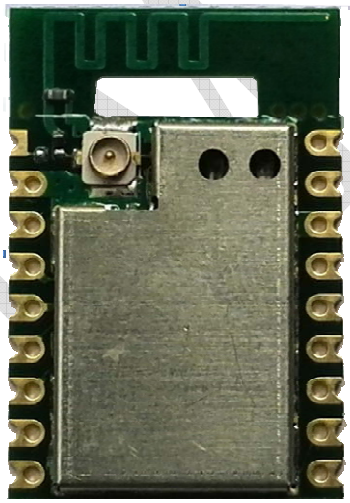
AirM2M_ESP8266_WiFi module user's manual V2.3



A6501

Directory

Table 1 Terms and abbreviations

Abbreviations	Descriptions
WiFi	Wireless Fidelity
UART	Universal Asynchronous Receiver & Transmitter
DTIM	Delivery Traffic Indication Message
SOC	System On a Chip
P2P	Point to Point
TCP	Transmission Control Protocol
IP	Internet Protocol
STBC	Space-Time Block Coding
MIMO	Multiple Input Multiple Output
MPDU	MAC Protocol Data Unit
MSDU	MAC Server Data Unit
IEEE	Institute Of Electrical And Electronics Engineers
bps	Bits Per Second
CCK	Corporate Control Key
DQPSK	Differential Quadrature Phase Shift Keying
DBPSK	Differential Binary Phase Shift Keying
QAM	Quadrature Amplitude Modulation
OFDM	Orthogonal Frequency Division Multiplexing
WPA	Wi-Fi Protected Access
WPS	Wi-Fi Protected Setup
TKIP	Temporal Key Integrity Protocol
WAPI	Wlan Authentication And Privacy Infrastructure
WEP	Wired Equivalent Privacy
CRC	Cyclic Redundancy Check

1.1. Product introduction

1.1.1. Summary

Up to now Shanghai AirM2M Communication Technology Co., Ltd has released 3 kinds of ESP8266 WiFi modules, including A6501、A6501S and A6502. There will be more kinds coming to meet different PCB footprint requirements of enterprise users and individual developers.

A650X series transparent transmission modules own competitive PCB footprint sizes in the industry and ultralow energy consumption technology. They are designed for mobile devices and internet applications, which can connect users' physical devices to WiFi wireless network. So communications happen on internet or local area network to make interconnection come true.

There are different kinds of PCB footprint sizes for A650X series transparent transmission modules. Some models of antennas can even support onboard PCB antennas, IPEX interfaces and stamp hole interfaces.

A650X series transparent transmission modules can be widely used on smart power grids, intelligent transportation, intelligent furniture, handheld devices, industrial control and so on.

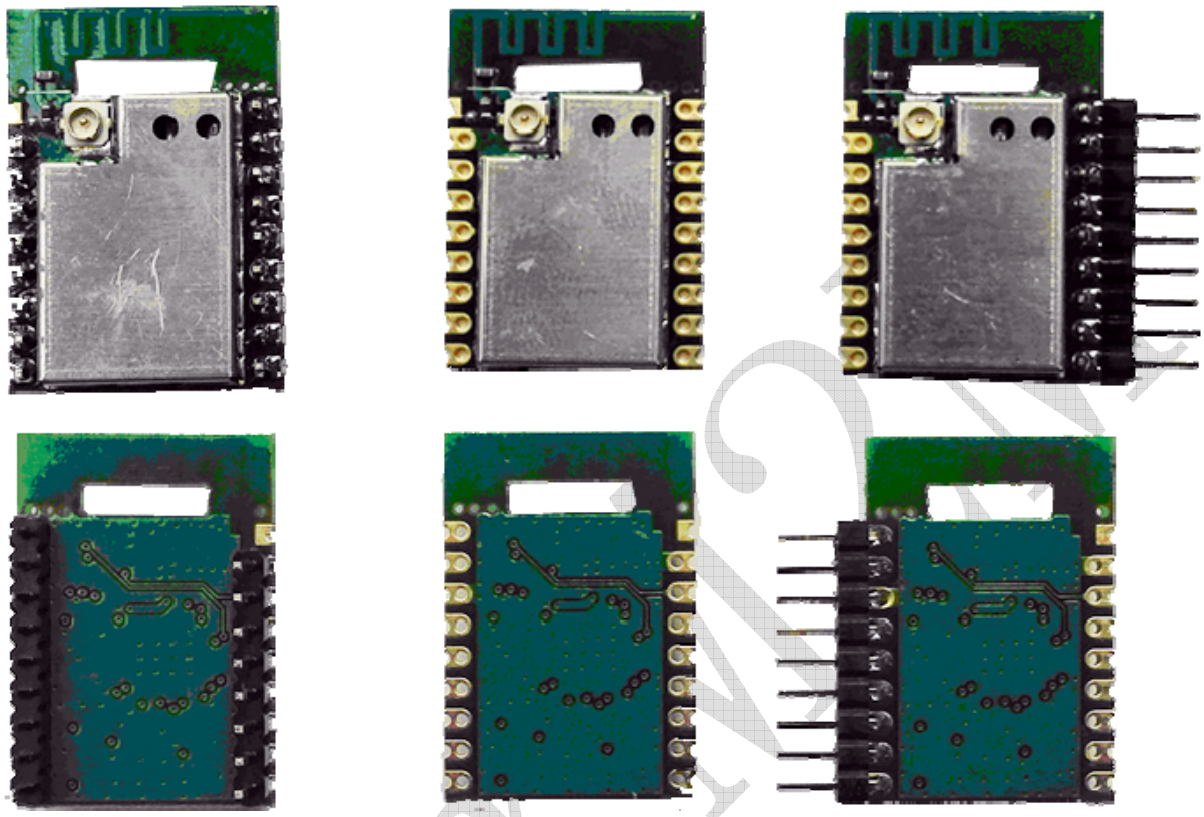
For more details please refer to http://www.luam2m.com/

Product features

- Support 802.11 b/g/n wireless standard
- Support STA/AP/STA+AP 3 working modes
- Built-in TCP/IP protocol stack, support multipath TCP Client connections
- Support abundant Socket AT commands.
- Built-in 32 bit MCU, can be used as the application processor
- Support UART/GPIO data communication interfaces
- Support PWM output
- Support I2C interfaces
- Support WiFi configuration parameters
- Support Smart Link intelligent networking function
- Support remote firmware upgrade （OTA）
- 3.3V single power supply
- Support multiple power consumption modes like Active、Modem sleep、Light sleep、Deep sleep
- Ultralow power consumption, suitable for battery-powered applications
- ESP8266 software owns industry leading stability

1.1.2 Module PCB footprint

AirM2M ESP8266 A650X series transparent transmission modules support multiple PCB footprints, which greatly enriches users' choice. They can be easily applied to a variety of networking hardware terminal situation.



A6501

Picture 1 A6501 PCB footprint modes A plan view and a bottom view

1.1.3 Module basic parameters

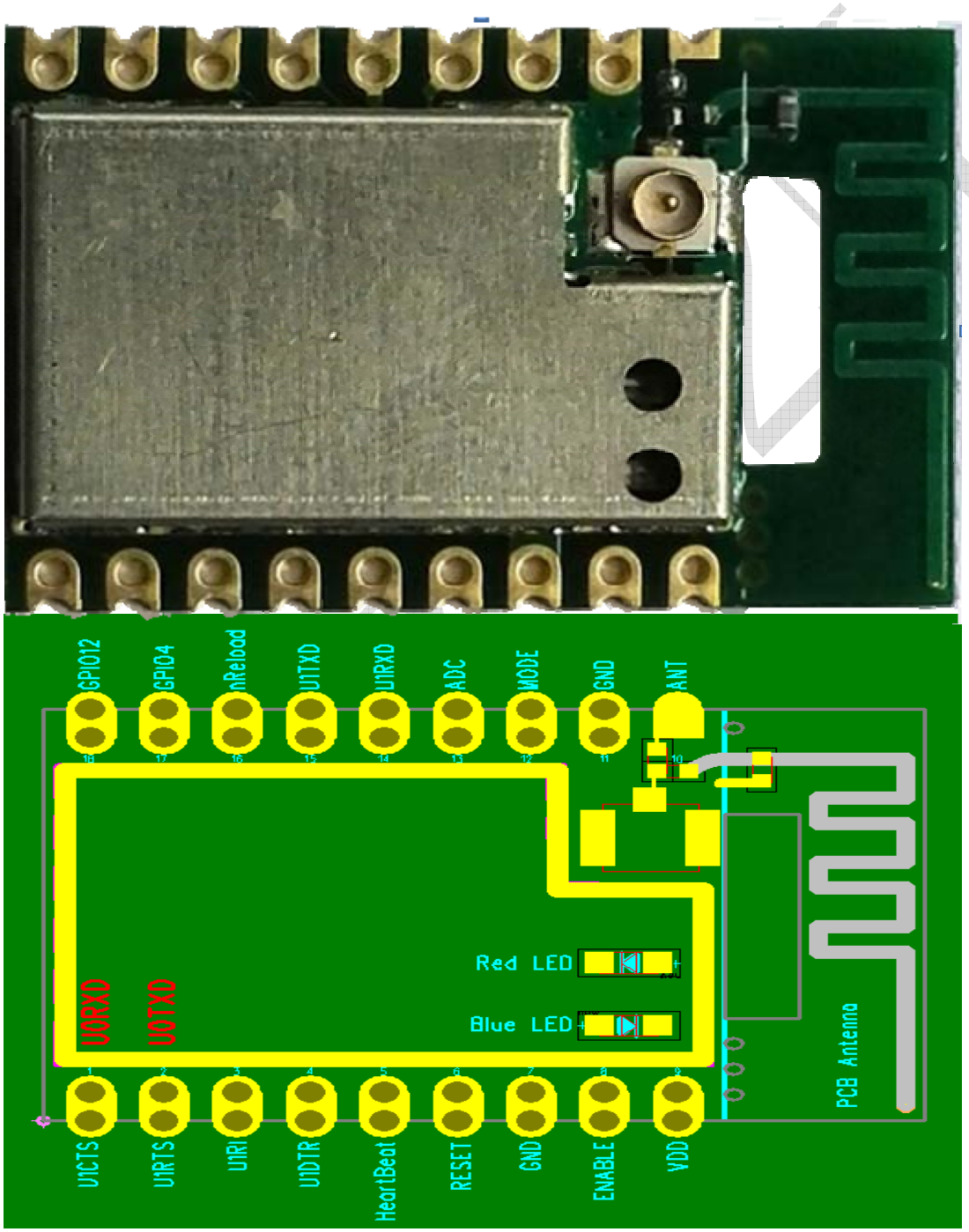
A650X	Mater chip	ESP8266
Wireless parameters	wireless standard	IEEE 802.11b/g/n
	frequency range	2.412GHz-2.462GHz
	transmitting power(PK)	802.11b: +16 +/-2dBm (@11Mbps)
		802.11g: +14 +/-2dBm (@54Mbps)
		802.11n: +13 +/-2dBm (@HT20, MCS7)
	receiving sensitivity	802.11b: -91 dBm (@11Mbps ,CCK)
		802.11g: -75dBm (@54Mbps, OFDM)
		802.11n: -71dBm (@HT20, MCS7)
Hardware parameters	hardware interface	UART，IIC，PWM，GPIO，ADC
	working voltage	3.0V--3.6V
	GPIO drive capability	Max：15ma
	working current	Active State, normal mode: average value：~70mA, peak value: 215mA； Modem Sleep, turnoff Modem，CPU work: average value： ~15mA； Light Sleep，turnoff Modem and CPU， wake up while break off：0.9mA； Deep Sleep，turnoff Modem and CPU， no wake up while break off：<15uA；
	operating temperature	-40℃~125℃
	storage environment	temperature：<40℃，relative humidity：<90%R.H.
	dimension	A6501： PCB onboard PCB antenna：16*24*3.3mm； built-out antenna：16*18.6*3.3mm；
		A6501S： PCB onboard PCB antenna：16*32*3.3mm； built-out antenna：16*24.6*3.3mm；
		A6502： PCB onboard PCB antenna：11.8*24*3mm；
Serial transmission	transmission rate	9600-460800bps
	TCP Client	4
Software parameters	wireless network types	STA/AP/STA+AP
	security mechanism	WEP/WPA-PSK/WPA2-PSK
	Encryption type	WEP64/WEP128/TKIP/AES
	firm upgrade	local serial, OTA remote upgrade
	networking protocol	IPv4, TCP/UDP/FTP/HTTP
	user configuration	AT+ instruction set Web page + PC configuration tool Android/iOS terminal Smart Link intelligent configuration APP

Table 2 Module technical specifications

1.2. Hardware introduction

A650X series modules using ESP8266 SOC processor with abundant hardware interface can support UART，IIC，PWM，GPIO，ADC, which can be widely used on different internet application situations.

1.2.1 A6501



Picture 2 A6501 pin configuration

A6501 Pin detailed descriptions:

PIN	Function	Description
1	U1CTS	1）UART_CTS， send request； 2）General Purpose Input/Output：GPIO3； 3）U U0TXD，ESP8266 using UART to receive message special for download；
2	U1RTS	1）UART_RTS， allow sending； 2）General Purpose Input/Output：GPIO1； 3) U0TXD，ESP8266 using UART to send message special for download； 4）Must remain vacant (internal pull up)when boot on, no pull down
3	U1RI	1）UART_RI， low level wake up opposite terminal MCU； 2）General Purpose Input/Output：GPIO5；
4	U1DTR	1）UART_DTR， low level wake up A650X； 2）General Purpose Input/Output：GPIO14；
5	HeartBeat	1）Deep Sleep mode： heart signal，need connect HeartBeat to RESET； 2）General Purpose Input/Output：GPIO16； 3）High level output is defaulted when boot up；
6	RESET	1）Non deepsleep mode: external reset signal, low level is effective 2）Deep sleep mode: RESET connect to Heartbeat
7	GND	GND
8	ENABLE	1)Module enable Pin, internal 10K ohm pull up, default power up for booting 2)Shutdown on low level, power consumption current is less than 5uA
9	VDD	3.3V，module power supply；
10	ANT	WiFi Antenna
11	GND	GND
12	MODE	1）Red LED: Indication light control signal in WiFI operating situation is defaulted. 2）General Purpose Input/Output：GPIO0； 3）mode selection： vancant：Flash Boot， normal operating mode； pull down：UART Download， serial download mode；
13	ADC	ADC， input range：0V-1V；
14	U1RXD	1）UART_RXD， receive signal； 2）General Purpose Input/Output：GPIO13；
15	U1TXD	1）UART_TXD， sending signal； 2)General Purpose Input/Output：GPIO15； 3)Module internal pull down at 10K ohm, no hardware pull up when boot up.
16	nReload	1）nReload, input interrupt， restore factory default settings: AP mode 2）General Purpose Input/Output：GPIO2； 3）Debug_TXD， used on output debugging log； 4）It must be high level when boot up, no hardware pull down
17	GPIO4	General Purpose Input/Output：GPIO4；
18	GPIO12	1)Blue LED：TCP/UDP indication light control signal is defaulted. 2)General Purpose Input/output：GPIO12(need software to setup in addition)

Table 1 A6501 Pin definition

Remarks

1)There are 3 serials of A6501. UART1（U1RXD、U1TXD、U1CTS、U1RTS、U1DTR、U1RI）are data serials, using on transparent transmission. UART0（U0RXD、U0TXD）are download serials, using on firmware updated. Debug_UART（Debug_TXD）are debugging serials, using on output debugging log.

UART0 and Debug_UART are marked by red and dark yellow respectively.

2) In order to avoid module cannot boot up or unstable phenomenon, close attention should be paid on request for initialization level on relative pins when boot up as most pins of A6501 are Multifunctional.

Pins that request initialization level when boot up already marked by blue color.

3) Signal“MODE”is used on module mode selection. Need connect “MODE”to“GND”when upgrading firmware.

4) Default functions are recommend to used on Pins of“MODE”、“nReload”、“GPIO12”, which are marked by purple.

5) Statements for“Red LED”和“Blue LED”：

Red LED：

TCP/UDP connect to indicator, blinks when connected, extinguishes when disconnect.

Blue LED：

（1）Boot constant is 2S,and then turn off；

（2）TCP/UDP data indicator blinks when transmitting, extinguishes when finish.

6) There are 3 GPIO pins of A6501which are completely unrestricted (no request on initialization level, no default function or default function can be modified). They are marked by green. First using these pins is recommended.

7) Difference between new version A6501_A11 and A6501_A10 are ：

（1）Module defaults power on for booting, pull down “ENABLE” when turn off.

（2）Pins location of GPIO12 and GPIO5（U0RI）are exchanged. GPIO5 is defined as U0RI signal.

（3）Optimized pin naming, e.g. changed “DEFAULT”to“nReload”

（4）Redefine UART0、UART1 and Debug_UART；

8)**Special reminding：**

Please remember that after software version **《SW_V0003_A6501_WEB》 of ESP8266** WiFi module of **Shanghai AirM2M,** firmware upgrading serial and data transparent transmission serial won’t use a same physical pin.

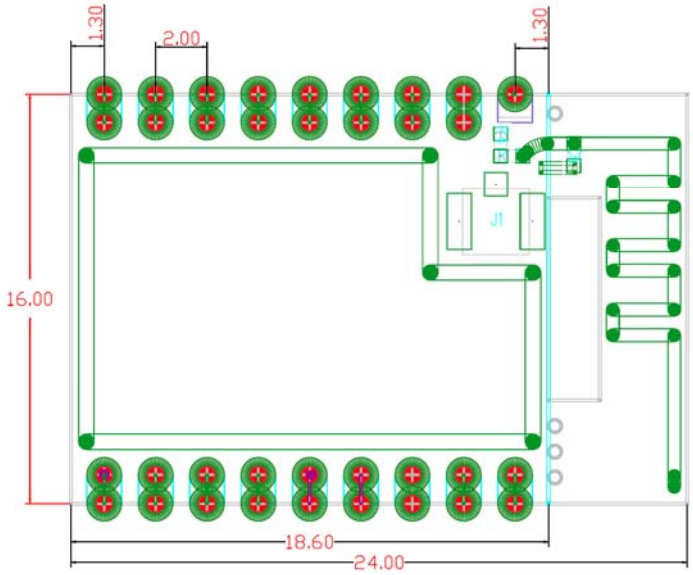
1.3. Power consumption

All the following power consumption data gained basing on 3.3V, 25° (temperature).
[1] All the tests are completed at the antenna interface.
[2] All emit data gained at continuous emission mode, basing on 90% of duty ratio.

Mode	Min	Normal	Max	unit
transmit 802.11b，CCK 1Mbps，Pout=+19.5dBm		215		mA
transmit 802.11b，CCK 11Mbps，Pout=+18.5dBm		197		mA
transmit 802.11g，OFDM54 Mbps，Pout=+16dBm		145		mA
transmit 802.11n，MCS7，Pout=+14dBm		135		mA
receive 802.11b，package length is 1024 bytes，-80dBm		60		mA
receive 802.11g，package length is 1024 bytes，-70dBm		60		mA
receive 802.11n，package length is 1024 bytes，-65dBm		62		mA
System standby mode		0.9		mA
Deep sleep		10		μA
Energy-saving mode DTIM1		1.2		mA
Energy-saving mode DTIM3		0.86		mA
Shutdown		0.5		μA

Table 5 Power consumption data

1.4. Dimension

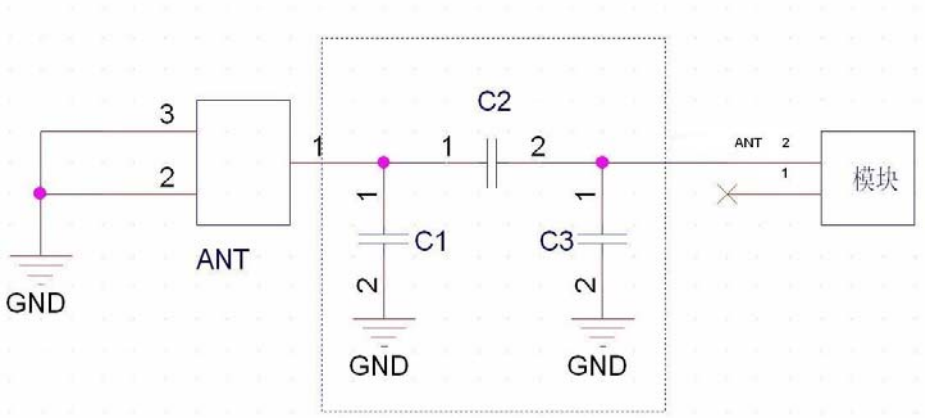


A6501

Picture 5 A6501 module Pin footprint

1.5. WiFi antenna

A650X series modules support 3 kinds of antenna interfaces. They are onboard PCB antenna, IPEX interface and stamp hole interface. Customers can use onboard PCB antenna and IPEX interface directly without adding any matching circuit. Customers can use A650X stamp hole antenna interface if they want to design antenna part on big board. In this case, matching circuit should be reserved on big board for this design. For example:



Picture 6 WiFi radio frequency reference circuit

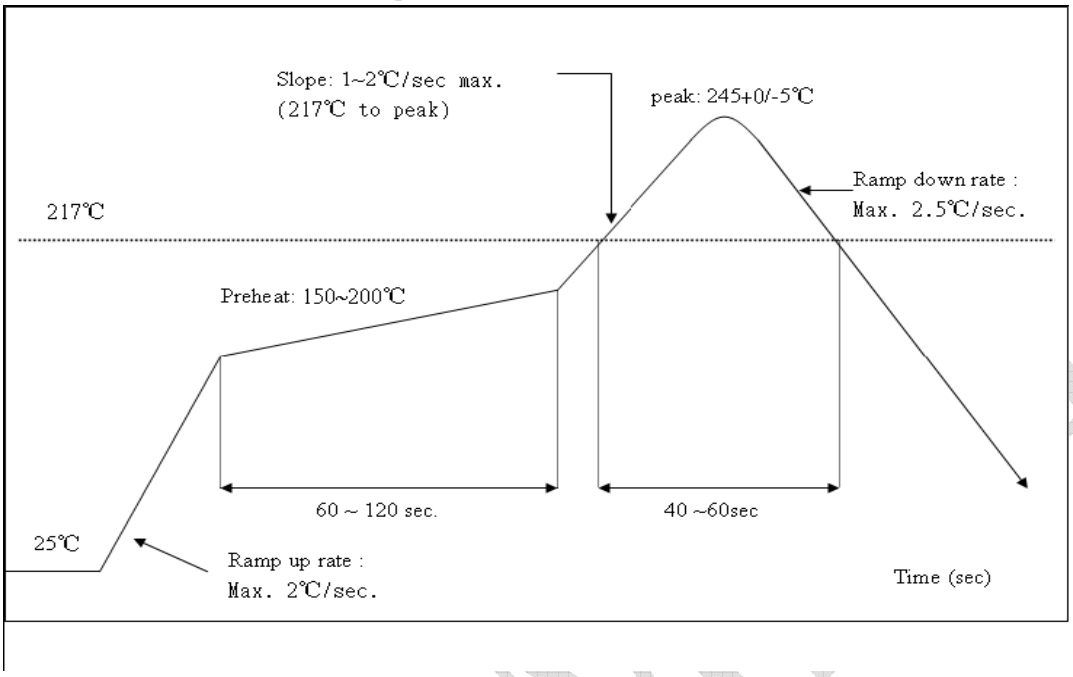
Note:

1、 Above dashed box parts need antenna matching. Should take actual electronic components' (who match with antenna) parameters as standard.

2、 Need 50 ohm resistance for RF wiring. 90 degree angle wiring is forbidden. No longer than 15mm.

1.6. Recommended furnace temperature curve

Refer to IPC/JEDEC standard；Peak Temperature : <250°C；Number of Times: ≤2 times；



Picture 7 Recommended back flow curve

2. Functional description

2.1. Main function

Main functions that A650X series modules can achieve include serial port transmission, PWM control and GPIO control.

Serial port transmission: data transmission, good reliability, stability, can reach 11Kbyte at serial rate 115200kbps

PWM control: light-control, three-colour LED control, motor speed control

GPIO control: control switch, relay and so on.

2.2. Operating mode

A650X modules support STA/AP/STA+AP 3 operating modes.

- STA mode: A6501 modules can connect to the internet by router. So cell phone or computer can remote control device by internet.
- AP mode: As hotspot, A6501 modules achieve that cell phone, computer can communicate with modules directly, which makes local area network wireless controlled come true.
- STA+AP mode: Coexist mode of these two modes. Can achieve seamless switching by internet control.

2.3. Application fields

- RS232 serial change to Wi-Fi
- industry transparent transmission
- Wi-Fi remote control
- toy field
- color LED control
- fire protection, security and intelligent integrated management
- Intelligent card terminal, wireless POS machine, Wi-Fi camera, handheld device.

2.4. AirM2M Cloud

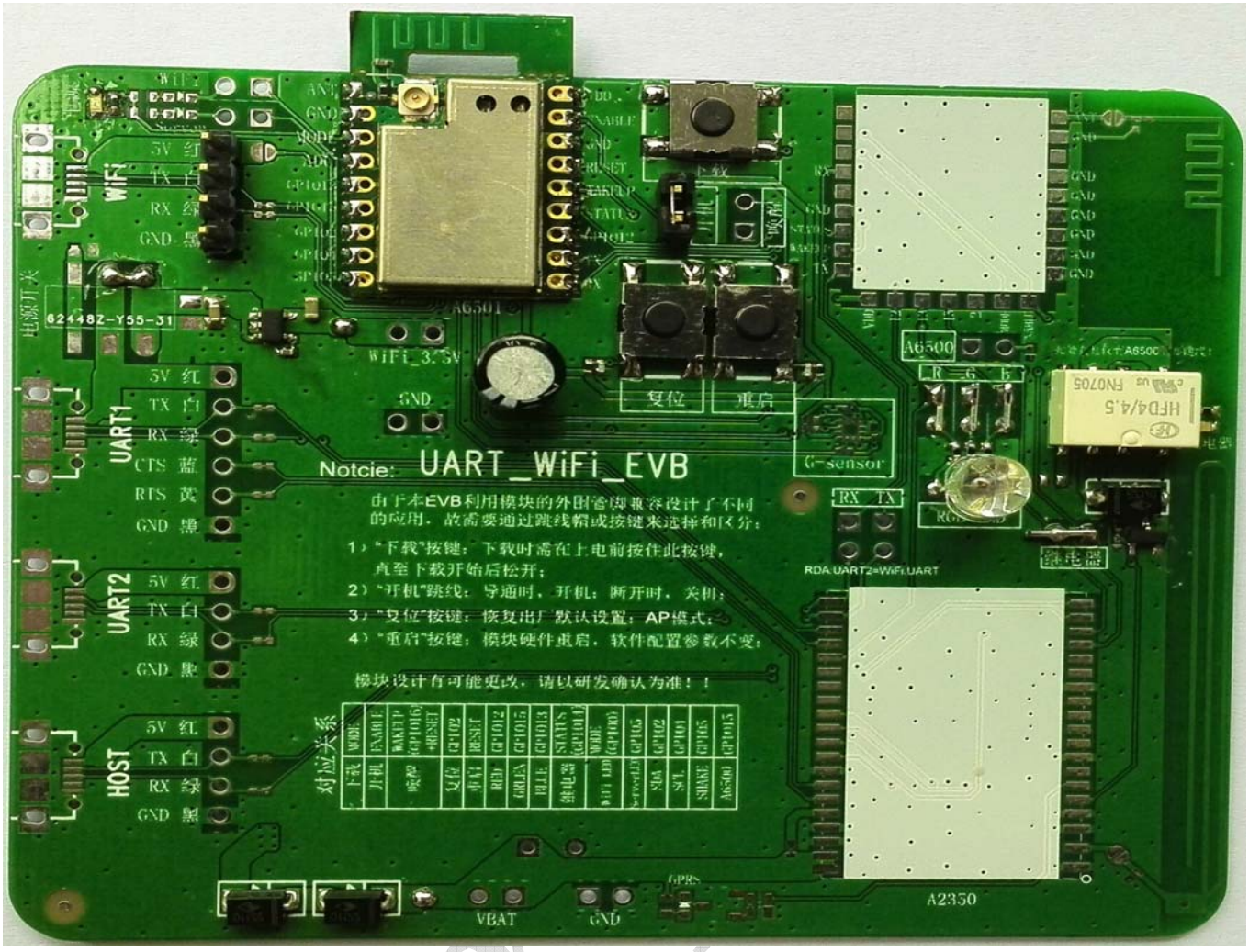
AirM2M Cloud is internet cloud platform service creative by Shanghai AirM2M（AirM2M）. Users can monitor and manage device on the platform to realize huge data management and analyze, which make device intelligent really come true.

AirM2M Cloud can accept customers' customized requests, including Web Page Configuration, Android/iOS platform and App.

3. EVB Introduction

AirM2M can offer special UART_WiFi_EVB developmwnt board for customers' to debug A650X. By this development board, traditional serial device or MCU device can easily connect to WiFi network to realize managing and controlling device by network.

This development board can offer hardware demonstration programs like UART serial port data transmission solution, RGB light-control and intelligent socket. At the same time, development board has reserved hardware circuit on which our GSM/GPRS standard module A2350 works as main control, MCU and A6501 serial port transmits data.



Picture 8 EVB front view

Notice：

Due to this EVB is compatible to different application by using module's periphery pins. So it need jumper cap or buttons for distinguishing.

1）Button "download": Press this button before power up for download until downloading begins.

2）Jumper wire "boot": when break over, turn on; when break, shutdown.

3）Button "reset": Restore factory default settings: AP mode.

4)"Button "reboot": module's hardware reboot, software configuration parameter won't change.

4. PC Configuration tool

4.1. Basic application methods

A650X support using PC tools: AirM2M_ESP8266_CONFIG_TOOL to configure relative parameters for easy using.

Here we take to set multilink transparent transmission mode(module working as STA+CLIENT) for example, introducing following configuration steps:

Step 1 A650X wireless module is at power-up operating mode, and at AP mode or AP+STATION (default is

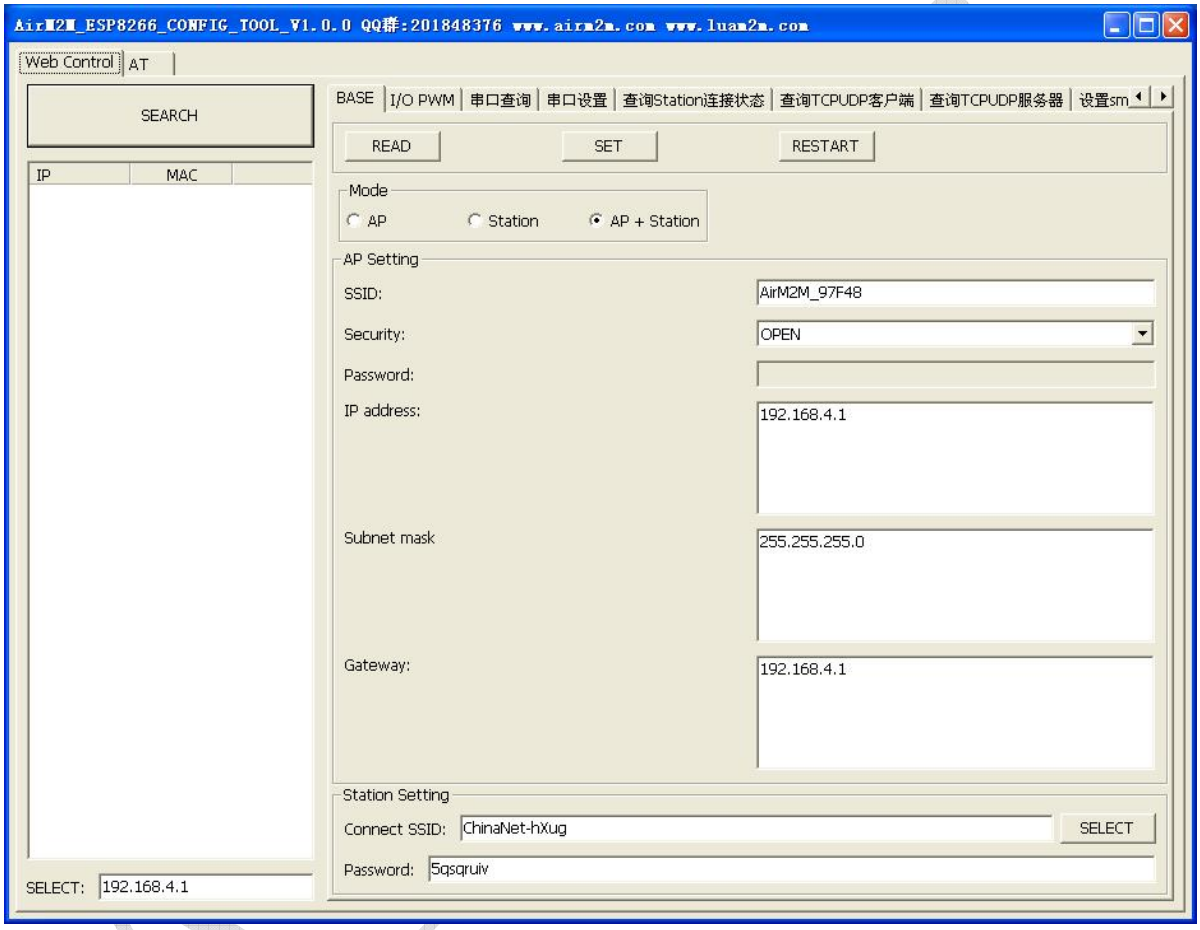
AP+STATION mode).

Step 2 Using computer's WIFI function, to search wireless module A650X (all SSID WiFi device of AirM2M are named as "AirM2M+MAC" e.g. AirM2M_97F502) and click connect.

Step 3 Open configuration tool AirM2M_WiFi_Tool. Please add trust if intercepted by security software.

Step 4 Turn on Web Control to Click "SEARCH", the searched module will be showed at the left-hand display column. If the module not searched, we can repeatedly click the button "SEARCH".(**Note: do not support SEARCH temperately)**

Step 5 Click "READ" in the "BASE" option card. Read basic data of AirM2M WiFi device. And to configure SSID and password for module connection in Station Setting, and then click "SET".



Step 6 Click "Setting TCPUDP client-side" option page to configure link. It supports 4 links most at the same time.

Every linking and configuring 4 parameters:

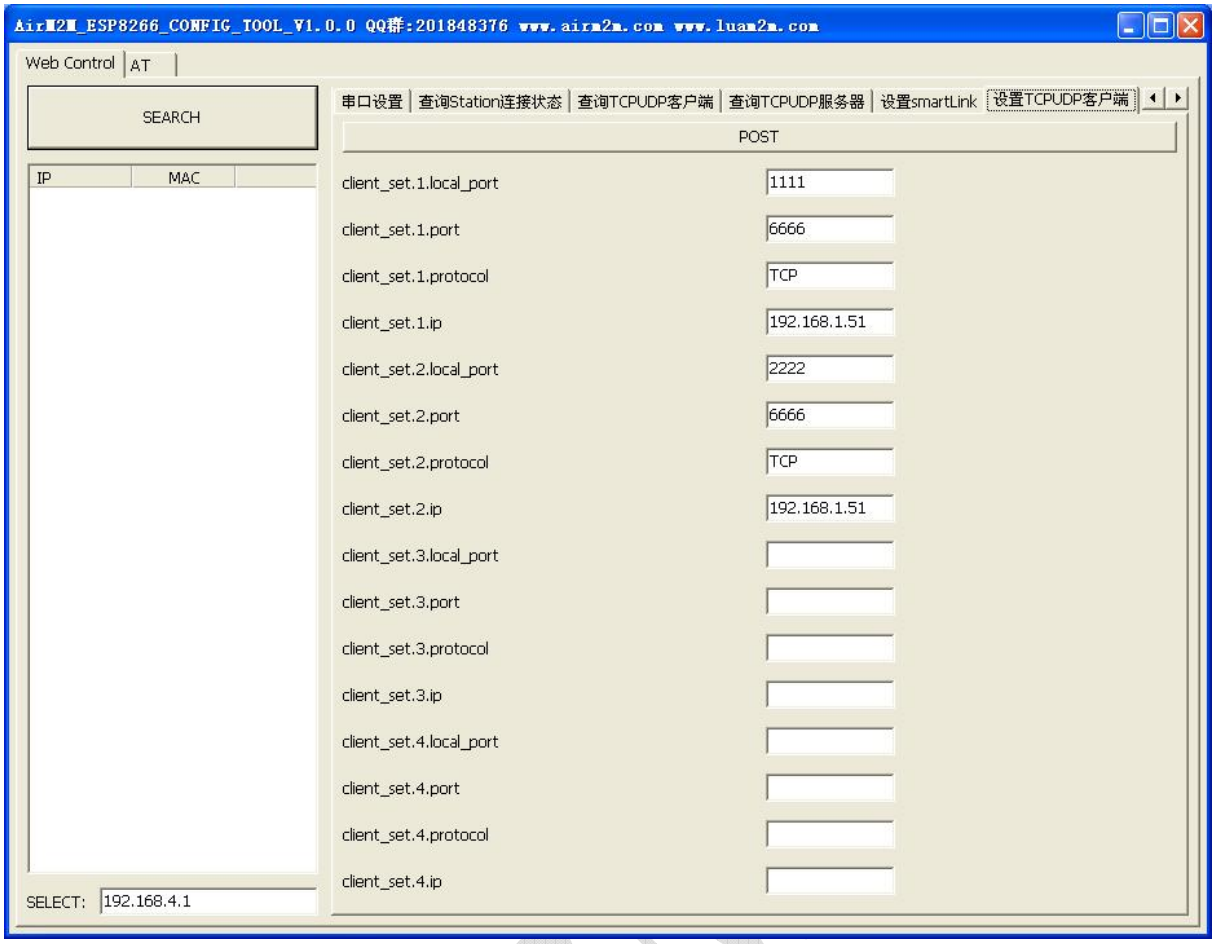
Local_port：local port

Port：server port

Protocol：TCP or UDP

Ip: server IP address

To configure links basing on need. Here take 2 for example and then click "POST"



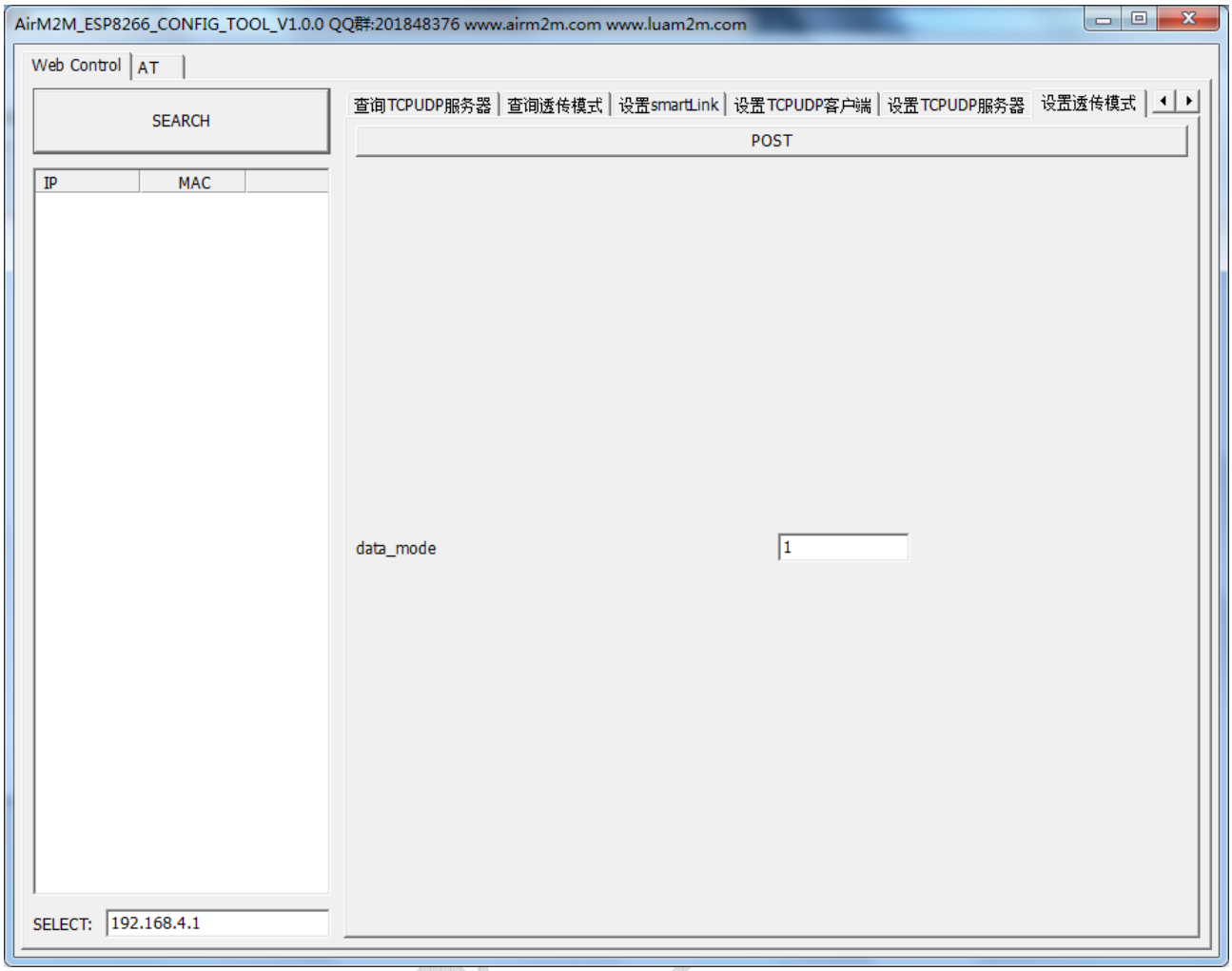
Note: If user want single link transparent transmission only, to configure one link at this step is enough.

Step 7 Click “set transparent transmission mode” option page to set transparent transmission mode.

Data_mode = 1 ， transparent transmission mode

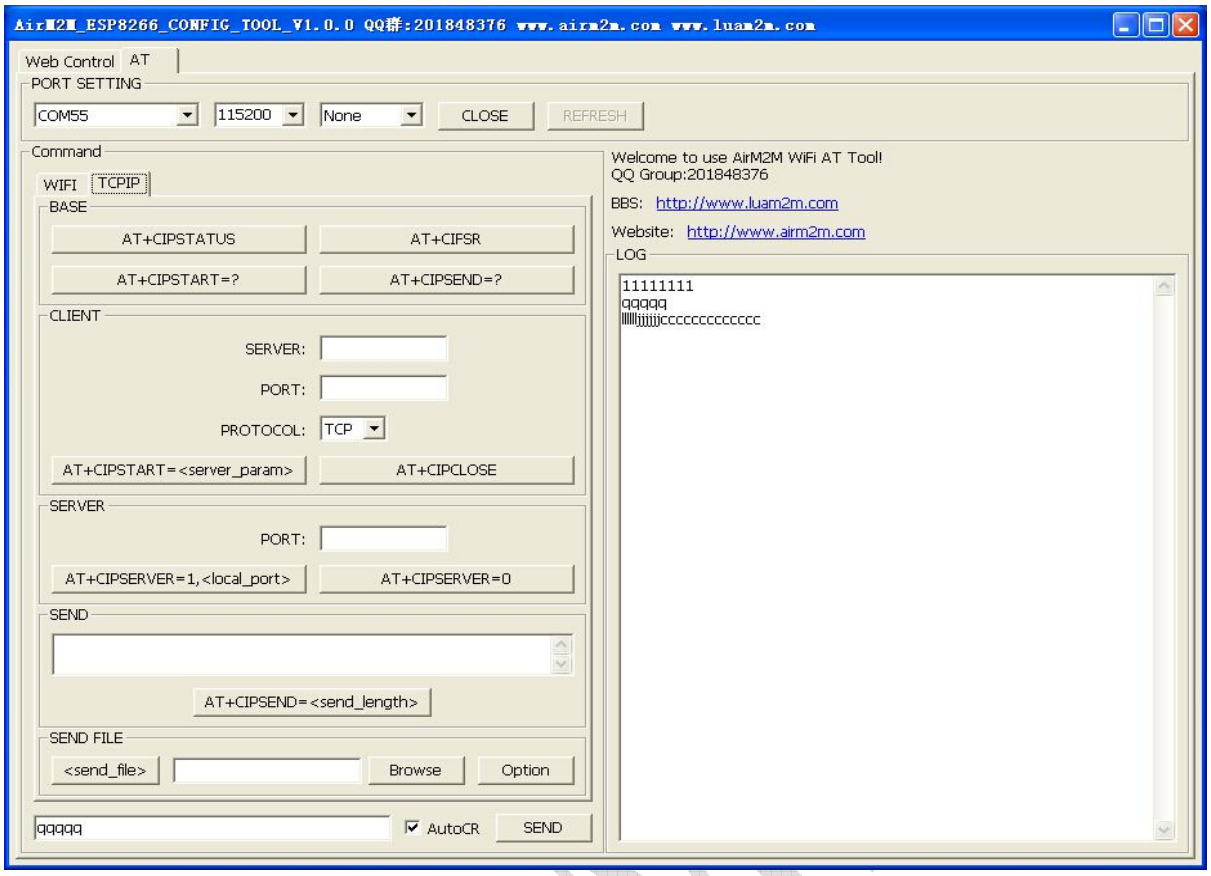
Data_mode = 0 ， non-transparent transmission mode，that is AT command mode

Set data_mode=1 over here，then POST



Step 8 Reboot module. After rebooting, module will turn into transparent transmission state. Look from the server side, there will be two links established

Step 9 Open AT option page, turn on interface, put in data in the input box at the left corner. Look from the server side, two links all received data “qqqqq”.



Step 10 Quit transparent transmission mode and go back to AT command mode.

Click “set transparent transmission mode” option page to set data_mode=0，POST, then reboot module will be OK.

Note:

There are 3 ways to quit under transparent transmission mode:

1) + + + quit transparent transmission, but it will go back to transparent transmission mode after rebooting.
2) + + +, then AT+CIPSCON=0, it will go back to transparent transmission mode after rebooting module.
3) At setting transparent transmission mode option page of AirM2M_ESP8266_CONFIG_TOOL ， WEB CONTROL->, set to 0 and then POST. Enter non-transparent transmission mode after rebooting module.

There are 2 ways to enter transparent transmission mode:

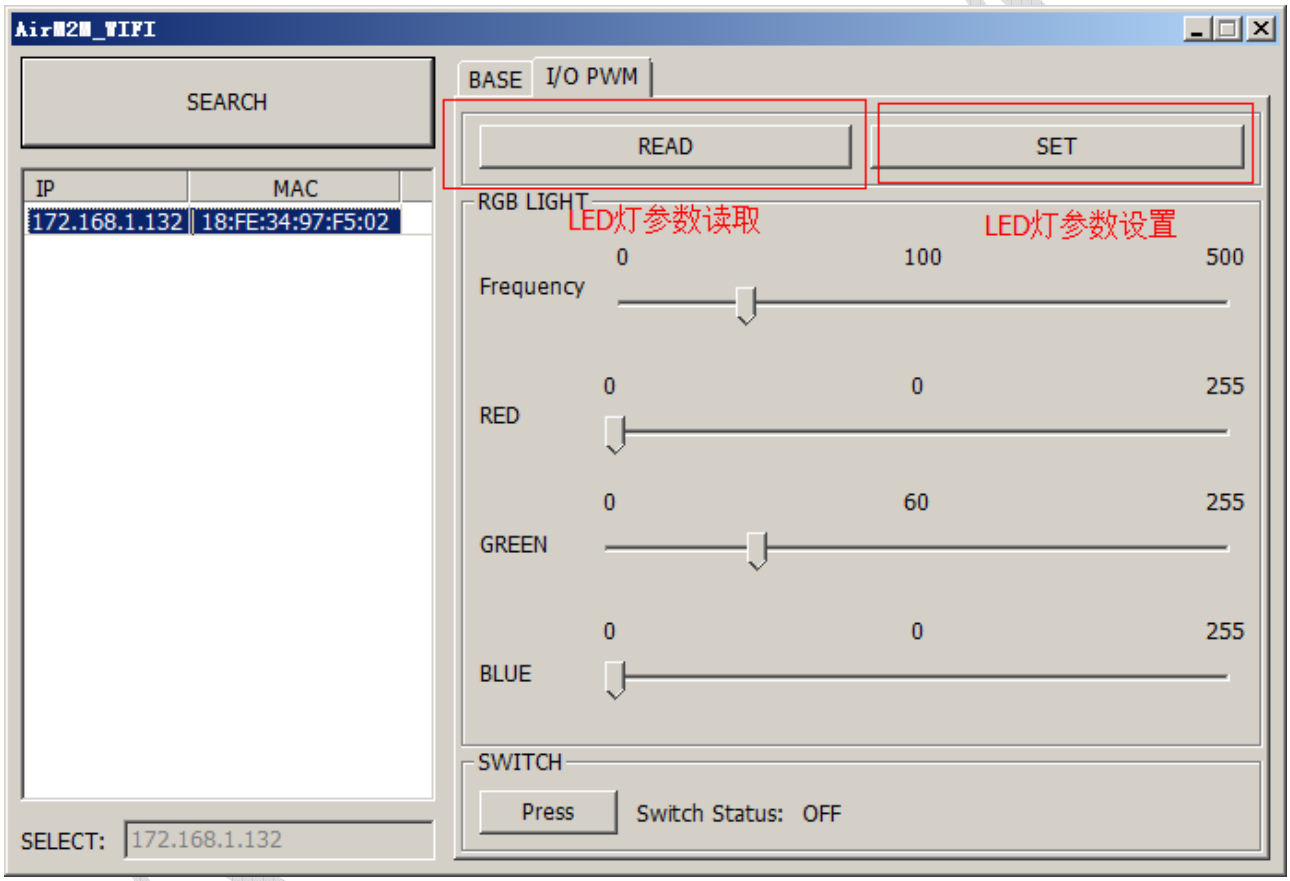
1) AT+CIPMODE=1，AT+CIPSERVER to configure server or AT+CIPSTAR to configure client side. AT+CIPSCON=1, reboot module and then enter transparent transmission, and automatically set up connection according to last saved data AT+CIPSCON=1.
2) At setting transparent transmission mode option page of AirM2M_ESP8266_CONFIG_TOOL ， WEB CONTROL->, set to 0 and then POST. At setting TCPUDP client side/server option page to configure CLIENT/SERVER, and then POST, reboot module to enter transparent transmission and automatically set up connection according to configuration.

4.2. IO/PWM function demonstration

Select “I/O PWM” option, access interface as picture 11 shows to control RBG tricolor light, and control relay switch.

Click “READ” to read current RGB LED, click “SET” to configure new parameters. Thereinto, “Frequency” is used to adjust PWM output frequency to control Brightness of RGB. Buttons of “RED”、“GREEN”、“BLUE” are used to adjust Color temperature value of RGB tricolor light. The color temperature value of red-green-blue light is the same as color palette of Windows’ drawing.

Click “PRESS” button on relay controller to budge relay’s on and off.



Note：

1）RGB LED and relay switch function can match our company’s development board UART_WiFi_EVB for demonstration.

2）Standard AT software version does not support this function.

5. AT Instructions Introduction

5.1. Basic AT instruction

5.1.1 Reboot module：AT+RST

Grammatical rules：

command type	grammar	backward and instruction
execute command	AT+RST	OK

5.2. WiFi function AT instruction

5.2.1 Select WiFi application mode：AT+CWMODE

Grammatical rules：

command type	grammar	backward and instruction
Set command	AT+CWMODE = <mode>	OK
		(AT+RST)Command will be in valid after reboot
check command	AT+CWMODE?	+CWMODE:<mode> OK
		What mode is currently in?
test command	AT+CWMODE = ?	+CWMODE:(<mode> value list) OK
		What mode it support currently?

Parameter definition：

parameter definition	definition	value	value instruction
<mode>	WiFi application mode	1	Station mode
		2	AP mode
		3	AP+Station mode

5.2.2 List current available access points: AT+CWLAP

Grammatical rules：

command type	grammar	backward and instruction
execute command	AT+CWLAP	+CWLAP: <ecn>,<ssid>,<rssi>[,<mode>] OK
		this command will go back to AP list

Parameter definition：

parameter	definition	value	value instruction
<ecn>	encryption way	0	OPEN
		1	WEP
		2	WPA_PSK
		3	WPA2_PSK
		4	WPA_WPA2_PSK
<ssid>	access point name		character string parameter
<rssi>	signal strength		
<mode>	connect mode	0	manual connect
		1	automatic connection

5.2.3 Add access point:AT+CWJAP

Grammatical rules：

command type	Grammar	Backward and instruction
Set command	AT+CWJAP=<ssid>,<pwd>	OK or ERROR
		Successfully join AP, go back to OK. If not, go back to ERROR
check command	AT+CWJAP?	+CWJAP:<ssid> OK
		go back to current choice AP

Parameter definition：

parameter	definition	value	value instruction
<ssid>	access point name		character string
<pwd>	password		character string, the longest is 64 byte, ASCII coding

5.2.4 Quit access point: AT+CWQAP

Grammatical rules：

command type	grammar	backward and instruction
execute command	AT+CWQAP	OK
		means exit this AP successfully
test command	AT+CWQAP=?	OK
		check whether this command support

5.2.5 Set parameter in AP mode: AT+CWSAP

Grammatical rules：

command type	grammar	backward and instruction
set command	AT+CWSAP=<ssid>,<pwd>,<chl>, <ecn>	OK
		successfully set parameters
check command	AT+CWSAP?	OK
		check current AP parameter

parameter definition：

parameter	definition	value	value instruction
<ecn>	encryption way	0	OPEN
		1	WEP
		2	WPA_PSK
		3	WPA2_PSK
		4	WPA_WPA2_PSK
<ssid>	access point name		character string parameter
<pwd>	password		character string, the longest is 64 byte, ASCII coding
<chl>	channel number		

5.2.6 Inquire MAC address: AT+AMMAC

Grammatical rules：

command type	grammar	backward and instruction
execute command	AT+AMMAC	If CWMODE set as 1（STA mode）： STATION_MAC: xx:xx:xx:xx:xx:xx OK

		If CWMODE set as 2（AP mode）： AP_MAC: xx:xx:xx:xx:xx:xx OK
		If CWMODE set as 3（STA+AP mode）： STATION_MAC: xx:xx:xx:xx:xx:xx AP_MAC: xx:xx:xx:xx:xx:xx OK
		Successfully set up parameters

Example：

command（→）/backward（←）	example	explanation and instruction
→	AT+CWMODE?	Inquire WIFI level working mode
←	+CWMODE::3 OK	3：AP+STA mode
→	AT+AMMAC	Inquire AC address
←	STATION_MAC: 18:fe:34:9e:ab:e4 AP_MAC: 1a:fe:34:9e:ab:e4 OK	
→	AT+CWMODE=2	Set working mode to 2，AP mode
←	OK	
→	AT+AMMAC	Inquire MAC adress
←	AP_MAC: 1a:fe:34:9e:ab:e4 OK	

5.2.7 Start SMART LINK: AT+AMSL

Grammatical rules：

Command type	grammar	Backward and instruction
Set up command	AT+AMSL=1	Go back： +AMSL:<ssid>,<password> OK
		instruction：successfully configured SMART LINK

parameter definition：

parameter	definition	value	Value instruction
<ssid>	AP hotspot name		
<password>	AP hotspot password		

example：

command（→）/backward（←）	example	Explanation and instruction
		● Turn on cell phone's WIFI function and link to router that module can search (name it A for easy understanding). ● Turn on SMARTLINK Android App that released by our company. Input name and password of router that we want module link to. And click below SMARTLINK button. Keep running state until configuration finish. Note: A and B can be a same one router. That is to say cell phone can directly link to router B that we want module link to. And configure router (that module links to) as B in APP.
→	AT+CWMODE?	Inquire WIFI level working mode
←	+CWMODE:1 OK	Mode 1：STA mode Mode 3：STA+AP mode Mode 1 and 3，both can use SMART LINK But mode 2(pure AP mode)cannot configure SMARTLINK.
→	AT+AMSL=1	Start SMART LINK configuration.
←	+AMSL: yh_taih,qin02666 OK	It means configuration is successful if backward of "wait for a moment" occurs. Ssid = yh_taih Password = qin02666
→	AT+CWJAP?	Inquire which hotspot module joins up currently.
←	+CWJAP:"yh_taih" OK	means successfully joined up yh_taih
		Click STOP，turn off APP

5.3. TCPIP AT command

5.3.1 set up TCP/UDP connection ： AT+CIPSTART

Grammatical rules：

command type	grammar	backward and instruction
set command	when single way connection (+CIPMUX=0)： AT+CIPSTART=<type>,<addr>,<port>,[local_port] When multichannel connection: AT+CIPSTART=<id>,<type>,<addr>,<port>,[local_port] And local port is optional parameters	if format is right, go back： OK otherwise go back： +CME ERROR: invalid input value successfully connect, go back： CONNECT OK (CPIMUX=0) <id>, CONNECT OK (CIPMUX=1) If connection already exist, go back： ALREADY CONNECT connection fail, go back： CONNECT FAIL (CIPMUX=0) <id>, CONNECT FAIL (CIPMUX=1)

parameter definition：

parameter	definition	value	value instruction
<id>	Link No.	0~4	means connected serial number. server / 0 connection can be connected by client or server, other ID can use on remote server connection only.
<type>	connection type	“TCP”/”UDP”	
<addr>	Remote server IP address		character string
<port>	remoter service port number		
[local_port]	local port		number, can select parameters

5.3.2 Get TCP/UDP connection mode ： AT+CIPSTATUS

Grammatical rules：

command type	grammar	backward and instruction
execute command	AT+CIPSTATUS	If it is single way connection (AT+CIPMUX=0), go back to OK STATE: <sl_state> C:<cid>, <TCP/UDP>, <IP address>, <port>, <client state> Note: If it is single way connection, it uses following way for connection:

		AT+CIPSTART=<type>,<addr>,<port>,[local_port] way, and one link can be set up only. Occupied <cid> = 0 If it is multiway connection (AT+CIPMUX=1), go back toOK STATE:<ml_state> S: <sid>,<port>,<server state> C:<cid>, <TCP/UDP>, <IP address>, <port>, <client state>
test command	AT+CIPSTATUS=?	backward： OK

parameter definition：

parameter	definition	value	value instruction
<sl_state>	single way connection state	IP INITIAL	initialization
		IP STATUS	obtain local IP status
		TCP CONNECTING/UDP CONNECTING	TCP connecting/UDP port registering
		CONNECT OK	successfully connect setup
		TCP CLOSING/UDP CLOSING	Shutting down TCP connection, and logging out UDP port.
<sid>	server ID	0~1	value 0 and 1
<server state>	server status	OPENING	turning on
		LISTENING	monitoring
		CLOSING	turning off
<cid>	customer's ID	0~3	value as 0,1,2,3
<IP address>	IP address	-	character string parameters(need quotation mark for character string)
<port>	server monitor port number	-	integer type
<client state>	customer's status	INITIAL	turn off state
		CONNECTING	connecting
		C O N N E C T E D	connected

example：

Command backward （←）	example	Explanation and instruction
→	AT+CWMODE?	Inquire WIFI level working mode
←	+CWMODE::3 OK	3：AP+STA mode
→	AT+CIPMUX?	
←	+CIPMUX:0 OK	Single way connection mode
→	AT+CIPSTATUS	
←	OK STATE:IP STATUS C:0,"","",,"INITIAL"	

	C:1,"","",,"INITIAL" C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	
→	AT+CIPSTART="TCP","192.168.1.51",6800	Set up one connection
←	OK CONNECT OK	
→	AT+CIPSTATUS	
←	OK STATE:CONNECT OK C:0,"TCP","192.168.1.51",6800,"CONNECTED" C:1,"","",,"INITIAL" C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	
→	AT+CIPSTART="TCP","192.168.1.51",7800	Want to set up one more connection
←	OK ALREADY CONNECT	Cannot set up.
→	AT+CIPSTATUS	inquire status
←	OK STATE:CONNECT OK C:0,"TCP","192.168.1.51",6800,"CONNECTED" C:1,"","",,"INITIAL" C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	Same as before
→	AT+CIPCLOSE	Turn off connection
←	CLOSE OK	
→	AT+CIPMUX=1	Must turn off all connection before configure multiway connection
←	OK	
→	AT+CIPSTART=0,"TCP","192.168.1.51",7800	
←	OK 0,CONNECT OK	
→	AT+CIPSTART=1,"TCP","192.168.1.51",6800	
←	OK 1,CONNECT OK	
→	AT+CIPSERVER=1,6000	
←	OK	
→	AT+CIPSTATUS	
←	OK STATE:IP STATUS S:0,6000,LISTENING C:0,"TCP","192.168.1.51",7800,"CONNECTED" C:1,"TCP","192.168.1.51",6800,"CONNECTED" C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	

5.3.3 Start multi-connection：AT+CIPMUX

Grammatical rules：

command type	grammar	backward and instruction
set command	AT+CIPMUX=<mode>	OK If it is at multiway connection mode, go back to +CME ERROR: The socket is already connected That is when there is already connection for module, <mode> value can not be modified. Only when all connection are turned off by using AT+CIPCLOSE, <mode> can be modified.
		Instruction: Start multi-connection successfully
check command	AT+CIPMUX?	+CIPMUX:<mode> OK
		Instruction: Check whether it is at multi-connection status or not.

parameter definition：

parameter	definition	value	value instruction
<mode>	whether to boot multi-connection mode	0	Configure to single way connection mode
		1	Configure to multi- way connection mode

5.3.4 sending data：AT+CIPSEND

Grammatical rues：

command type	Grammar	backward and instruction	
set command	When single way connect (+CIPMUX=0)： AT+CIPSEND=<length> When multi-way connect (+CIPMUX=1)： AT+CIPSEND=<id>,<length>	response	When module receives command it will line feed back to ”>” first, then start to receive serial port data. Data will be sent when it is full length. If connection is not setup or connection is interrupt, go back to ERROR; If successfully send data, go back to SEND OK. module executes wrongly, go back to ： parameter error： +CME ERROR: invalid input value no connection setup： +CME ERROR: The socket is not connected
		instruction	send assigned length data
test command	AT+CIPSEND=?	response	single way connection(AT+CIPMUX=0) back to ： +CIPSEND: <length>

			OK multi-way connection(AT+CIPMUX=1) back to ： +CIPSEND: <0-7>,<length> OK
execute command	AT+CIPSEND	instruction	AT+CIPMODE=1 and as customer-side mode, to access into transparent transmission mode(need support hardware flow control, otherwise data will be lost when there is mass data.) When module receives command it will line feed back to ”>” first, then start to send data which received by serial port.

Parameter definition：

parameter	definition	value	value instruction
<length>	data length		unit: byte
<id>	Link No.	0~3	connect serial number

5.3.5 Turn off TCP/UDP connection：AT+CIPCLOSE

Grammatical rules：

command type	grammar	backward and instruction
set command	Single way connection AT+CIPCLOSE=<id>	go back： CLOSE OK
	Multi-way connection AT+CIPCLOSE=<n>[,<id>]	go back： <n>,CLOSE OK
execute command	AT+CIPCLOSE	If shut down successfully, go back： CLOSE OK If shut down fail, go back： ERROR
test command	AT+CIPCLOSE=?	go back： OK
Items need attention	● Executing command is effective to single way connection, it will go back to ERROR when multi-way connection. ● When executing command AT+CIPCLOSE, connection will be shut down only when at TCP/UDP CONNECTING or CONNECT OK status, otherwise it will take shut down for fail and go back to ERROR. ● Status after shutting down is IP CLOSE when at single way connection mode.	

parameter definition：

parameter	definition	value	value instruction
<id>	shut down mode	0	slow shutdown(default value)
		1	quit shutdown
<n>	Link No.	0~3	integer type, means connected serial number

5.3.6 Obtain local IP address：AT+CIFSR

Grammatical rules：

command type	grammar	Response and instruction	
Execute command	AT+CIFSR	response	+ CIFSR:<IP address> OK or ERROR
test command	AT+CIFSR=?	response	OK

parameter definition：

parameter	definition	value	value instruction
<IP address>	Local current IP address(station)		

5.3.7 Configure as server：AT+CIPSERVER

Grammatical rules：

command type	grammar	Response and instruction	
		instruction	Need execute AT+CIPMUX=1 first, turn on mux mode. To monitor 2 ports at most, including TCP and UDP.
Execute command	AT+CIPSERVER=<mode>[,<port>]	response	After start server, it will automatically set up monitor by server. OK If there is client connect to the server of this module then it will go back to ： <linkid>,CONNECT OK If number 0 connection is occupied then it goes back to +CME ERROR: no change Note: Need reboot if use AT+CIPSERVER=0 to shut down server.

parameter definition：

parameter	definition	value	value instruction
<mode>	whether turn on server mode	0	turn off server mode
		1	turn on server mode
<port>	port number		default value is 333

5.3.8 Module receives data from server：+IPD and+RECEIVE

The command URC（Unsolicited Result Code） is sent by module to serial port. When module receives network data it sends data which uses +IPD and RECEIVE as header to serial port.

Grammatical rules：

command type	grammar	Response and instruction
When non-transparent transmission	when single way connect (+CIPMUX=0) +IPD,<len>:<data>	When single way, received data will take +IPD as beginning.
	when multi-way connect (+CIPMUX=1) +RECEIVE,<id>,<len>: <data>	<data> is received data, there is one \r\n between it and +RECEIVE, <n>,<length>: Note: Colon is English colon.
When transparent transmission	<data>	There is no data head, data come up directly when it is transparent transmission. **Note:** **Transparent transmission ways for multiway connection are: Every single data sent to module serial port will be automatically sent to all connections under CONNECTED state by module. Users can guarantee data's pertinence by application layer's protocol.**

Parameter definition：

parameter	definition	value	value instruction
<id>	Link No.	0~3	connected serial number
<len>	<data> length		unit: byte Note: The length is <data>'s length, not including data head.
<data>	burst		

Example of data receiving(take data receiving "TEST123" for example：

	Non-transparent transmission	Transparent transmission
single way link	+IPD,7:TEST123	TEST123
multilink	+RECEIVE,1,7: TEST123	TEST123(at every CONNECTED connection）

5.3.9 Select TCPIP application mode ： AT+CIPMODE

Grammatical rules：

command type	grammar	backward
set command	AT+CIPMODE=<mode>	OK

check command	AT+CIPMODE?	+CIPMODE: <mode> OK
test command	AT+CIPMODE=?	+CIPMODE: (0-NORMAL MODE,1-TRANSPARENT MODE) OK

parameter definition：

parameter	definition	value	value instruction
<mode>	TCPIP application mode	0	Non-transparent transmission mode, default mode.
		1	transparent transmission mode

5.3.10 Save transparent transmission configuration：AT+CIPSCON

Grammatical rules：

command type	grammar	backward
set command	AT+CIPSCON=<action>	OK
check command	AT+CIPSCON	C:<id>,<protocol>,<ip>,<remotePort>,<localPort> S:<id>,<localPort>,<timeout> DATA_MODE:<mode> OK
test command	AT+CIPSCON=?	+CIPSCON:(0,1) OK

parameter definition：

parameter	definition	value	value instruction
<action>	Save/delete transparent transmission parameter	1	Save transparent transmission parameter Module enter transparent transmission after booting. Set up connection according to data of C and monitor port according to data of S.
		0	delete transparent transmission parameter. Data of S and C is deleted. Module quits transparent transmission mode at the same time
<id>	connection marking		
<protocol>	value instruction	TCP	
		UDP	
<ip>	Ip address		x.x.x.x
<remotePort>	opposite port		means server port
<localPort>	local port		To C, that is module's local CLIENTport To S，that is module's loca SERVER port
<timeout>	Server link to time-out time	0~65535	Please refer to AT+CIPSTO. Default value is 180
<mode>	Transparent transmission mode	0	Common AT mode
		1	Transparent transmission mode

example：

Command （→） /backward	example	Explanation and instruction

（←）		
→	AT+CWMODE?	
←	+CWMODE:1 OK	
→	AT+CIPMODE=1	
←	OK	
→	AT+CIPMUX=1	
←	OK	
→	AT+CWJAP="ChinaNet-hXug","5qsqruiv"	
←	OK	
→	AT+CIPSTART=0,"TCP","192.168.1.51",6800	
←	OK 0,CONNECT OK	
→	AT+CIPSCON=1	
←	OK	
→	AT+CIPSCON	
←	C:0,"TCP","192.168.1.51",6800,1793 C:1,"","", , C:2,"","", , C:3,"","", , S:0, , S:1, , DATA_MODE:1 OK	
	AT+CIPSTATUS	
	OK STATE:IP STATUS C:0,"TCP","192.168.1.51",6800,"CONNECTED" C:1,"","",,"INITIAL" C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	
Then reboot module. Module will enter transparent transmission mode after reboot and automatically set up connection of 0,"TCP","192.168.1.51",6800,"CONNECTED"		

5.3.11 Time-out time set for server disconnected automatically：AT+CIPSTO

command type	grammar	backward and instruction
set command	AT+CIPSTO=<server timeout >	OK
check command	AT+CIPSTO?	+ CIPSTO:<server timeout> OK

parameter definition：

parameter	definition	value	value

<server timeout >	Use to set time-out time for server automatically disconnected.	0~65535	unit is 2s Using this command to set up time-out time, server will disconnect when times up. 0 is default value. Means initiatively disconnect. Default value is 180.

5.3.12 Set TE-TA Baud rate：AT+IPR

Grammatical rules：

command type	grammar	backward and instruction
set command	AT+IPR=<rate>	go back： OK
check command	AT+IPR?	go back： +IPR: <rate> OK
test command	AT+IPR=?	go back： +IPR: (<rate> value list) OK
Items need attention	When <rate> = 0, module self-adapt to baud rate. Two items need attention. 1. When main control serial port's baud rate is set as non-115200, it needs input at least two "AT" or "at" at the main control to boot the module. In this way module's baud rate will be trained the same as main control. Only when the main control 's serial port baud rate is the same as module's, serial port can normally input output AT command. There should be no more than 100ms between input "A" and "T". There is no limitation between two "AT". 2. When main control' s serial port baud rate is set as 115200, input command AT directly will be responded as it is at the self-adapted baud rate state.	

parameter definition：

parameter	definition	value	value instruction
< rate >	Baud rate unit: bps	0	self-adapt to baud rate
		2400	
		4800	
		9600	
		14400	
		19200	
		28800	
		38400	
		57600	
		115200	
		230400	
		460800	
		921600	

5.3.13 Select non-transparent transmission data sending mode：AT+CIPQSEND

Under non-transparent transmission data sending mode, there are two sending modes quick sending and slow sending also, using +CIPQSEND to set up.

Grammatical rules：

command type	grammar	Backward
set command	AT+CIPQSEND=<n>	OK
check command	AT+CIPQSEND?	+CIPQSEND: <n> OK
test command	AT+CIPQSEND=?	+CIPQSEND: (0,1) OK

parameter definition：

parameter	definition	value	value instruction
<n>	non-transparent transmission mode	0	Common mode, also called slow sending mode—when server receives TCP data, module goes back to SEND OK.
		1	Quick sending mode—when data sent to module, module goes back single way link：DATA ACCEP:<length> multilink：DATA ACCEPT:<id>,<length> but not SEND OK

5.3.14 Quit transparent transmission mode: + + +

There are 3 ways to quit under transparent transmission mode:

1) + + + quit transparent transmission, but it will go back to transparent transmission mode after rebooting.
2) + + +, then AT+CIPSCON=0, it will go back to non-transparent transmission mode after rebooting module.
3) At setting transparent transmission mode option page of AirM2M_ESP8266_CONFIG_TOOL ， WEB CONTROL->, set to 0 and then POST. Enter non-transparent transmission mode after rebooting module.

There are 2 ways to enter transparent transmission mode:

1) AT+CIPMODE=1，AT+CIPSERVER to configure server or AT+CIPSTAR to configure client side. AT+CIPSCON=1, reboot module and then enter transparent transmission, and automatically set up connection according to last saved data AT+CIPSCON=1.
2) At setting transparent transmission mode option page of AirM2M_ESP8266_CONFIG_TOOL ， WEB CONTROL->, set to 0 and then POST. At setting TCPUDP client side/server option page to configure CLIENT/SERVER, and then POST, reboot module to enter transparent transmission and automatically set up connection according to configuration.

Grammatical rules：

Command type	grammar	backward
execute command	+++	Attention：+++ no other character at front or back

5.3.15 Application example

Module can work as STATION on WIFI level, also can work as AP, and AP+ STATION. Module can work as SERVER on transmission level. While on application level, it can work as non-transparent data transmission and transparent data transmission also.

Examples：

command （→） / backward （←）	example	explanation and demonstration
1）Module works in STATION mode（mode=1）： The following words in light blue are the same common premise steps in this mode：		
→	AT+CWMODE?	Step1: check WIFI level working mode
←	+CWMODE::3 OK	Default mode is 3: AP+STA mode
→	AT+CWMODE=1	Step2: Set as Station mode
←	OK	
→	AT+RST	Step3: Need reboot after setting operation mode.
←	OK	**Note: Step2 and Step 3 can be ignored if after booting AT+CWMODE? is already Step 1.**
→	AT+CWLAP	Step 4: check current available access point
←	+CWLAP:(3,"yh_taihdhad",-45,"54:e6:fc:52:36:e6",9) +CWLAP:(4,"lemon",-86,"6c:e8:73:69:4b:76",10) +CWLAP:(4,"Xiaomi_yu",-85,"8c:be:be:2c:46:ea",11) OK	search result
→	AT+CWJAP="yh_taihdhad","UUU02339"	Step 5: add into one AP
←	OK	**Note: After finishing this step, it will automatically save the original mode after module reboot. It will be STA also if it is STA mode before reboot, and automatically add in hot spots of before reboot.**
1.1）Module work as CLIENT, single way connection, non-transparent transmission：		
→	AT+CIPSTATUS	Check current WIFI connected status
←	OK STATE:IP STATUS	Already joined up AP.
→	AT+CWJAP?	Check current joined AP.
←	+CWJAP:"yh_taihdhad" OK	
→	AT+CIFSR	Check current module IP address
←	172.168.1.114	
→	AT+CIPSTART="TCP","172.168.1.119",4321	Set up one TCP link
←	OK	
←	CONNECT OK	Means connection success.

→	AT+CIPSEND=5	Set sending length to 5, module will automatically send data after data up to 5 ones.
→	>hello	After > comes up, send 5 characters hello to serial port, module will automatically send data.
←	SEND OK	means sending is successful
←	+IPD,14:hello,thisispc	Module receives server's data and send to serial port directly.
→	AT+CIPCLOSE	Turn off connection
←	OK	
1.2）Module work as CLIENT, single way connection, data transparent transmission：		
→	AT+CIPMODE=1	set module to transparent transmission mode
←	OK	
→	AT+CIPSTART="TCP","172.168.1.119",4321	connect to one server
←	OK	
←	CONNECT OK	means connection is successful
→	AT+CIPSCON=1	Save parameter(link parameter and transparent transmission mode)
←	OK	
→	AT+CIPSEND	Enter transparent transmission. Enter transparent transmission directly without reboot.
←	>	After enter AT+CIPSEND, it will go back to >, then upper computer can send data to module from serial port, module can automatically send out data.
←	SSSSAAAAAAAA	After server sending "SSSSAAAAAAAA", module will display directly. **Note: No data head when received by transparent transmission.**
→	Reboot module	Module will enter transparent transmission directly after reboot and set up connection according to "TCP","172.168.1.119",4321.
→	+++	quit transparent transmission state Note: there should be no character in front or behind +++.
→	AT+CIPSCON	Inquire transparent transmission parameter and state
←	C:0,"TCP","172.168.1.119",4321,1793 C:1,"","",, C:2,"","",, C:3,"","",, S:0,, S:1,, DATA_MODE:1 OK	Inquire result is status and parameter of the last time when AT+CIPSCON=1.
→	AT+CIPSTATUS	
←	OK STATE:CONNECT OK C:0,"TCP","172.168.1.119",4321,"CONNECTED" C:1,"","",,"INITIAL"	

	C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	
→	AT+CIPMODE=1	
←	OK	
→	Reboot module. Reenter transparent transmission after reboot and set up one connection automatically. Connection parameter is the same as saved last time.	
1.3）Module work as CLIENT, single way connection, data transparent transmission.		
→	AT+CIPMUX=1	After CIPMUX=1, module can do transparent transmits only.
←	OK	
→	AT+CIPMODE=1	Transparent transmission mode
←	OK	
→	AT+CIPSTART=0,"TCP","172.168.1.119",4321	This is outer net server.
←	OK CONNECT OK	
→	AT+CIPSCON=1	Save parameter(connect parameter and transparent transmission mode)
←	OK	
	Reboot module. Reenter transparent transmission after reboot and set up two connections automatically. Connection parameter is the same as saved last time when +CIPSCON .	
1.4) Module work as SERVER, multi-way connection, non-transparent transmission.		
→	AT+CIPMUX=1	
←	OK	
→	AT+CIPSERVER=1,8800	Module works as SERVER
←	OK	configuration is successful, Can monitor TCP and UDP link at the same time
→	AT+CIPSTATUS	
←	OK STATE:IP STATUS S:0,8800,LISTENING C:0,"","",,"INITIAL" C:1,"","",,"INITIAL" C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	
←	0,CONNECT OK	CLIENT link to module (TCP)，linkid = 0
←	1,CONNECT OK	CLIENT link to module (UDP)，linkid = 1 **Note: No matter TCP or UDP, 4 links can be set up most.**
→	AT+CIPSTATUS	Inquire TCPIP level status
←	OK STATE:IP STATUS S:0,8800,LISTENING C:0,UDP,192.168.2.112,65074,CONNECTED	

	C:1,TCP,192.168.2.112,50114,CONNECTED C:2,"","",,"INITIAL" C:3,"","",,"INITIAL"	
←	+RECEIVE,0,10： 0123456789	Receive data from CLIENT，content is：0123456789
→	AT+CIPSEND=0,6	
←	>	Now data can be input. Input 6 characters, data will send initiatively.
←	0,SEND OK	Data sending successfully.
←	0,CLOSE OK	Opposite terminal （ CLIENT ） break connection initiatively.
→	AT+CIPSERVER=0	
←	OK we must restart	Need reboot module after shut down server.
1.5）Module works as SERVER, single way and multi-way connection, data		**transparent transmission.**
→	AT+CIPMUX=1	
←	OK	
→	AT+CIPSERVER=1,8800	Module works as SERVER
←	OK	Configuration is successful. can monitor TCP and UDP connection at the same time
→	AT+CIPMODE=1	
←	OK	
→	AT+CIPSCON=1	
←	OK	
	Enter transparent transmission mode after reboot. And wait for setting up CLIENT connection. There will be nothing reminder after connection set up successfully. Now data can be two-way transmitted.	模块作为SERVER可最多同时监听4条链接（TCP+UDP)
2）Module operates in AP mode： The following words in light blue are the same common premise steps in this mode:		
→	AT+CWMODE?	Step 1: check WIFI level operation mode
←	+CWMODE::3 OK	Default mode is 3: AP+STA mode
→	AT+CWMODE=2	Ste[2: Set as AP mode
←	OK	
→	AT+RST	Step 3: Need reboot after setting mode.
←	OK	Note: Step2 and Step 3 can be ignored if after booting AT+CWMODE it is already Step 1.
→	AT+CWSAP="aptest","123456",1,0	Step4: Set mode to AP and configure AP parameters.
←	OK	**Note: After finishing this step, it will automatically save the original mode after module reboot. It will be AP also if it is AP mode before reboot, and keeps the same AP parameters (SSID, password and so on) as before reboot.**
	STATION join uo aptest AP	

+CWMODE:
OK

2.1）CLIENT，single way connection， non-transparent transmission data：		
	Please refer to 1.1) partial examples	
2.2）CLIENT，single way connection， transparent transmission data：		
	Please refer to 1.2) partial examples	
2.3）SERVER，multi-way connection， non-transparent transmission data：		
	Please refer to 1.3) partial examples, different part is all connections are internet connection under this application.	
2.4) SERVER，multi-way connection， non-transparent transmission data：		
	Please refer to 1.4) partial examples	
2.5) SERVER，single way and multi-way connection, transparent transmission data：		
	Please refer to 1.5) partial examples	
3）Module work in AP+STATION mode ： The following words in light blue are the same common premise steps in this mode：：		
→	AT+CWMODE=3	Set as AP + STA mode
←	OK	
→	AT+RST	Need reboot after configuring mode.
←	OK	
→	AT+CWJAP="yh_taihdhad","UUU02339"	Add into one AP which is a router can access outer net.
←	OK	
→	AT+CWSAP="aptest","123456",1,0	Configure module as （name as aptest），and configure AP parameter
←	OK	
	There is one computer(name as user) using WIFI to join this aptest AP.	
→	AT+CIPMUX=1	Start multi-way connection mode
←	OK	
→	AT+CIPSTART=1,"TCP","120.209.197.145",8000	Set up TCP connection with one SERVER of outer net. Connection number is 1.
←	OK 1,CONNECT OK	Successfully connect to outer net SERVER.
3.1) Module works as CLIENT to link local server, meanwhile works as CLIENT to link outer net server.		
→	AT+CIFSR	
←	192.168.4.1 192.168.2.110 OK	192.168.2.110 is the allocated IP address when joins router which can surf outer net. 192.168.4.1 is the IP address and module works as AP itself.
→	AT+CIPSTART=2,"TCP","192.168.4.100",7777	To set up one connection with USER's server, connection number is 2.
←	OK 2,CONNECT OK	
←	+RECEIVE,2,11： hhhhhhfdddd	receive one pack of data from USER
→	AT+CIPSEND=1，11	Receive data from USER's server (link 2) and send to outer net's server (link 1)
←	>	input hhhhhhfdddd
←	1,SEND OK	When character's number is up to 11, it will send out by itself.
3.2）Besides linking to outer net's server, module also works as CLIENT to link local server, service including: multi-way link and transparent transmission.		

→	AT+CIPSTART=2,"TCP","192.168.4.100",7777	To set up one connection with USER's server, connection number is 2.
←	OK 2,CONNECT OK	
→	AT+CIPMODE=1	
←	OK	
→	AT+CIPSCON=1	
←	OK	
	Reboot module. Module will enter transparent transmission after reboot and automatically sets up 1 and 2 connections at the same time.	
→	+++	
→	AT+CIPSTATUS	Linking status can be checked after quiting transparent transmission mode
←	STATE:IP STATUS C:0,"","",,"INITIAL" C:1,"TCP","120.209.197.145",8000,"CONNECTED" C:2,"TCP","192.168.4.100",7777,"CONNECTED" C:3,"","",,"INITIAL"	
3.3 Besides linking to outer net's server, module also works as server to monitor local connection, service including: multi-way link and non- transparent transmission.		
→	AT+CIPSERVER=1,3366	
←	OK	
←	0,CONNECT OK	Working as CLIENT, USER set up one UDP connection joining up module's AP, linking number is 0.
←	2,CONNECT OK	Working as CLIENT, USER set up one TCP connection joining up module's AP, linking number is 1.
→	AT+CIPSEND=0,8	
←	>	After input 12345678, then data can send out by itself.
←	0，SEND OK	
→	AT+CIPSTATUS	Inquire TCPIP status
←	OK STATE:IP STATUS S:0,3366,LISTENING C:0,UDP,192.168.4.100,51733,CONNECTED C:1,TCP,120.209.197.156,8000,CONNECTED C:2,TCP,192.168.4.100,57957,CONNECTED	
3.4）Besides linking to outer net's server, module also works as server to monitor local connection , service including: multi-way link and transparent transmission.		
→	AT+CIPSERVER=1,3366	
←	OK	
→	AT+CIPMODE=1	
←	OK	
→	AT+CIPSCON=1	
←	OK	

	Reboot module. It will enter transparent transmission mode automatically and set up connection 1,"TCP","120.209.197.145",8000. Module works as SERVER at the same time to monitor port 3366, waiting for setting up connection.	

FCC Caution: Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.
This device complies with Part 15 of the FCC Rules.
Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.
This device and its antenna(s) must not be co-located or operating in conjunction with any other antenna or transmitter.

This equipment should be installed and operated with minimum distance 20cm between the radiator and your body.

15.105 Information to the user.
(b) For a Class B digital device or peripheral, the instructions furnished the user shall include the following or similar statement, placed in a prominent location in the text of the manual:
Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:
—Reorient or relocate the receiving antenna.
—Increase the separation between the equipment and receiver.
—Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
—Consult the dealer or an experienced radio/TV technician for help.

Radiation Exposure Statement:

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment.

This transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.

The availability of some specific channels and/or operational frequency bands are country dependent and are firmware programmed at the factory to match the intended destination.

The firmware setting is not accessible by the end user.

The final end product must be labelled in a visible area with the following:

“Contains Transmitter Module 2AEGGA6501”

”