E103- RTL8711 Series Product AT Command Manual

WiFi + BLE 5.0 2.4GHz + 5.8GHz WiFi serial port module



Chengdu Ebyte Electronic Technology Co.,Ltd.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Contents

1 AT commands 6
1.1 AT Command Description 6
1.2 General AT Command Set 6
1.2.1 AT+TEST Communication Test 6
1.2.2 Obtaining Version Information via AT+GMR 6
1.2.3 AT+LIST lists all supported AT commands. 7
1.2.4 AT+RST Reset Module 7
1.2.5 Use AT+STATE to obtain task status and heap information 8
1.2.6 AT+LOG 8
1.2.7 AT+UART Serial Port Parameters 9
1.3 WiFi AT Command Set 10
1.3.1 Connecting to the router using AT+WLCONN 10
1.3.2 Routers that have disconnected from AT+WLDISCONN 11
1.3.3 Configuring a Static IP Address in STA Mode using AT+WLSTATICIP 11
1.3.4 AT+WLSCAN Scan for Surrounding Wi-Fi Networks 11
1.3.5 Use AT+WLRSSI to obtain the signal strength of the currently connected AP. 12
1.3.6 AT+WLSTARTAP Enable AP Mode 12
1.3.7 AT+WLSTOPAP disables AP mode 13
1.3.8 Use AT+WLSTATE to query Wi-Fi status information 13
1.3.9 Configure automatic or fast reconnection strategy for AT+WLRECONN 16
1.3.10 AT+WLPROMISC Enable/Disable Wi-Fi Promiscuous Mode 16
1.3.11 AT+WLWPS executes Wi-Fi WPS function 16
1.3.12 AT+WLPS Configuration: Wi-Fi Power Saving Mode 17
1.4 Network AT Command Set 17
1.4.1 AT+PING test for domain name/IP address 17
1.4.2 AT+DNS resolves domain names to IP addresses 19
1.4.3 Use AT+QUERYDNSSRV to query the currently configured DNS server address. 19
1.4.4 AT+SETDNSSRV sets the DNS server address 19
1.4.5 Configuring SNTP Service with AT+SNTPCFG 20
1.4.6 Query the current SNTP time using AT+SNTPTIME 20
1.4.6 Use AT+SNTPQUERY to query the configuration information of the current SNTP service. 21
1.4.7 Enabling or disabling the AT+MDNS function 21
1.5 Socket AT Command Set 23
1.5.1 Configuring Global Options for TCP/UDP/SSL Sockets with AT+SKTCFG 23
1.5.2 AT+SKTQUERY Socket Global Options Configuration 24
1.5.3 Configuring TCP/UDP/SSL Server Listening Service on AT+SKTSERVER 25
1.5.4 Establishing TCP/UDP/SSL Client Connections with AT+SKTCLIENT 26
1.5.5 AT+SKTSEND sends data via a specified connection ID. 27
1.5.6 AT+SKTREAD reads cached data from a specified connection ID 27
1.5.7 AT+SKTSENDRAW transparent transmission mode sends data via a specified connection ID. 28

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.5.8 Use AT+SKTSTATE to query the status information of all active connection IDs.29
1.5.9 AT+SKTDEL deletes the specified connection ID 29
1.6 MQTT AT Command Set 31
1.6.1 Configuring MQTT Connection Parameters for a Specified Connection ID using AT+MQTTCFG 31
1.6.2 Connecting to the MQTT Proxy Server using AT+MQTTCONN 32
1.6.3 AT+MQTTDISCONN closes the specified MQTT connection and releases resources. 33
1.6.4 Subscribing to MQTT topics using AT+MQTTSUB 33
1.6.5 AT+MQTTUNSUB Unsubscribe from a Specified Topic 34
1.6.6 AT+MQTTPUB Message Publishing to Specific Topics 34
1.6.7 Publishing messages to a specific topic in AT+MQTTPUBRAW pass-through mode 35
1.6.8 AT+MQTTRESET resets all MQTT instances and releases resources. 36
1.6.9 Use AT+MQTTQUERY to query the status and configuration information of a specified connection. 36
Version 1.6.10 of AT+MQTTSUBQUERY lists all subscribed topics. 37
1.7 HTTP AT Command Set 37
1.7.1 Setting up global HTTP configuration using AT+HTTPCONF 37
1.7.2 Setting HTTP Global Request Headers in AT+HTTPHEADER Pass-through Mode38
1.7.3 AT+HTTPQUERY query HTTP global configuration 38
1.7.4 Retrieving HTTP Resources using AT+HTTPGET 38
1.7.5 Sending HTTP POST requests in AT+HTTPPOST pass-through mode 40
1.7.6 Sending HTTP PUT requests in AT+HTTPPUT pass-through mode 42
1.7.7 AT+HTTPDEL deletes the specified HTTP resource. 43
1.8 WebSocket AT Command Set 45
1.8.1 Configuring Global Parameters for the WebSocket Client using AT+WSGLCFG 45
1.8.2 Configuring WebSocket Client Parameters for a Specified Connection ID using AT+WSCFG 46
1.8.3 Configuring WebSocket Client Request Headers with AT+WSHEAD 47
1.8.4 Configuring WebSocket Client Request Headers in AT+WSHEADRAW Pass-Through Mode 47
1.8.5 Establishing a WebSocket Client Connection using AT+WSCONN 48
1.8.6 Sending WebSocket data using AT+WSSEND 49
1.8.7 Sending WebSocket data in AT+WSSENDRAW pass-through mode 49
1.8.8 AT+WSDISCONN Actively Closes WebSocket Client Connections with a Specified Connection ID 50
1.8.9 AT+WSQUERY query for WebSocket client-related information 50
1.8.10 AT+WSTXQUEUECHECK checks if the tx queue for the specified connection ID is full. 51
1.8.11 Configuring WebSocket Server Parameters with AT+WSSRVCFG 52
1.8.12 Start the WebSocket server with AT+WSSRVSTART 52
1.8.13 AT+WSSRVDISCONN disconnects the WebSocket client connection with the specified connection ID. 54

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Version 1.8.14 AT+WSSRVSTOP shuts down the WebSocket server. 54
1.8.15 AT+WSSRVSEND sends WebSocket data to the specified connection ID 55
1.8.16 AT+WSSRVSENDRAW Pass-through Mode: Send WebSocket data to a specified connection ID 55
1.8.17 Use AT+WSSRVQUERY to query WebSocket server-side related information. .56
1.9 Bluetooth Transmission AT Command Set 57
1.9.1 AT+BTDEMO=transfer_module Initializes or deinitializes the transfer module .. 57
1.9.2 AT+BTDEMO=transfer_module,get_name retrieves the device name. 57
1.9.3 AT+BTDEMO=transfer_module,set_name sets the device name. 58
1.9.4 AT+BTDEMO=transfer_module,get_uuid to obtain UUID 58
1.9.5 AT+BTDEMO=transfer_module,set_uuid sets the UUID. 59
1.9.6 AT+BTDEMO=transfer_module, read_val sets the read value for readable characteristics. 59
1.9.7 AT+BTDEMO=transfer_module,srv_add adds GATT service 60
Version 1.9.8 AT+BTDEMO=transfer_module, char_add adds a feature to the GATT service. 60
Version 1.9.9 `AT+BTDEMO=transfer_module,srv_end` completes the addition of the GATT service. 60
1.9.10 AT+BTDEMO=transfer_module,srv_del Delete all GATT services 62
1.9.11 AT+BTDEMO=transfer_module, char_read_val is the characteristic value set for GATT readable service. 62
Version 1.9.12 `AT+BLEGAP=addr` retrieves the Bluetooth MAC address. 62
1.9.13 AT+BLEGAP=rand_addr sets a random Bluetooth address. 63
Version 1.9.14 `AT+BLEGAP=mtu_size` retrieves the MTU size. 63
Version 1.9.15 AT+BLEGAP=set_mtu sets the MTU size. 64
Version 1.9.16 AT+BLEGAP=adv enables broadcasting and sets broadcast parameters.65
1.9.17 AT+BLEGAP=get_adv_param retrieves broadcast parameters 66
1.9.18 AT+BLEGAP=adv_data Sets broadcast data before starting the broadcast. 66
1.9.19 AT+BLEGAP=scan Starts or stops scanning 67
1.9.20 AT+BLEGAP=scan_param Sets scan parameters before scanning begins. 67
Version 1.9.21 `AT+BLEGAP=get_scan_param` retrieves scan parameters. 68
1.9.22 AT+BLEGAP=scan_rsp sets the scan response data before scanning begins. 68
Version 1.9.23 AT+BLEGAP=conn creates a connection. 69
1.9.24 AT+BLEGAP=disconn Close connection 70
Version 1.9.25 uses AT+BLEGAP=conn_info to retrieve connection information. 70
Version 1.9.26 `AT+BLEGAP=conn_update` updates connection parameters after the connection is established. 71
Version 1.9.27 `AT+BLEGAP=wl_add` adds the address to the whitelist. 72
Version 1.9.28 `AT+BLEGAP=wl_remove` removes addresses from the whitelist. 72
Version 1.9.29 AT+BLEGAP=wl_clear clears the blank list. 72
Version 1.9.30 AT+BLEGAP=sec_param configures encryption information. 73
Version 1.9.31 `AT+BLEGAP=get_sec_param` retrieves the encryption parameters. ... 74
1.9.32 AT+BLEGAP=sec Initiate the encryption process 74
1.9.33 AT+BLEGAP=pair_cfm confirms pair-only instruction. 75

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.9.34 AT+BLEGAP=auth_key Enter the authentication key 75
1.9.35 AT+BLEGAP=auth_oob Input OOB data 75
Version 1.9.36 uses AT+BLEGAP=bond_info to retrieve binding information. 76
1.9.37 AT+BLEGAP=auth_keycfm confirms the authentication key. 76
1.9.38 AT+BLEGAP=bond_del Deletes binding information 76
Version 1.9.39 AT+BLEGAP=bond_clear clears all bond information. 77
1.9.40 AT+BLEGAP=disc discovers services and features 77
1.9.41 AT+BLEGATTC=read (Reads feature values) 78
1.9.42 AT+BLEGATTC=write Write characteristic value 79
Version 1.9.43 AT+BLEGATTS=notify sends notifications from the GATT server. 80
Version 1.9.44 AT+BLEGATTS=indicate sends an instruction from the GATT server. . 80
1.10 File System AT Command Set 81
1.10.1 Performing file system operations with AT+FS 81
1.10.2 AT+CERT query server-side or client-side certificate components 82
1.10.3 AT+FSRGN Configuration File System Command Operations on Partitions 82
1.11 OTA AT Command Set 83
1.11.1 AT+OTA HTTP obtains OTA firmware via HTTP GET method 83
1.11.2 AT+OTAUSER Transparent Transmission Mode for Sending OTA Upgrade Firmware for Upgrade 84
1.12 EBYTE AT Command Set 85
1.12.1 AT+WEBCFG WiFi Configuration via Web 85
1.12.2 AT+RS485 Enable RS485 Compatibility Mode 85
1.12.3 AT+LPOWER Enable Low Power Mode 86
1.12.4 AT+RESTORE restores factory settings 86
1.12.5 AT+VERSION? Query version number 87
1.12.6 AT+BLERECONN disables power-on reconnection. 87
Revision History 88

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1 AT commands

1.1 AT Command Description

- AT commands use UART for communication by default, and the pin functions are defined as follows:

Pin	Function	Remark
PB30	RX	Baud rate: 115200 Data bits: 8 bits Stop bit: 1 bit Check digit: None Command suffix: Carriage Return/Line Feed (CRLF)
PB31	TX	
PB21	CTS	
PB20	RTS	

1.2 General AT Command Set

1.2.1 AT+TEST Communication Test

Instruction format	AT+TEST=[<mode>, <length>/<string>]
Execution Example	● Send: AT +TEST ● answer: OK
Parameter Description	● <mode> Test mode 0: Echo Test 1: Uplink speed test 2: Downlink speed test ● <length> The amount of data exchanged during rate testing ● <string> Echoing the input string during the test
Feature Description	

1.2.2 Obtaining Version Information via AT+GMR

Instruction format	AT +GMR
Execution Example	● Send: AT +GMR

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● answer: AMEBA-RTOS SDK VERSION: <SDK version number> ATCMD VERSION: <AT command version number> IMAGE VERSION: <Current firmware version number> <Chip Nickname>: <Chip Information> COMPILE TIME: <compile time> COMPILE USER: <Compiler User> COMPILE ENV: <Compilation Environment> OK
Feature Description	

1.2.3 AT+LIST lists all supported AT commands.

Instruction format	AT +LIST
Execution Example	● Send: AT +LIST ● answer: Common AT Command: AT+TEST // … followed by other common command list. Wi-Fi AT Command: AT+WLSOFTAP // … followed by other Wi-Fi command list. NETWORK AT command: AT+SKTGETERR // … followed by other TCP/IP command list. BT AT command: AT+BLECMODE // … followed by other BT command list. MQTT AT command: AT+MQTTOPEN // … followed by other MQTT command list. OK
Feature Description	

1.2.4 AT+RST Reset Module

Instruction format	AT +RST
Execution Example	● Send: AT +RST ● answer: OK
Feature Description	

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.2.5 Use AT+STATE to obtain task status and heap information

Instruction format	AT +STATE
Execution Example	● Send: AT +STATE ● answer: Task List: <Thread Name> <Status> <Priority> <Available Stack Level> <Task Number> ... HeapStats: Total Heap: <Total heap size> Heap Free Now: <Current available heap size> Heap Used Now: <Current heap size used> Heap Used Max: <Historical maximum heap usage> OK
Feature Description	● In the <Status> field: X indicates that it is running. R represents the ready state. B indicates blockage. S indicates suspension D indicates that it has been deleted.

1.2.6 AT+LOG

Instruction format	AT + LOG =< op > [, < module > , < log_level >]
Execution Example	● Send: AT + LOG = 1 , module1, 3 ● answer: OK
Parameter Description	● <op> Operation type 0: Get log level 1: Set the log level 2: Output all log levels 3: Clear all log levels ● <module> Specific module name String, * represents all modules The length should not exceed 9 bytes. ● <log_level> Log levels [0,5]
Feature Description	

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.2.7 AT+UART Serial Port Parameters

Instruction format	AT + UART =< store_in_flash > , < baudrate >
Execution Example	● Send: AT+UART=0,115200 ● answer: OK
Parameter Description	● <store_in_flash> parameters saved to FLASH 0 : Do not write to FLASH 1 : Write to FLASH ● <baudrate> baud rate [4800,6000000]
Feature Description	● A restart is required for the changes to take effect when saving to flash.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.3 WiFi AT Command Set

The error codes for Wi-Fi AT commands are defined as follows:

Error code	Definition
1	Required parameters not set
2	Invalid parameter value
3	Insufficient memory
4	Insufficient memory
5	Function not supported
6	Unknown error
7~16	reserve
17	Connection failed
18	Result failed to be retrieved.
19	Invalid Wi-Fi status
20	Abnormal scan parameters
21	AP startup failed
22	AP startup timeout

1.3.1 Connecting to the router using AT+WLCONN

Instruction format	AT+WLCONN=[<type>,<value>,<type>,<value>……]
Execution Example	● Send: AT+WLCONN=ssid,MyWiFi,pw,password123 ● answer: OK
Parameter Description	● <type> Parameter type, followed by the corresponding <value> SSID: followed by the SSID name bssid: followed by the BSSID address ch: followed by the channel number pw: Post-connection password key_id: followed by the WEP key ID ● The specific parameter corresponding to <type> for <value> ● <ssid> Wi-Fi SSID name Maximum length: 32 characters ● <bssid> MAC address of the access point (AP) Format: Colon-separated hexadecimal (e.g., 1a:2b:3c:4d:5e:6f) ● <ch> wireless channel Default value: 0 (auto-select) ● Wi - Fi connection password ● <key_id> Key index for WEP encryption mode Valid range: [0, 3]
Feature Description	● The order of parameters such as <ssid>, <bssid>, and <ch> can be arbitrary. ● Once the connection is established, a message indicating whether the connection

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	was successful or failed will be displayed. Wi-Fi connected Wi-Fi connection failed. ● When an IPv4 address is successfully requested, a success or timeout message will be displayed. [$]wifi got IP:"<ip>" [$]wifi got IP timeout ● A disconnection message will appear when the connection to the router is lost. Wi-Fi disconnected

1.3.2 Routers that have disconnected from AT+WLDISCONN

Instruction format	AT+WLDISCONN
Execution Example	● Send: AT+WLDISCONN ● answer: OK
Feature Description	

1.3.3 Configuring a Static IP Address in STA Mode using AT+WLSTATICIP

Instruction format	AT+WLSTATICIP=<ip>[,<gateway>,<netmask>]
Execution Example	● Send: AT+UART= 192.168.1.100 ● answer: OK
Parameter Description	● IPv4 address Format: IPv4 address string (e.g., 192.168.1.100) ● <gateway> gateway address Default value: Calculated automatically based on IP address ● <netmask> Subnet mask Default value: 255.255.255.0
Feature Description	● Entering this information after connecting will not change the IP status of an already connected network. ● The command becomes invalid after disconnect.

1.3.4 AT+WLSCAN Scan for Surrounding Wi-Fi Networks

Instruction format	AT+WLSCAN=[<type>,<value>,<type>,<value>]

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Execution Example	● Send: AT+WLSCAN=ch,1:6:11 ● answer: 1, a4:1a:3a:0d:0f:35, -26, 1, AX, "Open", "TP-LINK_0F35" 2, a6:1a:3a:7d:0f:35, -26, 1, AX, "WPA/WPA2 AES", "" 3, 78:60:5b:0f:b1:d7, -34, 6, AX, "WPA/WPA2 AES", "TP-LINK_B1D7" 4, 04:79:75:8e:3c:d5, -42, 11, AX, "WPA2/WPA3-SAE AES", "huaweipro4" OK
Parameter Description	● <type> Scan Filter Type SSID: followed by the SSID name ch: followed by a list of channels ● <value> Gateway address <ssid>: Target SSID name <chlist>: List of channels (e.g., 1:6:11)
Feature Description	● The order of parameters such as <ssid> and <ch> can be arbitrary. ● If the <ch> parameter is not specified, all available channels will be scanned. ● Response format <Serial Number>, <MAC Address>, <Signal Strength>, <Channel>, <Mode>, <Encryption Method>, <SSID> ... OK

1.3.5 Use AT+WLRSSI to obtain the signal strength of the currently connected AP.

Instruction format	AT+WLRSSI
Execution Example	● Send: AT+WLRSSI ● answer: RSSI=-50 DATA_RSSI=-52 BEACON_RSSI=-51 OK
Feature Description	● Response format RSSI = <current signal strength> DATA_RSSI = <data signal strength> BEACON_RSSI = <beacon signal strength> OK

1.3.6 AT+WLSTARTAP Enable AP Mode

Instruction format	AT+WLSTARTAP=[<type>,<value>,<type>,<value>……]
Execution Example	● Send: AT+WLSTARTAP=ssid,test_ssid,pw,12345678,sec,wpa2 ● answer:

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	OK
Parameter Description	● <type> Parameter type, consistent with the WiFi parameter mentioned earlier during connection. ● <value> Any type of <ssid>, <ch>, <pw>, <sec> ● <ssid> SSID cannot be empty. ● <ch> 2G configurable range [1,14], step size is 1. 5G configurable range [36, 177], with a stride of 4. Ultimately, it depends on the range supported by the configured channel plan. ● <sec> open/wep/tkip/wpa2/wpa3 ● <pw> Password length range: [8, 64] ● <ip> The IP address of the access point (AP), the default value is 192.168.43.1 ● The gateway address of the AP (Access Point), with a default value of 192.168.43.1. ● <msk> The subnet mask for the AP, with a default value of 255.255.255.0. ● <pl> AP IP address pool
Feature Description	

1.3.7 AT+WLSTOPAP disables AP mode

Instruction format	AT+WLSTOPAP
Execution Example	● Send: AT+WLSTOPAP ● answer: OK
Feature Description	

1.3.8 Use AT+WLSTATE to query Wi-Fi status information

Instruction format	AT+WLSTATE
Execution Example	● Send: AT+WLSTATE ● answer: AT+WLSTATE WLAN0 Status: Running ========== max_skbbuff_used_num=8, skbbuff_used_num=4 WLAN0 Setting: ========== MODE => STATION SSID => mowifi BSSID => e8:91:20:fa:8d:1d CHANNEL => 10 SECURITY => WPA2 AES PASSWORD => 12345678

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Interface (0) ========== MAC => 00:e0:4c:b7:23:12 IP => 192.168.55.206 GW => 192.168.55.1 MSK => 255.255.255.0 WLAN1 Status: Running ========== WLAN1 Setting: ========== MODE => AP SSID => test BSSID => 00:e1:4c:b7:23:12 CHANNEL => 10 SECURITY => OPEN PASSWORD => Interface (1) ========== MAC => 00:e1:4c:b7:23:12 IP => 192.168.43.1 GW => 192.168.43.1 MSK => 255.255.255.0 Associated Client List: ========== Client Num: 1 Client 1: IPv4 address: 192.168.43.101, MAC address: ae:a8:7f:ca:18:7b, RSSI: -48 OK
Feature Description	● Response format WLAN0 Status: Running ========== max_skbinfo_used_num=<maximum number of skbinfo instances used in history>, skbinfo_used_num=<current number of skbinfo instances used> max_skbdata_used_num=<maximum number of skbdata instances used in history>, skbdata_used_num=<current number of skbdata instances used> WLAN0 Setting: ========== MODE => <Mode: STATION or AP> SSID => <ssid>

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	BSSID => <bssid> CHANNEL => <Channel> SECURITY => <Security Mode: OPEN, WPA2, WPA3, or various hybrid encryption> PASSWORD => <Password> Interface (0) ========== MAC => <Local MAC address> IP => <Local IPv4 address> GW => <Local Gateway Address> MSK => <local subnet mask> WLAN1 Status: Running ========== WLAN1 Setting: ========== MODE => <Mode: STATION or AP> SSID => <ssid> BSSID => <bssid> CHANNEL => <Channel> SECURITY => <Security Mode: OPEN, WPA2, WPA3, or various hybrid encryption> PASSWORD => <Password> Interface (1) ========== MAC => <Local MAC address> IP => <Local IPv4 address> GW => <Local Gateway Address> MSK => <local subnet mask> Associated Client List: ========== Client Num: <Number of currently connected STAs> Client 1: IPv4 address: <IP address of STA 1>, MAC address: <MAC address of STA 1>, rssi: <signal strength of STA 1> Client 2: IPv4 address: <IP address of STA 2>, MAC address: <MAC address of STA 2>, RSSI: <signal strength of STA 2> ... OK

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.3.9 Configure automatic or fast reconnection strategy for AT+WLRECONN

Instruction format	AT+WLRECONN=<mode>,<value>
Execution Example	● Send: AT+WLRECONN =auto,1 ● answer: OK
Parameter Description	● <mode> Reconnection mode auto: Automatic reconnection after disconnection fast: Fast connection mode ● <value> mode parameters 0: Disable this mode 1: Enable this mode
Feature Description	● The following message will appear each time automatic reconnection is triggered at the start and end. [$]wifi reconnecting [$]wifi reconnect done

1.3.10 AT+WLPROMISC Enable/Disable Wi-Fi Promiscuous Mode

Instruction format	AT+WLPROMISC=<enable>[,<all/apall>]
Execution Example	● Send: AT+WLPROMISC=enable,apall ● answer: OK
Parameter Description	● <enable> Hybrid mode switch enable: Enable disable: disable ● <all/apall> Packet filtering mode all: Capture all packets apall: Only captures AP-related packets.
Feature Description	● When the <enable> parameter is set to enable, the <all/apall> parameter must be provided. ● Note: This is not supported when the parameter is set to 'all'.

1.3.11 AT+WLWPS executes Wi-Fi WPS function

Instruction format	AT+WLWPS=<pbc_pin>[,<pin_val>]
Execution Example	● Send: AT+WLWPS=pbc ● answer:

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	OK
Parameter Description	● <pbc_pin> WPS Mode Selection PBC: Keypad Mode pin: PIN code mode ● <pin_val> PIN code value Must comply with the verification rules When selecting pin mode, you must provide...
Feature Description	● The `<pin_val>` parameter only takes effect when `<pbc_pin>` is set to `pin`.

1.3.12 AT+WLPS Configuration: Wi-Fi Power Saving Mode

Instruction format	AT+WLPS=<mode>,<enable>[,<mode>,<enable>]
Execution Example	● Send: AT+WLPS=lps,0,ips,0 ● answer: OK
Parameter Description	● <mode> Power saving type, followed by the corresponding <enable> LPS: LPS power-saving mechanism, followed by the switch value. IPS: Power saving mechanism, followed by switch value ● <enable> Switch value 0: Disable 1: Enable
Feature Description	● The order of the <lps> and <ips> parameters can be arbitrary. ● LPS: Wi-Fi power-saving mechanism when connected to a router. ● IPS: Wi-Fi power saving mechanism when not connected to a router ● LPS and IPS are enabled by default.

1.4 Network AT Command Set

1.4.1 AT+PING test for domain name/IP address

Instruction format	AT+PING=<host>[,<-t>,<-n>,<count>,<-l>,<size>] AT+PING=<stop>
Execution Example	● Send: AT+PING=192.168.31.200 ● answer: OK [$][ping_test] 32 bytes from 192.168.31.1: icmp_seq=1 time=135 ms [$][ping_test] 32 bytes from 192.168.31.1: icmp_seq=2 time=5 ms [$][ping_test] 32 bytes from 192.168.31.1: icmp_seq=3 time=102 ms [$][ping_test] 32 bytes from 192.168.31.1: icmp_seq=4 time=3 ms [$][ping_test] 4 packets transmitted, 4 received, 0% packet loss, average 60 ms [$][ping_test] min: 3 ms, max: 135 ms
Parameter Description	● <host> Target host address

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Supports IPv4 addresses or domain names ● <-t> Ping the specified host until stopped ● <-n> Set the number of Ping packets Followed by the <count> parameter <count>: Number of Ping packets positive integer Default value: 4 ● <-l> Sets the data packet size Positive integer , default value: 32 ● <stop> Terminates the current Ping task.
Feature Description	● The order of the <-t>, <-l>, and <-n> parameters can be arbitrary.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.4.2 AT+DNS resolves domain names to IP addresses

Instruction format	AT+DNS=<host>
Execution Example	● Send: AT+DNS=www.baidu.com ● answer: +DNS:103.235.46.102 OK
Parameter Description	● <host> Target host domain
Feature Description	● Error code 1: Input empty parameters 2: Domain name resolution failed

1.4.3 Use AT+QUERYDNSSRV to query the currently configured DNS server address.

Instruction format	AT+QUERYDNSSRV
Execution Example	● Send: AT+QUERYDNSSRV ● answer: +QUERYDNSSRV:8.8.8.8, 8.8.4.4 OK
Feature Description	

1.4.4 AT+SETDNSSRV sets the DNS server address

Instruction format	AT+SETDNSSRV=<primary_dns>[,<secondary_dns>]
Execution Example	● Send: AT+SETDNSSRV=8.8.8.8 ● answer: OK
Parameter Description	● <primary_dns> First DNS server address ● <secondary_dns> Second DNS server address (optional)
Feature Description	● Error code 1: Incorrect number of parameters 2: Invalid DNS server address format

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.4.5 Configuring SNTP Service with AT+SNTPCFG

Instruction format	AT + SNTPCFG =< en > [, < timezone >][, < interval >][, < server1 >][, < server2 >][, < server3 >]
Execution Example	Send: AT + SNTPCFG =1 , 8 , 15 , pool . ntp . org,ntp . sjtu . edu . cn,ntp . aliyun . com ● answer: OK
Parameter Description	● <enable> Whether to enable SNTP service 0: Disable 1: Enable ● <timezone> Time zone offset, supports the following two formats: Format 1: In hours, range: [-12, 14] Format 2: UTC offset, formatted as [+|-][hh]mm, hours (hh) range: [-12, 14], minutes (mm) range: [00, 59] Default value: 8, which is China time zone (UTC+08:00) ● <interval> Synchronization interval (seconds) Range: [15,1073741] Default value: 15 ● <server1> First SNTP server address ● <server2> Second SNTP server address ● <server3> The third SNTP server address Supports up to 3 servers If no server parameters are specified, pool.ntp.org will be used by default.
Feature Description	● Error code 1: Incorrect number of parameters 2: Incorrect parameter format

1.4.6 Query the current SNTP time using AT+SNTPTIME

Instruction format	AT+SNTPTIME
Execution Example	● send: AT+SNTPTIME ● answer: +SNTPTIME:2025-10-20 18:30:30 UTC+08:00 OK
Feature Description	

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.4.6 Use AT+SNTPQUERY to query the configuration information of the current SNTP service.

Instruction format	AT + SNTPQUERY
Execution Example	● send: AT + SNTPQUERY ● answer: enable: <SNTP service enabled status, 0: disabled, 1: enabled> timezone: <time zone> interval: <synchronization interval (seconds)> server_count: <Number of servers configured> server0: <SNTP server address 1> server1: <SNTP server address 2> server2: <SNTP server address 3> OK
Feature Description	● Error code 1: Parameter error

1.4.7 Enabling or disabling the AT+MDNS function

Instruction format	AT+MDNS=<mode>[,<hostname>,<service_type>,<port>][,<service_name>][,<protocol>][,<txt_record_cnt>,<key>,<value>...<key>,<value>]
Execuți	● send: AT+MDNS=1,ameba,_service1,9900 ● answer: OK

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

onExample	
ParameterDescription	● <mode> Enable mode, value range [0,1] 1: Enable mDNS service. The following parameters, hostname, service_type, and port, are required. 0: Disable mDNS service; no further parameters are required. ● <hostname> mDNS hostname ● <service_type> The mDNS service type. For example, "_http". ● <port> mDNS service port Value range [1, 65535] ● <service_name> The name of the mDNS service. If omitted, it defaults to <hostname>. ● <protocol> mDNS transport protocol 1: TCP 2: UDP It is recommended to use "_tcp" or "_udp" whenever possible. If using other transport protocols, please select option 2 and include the specific protocol information in the txt record. If omitted, the default is 1: tcp. ● <txt_record_cnt> The total number of mDNS TXT records, with a value range of [1-6]. The numerical values must correspond to the following "key" and "value". An error will occur if the number of logarithms is less than this number. If the number of logarithms is greater than this number, the excess will be ignored. ● <key> The key name of the TXT record. ● <value> The value recorded in the TXT file
FeatureDescription	● Error code 1: Parameter error 2: Memory allocation failed 3: Repeatedly starting the service 4: Network connection error 5: Failed to start mDNS service

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.5 Socket AT Command Set

1.5.1 Configuring Global Options for TCP/UDP/SSL Sockets with AT+SKTCFG

Instruction format	AT+SKTCFG=[<so_sndtimeo>],[<so_rcvtimeo>],[<tcp_nodelay>],[<so_keepalive>],[<tcp_keepidle>],[<tcp_keepintvl>],[<tcp_keepcnt>]
Execution Example	● Send: AT+SKTCFG=2000,2000,,1,60,10,3 ● answer: OK
Parameter Description	● The `<so_sndtimeo>` option sets the `SO_SNDTIMEO` option to cause socket send operations to return within the `<so_sndtimeo>` timeout period, regardless of success or failure. Unit: milliseconds non-negative integers Default value: 0 ● The `<so_rcvtimeo>` option sets the `SO_RCVTIMEO` option to cause socket receive operations to return within the `<so_rcvtimeo>` timeout period, regardless of success or failure. Unit: milliseconds non-negative integers Default value: 0 ● The `<tcp_nodelay>` option sets the TCP_NODELAY option for the socket to enable or disable the Nagle algorithm in the TCP connection. Default value: 0 0: Enable Nagle's algorithm 1: Disable Nagle's algorithm ● The `<so_keepalive>` option sets the `SO_KEEPALIVE` option for a socket to enable or disable keepalive functionality in a TCP connection. Default value: 0 0: Disable socket keep-alive function 1: Enable socket keep-alive function ● The `<tcp_keepidle>` option sets the TCP_KEEPIDLE option to configure the idle time before sending the first keepalive probe packet to check if the connection is still active. Unit: seconds Value range: [0, 7200], positive integers Default value: 0 ● The `<tcp_keepintvl>` option sets the TCP_KEEPINTVL option to configure the time interval between consecutive keepalive probe packets when no response is received. Unit: seconds Value range: [0, 75], positive integers Default value: 0 ● The `<tcp_keepcnt>` option sets the TCP_KEEPCNT option to determine how many unanswered probe packets to send

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	before declaring a disconnection. Value range: [0, 10], positive integers Default value: 0
Feature Description	

1.5.2 AT+SKTQUERY Socket Global Options Configuration

Instruction format	AT+SKTQUERY
Execution Example	● Send: AT+SKTQUERY ● answer: Global SOCKET configuration: so_sndtimeo: 0 so_rcvtimeo: 0 tcp_nodelay: 0 so_keepalive: 0 tcp_keepidle: 0 tcp_keepintvl: 0 tcp_keepcnt: 0 OK
Feature Description	● Feedback Format Global SOCKET configuration: so_sndtimeo: <socket send timeout> so_rcvtimeo: <socket receive timeout> tcp_nodelay: <Disable/Enable Nagle Algorithm> so_keepalive: <Disables/enables socket keep-alive functionality> tcp_keepidle: <TCP keep-alive probe start time> tcp_keepintvl: <TCP keep-alive probe interval> tcp_keepcnt: <Number of TCP keep-alive probes> OK ● Error code 1: No parameters required

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.5.3 Configuring TCP/UDP/SSL Server Listening Service on AT+SKTSERVER

Instruction format	AT+SKTSERVER=<link_id>,<conn_type>[,<cert_index>],<src_port>,<auto_rcv>
Execution Example	● Send: AT+SKTSERVER=1,1,,1234,1 ● answer: OK
Parameter Description	● <link_id> is the connection ID specified by the user. Value range: [0, MEMP_NUM_NETCONN-1], currently MEMP_NUM_NETCONN is configured as 20 in the lwip library. ● <conn_type> Connection type 0: UDP-based socket server 1: TCP-based socket server 2: TLS-based socket server (provides server certificates for client verification) 3: TLS-based socket server (provides server certificates for client verification and uses CA certificates to verify client certificates) ● <cert_index> Security certificate suite index number, referencing AT security certificates. Value range: [1-10] This parameter is required only if <conn_type> is 2 or 3. ● <src_port> The local port that the server to be created will listen on. Value range: [1, 65535] ● <auto_rcv> Enables/disables automatic reception. 0: Not enabled 1: Enable
Feature Description	● Error code 1: Input parameter error 2: Failed to create TCP server 3: The task to create automatic TCP client connection reception and automatic data reception from all TCP clients failed. 4: Failed to create UDP server 5: Failed to create a task to automatically receive UDP client data. 6: Failed to retrieve SERVER_CERT/SERVER_KEY/SERVER_CA from Flash. 7: Memory allocation failed 8: Execution of mbedtls_ssl_config_defaults failed. 9: Failed to execute mbedtls_x509_crt_parse 10: Execution of mbedtls_ssl_conf_own_cert failed. 11: Execution of mbedtls_net_bind failed. 12: The task to create automatic TLS client connection reception and automatic data reception from all TLS clients failed. ● If a client connects to this TLS server, an unsolicited message will pop up in the log window.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	[$][SKT][EVENT]: A client[link_id:1,seed,tls,dst_address:192.168.0.103,dst_port:49946] connected to server[link_id:0]

1.5.4 Establishing TCP/UDP/SSL Client Connections with AT+SKTCLIENT

Instruction format	AT+SKTCLIENT=<link_id>,<conn_type>[,<cert_index>],<dst_host>,<dst_port>[,<src_port>],<auto_rcv>
Execution Example	● Send: AT+SKTCLIENT=1,1,,192.168.0.100,6666,,1 ● answer: OK
Parameter Description	● <link_id> is the connection ID specified by the user. Value range: [0, MEMP_NUM_NETCONN-1], currently MEMP_NUM_NETCONN is configured as 20 in the lwip library. ● <conn_type> Connection type 0: UDP-based socket client 1: TCP-based socket client 2: TLS-based socket clients (without certificate verification) 3: TLS-based socket client (verifies server certificate using CA certificate) 4: TLS-based socket client (provides client certificate for server verification) 5: TLS-based socket clients (two-way authentication, including 3 and 4) ● <cert_index> Security certificate suite index number, referencing AT security certificates. Value range: [1-10] This parameter is required only if <conn_type> is 3, 4, or 5. ● <dst_host> Remote server address Supports IPv4 address or domain name formats (e.g., 192.168.1.100 or www.example.com). The maximum length of a domain name is 255 characters. ● <dst_port> Remote server port Value range: [1, 65535] ● <src_port> The local port that the UDP client binds to (for UDP only). Value range: [1, 65535] ● <auto_rcv> Enables/disables automatic receive function 0: Not enabled 1: Enable
Feature Description	● Error code 1: Input parameter error 2: Failed to create TCP client 3: Failed to create a task to automatically receive TCP data as a TCP client. 4: Failed to create UDP client 5: Failed to create a task to automatically receive UDP data as a UDP client. 6: Failed to retrieve CLIENT_CERT/CLIENT_KEY/CLIENT_CA from Flash.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	7: Memory allocation failed 8: Execution of mbedtls_ssl_config_defaults failed. 9: Failed to execute mbedtls_x509_crt_parse 10: Execution of mbedtls_ssl_conf_own_cert failed. 11: Execution of mbedtls_ssl_setup failed. 12: Execution of inet_ntoa_r failed. 13: Execution of mbedtls_net_connect failed. 14: Execution of mbedtls_ssl_handshake failed. 15: Failed to create a task to automatically receive TLS data as a TLS client.

1.5.5 AT+SKTSEND sends data via a specified connection ID.

Instruction format	AT+SKTSEND=<link_id>,<data_size>[,<dst_ip>,<dst_port>],<data>
Execution Example	● Send: AT+SKTSEND=0,6,,,Hello! ● answer: OK
Parameter Description	● <link_id> A connection ID that has been successfully created Value range: [0, MEMP_NUM_NETCONN-1], currently MEMP_NUM_NETCONN is configured as 20 in the lwip library. ● <data_size> The number of bytes of data the user wants to send. Maximum value: Determined by subtracting the number of characters before the <data> field from UART_LOG_CMD_BUFLEN. The maximum value of UART_LOG_CMD_BUFLEN is 2000. Refer to the configuration project (menuconfig), enter the CONFIG AT CMD configuration and check Enable Longer CMD. Refer to the command length section for information on command length. ● <dst_ip> Target IPv4 address (for UDP servers only) For example: 192.168.1.100 ● <dst_port> Target port number (for UDP servers only) Value range: [1, 65535] ● <data> The actual data to be sent, the length of which is specified by <data_size>.
Feature Description	● Error code 1: Input parameter error 2: Data transmission failed.

1.5.6 AT+SKTREAD reads cached data from a specified connection ID

Instruction format	AT+SKTREAD=<link_id>,<data_size>
Execution	● Send: AT+SKTREAD=1,10

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Example	● answer: +SKTREAD:1,6,Hello! OK
Parameter Description	● <link_id> A connection ID that has been successfully created Value range: [0, MEMP_NUM_NETCONN-1], currently MEMP_NUM_NETCONN is configured as 20 in the lwip library. ● <data_size> The number of bytes of data the user wants to read. ● <actual_recv_size> The number of bytes of data actually read ● <udp_dstip> The IPv4 address of the target UDP client/server (UDP only). For example: 192.168.1.100 ● <dst_port> The port number of the target UDP client/server (UDP only) Value range: [1, 65535] ● <data> represents the actual data that has been read; its length is given by <actual_recv_size>.
Feature Description	● Error code 1: Input parameter error 2: Memory allocation failed 3: select() failed 4: Data read failed ● Successful read feedback format +SKTREAD:<link_id>,<actual_recv_size>[,<udp_dstip>,<udp_dstport>],<data> OK

1.5.7 AT+SKTSENDRAW transparent transmission mode sends data via a specified connection ID.

Instruction format	AT+SKTSENDRAW=<link_id>,<data_size>[,<dst_ip>,<dst_port>]
Execution Example	● Send: AT+SKTSENDRAW=0,6 ● answer: >>> Hello! OK
Parameter Description	● <link_id> A connection ID that has been successfully created Value range: [0, MEMP_NUM_NETCONN-1], currently MEMP_NUM_NETCONN is configured as 20 in the lwip library. ● <data_size> The number of bytes of data the user wants to send . ● <dst_ip> Target IPv4 address (for UDP servers only) ● <dst_port> Target port number (for UDP servers only) Value range: [1, 65535] ● <data> The actual data to be sent, the length of which is determined by data_size.
Feature Description	● Error code 1: Input parameter error

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	2: Data transmission failed. 3: Memory allocation failed 4: Failed to acquire data in Transparent Transmission (TT) mode

1.5.8 Use AT+SKTSTATE to query the status information of all active connection IDs.

Instruction format	AT+SKTSTATE
Execution Example	● Send: AT+SKTSTATE ● answer: link_id:0,server,tcp,src_address:192.168.0.109,src_port:6666,dst_address:0.0.0.0,dst_port:0,socket_fd:0 link_id:3,seed,tcp,src_address:0.0.0.0,src_port:0,dst_address:192.168.0.103,dst_port:49941,socket_fd:3 link_id:1,server,udp,src_address:192.168.0.109,src_port:12345,dst_address:0.0.0.0,dst_port:0,socket_fd:1 link_id:2,client,ssl,src_address:0.0.0.0,src_port:0,dst_address:192.168.0.103,dst_port:4433,socket_fd:2 OK
Parameter Description	● <link_id> A connection ID that has been successfully created Value range: [0, MEMP_NUM_NETCONN-1], currently MEMP_NUM_NETCONN is configured as 20 in the lwip library. ● <role> Socket role, in string format server client Seed ● <protocol> String format, corresponding to the protocol used by the connection ID. TCP UDP tls ● <ip> IPv4 address (e.g., 192.168.1.100) ● <port> Port number Value range: [1, 65535] ● <fd> The socket fd value created by the lwip library.
Feature Description	

1.5.9 AT+SKTDEL deletes the specified connection ID

Instruction format	AT+SKTDEL=<link_id>
Execution Example	● Send: AT+SKTDEL=0 ● answer: OK

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Parameter Description	● <link_id> A connection ID that has been successfully created Value range: [0, MEMP_NUM_NETCONN-1], currently MEMP_NUM_NETCONN is configured as 20 in the lwip library.
Feature Description	● Error code 1: Input parameter error

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.6 MQTT AT Command Set

MQTT AT error codes

Error code	describe
1	Parameter error
2	Memory allocation failed
3	Connection error
4	Disconnection error
5	Failed to enter TT mode
6	Certificate suite acquisition failed
7	Not connected
8	Connected
9	Release failed
10	Subscription failed
11	Unsubscription failed
12	Subscribed
13	Waiting timeout
14	Thread creation failed
15	Network connection error

1.6.1 Configuring MQTT Connection Parameters for a Specified Connection ID using AT+MQTTCFG

Instruction format	AT+MQTTCFG=<link_id>[,<keepalive>][,<cmd_timeout>][,<clean_session>][,<lwt_topic>,<lwt_msg>][,<lwt_qos>][,<lwt_retain>]
Execution Example	● Send: AT+MQTTCFG=0,20,60,0,topic/lwt,This is a lwt message,0,0 ● answer: OK
Parameter Description	● <link_id> Connection ID Value range [0,3] ● <keepalive> Ping timeout (seconds) Default value: 60 (heartbeat cycles) ● <cmd_timeout> Connection/subscription timeout (seconds) Default value: 60 ● <clean_session> Clean up session options 0: Disable session cleanup 1: Enable session cleanup Default value: 1 ● <lwt_topic> LWT Message Topic ● <lwt_msg> LWT message

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● <lwt_qos> LWT message QoS level Value range [0,2] Default value: 0 ● <lwt_retain> LWT message retention option Value range [0,1] Default value: 0
Feature Description	● Error code 1: Parameter error 2: Memory allocation failed 8: Proxy server connected

1.6.2 Connecting to the MQTT Proxy Server using AT+MQTTCONN

Instruction format	AT+MQTTCONN=<link_id>,<host>,[<port>],<client_id>,<conn_type>[,<cert_index>,<username>,<password>]
Execution Example	● Send: AT+MQTTCONN=0,broker.hivemq.com,1883,publisher,1 ● answer: OK [$][MQTT][EVENT]:linkid:0, connected
Parameter Description	● <link_id> Connection ID Value range [0,3] ● <host> Proxy server domain Maximum length: 128 bytes ● <port> Proxy server port Value range [0, 65535] Default value: 0 0: The default port for MQTT is 1883, and the default port for MQTTS is 8883. ● <client_id> Client ID Maximum length: 256 bytes ● <conn_type> Connection type 1: MQTT 2: MQTTS (No certificate verification) 3: MQTTS (Verify server certificate using CA certificate) 4: MQTTS (provides client certificates for server verification) 5: MQTTS (Two-way authentication, a combination of 3 and 4) ● <cert_index> Index of security certificate suites, see AT Security Certificates. Value range [1-10] It is only needed when <conn_type> is 3-5. ● <username> Login Username Maximum length: 64 bytes ● <password> Login password

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Maximum length: 64 bytes
Feature Description	● Error code 1: Parameter error 2: Memory allocation failed 3: Connection error 8: Proxy server connected 14: Main task creation failed. ● Information Feedback Instructions A successful connection will display the following message: [$][MQTT][EVENT]:linkid:<link_id>, connected If the connection fails or an error occurs, the following message will be displayed: [$][MQTT][EVENT]:linkid:<link_id>, ERROR:<err_no>

1.6.3 AT+MQTTDISCONN closes the specified MQTT connection and releases resources.

Instruction format	AT+MQTTDISCONN=<link_id>
Execution Example	● Send: AT+MQTTDISCONN=0 ● answer: OK
Parameter Description	● <link_id> Connection ID Value range [0,3]
Feature Description	● Error code 1: Parameter error 4: Connection failed.

1.6.4 Subscribing to MQTT topics using AT+MQTTSUB

Instruction format	AT+MQTTSUB=<link_id>,<topic>,<qos>
Execution Example	● Send: AT+MQTTSUB=0,LASS/Test/Pm25Ameba/#,1 ● answer: OK [$][MQTT][EVENT]:linkid:0, subscribed
Parameter Description	● <link_id> Connection ID Value range [0,3] ● <topic> Topics to be subscribed to Maximum length: 128 bytes ● QoS of the topic to be subscribed to Value range [0,2]

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Feature Description	● Supports up to 5 themes to subscribe to ● If you have already subscribed to this topic, you will be returned with: OK ALREADY SUBCRIBE ● Feedback message Upon successful subscription, the following message will be displayed: [$][MQTT][EVENT]:linkid:<link_id>, subscribed When you receive an MQTT message from the subscribed topic, you will see the following notification: [$][MQTT][DATA][<link_id>][<topic>][<message ID>][<message length>]:<message> If an error occurs, the following message will appear. [$][MQTT][EVENT]:linkid:<link_id>, ERROR:<err_no>

1.6.5 AT+MQTTUNSUB Unsubscribe from a Specified Topic

Instruction format	AT+MQTTUNSUB=<link_id>,<topic>
Execution Example	● Send: AT+MQTTUNSUB=0,LASS/Test/Pm25Ameba/# ● answer: OK [$][MQTT][EVENT]:linkid:0, unsubscribed
Parameter Description	● <link_id> Connection ID Value range [0,3] ● <topic> Topics that need to be unsubscribed Maximum length: 128 bytes
Feature Description	● Feedback message When you successfully unsubscribe, you will see the following message: [$][MQTT][EVENT]:linkid:<link_id>, unsubscribed If an error occurs, the following message will appear: [$][MQTT][EVENT]:linkid:<link_id>, ERROR:<err_no>

1.6.6 AT+MQTTPUB Message Publishing to Specific Topics

Instruction format	AT+MQTTPUB=<link_id>,<msg_id>,<topic>,<msg>,<qos>,<retain>
Execution Example	● Send: AT+MQTTPUB=0,1,LASS/Test/Pm25Ameba/FT1_018,This is a test,1,1 ● answer: OK [$][MQTT][EVENT]:linkid:0, published
Parameter Description	● <link_id> Connection ID Value range [0,3] ● <msg_id> Message ID Value range [0, 65535] ● <topic> Target topic

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Maximum length: 128 bytes It is recommended to fill in the topics you have previously subscribed to. ● <msg> Message content Maximum length: 512 bytes ● <qos> Message QoS Level It can be set to 0, 1, 2 Default value: 0 ● <retain> Retain value Can be set to 0 or 1 Default value: 0
Feature Description	● Feedback message When the placement is successful, the following message will appear: [$][MQTT][EVENT]:linkid:<link_id>, published If an error occurs, the following message will appear. [$][MQTT][EVENT]:linkid:<link_id>, ERROR:<err_no>

1.6.7 Publishing messages to a specific topic in AT+MQTTPUBRAW pass-through mode

Instruction format	AT+MQTTPUBRAW=<link_id>,<msg_id>,<topic>,<length>,<qos>,<retain>
Execution Example	● Send: AT+MQTTPUBRAW=0,2,LASS/Test/Pm25Ameba/FT1_018,14,1,1 ● answer: >>> This is a test OK [$][MQTT][EVENT]:linkid:0, published
Parameter Description	● <link_id> Connection ID Value range [0,3] ● <msg_id> Message ID Value range [0, 65535] ● <topic> Target topic Maximum length: 128 bytes It is recommended to fill in the topics you have previously subscribed to. ● <length> Message length (bytes) Maximum length: 10240 bytes ● <qos> Message QoS Level It can be set to 0, 1, 2 Default value: 0 ● <retain> Retain value

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Can be set to 0 or 1 Default value: 0
Feature Description	● It will wait for the data sent to match the settings before automatically exiting the pass-through mode.

1.6.8 AT+MQTTRESET resets all MQTT instances and releases resources.

Instruction format	AT+MQTTRESET
Execution Example	● Send: AT+MQTTRESET ● answer: OK
Feature Description	● If a running MQTT instance exists, the connection will be immediately closed and all related resources will be released.

1.6.9 Use AT+MQTTQUERY to query the status and configuration information of a specified connection.

Instruction format	AT+MQTTQUERY=<link_id>
Execution Example	● Send: AT+MQTTQUERY=0 ● answer: link_id: 0 state: 1 conn_type: 1 host: broker.hivemq.com port: 1883 client_id: publisher username: rs password: 12345678 cmd_timeout: 60 keepalive: 20 disable_clean_session: 0 LWT: 1 lwt_qos: 0 lwt_retain: 0 lwt_topic: topic/lwt lwt_message: This is a lwt message OK

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Parameter Description	● <link_id> Connection ID Value range [0,3]
Feature Description	

Version 1.6.10 of AT+MQTTSUBQUERY lists all subscribed topics.

Instruction format	AT+MQTTSUBQUERY=<link_id>
Execution Example	● Send: AT+MQTTSUBQUERY=0 ● answer: link_id: 0 state: 1 topic: LASS/Test/Pm25Ameba/#, qos: 1 OK
Parameter Description	● <link_id> Connection ID Value range [0,3]
Feature Description	

1.7 HTTP AT Command Set

1.7.1 Setting up global HTTP configuration using AT+HTTPCONF

Instruction format	AT+HTTPCONF=<timeout>[,<port>]
Execution Example	● Send: AT+HTTPCONF=1,12345 ● answer: OK
Parameter Description	● <timeout> HTTP receive/send timeout (seconds) Default value: 10 ● <port> HTTP server port Range: [0, 65535] (default value is 0) 0: HTTP uses port 80, HTTPS uses port 443.
Feature Description	● Error code 1: Parameter error

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.7.2 Setting HTTP Global Request Headers in AT+HTTPHEADER Pass-through Mode

Instruction format	AT+HTTPHEADER=<req_header_len>
Execution Example	● Send: AT+HTTPHEADER=17 ● answer: >>> Connection: close OK
Parameter Description	● <req_header_len: The length of a single HTTP request header 0: Clear and reset all set HTTP request headers. No length limit ● <req_header> The actual HTTP request header string.
Feature Description	● Error code 1: Parameter error 2: Memory allocation failed 3: Data acquisition failed in pass-through mode.

1.7.3 AT+HTTPQUERY query HTTP global configuration

Instruction format	AT+HTTPQUERY
Execution Example	● Send: AT+HTTPQUERY ● answer: Global HTTP configuration: http_timeout: 10 http_server_port: 0 http_req_header_cnt: 2 http_req_header_list: Connection: close Accept-Encoding: gzip, deflate OK
Feature Description	● Error code 1: No parameters required

1.7.4 Retrieving HTTP Resources using AT+HTTPGET

Instruction	AT+HTTPGET=<host>,<path>,<conn_type>[,<cert_index>][,<req_header_cnt>,<req_header>...<req_heade

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

format	r>]

Execution Example	● Send: AT+HTTPGET=httpbin.org,/get,1
	● answer:
	+HTTPGET: HEADER LEN = 230
	+HTTPGET: HEADER DUMP START
	HTTP/1.1 200 OK
	Date: Mon, 06 Jan 2025 07:53:34 GMT
	Content-Type: application/json
	Content-Length: 196
	Connection: keep-alive
	Server: gunicorn/19.9.0
	Access-Control-Allow-Origin: *
	Access-Control-Allow-Credentials: true
	
	+HTTPGET: HEADER DUMP END
	+HTTPGET: BODY DUMP START
	{
	"args": {},
	"headers": {
	"Host": "httpbin.org",
	"X-Amzn-Trace-Id": "Root=1-677b8bfe-6faa4a9d7264e26b46397567"
	},
	"origin": "98.98.48.11",
	"url": "http://httpbin.org/get"
	}
	+HTTPGET: BODY DUMP END
	+HTTPGET: BODY LEN = 196
	
	OK

Parameter Description	● <host> Target host. The http:// or https:// prefix is not required in the input parameter. Supports domain names or IPv4 addresses Maximum length: 256 characters ● <path> resource path default value:"/" ● <conn_type> HTTP connection type 1: HTTP 2: HTTPS (no certificate verification) 3: HTTPS (verify server certificates using CA certificates) 4: HTTPS (provides the client certificate to the server for verification) 5: HTTPS (mutual authentication, a combination of 3 and 4) ● <cert_index> Index of security certificate suites, see AT Security Certificates. [1-10], only required when <conn_type> is 3, 4, or 5. ● <req_header_cnt> The total number of actual request headers. ● <req_header> The actual request header string. The number of request headers is determined by

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	<req_header_cnt>, and it is only carried in the headers of the current method.
Feature Description	● Error code 1: Parameter error 2: Memory allocation failed 3: Failed to read security certificate suite 4: Failed to send HTTP GET header. 5: Failed to read HTTP response.

1.7.5 Sending HTTP POST requests in AT+HTTPPOST pass-through mode

Instruction format	AT+HTTPPOST=<host>,<path>,<conn_type>[,<cert_index>],<body_size>[,<req_header_cnt>,<req_header>...<req_header>]

Execution Example	● Send: AT+HTTPPOST=httpbin.org,/post,2,,35,2,Connection: close,User-Agent: Ameba_iot
	● answer:
	>>>
	param1=test_data1&param2=test_data2
	
	+HTTPPOST: HEADER LEN = 225
	+HTTPPOST: HEADER DUMP START
	HTTP/1.1 200 OK
	Date: Mon, 06 Jan 2025 07:49:09 GMT
	Content-Type: application/json
	Content-Length: 359
	Connection: close
	Server: gunicorn/19.9.0
	Access-Control-Allow-Origin: *
	Access-Control-Allow-Credentials: true
	
	+HTTPPOST: HEADER DUMP END
	+HTTPPOST: BODY DUMP START
	{
	"args": {},
	"data": "param1=test_data1&param2=test_data2",
	"files": {},
	"form": {},
	"headers": {
	"Content-Length": "35",
	"Host": "httpbin.org",
	"User-Agent": "Ameba_iot",
	"X-Amzn-Trace-Id": "Root=1-677b8af2-73ea1c6e376e5d1e672893b5"
	},

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	"json": null, "origin": "98.98.48.11", "url": "https://httpbin.org/post" } +HTTPPOST: BODY DUMP END +HTTPPOST: BODY LEN = 357 OK
Parameter Description	● <host> Target host. The http:// or https:// prefix is not required in the input parameter. Supports domain names or IPv4 addresses Maximum length: 256 characters ● <path> resource path default value:"/" ● <conn_type> HTTP connection type 1: HTTP 2: HTTPS (no certificate verification) 3: HTTPS (verify server certificates using CA certificates) 4: HTTPS (provides the client certificate to the server for verification) 5: HTTPS (mutual authentication, a combination of 3 and 4) ● <cert_index> Index of security certificate suites, see AT Security Certificates. [1-10], only required when <conn_type> is 3, 4, or 5. ● <body_size> The length of the HTTP data to be POSTed. Length unlimited ● <req_header_cnt> The total number of actual request headers. ● <req_header> The actual request header string. The number of request headers is determined by <req_header_cnt>, and it is only carried in the headers of the current method.
Feature Description	● Error code 1: Parameter error 2: Memory allocation failed 3: Failed to read security certificate suite 4: Failed to send HTTP POST header. 5: Data retrieval failed in TT mode. 6: Failed to send HTTP POST body data. 7: Failed to read HTTP response ● If the command includes request headers in its parameters, these headers will be added to the HTTP message along with the global request headers configured in AT+HTTPHEADER.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.7.6 Sending HTTP PUT requests in AT+HTTPPUT pass-through mode

Instruction format	AT+HTTPPUT=<host>,<path>,<conn_type>[,<cert_index>],<body_size>[,<req_header_cnt>,<req_header>...<req_header>]

Execution Example	● Send: AT+HTTPPUT=httpbin.org,/put,2,,35,2,Connection: close,User-Agent: Ameba_iot
	● answer:
	>>>
	param1=test_data1&param2=test_data2
	
	+HTTPPUT: HEADER LEN = 225
	+HTTPPUT: HEADER DUMP START
	HTTP/1.1 200 OK
	Date: Mon, 06 Jan 2025 07:50:05 GMT
	Content-Type: application/json
	Content-Length: 358
	Connection: close
	Server: gunicorn/19.9.0
	Access-Control-Allow-Origin: *
	Access-Control-Allow-Credentials: true
	
	+HTTPPUT: HEADER DUMP END
	+HTTPPUT: BODY DUMP START
	{
	"args": {},
	"data": "param1=test_data1&param2=test_data2",
	"files": {},
	"form": {},
	"headers": {
	"Content-Length": "35",
	"Host": "httpbin.org",
	"User-Agent": "Ameba_iot",
	"X-Amzn-Trace-Id": "Root=1-677b8b2a-30de8616008ca27e176ccda2"
	},
	"json": null,
	"origin": "98.98.48.11",
	"url": "https://httpbin.org/put"
	}
	+HTTPPUT: BODY DUMP END
	+HTTPPUT: BODY LEN = 356
	
	OK

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Parameter Description	● <host> Target host. The http:// or https:// prefix is not required in the input parameter. Supports domain names or IPv4 addresses Maximum length: 256 characters ● <path> resource path default value:"/" ● <conn_type> HTTP connection type 1: HTTP 2: HTTPS (no certificate verification) 3: HTTPS (verify server certificates using CA certificates) 4: HTTPS (provides the client certificate to the server for verification) 5: HTTPS (mutual authentication, a combination of 3 and 4) ● <cert_index> Index of security certificate suites, see AT Security Certificates. [1-10], only required when <conn_type> is 3, 4, or 5. ● <body_size> The length of the HTTP data to be PUT Length unlimited ● <req_header_cnt> The total number of actual request headers. ● <req_header> The actual request header string. The number of request headers is determined by <req_header_cnt>, and it is only carried in the headers of the current method.
Feature Description	● Error code 1: Parameter error 2: Memory allocation failed 3: Failed to read security certificate suite 4: Failed to send HTTP PUT header. 5: Data retrieval failed in TT mode. 6: Failed to send HTTP PUT body data 7: Failed to read HTTP response ● If the command includes request headers in its parameters, these headers will be added to the HTTP message along with the global request headers configured in AT+HTTPHEADER.

1.7.7 AT+HTTPDEL deletes the specified HTTP resource.

Instruction format	AT+HTTPDEL=<host>,<path>,<conn_type>[,<cert_index>][,<req_header_cnt>,<req_header>...<req_header>]
Execution Example	● Send: AT+HTTPDEL=httpbin.org,/delete,2 ● answer: +HTTPDEL: HEADER LEN = 230 +HTTPDEL: HEADER DUMP START HTTP/1.1 200 OK Date: Mon, 06 Jan 2025 07:51:15 GMT Content-Type: application/json Content-Length: 263 Connection: keep-alive

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Server: gunicorn/19.9.0
	Access-Control-Allow-Origin: *
	Access-Control-Allow-Credentials: true
	
	+HTTPDEL: HEADER DUMP END
	+HTTPDEL: BODY DUMP START
	{
	"args": {},
	"data": "",
	"files": {},
	"form": {},
	"headers": {
	"Host": "httpbin.org",
	"X-Amzn-Trace-Id": "Root=1-677b8b73-43851d66543b1a534698c987"
	},
	"json": null,
	"origin": "98.98.48.11",
	"url": "https://httpbin.org/delete"
	}
	+HTTPDEL: BODY DUMP END
	+HTTPDEL: BODY LEN = 263
	
	OK

Parameter Description	● <host> Target host. The http:// or https:// prefix is not required in the input parameter. Supports domain names or IPv4 addresses Maximum length: 256 characters ● <path> resource path default value:"/" ● <conn_type> HTTP connection type 1: HTTP 2: HTTPS (no certificate verification) 3: HTTPS (verify server certificates using CA certificates) 4: HTTPS (provides the client certificate to the server for verification) 5: HTTPS (mutual authentication, a combination of 3 and 4) ● <cert_index> Index of security certificate suites, see AT Security Certificates. [1-10], only required when <conn_type> is 3, 4, or 5. ● <req_header_cnt> The total number of actual request headers. ● <req_header> The actual request header string. The number of request headers is determined by <req_header_cnt>, and it is only carried in the headers of the current method.
Feature Description	● Error code 1: Parameter error 2: Memory allocation failed 3: Failed to read security certificate suite 4: Failed to send HTTP DELETE header.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	5: Failed to read HTTP response. ● If the command includes request headers in its parameters, these headers will be added to the HTTP message along with the global request headers configured in AT+HTTPHEADER.

1.8 WebSocket AT Command Set

1.8.1 Configuring Global Parameters for the WebSocket Client using AT+WSGLCFG

Instruction format	AT+WSGLCFG=[<connect_timeout>][,<recv_timeout>][,<send_timeout>][,<send_blocktime>][,<keepalive_idle>][,<keepalive_interval>][,<keepalive_count>]
Execution Example	● Send: AT+WSGLCFG=5,2000,2000,0,300,5,3 ● answer: OK
Parameter Description	● <connect_timeout> Connection timeout (milliseconds) non-negative integers Default value: 0 (infinite wait) ● <recv_timeout> Receive timeout (milliseconds) non-negative integers Default value: 0 (no timeout) ● <send_timeout> Send timeout (milliseconds) non-negative integers Default value: 0 (no timeout) ● <send_blocktime> Send blocking time (milliseconds) non-negative integers The special value 0 indicates an immediate return. ● <keepalive_idle> TCP keepalive probe start time (seconds) Range [1, 7200] Default value: 0 (disabled) ● <keepalive_interval> Keep-alive detection interval (seconds) Range [1, 75] Default value: 0 (disabled) ● <keepalive_count> Keepalive probe count Range [1, 10], Default: 0 (Disabled)
Feature Description	● Error code 1: Parameter range error

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.8.2 Configuring WebSocket Client Parameters for a Specified Connection ID using AT+WSCFG

Instruction format	AT+WSCFG=<link_id>,<ping_interval>,<ping_timeout>[,<tx_buffer_size>][,<rx_buffer_size>][,<max_queue_size>][,<protocol>][,<version>][,<stable_buf_num>]
Execution Example	● Send: AT+WSCFG=0,30,120,2048,1500,5,"chat",13,3 ● answer: OK
Parameter Description	● <link_id> Connection ID Valid range: 0-2 ● <ping_interval> Ping heartbeat interval (seconds) Range [1, 7200] Default value: 10 ● <ping_timeout> Ping timeout duration (seconds) Range [1, 7200] Default value: 120 ● <tx_buffer_size> Size of the send data buffer (in bytes) Range [1500, 16384] Default value: 1500 ● <rx_buffer_size> Size of the receive data buffer (in bytes) Range [1500, 16384] Default value: 1500 ● <max_queue_size> Maximum queue depth Range [1,50] Default value: 3 ● <protocol> Subprotocol Type String format (e.g., "chat", "superchat") Default value: empty ● <version> Protocol version Currently only "13" is supported. Default value: 13 ● <stable_buf_num> Number of stable buffers Range [1, <max_queue_size>] Default value: 3
Feature Description	● Error code 1: Parameter out of range 2: Memory allocation failed

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.8.3 Configuring WebSocket Client Request Headers with AT+WSHEAD

Instruction format	AT+WSHEAD=<link_id>,<header_len>[,<header>]
Execution Example	● Send: AT+WSHEAD=1,26,Origin: http://example.com ● answer: OK
Parameter Description	● <link_id> Connection ID Valid range: 0-2 ● <header_len> Request header length The entire command is subject to command length limitations. 0: Clear all request headers ● <header> Request header data
Feature Description	● Error code 1: Parameter out of range 2: Memory allocation failed 3: The number of connectors exceeds the maximum value. ● Each connection ID supports a maximum of 10 request headers.

1.8.4 Configuring WebSocket Client Request Headers in AT+WSHEADRAW Pass-Through Mode

Instruction format	AT+WSHEADRAW=<link_id>,<header_len>
Execution Example	● Send: AT+WSHEADRAW=1,39 ● answer: >>> Sec-WebSocket-Protocol: chat, superchat OK
Parameter Description	● <link_id> Connection ID Valid range: 0-2 ● <header_len> Request header length The entire command is subject to command length limitations. 0: Clear all request headers
Feature Description	● Error code 1: Parameter out of range 2: Memory allocation failed

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	3: Failed to enter pass-through mode 4: The number of connectors exceeds the maximum value.

1.8.5 Establishing a WebSocket Client Connection using AT+WSCONN

Instruction format	AT+WSCONN=<link_id>,<host>[,<path>][,<port>],<conn_type>[,<cert_index>]
Execution Example	● Send: AT+WSCONN=0,echo.websocket.org,/chat,8433,3,1 ● answer: [$][WS][EVENT]:linkid:0, connected OK
Parameter Description	● <link_id> Connection ID Valid range: 0-2 ● <host> Target host, no ws:// or wss:// prefix is needed. Supports domain names or IPv4 addresses Maximum length: 128 characters ● <path> resource path default value:"/" Maximum length: 256 characters ● <port> Port Range: [0, 65535] (Default value 0) 0: WS uses port 80, WSS uses port 443. ● <conn_type> Connection type 1: WebSocket over TCP 2: Websocket over TLS (without certificate verification) 3: Websocket over TLS (Verifying server certificates using CA certificates) 4: Websocket over TLS (providing client certificates for server verification) 5: Websocket over TLS (mutual authentication, a combination of 3 and 4) ● <cert_index> Index of security certificate suites, see AT Security Certificates. [1-10], only required when <conn_type> is 3-5.
Feature Description	● Error code 1: Parameter out of range 2: Memory allocation failed 3: Socket creation failed. 4: Certificate loading failed. 5: Connection to server failed. 6: Connection task creation failed. ● Feedback message When the connection is successful, the following message will appear: [$][WS][EVENT]:linkid:<link_id>, connected

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	When the connection is lost, the following message will appear: [$][WS][EVENT]:linkid:<link_id>, disconnect When a PONG message is received, the following message will be displayed: [$][WS][EVENT]:linkid:%d, pong When a data packet is received, the following message will be displayed: [$][WS][DATA][<link_id>][<data length>]:<data>

1.8.6 Sending WebSocket data using AT+WSSEND

Instruction format	AT+WSSEND=<link_id>,<length>[,<use_mask>][,<opcode>],<data>
Execution Example	● Send: AT+WSSEND=0,11,1,1,HelloWorld! ● answer: OK
Parameter Description	● <link_id> Connection ID Valid range: 0-2 ● <length> Data length Maximum value: The size defined by the <buffer_size> parameter in AT+WSCFG ● <use_mask> data mask 0: Disable 1: Enabled (default) ● <opcode> Frame type 0: CONTINUATION frame 1: Text frame (default) 2: Binary frame 8: Close frames 9: PING frame 10: PONG frame ● <data> Data to be sent
Feature Description	● Error code 1: Parameter out of range 2: Connection unavailable

1.8.7 Sending WebSocket data in AT+WSSENDRAW pass-through mode

Instruction format	AT+WSSENDRAW=<link_id>,<length>[,<use_mask>][,<opcode>]
Execution Example	● Send: AT+WSSENDRAW=0,10,1,1 ● answer: >>> 1234567890

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	OK
Parameter Description	● <link_id> Connection ID Valid range: 0-2 ● <length> Data length No length limit ● <use_mask> data mask 0: Disable 1: Enabled (default) ● <opcode> Frame type 0: CONTINUATION frame 1: Text frame (default) 2: Binary frame 8: Close frames 9: PING frame 10: PONG frame
Feature Description	● Error code 1: Parameter out of range 2: Connection unavailable 3: Memory allocation failed 4: Failed to enter pass-through mode

1.8.8 AT+WSDISCONN Actively Closes WebSocket Client Connections with a Specified Connection ID

Instruction format	AT+WSDISCONN=<link_id>
Execution Example	● Send: AT+WSDISCONN=0 ● answer: OK [$][WS][EVENT]:linkid:0, disconnect
Parameter Description	● <link_id> Connection ID Valid range: 0-2
Feature Description	● Error code 1: Parameter out of range 2: Connection unavailable

1.8.9 AT+WSQUERY query for WebSocket client-related information

Instruction	AT+WSQUERY=[<link_id>]

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

format	
Execution Example	● Send: AT+WSQUERY=0 ● answer: Global config: connect_timeout: 3000 recv_timeout: 3000 send_timeout: 3000 send_blocktime: 4000 keepalive_idle: 0 keepalive_interval: 0 keepalive_count: 0 link_id: 0 status: 3 ping_interval: 20 ping_timeout: 60 buffer_size: 4096 max_queue_size: 12 protocol: version: 13 stable_buf_num: 8 header: Sec-WebSocket-Protocol: chat, superchat Origin: http://example.com OK
Parameter Description	● <link_id> Connection ID Valid range: 0-2
Feature Description	● Error code 1: Parameter out of range

1.8.10 AT+WSTXQUEUECHECK checks if the tx queue for the specified connection ID is full.

Instruction format	AT+WSTXQUEUECHECK=<link_id>
Execution Example	● Send: AT+WSQUERY=0 ● answer: OK Tx queue is not full
Parameter Description	● <link_id> Connection ID Valid range: 0-2
Feature	● Error code

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Description	1: Parameter out of range 2: Connection unavailable

1.8.11 Configuring WebSocket Server Parameters with AT+WSSRVCFG

Instruction format	AT+WSSRVCFG=[<max_conn>][,<ping_interval>][,<idle_timeout>][,<tx_size>][,<rx_size>]
Execution Example	● Send: AT+WSSRVCFG=2,30,90,16384,16384 ● answer: OK Tx queue is not full
Parameter Description	● <max_conn> Maximum number of connections Range [1,3] Default value: 3 ● <ping_interval> Ping packet interval (seconds) Range [1, 7200] Default value: 30 ● <idle_timeout> Idle timeout (seconds) The connection will be actively closed if there is no continuous communication for more than the idle timeout period. Range [0, 7200] Default value: 0 (no timeout) ● <tx_size> Send buffer (bytes) Range [1, 16384] Default value: 4096 ● <rx_size> Receive buffer (bytes) Range [1, 16384] Default value: 4096
Feature Description	● Error code 1: Parameter out of range

1.8.12 Start the WebSocket server with AT+WSSRVSTART

Instruction format	AT+WSSRVSTART=[<port>],<conn_type>[,<cert_index>]
Execution Example	● Send: AT+WSSRVSTART=0,1 ● answer: OK
Parameter Description	● <port> Port Range: [0, 65535] (Default value 0) 0: WS uses port 80, WSS uses port 443.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● <conn_type> Connection type 1: WebSocket Server over TCP 2: Websocket Server over TLS (provides server certificates for client verification) 3: Websocket Server over TLS (provides server certificate verification to client and verifies client certificate using CA certificate) ● <cert_index> Index of security certificate suites, see AT Security Certificates. [1-10], only required when <conn_type> is 2 or 3.
Feature Description	● Error code 1: Incorrect parameter range 2: The server is running. 3: Failed to obtain certificate suite 4: Memory allocation failed 5: Certificate setup failed 6: Server startup failed. ● Disconnection Reason List 1: Connection creation failed. 2: Connection initialization failed. 3: Client actively disconnects 4: Server actively disconnects 5: WS server handshake failed. 6: Select timeout 7: FD_ISSET failed 8: TLS handshake failed 9: Setting NONBLOCK failed 10: Read failed 11: Sending failed 12: Failed to set socket options 13: Timeout when obtaining PONG 14: Idle timeout 15: Server failed to send PONG. ● Feedback message After the server starts, the following message will appear. [$][WSSRV][EVENT]:start When a client successfully connects, the following message will appear. [$][WSSRV][EVENT]:linkid:<link_id>, connected When the connection is lost, the following message will appear. [$][WSSRV][EVENT]:linkid:<link_id>, disconnected, reason:<reason for disconnection> When a data packet is received, the following message will be displayed. [$][WSSRV][DATA][<link_id>][<opcode>][<data length>]:<data>

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.8.13 AT+WSSRVDISCONN disconnects the WebSocket client connection with the specified connection ID.

Instruction format	AT+WSSRVDISCONN=<link_id>
Execution Example	● Send: AT+WSSRVDISCONN=2 ● answer: OK
Parameter Description	● <link_id> Connection ID Range: [0,2]
Feature Description	● Error code 1: Incorrect parameter range 2: Connection unavailable

Version 1.8.14 AT+WSSRVSTOP shuts down the WebSocket server.

Instruction format	AT+WSSRVSTOP
Execution Example	● Send: AT+WSSRVSTOP ● answer: OK
Feature Description	● Feedback message After the server releases its resources, the following message will appear. If you need to restart the server, please ensure that you execute AT+WSSRVSTART after the message appears. [$][WSSRV][EVENT]:stop

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.8.15 AT+WSSRVSEND sends WebSocket data to the specified connection ID

Instruction format	AT+WSSRVSEND=<link_id>,<length>[,<opcode>],<data>
Execution Example	● Send: AT+WSSRVSEND=1,11,1,HelloWorld! ● answer: OK
Parameter Description	● <link_id> Connection ID Range: [0,2] ● <length> Data length Maximum value: The size defined by the <tx_buff> parameter in AT+WSSRVCFG ● <opcode> Frame type 0: CONTINUATION frame 1: Text frame (default) 2: Binary frame 8: Close frames 9: PING frame 10: PONG frame ● <data> Data to be sent
Feature Description	● Error code 1: Incorrect parameter range 2: Memory allocation failed

1.8.16 AT+WSSRVSENDRAW Pass-through Mode: Send WebSocket data to a specified connection ID

Instruction format	AT+WSSRVSENDRAW=<link_id>,<length>[,<opcode>]
Execution Example	● Send: AT+WSSRVSENDRAW=0,10,1 ● answer: >>> 1234567890 OK
Parameter Description	● <link_id> Connection ID Range: [0,2] ● <length> Data length Maximum value: The size defined by the <tx_buff> parameter in AT+WSSRVCFG

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● <opcode> Frame type 0: CONTINUATION frame 1: Text frame (default) 2: Binary frame 8: Close frames 9: PING frame 10: PONG frame
Feature Description	● Error code 1: Incorrect parameter range 2: Memory allocation failed 3: Failed to enter pass-through mode

1.8.17 Use AT+WSSRVQUERY to query WebSocket server-side related information.

Instruction format	AT+WSSRVQUERY
Execution Example	● Send: AT+WSSRVQUERY ● answer: port: 1000 max_conn: 3 ping_interval: 30 idle_timeout: 0 tx_size: 4096 rx_size: 4096 link_id: 0 status: 2 IP: 192.168.1.105 OK
Feature Description	

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.9 Bluetooth Transmission AT Command Set

Before initializing the transfer module, only commands starting with AT+BTDEMO are supported. To use other commands, the transfer module must be initialized first. It is recommended to use transfer_module to initialize the module .

Bluetooth transmission AT error code:

Error code	describe
1	General Error
2	Bluetooth protocol stack not enabled
3	Insufficient memory
4	Operating system operation error
5	Invalid AT command parameters
6	AT command broadcast length is invalid.
7	No BLE connection error
8	IRK not found error
9	No binding error
10	Bluetooth lower-layer protocol stack error
11	BLE connection creation timed out
12	Bluetooth transmission of AT commands is invalid.
13	Bluetooth AT command transmission timeout

1.9.1 AT+BTDEMO=transfer_module Initializes or deinitializes the transfer module

Instruction format	AT+BTDEMO=transfer_module,<op>
Execution Example	● Send: AT+BTDEMO=transfer_module, 1 ● answer: OK
Parameter Description	● <op> Initialize or deinitialize the transmission module 0: Deinitialize the transmission module 1: Initialize the transmission module
Feature Description	● Once enabled, it can be used as a master or slave device.

1.9.2 AT+BTDEMO=transfer_module,get_name retrieves the device name.

Instruction	AT+BTDEMO=transfer_module,get_name

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

format	
Execution Example	● Send: AT+BTDEMO=transfer_module,get_name ● answer: +BTDEMO:transfer_module,get_name, test OK
Parameter Description	● <name> Device Name
Feature Description	

1.9.3 AT+BTDEMO=transfer_module,set_name sets the device name.

Instruction format	AT+BTDEMO=transfer_module,set_name,<name>
Execution Example	● Send: AT + BTDEMO=transfer_module, set_name , test ● answer: +BTDEMO:transfer_module,get_name, test OK
Parameter Description	● <name> Device Name The length should not exceed 40 characters.
Feature Description	● This command only sets the device name. To modify the name in the broadcast data, use the command AT+BLEGAP=adv_data[,<data>] before starting the broadcast.

1.9.4 AT+BTDEMO=transfer_module,get_uuid to obtain UUID

Instruction format	AT+BTDEMO=transfer_module,get_uuid,<attr>
Execution Example	● Send: AT+BTDEMO=transfer_module,get_uuid,1 ● answer: +BTDEMO:transfer_module,get_uuid,1,0xd001 OK
Parameter Description	● <attr> Attribute Index 0: Main Service 1: Read characteristics 2: Write characteristics 3: Notification features 4: Indicator Characteristics ● UUID of the <uuid> attribute

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Feature Description	

1.9.5 AT+BTDEMO=transfer_module,set_uuid sets the UUID.

Instruction format	AT+BTDEMO=transfer_module,set_uuid,<attr>,<uuid>
Execution Example	● Send: AT+BTDEMO=transfer_module,set_uuid,1,0xd005 ● answer: OK
Parameter Description	● <attr> Attribute Index 0: Main Service 1: Read characteristics 2: Write characteristics 3: Notification features 4: Indicator Characteristics ● <uuid> The UUID that needs to be set
Feature Description	

1.9.6 AT+BTDEMO=transfer_module, read_val sets the read value for readable characteristics.

Instruction format	AT+BTDEMO=transfer_module,read_val,<length>,<data>
Execution Example	● Send: AT+BTDEMO=transfer_module,read_val, 10 , 00112233445566778899 ● answer: OK
Parameter Description	● <length Length of the value to be read The maximum value is 30 ● <data> Read value Hexadecimal numbers that do not contain 0x, for example: 112233445566778899aabb
Feature Description	● This command is used to set the read characteristic value for the module's default service. For user-defined read characteristic values, please use the command `AT+BTDEMO=transfer_module,char_read_val`.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.9.7 AT+BTDEMO=transfer_module,srv_add adds GATT service

Instruction format	AT+BTDEMO=transfer_module,srv_add,<service_uuid>
Execution Example	● Send: AT+BTDEMO=transfer_module,srv_add,0x7000 ● answer: OK
Parameter Description	● <service_uuid> Service UUID
Feature Description	● This means adding a GAT service under the previously defined service.

Version 1.9.8 AT+BTDEMO=transfer_module, char_add adds a feature to the GATT service.

Instruction format	AT+BTDEMO=transfer_module,char_add,<characteristic_uuid>,<characteristic_property>
Execution Example	● Send: AT+BTDEMO=transfer_module,char_add,0x6005,0x0A ● answer: OK
Parameter Description	● <characteristic_uuid> Characteristic UUID ● <characteristic_property> Characteristic Attribute 0x02: Reading Attributes 0x04: No response when writing property 0x08: Responsive write property 0x10: Notification Attributes 0x20: Indicator Attribute
Feature Description	● Please use this command after using AT+BTDEMO=transfer_module,srv_add and before initializing the transfer module.

Version 1.9.9 `AT+BTDEMO=transfer_module,srv_end` completes the addition of the GATT service.

Instruction format	AT+BTDEMO=transfer_module,srv_end
Execution Example	● Send: AT+BTDEMO=transfer_module,srv_end ● answer: +BTDEMO:transfer_module,srv_end,0,128

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	+BTDEMO:transfer_module,srv_end,1,2,3 +BTDEMO:transfer_module,srv_end,1,5 +BTDEMO:transfer_module,srv_end,1,7,8 OK
Parameter Description	● <flag> APP ID flag or feature index flag 0: App ID identifier 1: Feature Index Flag ● <app_id> Application ID ● <value_index> Feature index ● <cccd_index> Feature CCCD Index
Feature Description	● Please use this command after using AT+BTDEMO=transfer_module,srv_add and before initializing the transfer module. ● The response to this command includes the app_id and characteristic index for each feature added using the AT+BTDEMO=transfer_module,char_add command. The response distinguishes the app_id and characteristic index using a <flag>. A flag of 1 corresponds to the features added using the AT+BTDEMO=transfer_module,char_add command in the order they were added. For example, if the AT+BTDEMO=transfer_module,char_add command is used three times to add characteristic_1, characteristic_2, and characteristic_3 respectively, the response to the AT+BTDEMO=transfer_module,srv_end command will contain three response messages starting with +BTDEMO:transfer_module,srv_end,1. These three responses correspond to the characteristic indices of characteristic_1, characteristic_2, and characteristic_3, respectively.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.9.10 AT+BTDEMO=transfer_module,srv_del Delete all GATT services

Instruction format	AT+BTDEMO=transfer_module,srv_del
Execution Example	● Send: AT+BTDEMO=transfer_module,srv_del ● answer: OK
Feature Description	● Please use this command before initializing the transport module.

1.9.11 AT+BTDEMO=transfer_module, char_read_val is the characteristic value set for GATT readable service.

Instruction format	AT+BTDEMO=transfer_module,char_read_val,<app_id>,<char_index>,<read_len>,<read_value>
Execution Example	● Send: AT+BTDEMO=transfer_module,char_read_val,128,2,10,33445566778899001122 ● answer: ● OK
Parameter Description	● <app_id> Application ID ● <char_index> Read feature index ● <read_len> is the length to be read. The maximum value is 30 ● <read_value> Reads the value Hexadecimal numbers that do not contain 0x, for example: 112233445566778899aabb
Feature Description	● Please use this command before adding the user-defined GATT service and initializing the transport module.

Version 1.9.12 `AT+BLEGAP=addr` retrieves the Bluetooth MAC address.

Instruction format	AT+BLEGAP=addr
Execution Example	● Send: AT+BLEGAP=addr ● answer: +BLEGAP:addr,00:11:22:33:44:55(public) OK
Parameter Description	● <addr> Local address string For example: 00:11:22:33:44:55

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● <addr_type> Local address type string public: public address random: random address public-id: Public identity address type random-id: Random identity address type
Feature Description	

1.9.13 AT+BLEGAP=rand_addr sets a random Bluetooth address.

Instruction format	AT+BLEGAP=rand_addr,<addr>[,<type>]
Execution Example	● Send: AT+BLEGAP=rand_addr,001122334455 ● answer: ● +BLEGAP:rand_addr,001122334455 ● OK
Parameter Description	● <addr> The random address to set Address string, for example: 001122334455 ● <type> Address type 0: Static random device address 1: Unresolved random device address 2: Resolves random device addresses ● <rand_addr> successfully set the random address Address string, for example: 00:11:22:33:44:55
Feature Description	

Version 1.9.14 `AT+BLEGAP=mtu_size` retrieves the MTU size.

Instruction format	AT+BLEGAP=mtu_size,<conn_handle>
Execution Example	● Send: AT+BLEGAP=mtu_size,16 ● answer: +BLEGAP:mtu_size,16,256 OK
Parameter Description	● <conn_handle> connection handle ● <mtu_size> The MTU size of this connection
Feature	

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Description	

Version 1.9.15 AT+BLEGAP=set_mtu sets the MTU size.

Instruction format	AT+BLEGAP=set_mtu,<mtu_size>
Execution Example	● Send: AT+BLEGAP=set_mtu,512 ● answer: ● OK
Parameter Description	● <mtu_size> The size of the MTU to set.
Feature Description	● Set the MTU size before establishing a connection.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Version 1.9.16 AT+BLEGAP=adv enables broadcasting and sets broadcast parameters.

Instruction format	AT+BLEGAP=adv,<op>[,<type>,<own_addr_type>,<filter_policy>[,<peer_addr_type>,<peer_addr>[,<interval_min>,<interval_max>,<channel_map>]]]
Execution Example	● Send: AT+BLEGAP=adv,1 ● answer: ● OK
Parameter Description	● <op> Broadcast Options 1: Start broadcasting 0: Stop broadcasting ● <type> Broadcast type 0: Can connect to non-directional broadcasts 1: Can connect to high duty cycle directional broadcasting 2: Can scan non-directional broadcasts 3: Cannot connect to non-directional broadcasts 4: Can connect to low duty cycle directional broadcasting ● <own_addr_type> Local address type 0: Public Address 1: Random address 2: Publicly Identifiable Address Type 3: Random Identity Address Type ● <filter_policy> Broadcast filtering policy 0: Handle scanning and connection requests from all devices. 1: Process connection requests from all devices, but only process scan requests from devices on the whitelist. 2: Process scan requests from all devices, but only process connection requests from devices on the whitelist. 3: Only process scan and connection requests from devices on the whitelist. ● <peer_addr_type> Peer address type 0: Public Address 1: Random address 2: Publicly Identifiable Address Type 3: Random Identity Address Type ● <peer_addr> peer address ● <interval_min> Minimum broadcast interval [0x20, 0x4000] Minimum broadcast interval = <interval_min> * 0.625ms, ranging from 20ms to 10.24s. ● <interval_max> Maximum broadcast interval [0x20, 0x4000] Maximum broadcast interval = <interval_max> * 0.625ms, ranging from 20ms to 10.24s. ● <channel_map> Broadcast channel mapping 0x01: Broadcast Channel 37 0x02: Broadcast Channel 38

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	0x04: Broadcast Channel 39 0x07: All broadcast channels enabled
Feature Description	

1.9.17 AT+BLEGAP=get_adv_param retrieves broadcast parameters

Instruction format	AT+BLEGAP=get_adv_param
Execution Example	● Send: AT+BLEGAP=get_adv_param ● answer: +BLEGAP:get_adv_param,<type>,<own_addr_type>,<filter_policy>,<peer_addr_type>,<peer_addr>,<interval_min>,<interval_max>,<channel_map> OK
Parameter Description	● Refer to the parameters of the AT+BLEGAP=adv,<op> instruction.
Feature Description	

1.9.18 AT+BLEGAP=adv_data Sets broadcast data before starting the broadcast.

Instruction format	AT+BLEGAP=adv_data[,<data>]
Execution Example	● Send: AT+BLEGAP=adv_data,02010508095265616c74656b ● answer: ● OK
Parameter Description	● <data> Broadcast data ● If <data> is not provided, the default broadcast data will be used.
Feature Description	● Broadcast Data Description 02: Length of the logo section 01: Type is a marker 05: LE restricts discoverable modes and BR/EDR are not supported. 08: Length of the local name portion 09: The type is a fully local name 5265616c74656b: The ASCII value of the string "Realtek", which is the local name in the broadcast data.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.9.19 AT+BLEGAP=scan Starts or stops scanning

Instruction format	AT+BLEGAP=scan,<op>
Execution Example	● Send: AT+BLEGAP=scan,0 ● answer: OK
Parameter Description	● <op> Scan Options 1: Start scanning 0: Stop scanning ● <addr> Broadcast address ● <event_type> Event type 0: Connectable and scannable non-directional broadcast (ADV_IND) 1: Connectable Directed Broadcast (ADV_DIRECT_IND) 2: Scannable non-directional broadcast (ADV_SCAN_IND) 3: Unconnectable non-directional broadcast (ADV_NONCONN_IND) 4: Scan Response (SCAN_RSP) ● <rssi> RSSI [-127, 20], where 127 indicates that RSSI is unavailable. ● <adv_report_len> Broadcast data length
Feature Description	● The parameters are the filtering conditions.

1.9.20 AT+BLEGAP=scan_param Sets scan parameters before scanning begins.

Instruction format	AT+BLEGAP=scan_param[,<type>,<own_addr_type>,<filter_policy>,<duplicate_opt>[,<interval>,<window>]]
Execution Example	● Send: AT+BLEGAP=scan_param,1,0,0,0,0x4,0x4 ● answer: OK
Parameter Description	● <type> Scan type 0: Passive scanning 1: Active scanning ● <own_addr_type> Local address type 0: Public Address 1: Random address 2: Publicly Identified Address Type 3: Random Identity Address Type ● <filter_policy> Scan filtering strategy 0: Accept all broadcast packets, except for targeted broadcast packets not directed to this

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	device (default). 1: Only accept broadcast packets from broadcaster addresses in the whitelist. Targeted broadcast packets not directed to this device will be ignored. 2: Accept all broadcast packets, except for targeted broadcast packets whose initiator's identity address is not specific to this device. 3: Accept all broadcast packets, except for targeted broadcast packets whose broadcaster's address is not in the whitelist and whose initiator's address is not specific to this device. ● <duplicate_opt> Duplicate scanning filtering option 0: Disable duplicate filtering 1: Enable duplicate filtering 2: Enable duplicate filtering, reset every scan cycle. ● <interval> Scan interval [0x4, 0x4000] Scan interval = <interval> * 0.625ms, ranging from 2.5ms to 10.24s. ● <window> Scan window [0x4, 0x4000] Scan window = <window> * 0.625ms, ranging from 2.5ms to 10.24s
Feature Description	● The scan interval cannot be less than the scan window.

Version 1.9.21 `AT+BLEGAP=get_scan_param` retrieves scan parameters.

Instruction format	AT+BLEGAP=get_scan_param
Execution Example	● Send: AT+BLEGAP=get_scan_param ● answer: +BLEGAP:get_scan_param,<scan_type>,<local_addr_type>,<filter_policy>,<duplicate_opt>,<interval>,<window> OK
Parameter Description	● Refer to the AT+BLEGAP=scan_param command
Feature Description	

1.9.22 AT+BLEGAP=scan_rsp sets the scan response data before scanning begins.

Instruction format	AT+BLEGAP=scan_rsp[,<data>]
Execution	● Send: AT+BLEGAP=scan_rsp,03194000

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Example	● answer: OK
Parameter Description	● <data> Scan response data If <data> is not provided, the default scan response data will be used.
Feature Description	● Scan response data description 03: Length of the exterior portion 19: Broadcast data type appearance 4000: General-purpose mobile phone (0x0040)

Version 1.9.23 AT+BLEGAP=conn creates a connection.

Instruction format	AT+BLEGAP=conn,<peer_addr_type>,<peer_addr>[,<own_addr_type>,<filter_policy>[,<scan_interval>,<scan_window>,<conn_interval_min>,<conn_interval_max>,<conn_latency>,<supv_timeout>]]
Execution Example	● Send: AT+BLEGAP=conn,0,701d080bdc77 ● answer: +BLEGAP:conn,0,17,70:1D:08:0B:DC:77(public) OK
Parameter Description	● <peer_addr_type> Peer address type 0: Public Address 1: Random address 2: Publicly Identifiable Address Type 3: Random Identity Address Type ● <own_addr_type> Local address type 0: Public Address 1: Random address 2: Publicly Identified Address Type 3: Random Identity Address Type ● <filter_policy> Connection filtering policy 0: Do not use whitelist filtering 1: Use whitelist filtering ● <scan_interval> scan interval [0x4, 0x4000], scan interval = <scan_interval> * 0.625ms, ranging from 2.5ms to 10.24s. ● <scan_window> Scan window [0x4, 0x4000], scan window = <scan_window> * 0.625ms, range 2.5ms to 10.24s ● <conn_interval_min> Minimum connection interval [0x6, 0x0C80], Minimum connection interval = <conn_interval_min> * 1.25ms, ranging from 7.5ms to 4s. ● <conn_interval_max> Maximum connection interval [0x6, 0x0C80], Maximum connection interval = <conn_interval_max> * 1.25ms, ranging from 7.5ms to 4s. ● <conn_latency> Slave connection latency (number of connection events) [0, 0x01F3]

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● <supv_timeout> Supervision timeout for the LE link [0xa, 0x0C80], Supervision timeout = <supv_timeout> * 10ms, ranging from 100ms to 32s. ● <conn_handle> connection handle ● <peer_device_addr> The address string of the peer device (e.g., 00:11:22:33:44:55) ● <peer_device_addr_type> Peer device address type string public: public address random: random address public-id: Public identity address type random-id: Random identity address type
Feature Description	● The scan interval cannot be less than the scan window. ● Message Feedback If the BLE connection is successfully established, the peer device will display the following message: [$]+BLEGAP:conn,<conn_handle>,<peer_device_addr>

1.9.24 AT+BLEGAP=disconn Close connection

Instruction format	AT+BLEGAP=disconn,<conn_handle>
Execution Example	● Send: AT+BLEGAP=disconn,16 ● answer: OK
Parameter Description	● <conn_handle> connection handle ● <peer_addr> The address string of the peer device (e.g., 00:11:22:33:44:55) ● <peer_addr_type> Peer device address type string public: public address random: random address public-id: Public identity address type random-id: Random identity address type
Feature Description	● Feedback message If the BLE connection is lost, the peer device will display the following message: [$]+BLEGAP:disconn,<conn_handle>,<peer_addr>(<peer_addr_type>) If the local device does not execute this command and the local device's BLE connection is lost, the local device will display the following message: [$]+BLEGAP:disconn,<conn_handle>,<peer_addr>(<peer_addr_type>)

Version 1.9.25 uses AT+BLEGAP=conn_info to retrieve connection information.

Instruction format	AT+BLEGAP=conn_info,<conn_handle>
Execution	● Send: AT+BLEGAP=conn_info,16

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Example	● answer: +BLEGAP:conn_info, 16 ,<role>,<remote_addr>(<remote_addr_type>),<interval>,<latency>,<supv_timeout>,<tx_phy>,<rx_phy> OK
Parameter Description	● <conn_handle> connection handle ● <role> Master or slave device ● <remote_addr> Remote device address ● <interval> Connection interval ● <latency> Slave connection latency (number of connection events) ● <supv_timeout> Supervision timeout for the LE link ● <tx_phy> Send physical layer 1:1M 2: 2M 3: Encoding ● <rx_phy> Receive physical layer 1:1M 2: 2M 3: Encoding
Feature Description	

Version 1.9.26 `AT+BLEGAP=conn_update` updates connection parameters after the connection is established.

Instruction format	AT+BLEGAP=conn_update,<conn_handle>,<conn_interval_min>,<conn_interval_max>,<conn_latency>,<supv_timeout>
Execution Example	● Send: AT+BLEGAP=conn_update,16,0x6,0x6,0,0xa ● answer: OK
Parameter Description	<conn_handle> connection handle <conn_interval_min> Minimum connection interval [0x6, 0x0C80] Minimum connection interval = <conn_interval_min> * 1.25ms, ranging from 7.5ms to 4s. <conn_interval_max> Maximum connection interval [0x6, 0x0C80] Maximum connection interval = <conn_interval_max> * 1.25ms, ranging from 7.5ms to 4s. <conn_latency> Slave connection latency (number of connection events) [0, 0x01F3] <supv_timeout> Supervision timeout for the LE link [0xa, 0x0C80] Supervision timeout = <supv_timeout> * 10ms, ranging from 100ms to 32s.
Feature	

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Description	

Version 1.9.27 `AT+BLEGAP=wl_add` adds the address to the whitelist.

Instruction format	AT+BLEGAP=wl_add,<addr_type>,<addr>
Execution Example	● Send: AT+BLEGAP=wl_add,0,112233445566 ● answer: OK
Parameter Description	● <addr_type> Remote address type 0: Public Address 1: Random address 2: Publicly Identified Address 3: Random identity address ● <addr> The address to be added to the whitelist
Feature Description	

Version 1.9.28 `AT+BLEGAP=wl_remove` removes addresses from the whitelist.

Instruction format	AT+BLEGAP=wl_remove,<addr_type>,<addr>
Execution Example	● Send: AT+BLEGAP=wl_remove,0,112233445566 ● answer: OK
Parameter Description	● <addr_type> Remote address type 0: Public Address 1: Random address 2: Publicly Identified Address 3: Random identity address ● <addr> The address to be removed from the whitelist
Feature Description	

Version 1.9.29 AT+BLEGAP=wl_clear clears the blank list.

Instruction format	AT+BLEGAP=wl_clear

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Execution Example	● Send: AT+BLEGAP=wl_clear ● answer: OK
Feature Description	

Version 1.9.30 AT+BLEGAP=sec_param configures encryption information.

Instruction format	AT+BLEGAP=sec_param[,<io_cap>,<oob_data_flag>,<bond>,<mitm>,<sec_pair>,<use_fixed_key>,<fixed_key>,<sec_pair_only>,<auto_sec_req>]
Execution Example	● Send: AT+BLEGAP=sec_param,4 ● answer: OK
Parameter Description	● <io_cap> Input/output capabilities 0: Display only, no keyboard or /no key. 1: There is a display and a yes/no button. 2: Only keyboard, no display 3: No input/output capability 4: It has a keyboard and display. ● <oob_data_flag> OOB data flag Default: all zeros ● <bond> binding flag ● <mitm> MITM protection flag ● <sec_pair> Secure pairing flag ● <use_fixed_key> Use a fixed key when pairing 0: Disabled (default) 1: Enable ● <fixed_key> A fixed key protected by MITM. [0, 999999] The default value is 0. ● <sec_pair_only> Secure pairing only flag ● `<auto_sec_req>` automatically sends an SMP security request when connecting. 0: Disabled (default) 1: Enable ● <sign_key_flag> Signature key flag 0: Disabled (default) 1: Enable
Feature Description	

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Version 1.9.31 `AT+BLEGAP=get_sec_param` retrieves the encryption parameters.

Instruction format	AT+BLEGAP=get_sec_param
Execution Example	● Send: AT+BLEGAP=sec_param,4 ● answer: +BLEGAP:get_sec_param,<io_cap>,<oob_data_flag>,<bond>,<mitm>,<sec_pair>,<use_fixed_key>,<fixed_key> OK
Parameter Description	● Refer to the AT+BLEGAP=sec_param command parameters
Feature Description	

1.9.32 AT+BLEGAP=sec Initiate the encryption process

Instruction format	AT+BLEGAP=sec,[conn_handle]
Execution Example	● Send: AT+BLEGAP=sec,16 ● answer: OK
Parameter Description	● <conn_handle> connection handle ● <sec_result> 0: Success 1: Failure ● <passkey> The entered key
Feature Description	● Feedback message If BLE pairing is successful, both the local device and the peer device will display the following prompts: [$]+BLEGAP:authcmpl,<conn_hdl>,<sec_result> If the peer device requires a working pair confirmation during pairing, the peer device will display the following prompt: [$]+BLEGAP:pair_cfm,<conn_handle> If the device requires an authentication key during the pairing process, the device will display the following prompt: [$]+BLEGAP:passkey_input,<conn_handle> If the device needs to display a key during the pairing process, the device will display the following prompt: [$]+BLEGAP:passkey_display,<conn_handle>,<passkey> If the device requires an OOB key during the pairing process, the device will display the

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	following prompt: [$]+BLEGAP:oobkey_input,<conn_handle>

1.9.33 AT+BLEGAP=pair_cfm confirms pair-only instruction.

Instruction format	AT+BLEGAP=pair_cfm,<conn_handle>,<op>
Execution Example	● Send: AT+BLEGAP=pair_cfm,16,1 ● answer: OK
Parameter Description	● <conn_handle> connection handle ● <op> Confirm pairing instructions 1: Accept 2: Refusal
Feature Description	

1.9.34 AT+BLEGAP=auth_key Enter the authentication key

Instruction format	AT+BLEGAP=auth_key,<conn_handle>,<passkey>
Execution Example	● Send: AT+BLEGAP=auth_key,16,1234 ● answer: OK
Parameter Description	● <conn_handle> connection handle ● <passkey> Authentication key [0, 999999]
Feature Description	

1.9.35 AT+BLEGAP=auth_oob Input OOB data

Instruction format	AT+BLEGAP=auth_oob,<conn_handle>,<tk>
Execution Example	● Send: AT+BLEGAP=auth_oob,16,0x12121212121212121212121212121212 ● answer: OK
Parameter Description	● <conn_handle> connection handle ● <tk> OOB Data
Feature	

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Description	

Version 1.9.36 uses AT+BLEGAP=bond_info to retrieve binding information.

Instruction format	AT+BLEGAP=bond_info
Execution Example	● Send: AT+BLEGAP=bond_info ● answer: +BLEGAP:bond_info,<idx_1>,<remote_addr_1>,<identity_addr_1>,<idx_2>,,<remote_addr_2>,<identity_addr_2>,… OK
Parameter Description	● <idx> Index number of the binding information ● <remote_addr> Remote device address ● <identity_addr> Remote device identity address
Feature Description	

1.9.37 AT+BLEGAP=auth_keycfm confirms the authentication key.

Instruction format	AT+BLEGAP=auth_keycfm,<conn_handle>,<op>
Execution Example	● Send: AT+BLEGAP=auth_keycfm,16,1 ● answer: OK
Parameter Description	● <conn_handle> connection handle ● <op> Confirm authentication key 1: Accept 2: Refusal
Feature Description	

1.9.38 AT+BLEGAP=bond_del Deletes binding information

Instruction format	AT+BLEGAP=bond_del,<addr_type>,<addr>
Execution Example	● Send: AT+BLEGAP=bond_del,0,112233445566 ● answer: OK

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Parameter Description	● <addr_type> Address type 0: Public Address 1: Random address 2: Publicly Identified Address 3: Random identity address ● <addr> The address to be deleted
Feature Description	

Version 1.9.39 AT+BLEGAP=bond_clear clears all bond information.

Instruction format	AT+BLEGAP=bond_clear
Execution Example	● Send: AT+BLEGAP=bond_del,0,112233445566 ● answer: OK
Feature Description	

1.9.40 AT+BLEGAP=disc discovers services and features

Instruction format	AT+BLEGATTC=disc,<conn_handle>,<discover_type>[,<start_handle>,<end_handle>,<uuid16/uuid128>]
Execution Example	● Send: AT+BLEGATTC=disc,17,3,0x1,0xFFFF ● answer: +BLEGATTC:disc,3,17,0x0002,0x20,0x0003,2a05 +BLEGATTC:disc,3,17,0x0006,0x02,0x0007,2a00 +BLEGATTC:disc,3,17,0x0008,0x02,0x0009,2a01 +BLEGATTC:disc,3,17,0x000a,0x02,0x000b,2a04 +BLEGATTC:disc,3,17,0x000d,0x02,0x000e,d001 +BLEGATTC:disc,3,17,0x000f,0x0c,0x0010,d002 +BLEGATTC:disc,3,17,0x0011,0x10,0x0012,d003 +BLEGATTC:disc,3,17,0x0014,0x20,0x0015,d004 OK
Parameter Description	<conn_handle> connection handle <discover_type> Discovery type 0: Discover all main services 1: Discover the main service based on the service UUID 2: Find the included services 3: Discover all features of the service 4: Discover features based on UUID

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	5: Discover all feature descriptors <start_handle> The start attribute handle for the discovery range. <end_handle> Discovers the end-of-range attribute handle. <uuid16/uuid128> UUID Target Discovery `<included_service_handle>` contains the handle of the service declaration. <characteristic_handle> Characteristic declaration handle <properties> Feature attributes <value_handle> Feature attribute handle <descriptor_handle> Feature descriptor handle
Feature Description	● Feedback message If <discover_type> is 0, the response is as follows: +BLEGATTC:disc,<discover_type>,<conn_handle>,<start_handle>,<end_handle>,<UUID> OK If <discover_type> is 1, the response is as follows: +BLEGATTC:disc,<discover_type>,<conn_handle>,<start_handle>,<end_handle> OK If <discover_type> is 2, the response is as follows: +BLEGATTC:disc,<discover_type>,<conn_handle>,<included_service_handle>,<start_handle>,<end_handle>,<UUID> OK If <discover_type> is 3, the response is as follows: +BLEGATTC:disc,<discover_type>,<conn_handle>,<characteristic_handle>,<properties>,<value_handle>,<UUID> OK If <discover_type> is 4, the response is as follows: +BLEGATTC:disc,<discover_type>,<conn_handle>,<characteristic_handle>,<properties>,<value_handle>,<UUID> OK If <discover_type> is 5, the response is as follows: +BLEGATTC:disc,<discover_type>,<conn_handle>,<descriptor_handle>,<UUID> OK

1.9.41 AT+BLEGATTC=read (Reads feature values)

Instruction format	AT+BLEGATTC=read,<conn_handle>,<type>[,…]
Execution Example	● Send: AT+BLEGATTC=read,17,0,0xc ● answer: +BLEGATTC:read,17,0,0x000e,10,01020304050607080 90a //The value read is a hexadecimal number OK [$]+BLEGATTS:read,12,23,2
Parameter	● <conn_handle> connection handle

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Description	● <type> Read type 0: Read feature values via handle, [...]=<handle> 1: Read by UUID feature, [...]=<start_handle>,<end_handle>,<uuid16/uuid128> 2: Read multiple feature values (not supported) 3: Read multiple variable feature values, [...]=<handle_count>,<handles_arr_1>[,...] ● <handle> property handle ● <length> is the length of the value read. ● <data> The value to read, excluding the hexadecimal number 0x. ● <service_app_id> Service Application ID ● <charac_index> Feature Index
Feature Description	● Feedback message If <type> is 0, the response is as follows: +BLEGATTC:read,<conn_handle>,<type>,<handle>,<length>,<data> OK If <type> is 1, the response is as follows: +BLEGATTC:read,<conn_handle>,<type>,<handle>,<length>,<data> OK If <type> is 3, the response is as follows: +BLEGATTC:read,<conn_handle>,<type>,<length>,<data> OK If the GATT client successfully reads the feature value, the GATT server will display the following message: [$]+BLEGATTS:read,<service_app_id>,<conn_handle>,<charac_index>

1.9.42 AT+BLEGATTC=write Write characteristic value

Instruction format	AT+BLEGATTC=write,<conn_handle>,<type>,<handle>,<length>,<data>
Execution Example	● Send: AT+BLEGATTC=write,17,1,0x10,1,01 ● answer: OK [$]+BLEGATTS:write,12,23,4,1,1,01
Parameter Description	● <conn_handle> connection handle ● <type> Write type 0: Write feature request 1: No response when writing feature value 2: No response and signed data written to the feature value ● <handle> Attribute handle of the peer device ● <length> The length of the data written ● The data written in `<data>` does not include hexadecimal numbers containing 0x, such as 0a0b. ● <service_app_id> Service Application ID

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● <len> data length ● <data> The data value to write, for example: 0a0b
Feature Description	● If the GATT client successfully writes the feature value, the GATT server will display the following message: [$]+BLEGATTS:write,<service_app_id>,<conn_handle>,<charac_index>,<type>,<length>,<data> ● You should use this command to enable/disable notifications/instructions. The format is as follows: AT+BLEGATTC=write,17,0,0x17,2,0100 17: Connection handle 0: Write request type, must be 0 0x17: CCCD handle 2: The length of the data to be written must be 2. 0100: Write data; 0100 enables notifications; 0200 enables indications; 0000 disables notifications/indications.

Version 1.9.43 AT+BLEGATTS=notify sends notifications from the GATT server.

Instruction format	AT+BLEGATTS=notify,<conn_handle>,<service_app_id>,<charac_index>,<length>,<data>
Execution Example	● Send: AT+BLEGATTS=notify,23,12,6,4,11223344 ● answer: OK [$]+BLEGATTC:notify,17,0x16,4,11223344
Parameter Description	● <conn_handle> connection handle ● <service_app_id> Service application ID, fixed value 12 ● <charac_index> Feature index, fixed value 6 ● <length> The length of the notification data ● <data> Notification data, a hexadecimal number excluding 0x, for example: 0a0b ● <value_handle> Feature attribute handle
Feature Description	● If the GATT server successfully sends the notification, the GATT client will display the following message: [$]+BLEGATTC:notify,<conn_handle>,<value_handle>,<length>,<data>

Version 1.9.44 AT+BLEGATTS=indicate sends an instruction from the GATT server.

Instruction format	AT+BLEGATTS=indicate,<conn_handle>,<service_app_id>,<charac_index>,<length>,<data>

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Execution Example	● Send: AT+BLEGATTS=indicate,23,12,9,4,44332211 ● answer: OK [$]+BLEGATTC:indicate,17,0x19,4,44332211
Parameter Description	● <conn_handle> connection handle ● <service_app_id> Service application ID, fixed value 12 ● <charac_index> Feature index, fixed value 9 ● <length> The length of the notification data ● <data> Notification data, a hexadecimal number excluding 0x, for example: 0a0b ● <value_handle> Feature attribute handle
Feature Description	● If the GATT server successfully sends the notification, the GATT client will display the following message: [$]+BLEGATTC:notify,<conn_handle>,<value_handle>,<length>,<data>

1.10 File System AT Command Set

- The AT+FS command performs corresponding processing in the AT directory of the file system.
- The AT+CERT command performs the corresponding processing in the CERT directory of the file system.
- The AT+FSRGN command configures the file system partition operated on by the above commands.

1.10.1 Performing file system operations with AT+FS

Instruction format	AT+FS=<operation> [,<filename>,<offset>,<length>]
Execution Example	● Send: AT+FS=4,"new.dat",0,100 ● answer: >>> Hello OK
Parameter Description	● <operation> File operation type 0: List the files in the directory 1: Delete files 2: Get file size 3: Reading files 4: Write to file ● <filename> filename string No more than 125 bytes in length ● <offset> The starting position for reading or writing.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	● <length> The length to read or write
Feature Description	● Error code 1: Formatting error 2: Operation failed ● When <operation> is set to write, it will enter pass-through mode.

1.10.2 AT+CERT query server-side or client-side certificate components

Instruction format	AT+CERT=<role>,<index>
Execution Example	● Send: AT+CERT=0,1 ● answer: <Certificate Name 1> <Certificate Content 1> <Certificate Name 2> <Certificate Content 2> <Certificate Name 3> <Certificate Content 3> OK
Parameter Description	● <role> Certificate Component Type 0: Client 1: Server side ● <index> Certificate Component Number The range is 1~10
Feature Description	● Error code 1: Parameter error

1.10.3 AT+FSRGN Configuration File System Command Operations on Partitions

Instruction format	AT+FSRGN=<vfs region>
Execution Example	● Send: AT+FSRGN=1 ● answer: OK
Parameter Description	● <vfs region> file system partition 1: Partition 1

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	2: Partition 2
Feature Description	● Error code 1: Parameter error

1.11 OTA AT Command Set

1.11.1 AT+OTA HTTP obtains OTA firmware via HTTP GET method

Instruction format	AT+OTAHTTP=<sysrst>,<host>,<path>[,<port>],<conn_type>[,<cert_index>]
Execution Example	● Send: AT+OTAHTTP=1,192.168.137.1,,8082,ota_all.bin ● answer: OK [$][OTA][HTTP]: UPDATE 10% [$][OTA][HTTP]: UPDATE 20% [$][OTA][HTTP]: UPDATE 30% [$][OTA][HTTP]: UPDATE 40% [$][OTA][HTTP]: UPDATE 50% [$][OTA][HTTP]: UPDATE 60% [$][OTA][HTTP]: UPDATE 70% [$][OTA][HTTP]: UPDATE 80% [$][OTA][HTTP]: UPDATE 90% [$][OTA][HTTP]: UPDATE 100% [$][OTA][HTTP]: UPDATE SUCCESS
Parameter Description	● Does `<sysrst>` automatically restart after the upgrade is complete? 0: Do not automatically restart after upgrade. 1: Automatic restart after upgrade complete ● <host> Target host, no http:// or https:// prefix is needed. Supports domain names or IPv4 addresses Maximum length: 128 characters ● <path> resource path Maximum length: 256 characters ● <port> HTTP server port Range: [0, 65535] (default value 0) 0: HTTP uses port 80, HTTPS uses port 443. ● <conn_type> HTTP connection type 1: HTTP 2: HTTPS (no certificate verification) 3: HTTPS (using a CA certificate to verify the server certificate) 4: HTTPS (provides the client's certificate to the server for verification)

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	5: HTTPS (mutual authentication, a combination of 3 and 4) ● <cert_index> Index of security certificate suites, see AT Security Certificates. [1-10], only required when <conn_type> is 3~5.
Feature Description	● Error code 1: Parameter error 2: Memory allocation error 3: OTA initialization error 4: OTA is underway 5: OTA task creation error 6: Certificate suite acquisition failed

1.11.2 AT+OTAUSER Transparent Transmission Mode for Sending OTA Upgrade Firmware for Upgrade

Instruction format	AT+OTAUSER=<sysrst>,<length>
Execution Example	● Send: AT+OTAUSER=0,1120704 ● answer: >>> <Send OTA firmware content> OK
Parameter Description	● Does `<sysrst>` automatically restart after the upgrade is complete? 0: Do not automatically restart after upgrade. 1: Automatic restart after upgrade complete ● <length> Total length of OTA firmware
Feature Description	● Error code 1: Parameter error 2: Memory allocation error 3: OTA initialization error 4: OTA is underway 5: Pass-through mode error 6: An error occurred during the upgrade process. ● To use in UART mode, please enable hardware flow control. ● When upgrading using UART, hardware flow control (RTS and CTS) is required. After sending the command AT+OTAUSER=0,1120704, first send the RTS signal. The module will then enter transparent transmission mode, at which point the firmware can be directly transmitted transparently. During the upgrade process, LOG_UART will output upgrade information in real time. ● Please refer to the Quick Start Guide for details.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1.12 EBYTE AT Command Set

1.12.1 AT+WEBCFG WiFi Configuration via Web

Instruction format	AT+WEBCFG=<en>[,<port>]
Execution Example	● Send: AT+WEBCFG =1 ● answer: please access in a browser: http://192.168.43.1:80 OK
Parameter Description	● <en> 0: Disable web-based network configuration 1: Enable web-based network configuration ● <port> Port number, default is 80
Feature Description	● After successful network configuration, you need to manually close the web-based network configuration to release resources. ● Disabling web-based network configuration will also disable Wi-Fi access point mode. ● Error code 1. ERROR: 17 AP mode not enabled 2. ERROR: 20 Web configuration failed to shut down 3. ERROR: 3 Incorrect number of parameters 4. ERROR: Port number 18 is incorrect 5. ERROR: 21 Web page callback registration failed. 6. ERROR: 19 Web network configuration failed to start. 7. ERROR: 2 Parameter values are incorrect ● Parameters are not saved when power is off

1.12.2 AT+RS485 Enable RS485 Compatibility Mode

Instruction format	AT+RS485=<en>[<port>,<pin>]
Execution Example	● Send: AT+RS485 = 1,B,19 ● answer: OK
Parameter Description	● Enable 0: Disable RS485 compatibility 1: Enable RS485 compatibility ● <port> RS485-EN pin port group A: Group A B: Group B, default value ● <pin> RS485-EN pin

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

	Select based on the pinout of the module; default: 19.
Feature Description	● No subsequent parameters are needed when en is 0. ● When en is 1, all parameters need to be filled in. ● Parameters are saved even after power failure ● Do not use pins that have been defined for other functions, otherwise unpredictable problems may occur. For example, AT communication serial ports tx, rx, cts, rts ● It is recommended to use the pins specified on the module; using them arbitrarily may cause module malfunctions.

1.12.3 AT+LPOWER Enable Low Power Mode

Instruction format	AT+LPOWER=<en>[,<mode>,<param>]
Execution Example	● Send: AT+LPOWER =1,0,0 ● answer: OK >>
Parameter Description	● Enable 0: Off 1: Enable ● <mode> Low power mode 0: Maintains UART baud rate during sleep, supports UART wake-up. 1: Wake up using GPIO while sleeping ● <param> Additional parameters 0: sleep pg (power off the SOC domain) 1: sleep cg (turns off the clock in the SOC domain while retaining its power supply) 2: Deep sleep (GPIO only, wake-up from low level)
Feature Description	● When en is 0, no further parameters are needed; when en is 1, all parameters are required. Parameters are not saved upon power failure. ● Deep sleep will begin after 5 seconds once enabled, and the system will reset upon waking. ● Serial port wake-up supports a maximum baud rate of 115200. Do not modify the baud rate beyond 115200 in low power mode. ● After waking up in CG or PG mode, the system will re-enter low-power mode after 3 seconds until low-power mode is turned off. ● Sleep feedback >>, Exit sleep feedback << ● Note: Please disconnect your WiFi connection before using this command, as WiFi's built-in low-power mechanism may conflict with this command.

1.12.4 AT+RESTORE restores factory settings

Instruction format	AT+RESTORE
Execution	● Send: AT+RESTORE

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Example	● answer: OK
Feature Description	● A reset will occur upon successful execution. ● Restore serial port baud rate, RS485, and WiFi connection information parameters.

1.12.5 AT+VERSION? Query version number

Instruction format	AT+VERSION
Execution Example	● Send: AT+VERSION ● answer: VERSION: V1.0 OK
Feature Description	

1.12.6 AT+BLERECONN disables power-on reconnection.

Instruction format	AT+BLERECONN=<en>
Execution Example	● Send: AT+BLERECONN=0 ● answer: OK
Parameter Description	● E n 0: Disable BLE reconnection
Feature Description	● Automatic reconnection is always enabled every time the AT+BLEGAP=conn command is used to connect to another Bluetooth device.

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Revision History

Version	Revision Date	Revision Notes	maintainer
1.0	2026-06-12	Initial version	ALL

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000