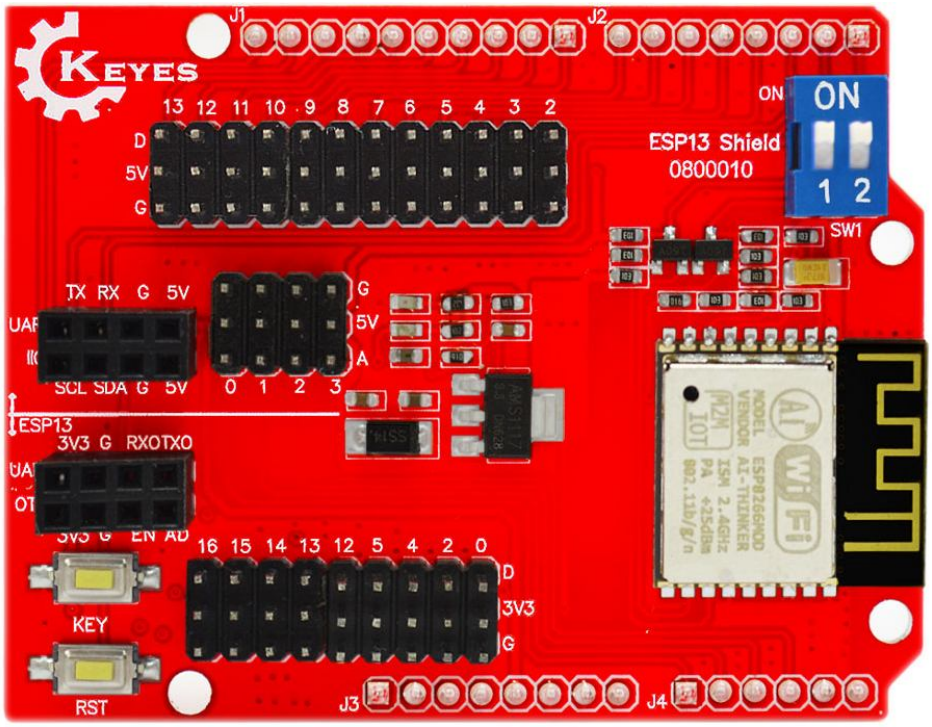
Arduino UNO R3 ESP8266 Web Server Serial Port WiFi Shield ESP13



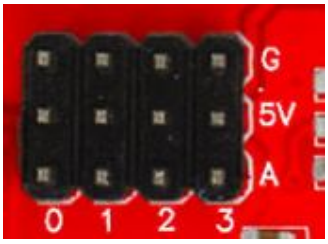
1. Pin Explanation

G: GND

5V: 5V Power Output

D：connecting to digital port（D2-D13）of arduino control board

Control Switch: to control the serial connection between ESP8266 and arduino control board

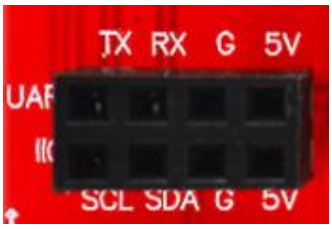


、

G：GND

5V：5V Power Output

A：connecting to analog port（A0-A3）of arduino control board



UART: connecting to serial communication port（5V GND RX TX）of arduino control board

IIC: connecting to IIC communication port（5V, GND, SDA, SCL）of arduino control board

UART: connecting to serial communication port（3.3V, GND, RX, TX）of ESP8266 module

OTH

3V3：3.3V Power Output

G：GND

EN：ESP8266 EN

AD：ESP8266 AD

G：GND

3V3：3.3V Power Output

D：lead ESP8266 GPIO（GPIO 0/2/4/5/12/13/14/15/16）

Buttons for burning the firmware

Finally, click on flash to enter the programming wait mode, and at this moment keep pressing the RST button of firmware pass-through board, then press down the KEY key and release it. Waiting for a moment, you can find that the programming progress bar starts to roll.

2. Product Features

1) WiFi adopts industrial chip ESP8266. ESP-13 module with a metal shield has strong anti-interference ability.

2) Standard compatible with pin of Arduino Uno, Mega2560 and other control board.

Connecting to Arduino Uno using voltage conversion chip, making 3.3V and 5V compatible.

3) The serial port is controlled by dual DIP switch, so that the shield can be used alone as an Arduino Uno expansion board or as an ESP8266 expansion board.

4) Configure WiFi and serial port parameters based on Web Server.

5) Can be used as an independent ESP8266 development board, to download the official AT command firmware, NodeMCU open source firmware, etc.

6) Can be used as a standalone Arduino Uno shield, leading out all the pins.

3. Technical Specifications

1) Supporting wireless 802.11 b / g / n standards.

2) Supporting STA / AP two operating modes

3) Built-in TCP / IP protocol stack, you can configure a socket.

4) Supporting standard TCP / UDP Server and Client.

5) Supporting the baud rate of serial port as

6) 1200/2400/4800/9600/19200/38400/57600/74800/115200 bps.

7) Serial data: 5/6/7/8 bit.

8) Serial parity: without

9) Serial port stop bit: 1/2 bit

10) Standard Arduino Uno, Mega pin pitch

11) Lead Arduino Pin 2/3/4/5/6/7/8/9/10/11/12/13

12) Lead ESP8266 GPIO 0/2/4/5/9/10/12/13/14/15/16 / ADC / EN / UART / TX / UART RX / UART

13) RESET button

14) KEY button for multiplexing configuration function.

15) Dual DIP switch can achieve the Arduino and ESP8266 serial port expansion switching.

16) WiFi operating current: continuous transmission ≈ 700mA (200mA MAX), standby <200uA.

17) Wireless transmission rate: 110-460800bps.

18) Operating temperature: -40 □ - +125 □

4. Using Method

The board is initially by default the firmware which has been programmed to support the AT command, and the firmware version is v0.9.5.2 AT Firmware.

Open the serial debugging tool, set up it well as the figure shown below, then click to test AT. Under normal circumstances, the board will return OK to response AT. When Arduino UNO has been communicated with this board via serial port, testing code is as below.

```
*********************************************************************************
 void setup() {
 Serial.begin(115200);
 delay(100);
 Serial.println("AT");
 delay(5000);
 Serial.println("AT+CWMODE=3");     //mode:3 is SerialNet mode STA+AP
 delay(5000);
 Serial.println("AT+RST");     //reset
 delay(5000);
 //*****************************change WiFi name
 Serial.println("AT+CWSAP=\"xingxing\",\"123456789\",11,0");     //change WiFi name as xingxing,  password 123456789
 delay(5000);
 //*****************************connect to WIFI
 // Serial.println("AT+CWJAP=\"xingxing\",\"123456789\"");   //connect to wifi
 //delay(5000);
 //Serial.println("AT+CIFSR");     //inquiry IP
```

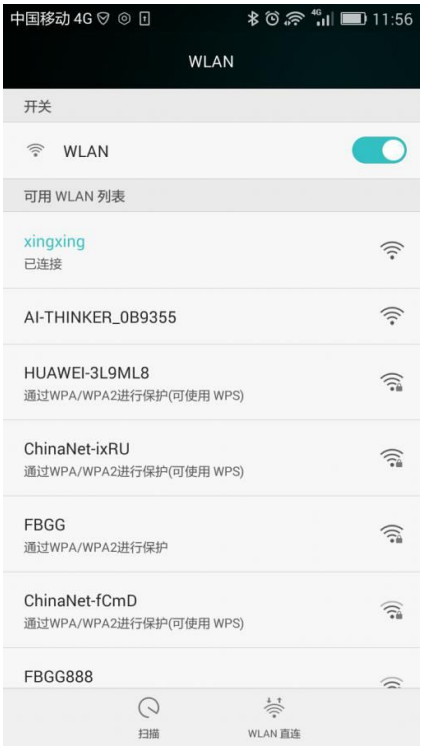
```
 //delay(5000);
 //Serial.println("AT+CIPMUX=0");     //set as single link mode
 //delay(5000);
 //Serial.println("AT+CIPMODE=1");     //set as SerialNet mode
 //delay(5000);
 //Serial.println("AT+CIPSTART=\"TCP\",\"172.19.27.1\",8080"); //connect to server
 //delay(5000);
 //Serial.println("AT+CIPSEND");     //start SerialNet mode
 //delay(5000);
}

void loop() {
 //Serial.println("hello!");
 delay(1000);
}
```

****

In the code,we have set the working mode、name、password and so on via AT command. If needed to change the mode, you can directly find the corresponding AT command from this tool esp8266调试工具-v1.1

Uploading the code to Arduino UNO board, stack the shield onto Arduino UNO board, powered on, you can connect this WIFI on the phone without password, as the figure shown below.



5. Burning firmware

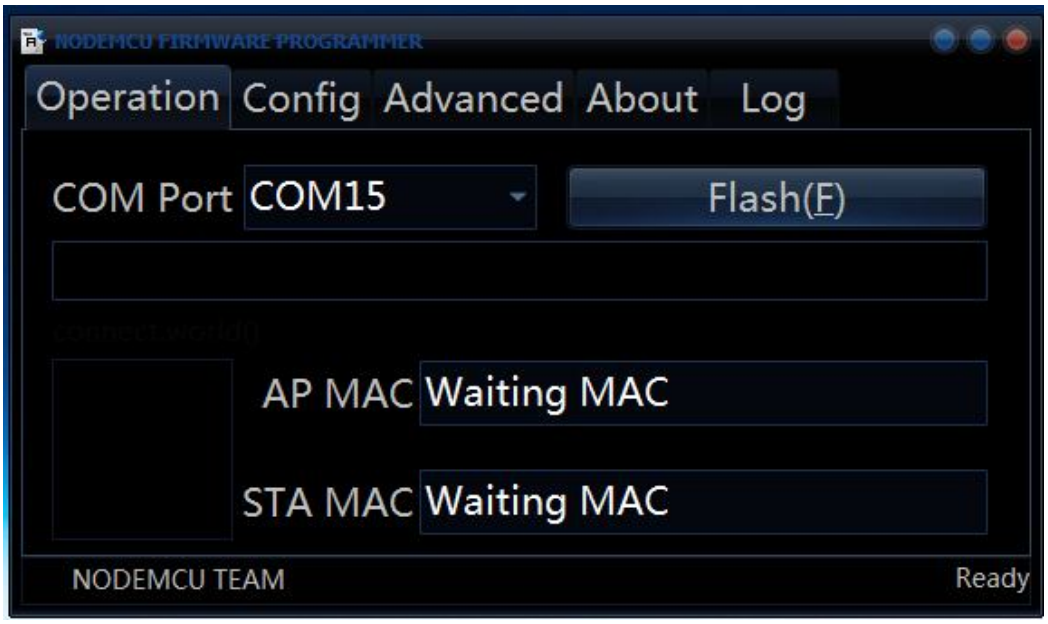
Connect external power supply to the arduino control board, with the range of DC 7-12V. And its GND and 5V interfaces are separately connected to the GND and 5V interfaces of WiFi shield. Connect the GND RX TX of FT232 module to the shield, and the wiring method is as follows:

FT232_TXD------Digital Port1

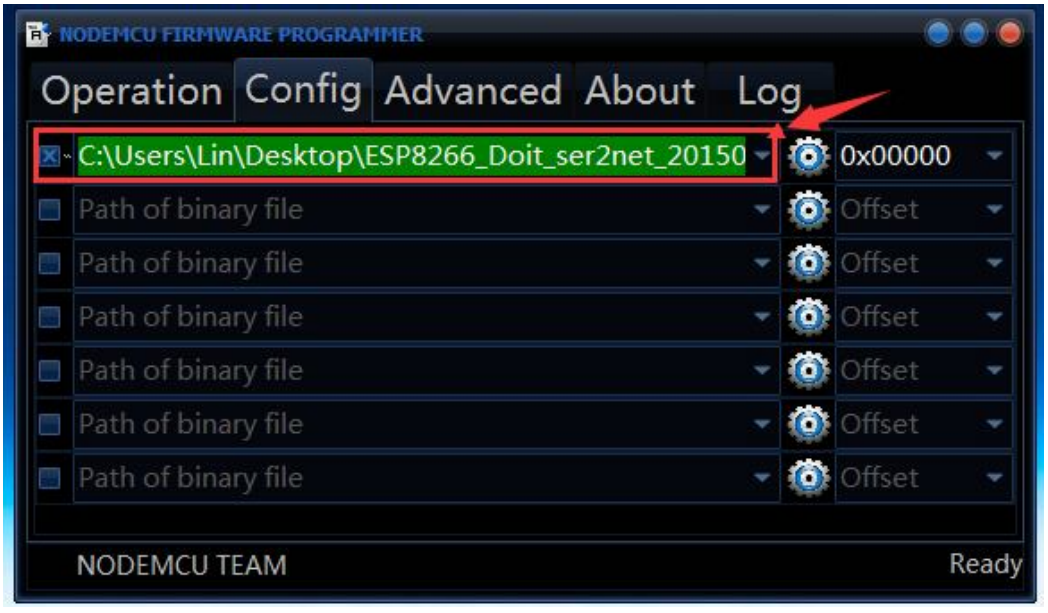
FT232_RXD------Digital Port0

GND----------GND

a. Open ESP8266Flasher program, select COM Port.

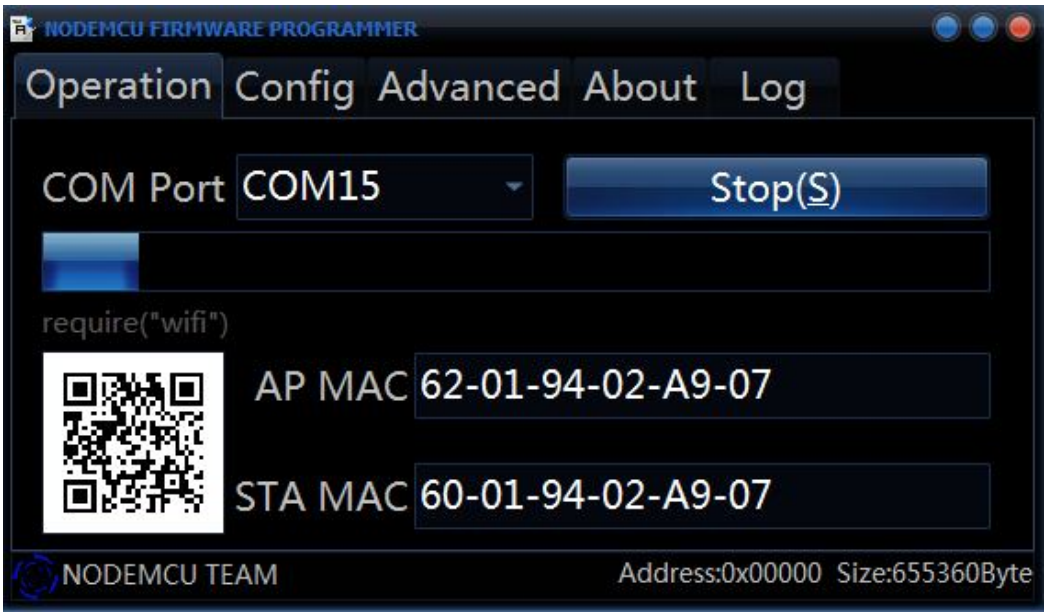


b. Click **Config**, select the firmware, and firmware version is ESP8266_Doit_ser2net_20150723.



c. Click **Operation**, click on 

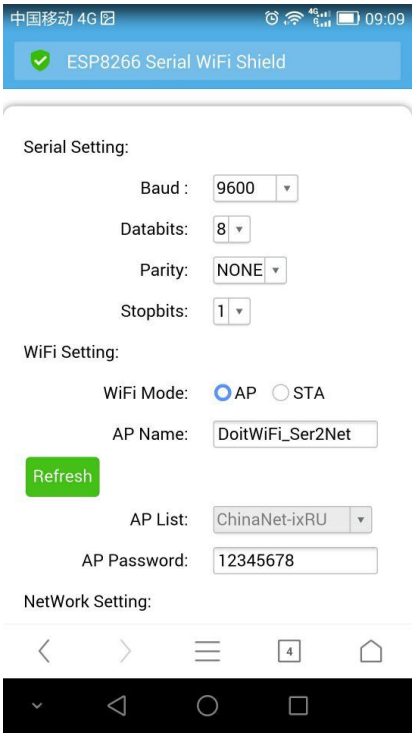
, press the RST button, then press the KEY button, starting to burn the firmware as shown below.



6. Configuration Method

After successful programming, power off to restart, arduino control board is connected to external power supply, with the range of DC 7-12V. And its GND and 5V interfaces are separately connected to GND and 5V interfaces of WiFi shield.

Then open the phone's WIFI to search for WIFI DoitWiFi_Config, and connect to it using password 12345678. In the browser input the address of WiFi shield, http: //192.168.4.1, as shown below.


中国移动 4G
09:09
ESP8266 Serial WiFi Shield
Serial Setting:
Baud :
9600
Databits:
8
Parity:
NONE
Stopbits:
1
WiFi Setting:
WiFi Mode:
AP
STA
AP Name:
DoitWiFi_Ser2Net
Refresh
AP List:
ChinaNet-ixRU
AP Password:
12345678
NetWork Setting: