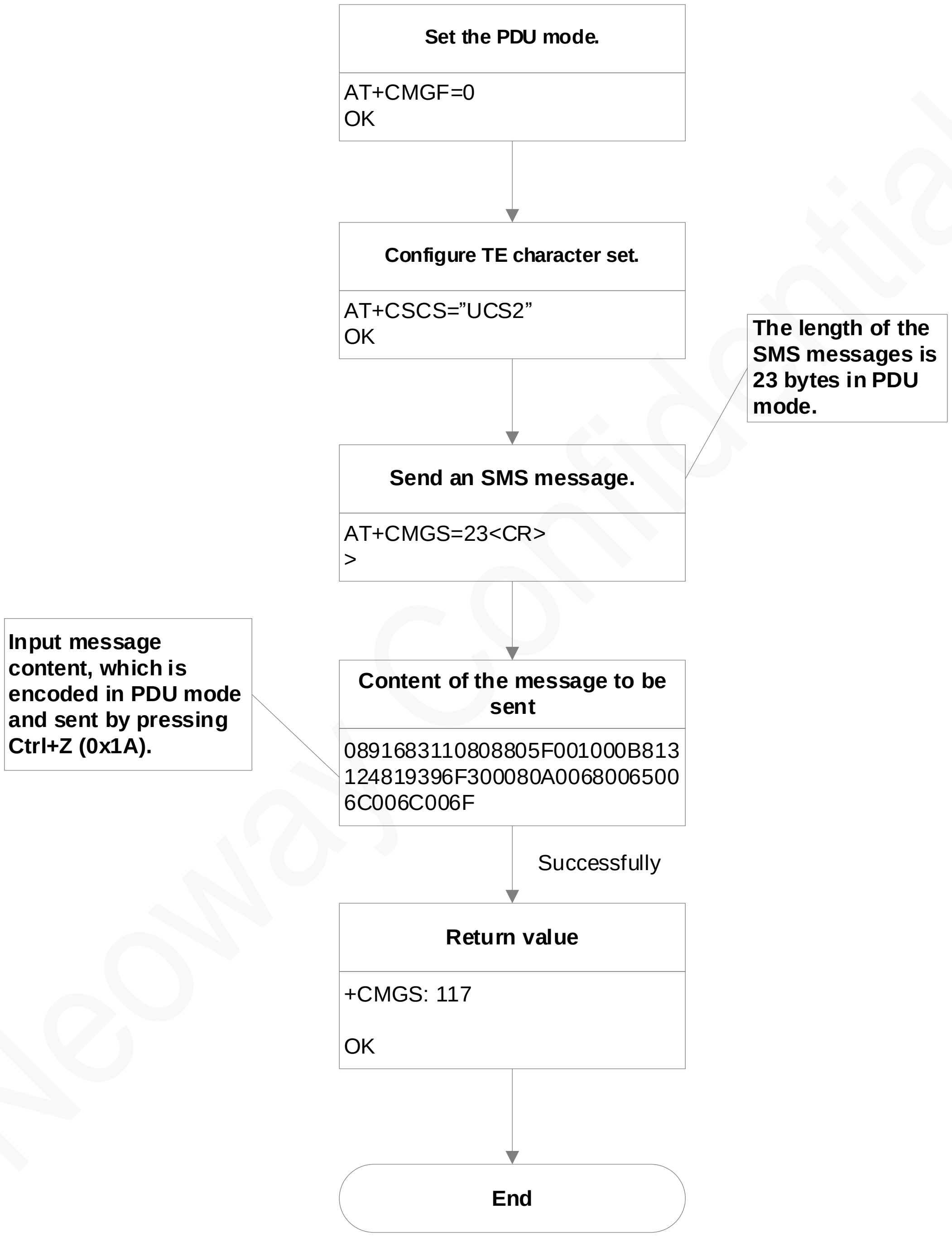
B.2 Flowchart of Sending Text SMS Messages (Through UART)

Figure B-1 Flowchart of sending text format SMS messages


Set the text mode
AT+CMGF=1
OK
Configure the TE character set
AT+CSCS=”GSM”
OK
Press **Enter**. After the > symbol is displayed, you input the message.
Send an SMS message.
AT+CMGS="13430981508"
>
Enter **Shenzhen Neoway Technology** and press **Ctrl+Z (**character 0x1A) to send the message.
Content of the message to be sent.
Shenzhen Neoway Technology
Successfully
Return value
+CMGS: 116
OK
End

B.3 Flowchart of Sending PDU SMS Messages (Through UART)

Figure B-2 Flowchart of Sending PDU SMS messages


Set the PDU mode.
Configure TE character set.
The length of the SMS messages is 23 bytes in PDU mode.
Send an SMS message.
Input message content, which is encoded in PDU mode and sent by pressing Ctrl+Z (0x1A).
Content of the message to be sent
Successfully
Return value
End

```
AT+CMGF=0
OK
```

```
AT+CSCS=”UCS2”
OK
```

```
AT+CMGS=23<CR>
>
```

```
0891683110808805F001000B813
124819396F300080A0068006500
6C006C006F
```

```
+CMGS: 117

OK
```

C Support Band List

C.1 GSM Bands

GSM Band Name	Bit	HEX Band Mask
GSM_450	1	1
GSM_480	2	2
GSM_900P	3	4
GSM_900E	4	8
GSM_900R	5	10
GSM_850	6	20
GSM_1800	7	40
GSM_1900	8	80

C.2 LTE Bands

LTE Band Name	Bit	HEX Band Mask
EUTRAN_BAND1	1	1
EUTRAN_BAND2	2	2
EUTRAN_BAND3	3	4
EUTRAN_BAND4	4	8
EUTRAN_BAND5	5	10
EUTRAN_BAND6	6	20
EUTRAN_BAND7	7	40
EUTRAN_BAND8	8	80
EUTRAN_BAND9	9	100
EUTRAN_BAND10	10	200
EUTRAN_BAND11	11	400
EUTRAN_BAND12	12	800
EUTRAN_BAND13	13	1000

EUTRAN_BAND14	14	2000
EUTRAN_BAND17	17	10000
EUTRAN_BAND33	33	100000000
EUTRAN_BAND34	34	200000000
EUTRAN_BAND35	35	400000000
EUTRAN_BAND36	36	800000000
EUTRAN_BAND37	37	1000000000
EUTRAN_BAND38	38	2000000000
EUTRAN_BAND39	39	4000000000
EUTRAN_BAND40	40	8000000000
EUTRAN_BAND41	41	10000000000
EUTRAN_BAND42	42	20000000000
EUTRAN_BAND43	43	40000000000
EUTRAN_BAND44	44	80000000000
EUTRAN_BAND65	51	4000000000000
EUTRAN_BAND66	52	8000000000000
EUTRAN_BAND71	60	800000000000000
EUTRAN_BAND252	61	1000000000000000
EUTRAN_BAND253	62	2000000000000000
EUTRAN_BAND255	64	8000000000000000

D Result Codes

<err>	Description
0	Operation successful
601	Unknown error
602	FTP(S) server blocked
603	FTP(S) server busy
604	DNS parse failed DNS
605	Network error
606	Control connection closed.
607	Data connection closed
608	Socket closed by peer
609	Timeout error
610	Invalid parameter
611	Failed to open file
612	File position invalid
613	File error
614	Service not available, closing control connection
615	Open data connection failed
616	Connection closed; transfer aborted
617	Requested file action not taken
618	Requested action aborted: local error in processing
619	Requested action not taken: insufficient system storage
620	Syntax error, command unrecognized
621	Syntax error in parameters or arguments
622	Command not implemented
623	Bad sequence of commands
624	Command parameter not implemented
625	Not logged in
626	Need account for storing files

627	Requested action not taken
628	Requested action aborted: page type unknown
629	Requested file action aborted
631	SSL authentication failed SSL

E FTP(S) Error Codes

<protocol_error>	Description
0	Invalid result
200	Command okay
421	Service not available, closing control connection
425	Open data connection failed
426	Connection closed; transfer aborted
450	Requested file action not taken
451	Requested action aborted: local error in processing
452	Requested action not taken: insufficient system storage
500	Syntax error, command unrecognized
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command parameter not implemented
530	Not logged in
532	Need account for storing files
550	Requested action not taken: file unavailable
551	Requested action aborted: page type unknown
552	Requested file action aborted: exceeded storage allocation
553	Requested action not taken: file name not allowed