SKU:TEL0092 WiFi Bee-ESP8266 Wirelss module

Contents

(/wiki/index.php/File:TEL0092_frontpage.jpg)

WiFi Bee-ESP8266 SKU:TEL0092

Introduction

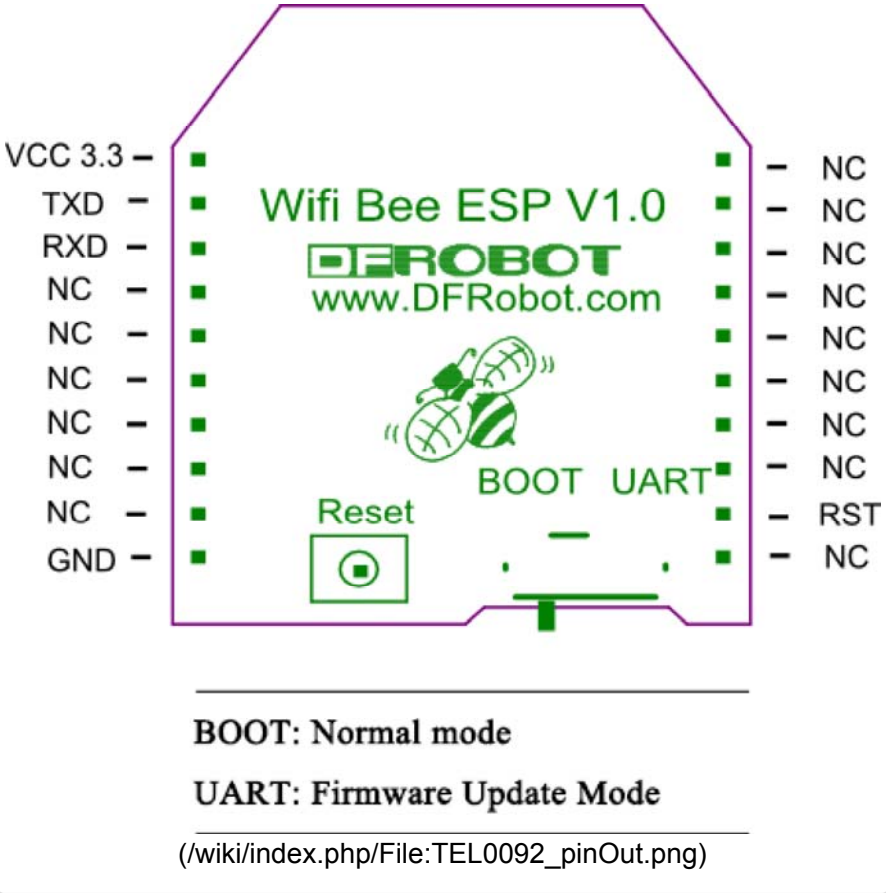
Wifi Bee-ESP8266 is a Serial-to-WIFI module using XBEE design in a compact size, compatible with XBEE expansion base, applicable to a variety of 3.3V single-chip system. It can be used for Arduino, wireless data transfer, remote control. On-board switch can be used to easily select the Startup module or Upgrade firmware.

ESP8266 has a powerful on-chip processing and storage capacity, built-in 32-bit processor, built-in Lwip protocol stack. Support AP+STA mode co-exist. And you could configure various parameters via AT commands.

Specifications

1. Wi-Fi Direct (P2P),soft-AP
2. Built-in TCP/IP protocol stack
3. Built-in low-power 32-bit CPU: can work as an application processor
4. Support WPA WPA2/WPA2-PSK encryption
5. Support UART interface
6. Support for TTL serial port to wireless application
7. Working voltage: 3.3V power <240Ma
8. Wireless standard: IEEE802.11b/g/n
9. Frequency: 2.4 GHz

Pin out



(/wiki/index.php/File:TEL0092_pinOut.png)

Tutorials

These stuffs are needed:

1 Software

1. ESP_Flasher
2. Arduino IDE 1.0.6
3. CoolTerm
4. NetAssist

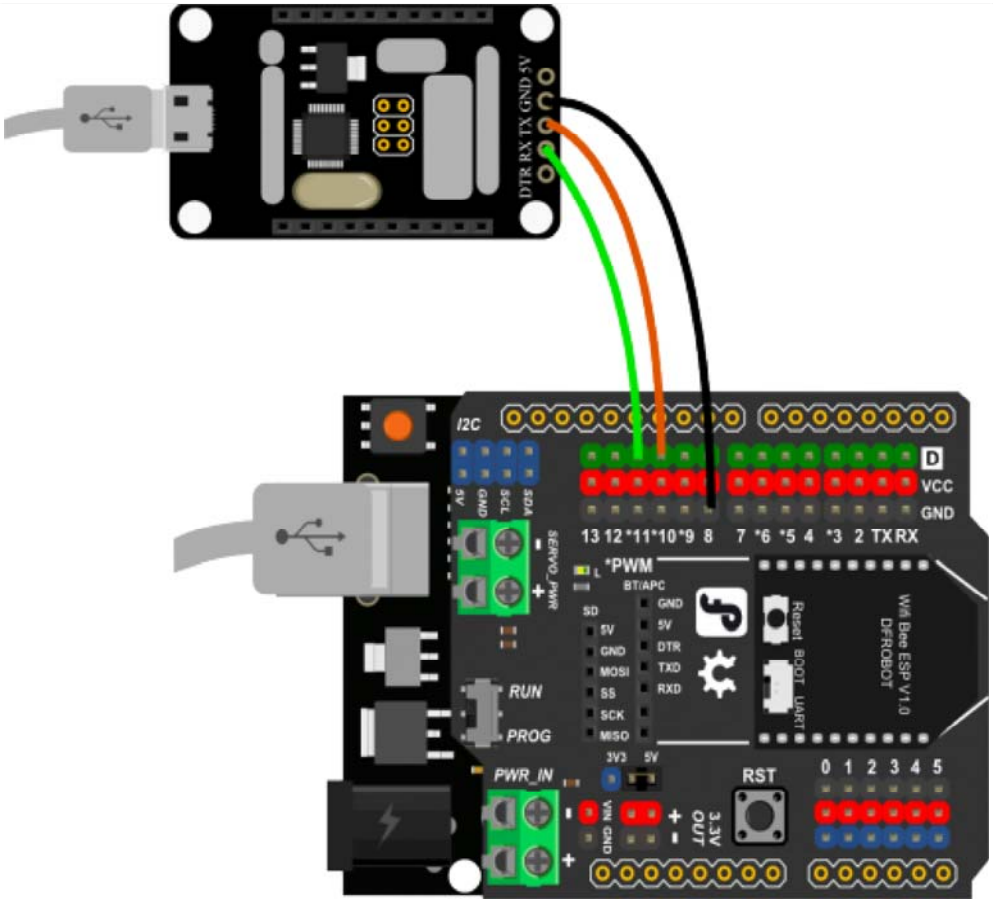
2 Hardware

1. DFRduino UNO R3 (http://www.dfrobot.com/index.php?route=product/product&product_id=838&search=uno&description=true)
2. Xbee USB adapter (FTDI ready) (http://www.dfrobot.com/index.php?route=product/product&product_id=72&search=USB&description=true&page=2)
3. IO Expansion Shield for Arduino (V6) (http://www.dfrobot.com/index.php?route=product/product&product_id=1134&search=expansion&description=true)
4. USB Cable A-B for Arduino (http://www.dfrobot.com/index.php?route=product/product&product_id=134&search=usb&description=true)
5. Mini USB cable (http://www.dfrobot.com/index.php?route=product/product&product_id=215&search=usb&description=true&page=2&description=true)

How to Use ?

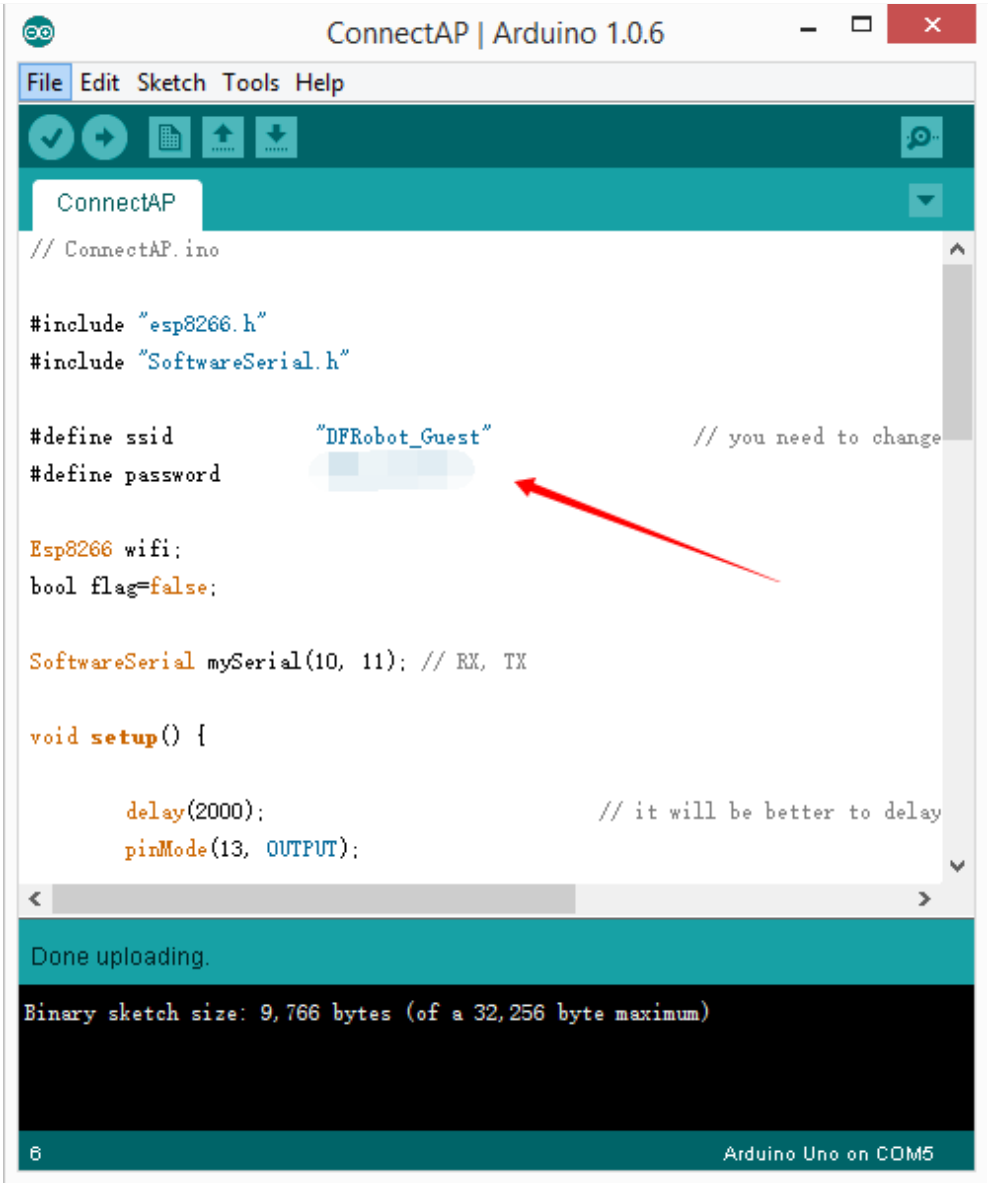
1 Connect AP

1 Download the ESP8266 library (http://www.dfrobot.com.cn/images/upload/File/TEL0092/2015050816413167nsek.rar), and unzip it to "C:\Users***\Documents\Arduino\libraries"
2 Insert the expansion shield on UNO, and plug ESP8266 in the socket on the expansion shield.
3 Note the switch: “RUN/Prog” at “Prog” side; “BOOT/UART” at "BOOT" side;
4 Wire adapter to the expansion shield: **TX- PIN10, RX- PIN11, GND - GND**;



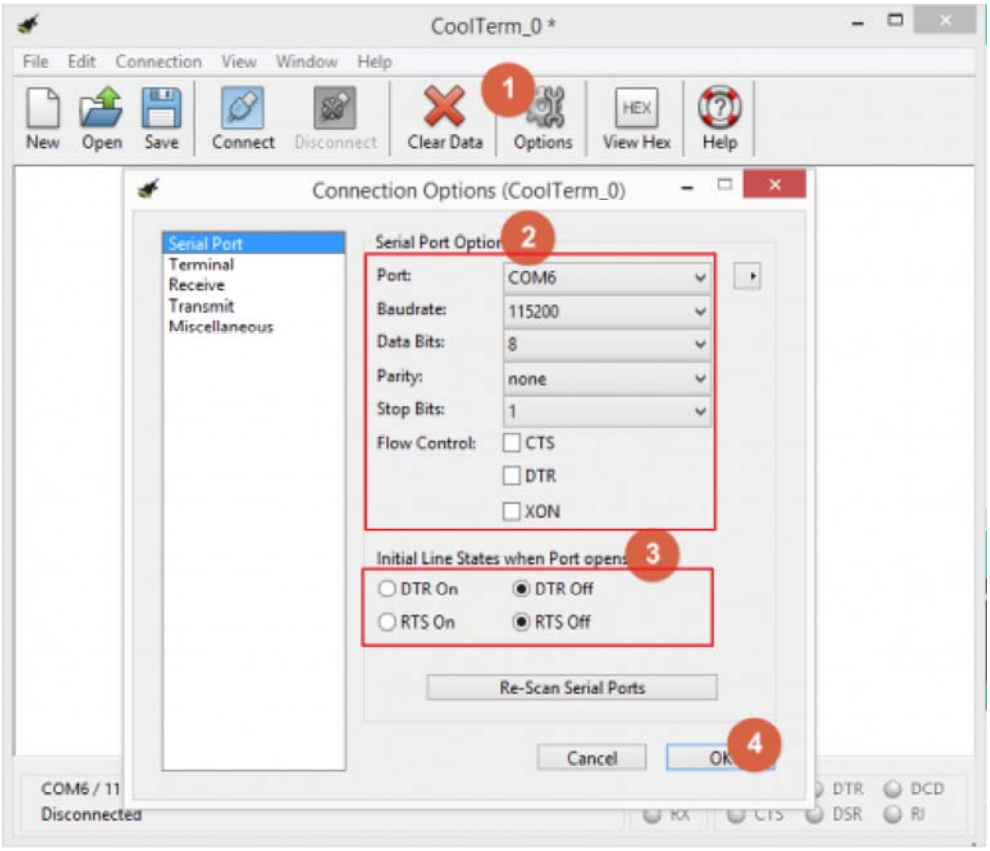
(/wiki/index.php/File:TEL0092_Connectiondiagram.png)

5 Open the sketch "Connect AP", and modify the wifi AP ssid and password of yours;

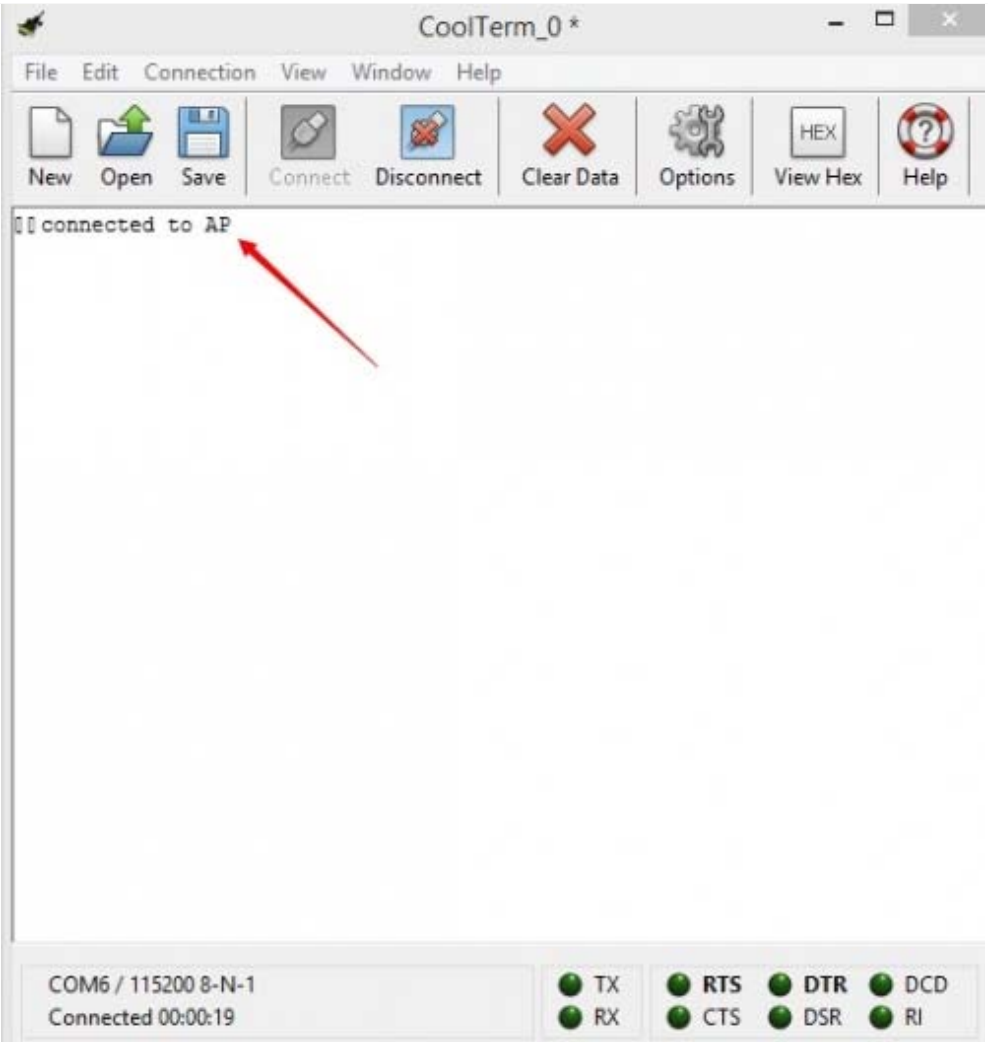


(/wiki/index.php/File:TEL0092_Connect_AP_chan.png)

6 Upload the sketch;
7 Trun the switch of the expansion shield “RUN/Prog” to “Run” side;
8 Use software "CoolTerm" to monitor if the AP connection is done.(Configure as the follow picture)



(/wiki/index.php/File:TEL0092_Connect_AP_para.png)

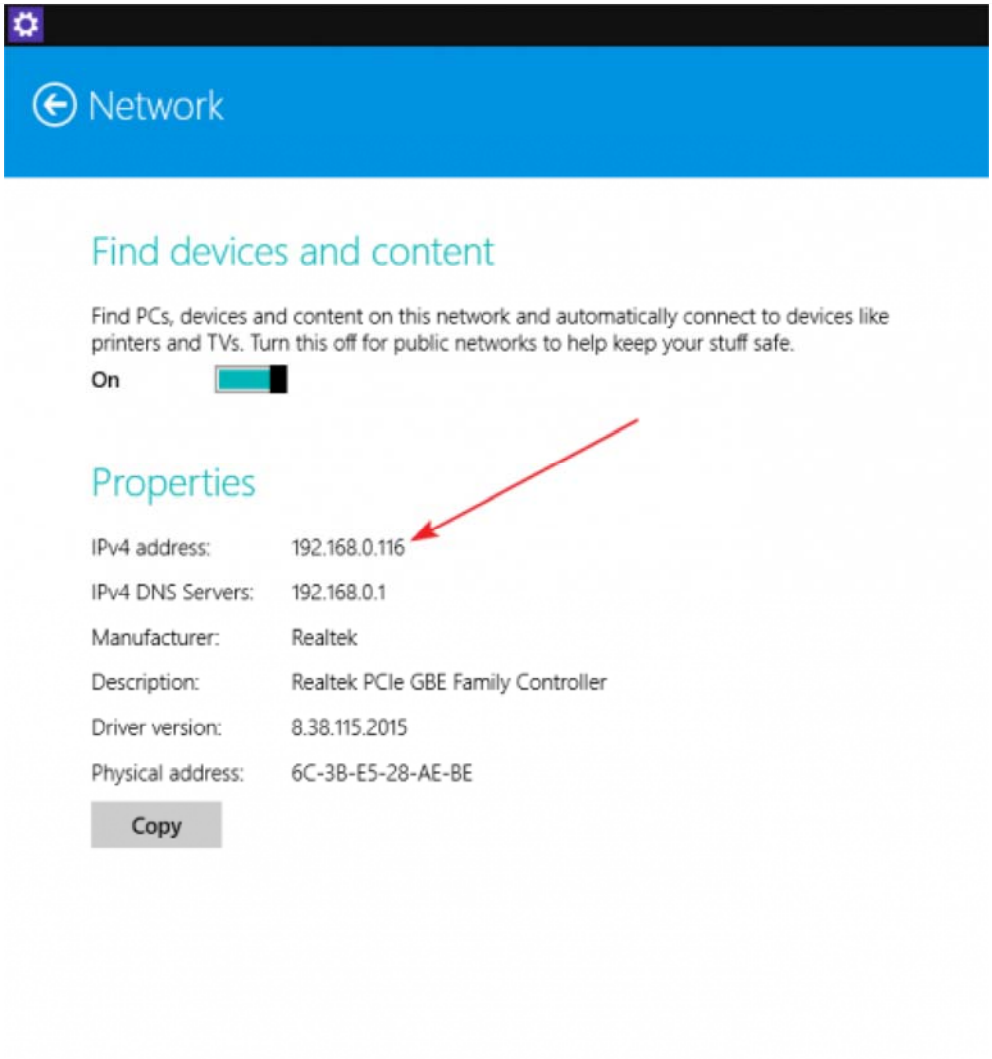


(/wiki/index.php/File:TEL0092_ctmode_choo2asd.jpg)

2 TCP_Client_Single

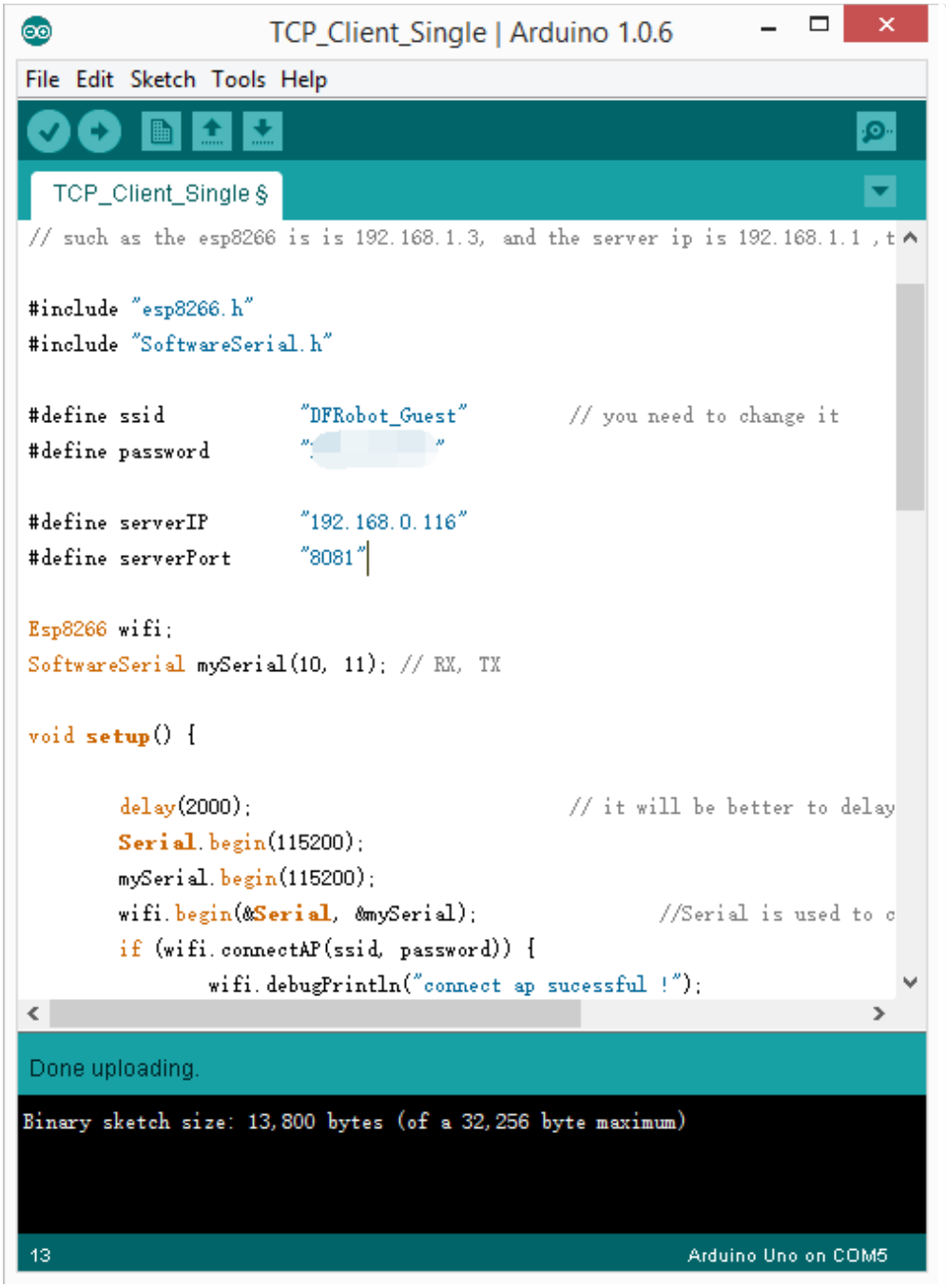
1 Pull the switch"RUN/Prog" to "Prog";

2 Check the Ip address of your computer, we will use it as the Server Ip address;



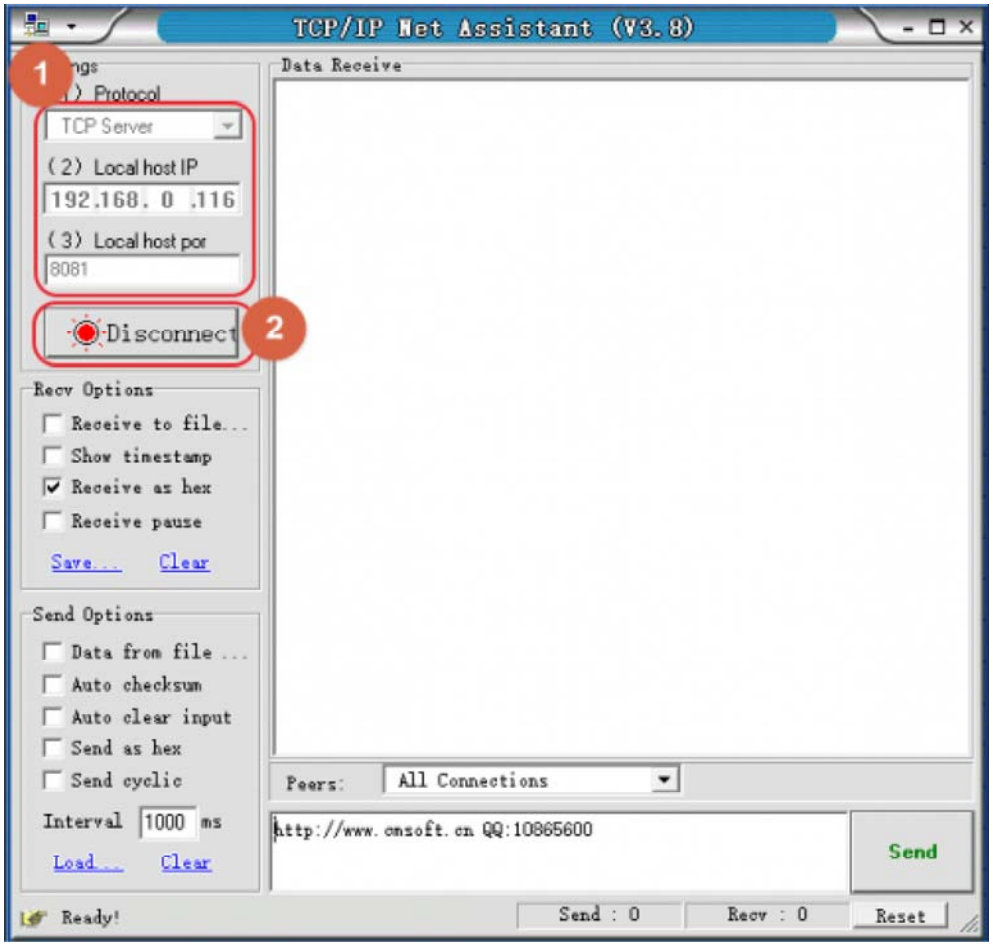
(/wiki/index.php/File:TEL0092TCP_Client_Single_checkIPpc.png)

3 Open sample sketch **"TCP_Client_Single"**, write your **wifi's ssid, password , Server Ip address, port**;



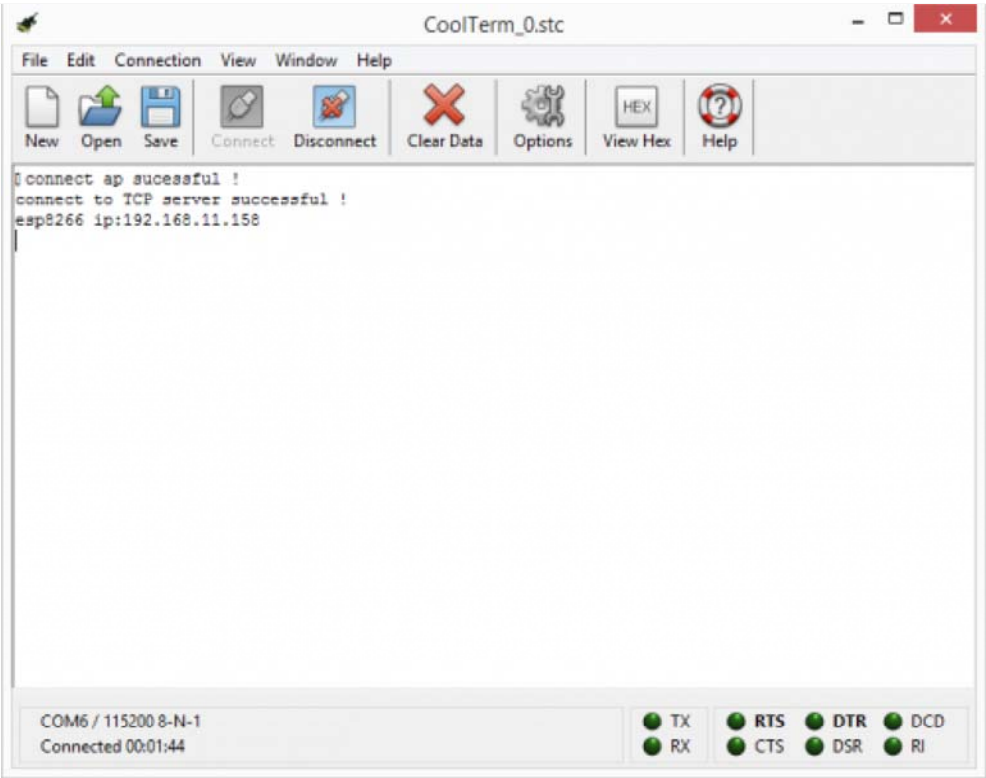
(/wiki/index.php/File:TEL0092TCP_Client_Single_sampleMod.png)

4 Open the software"TCP/IP Net Assistant V3.8", configure as follow, and click **Connect** to monitor the PC doing as Server;



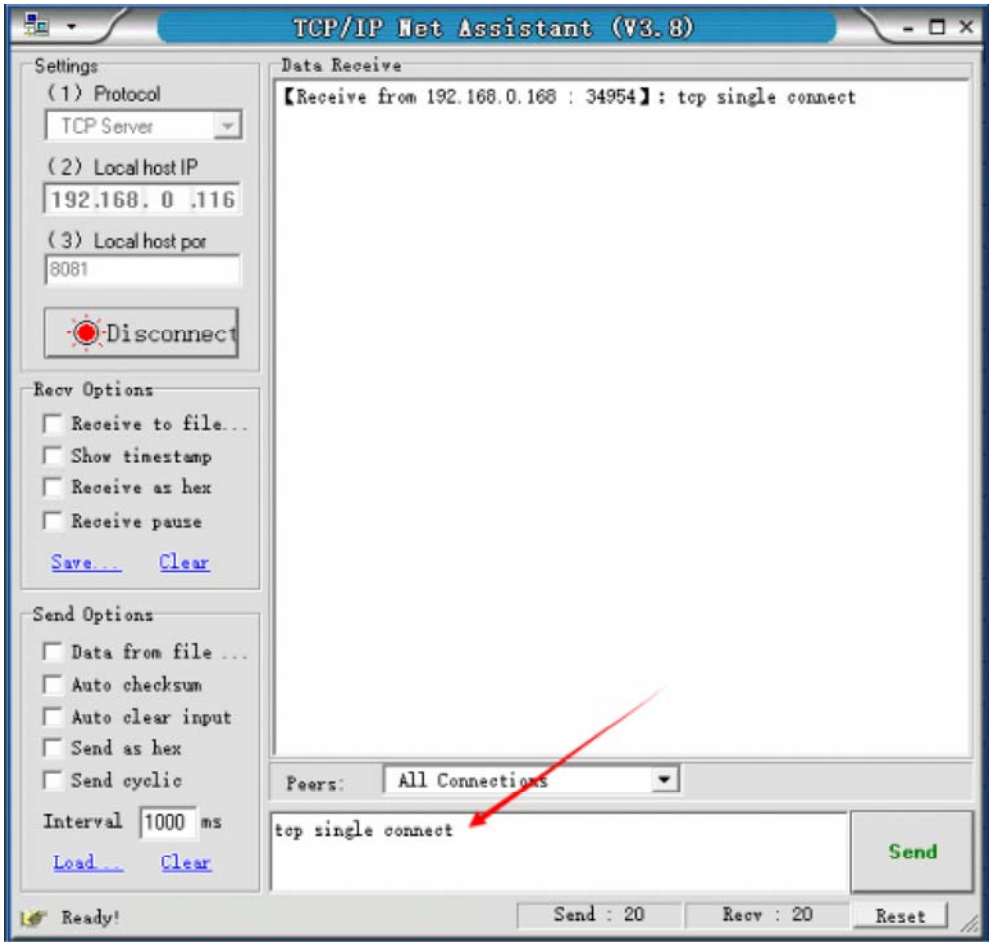
(/wiki/index.php/File:TEL0092TCP_Client_Single_assisP.png)

5 Upload the modified sketch, and then pull the switch"RUN/Prog" to "RUN";
6 Open **"CoolTerm"** and monitor the if the AP connection was good;

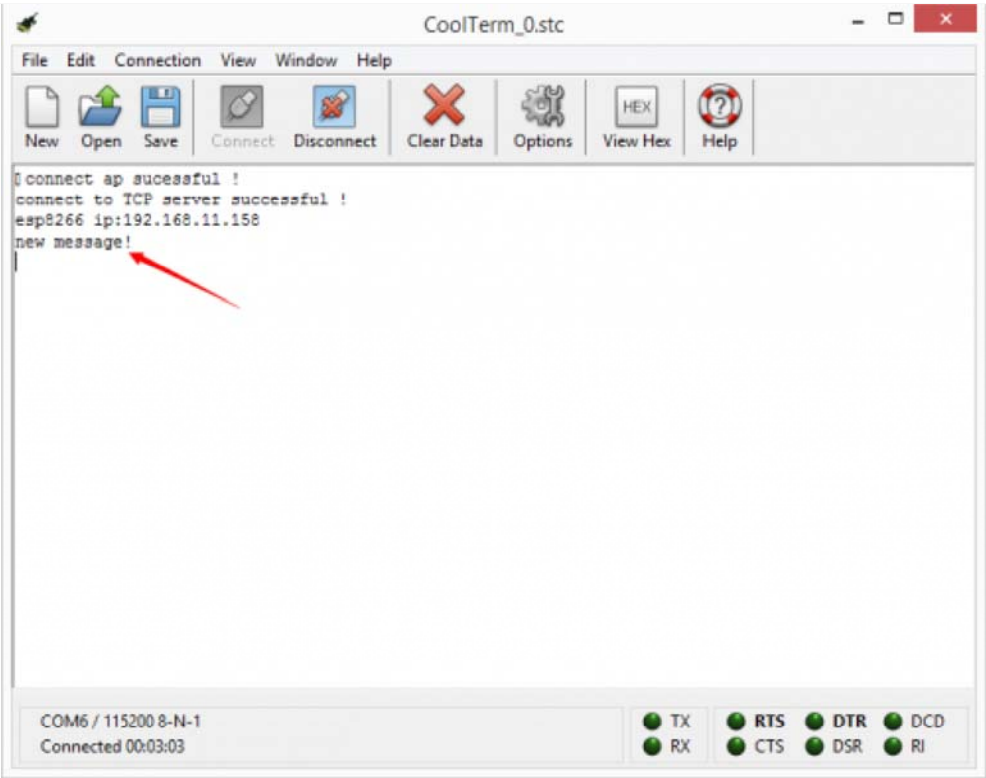


(/wiki/index.php/File:TEL0092TCP_Client_Single_cooltC.png)

7 Send a message from **"TCP/IP Net Assistant V3.8"**, you could see "New message" appear on **"CoolTerm"** .



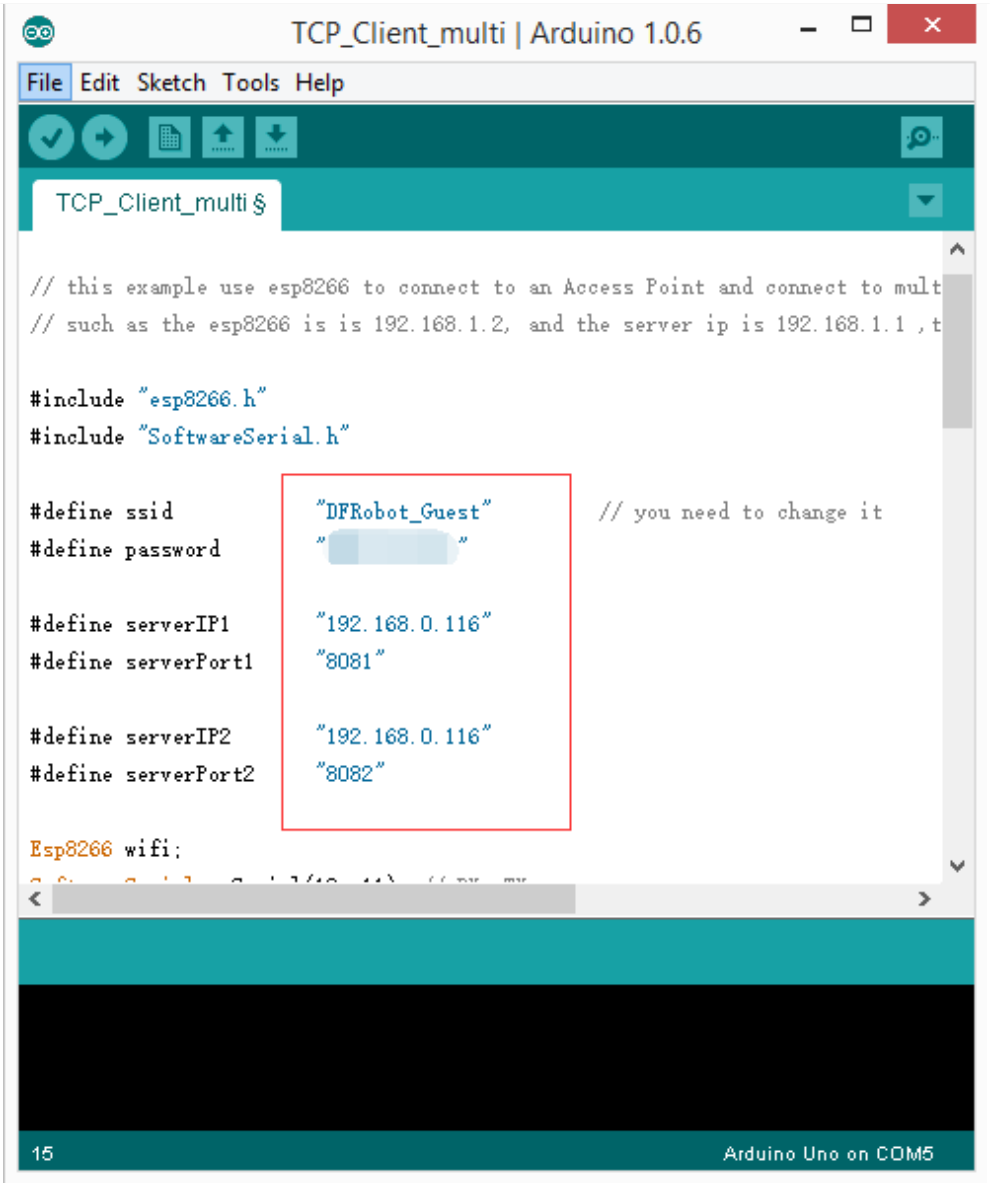
(/wiki/index.php/File:TEL0092TCP_Client_Single_Send.png)



(/wiki/index.php/File:TEL0092TCP_Client_Single_recv.png)

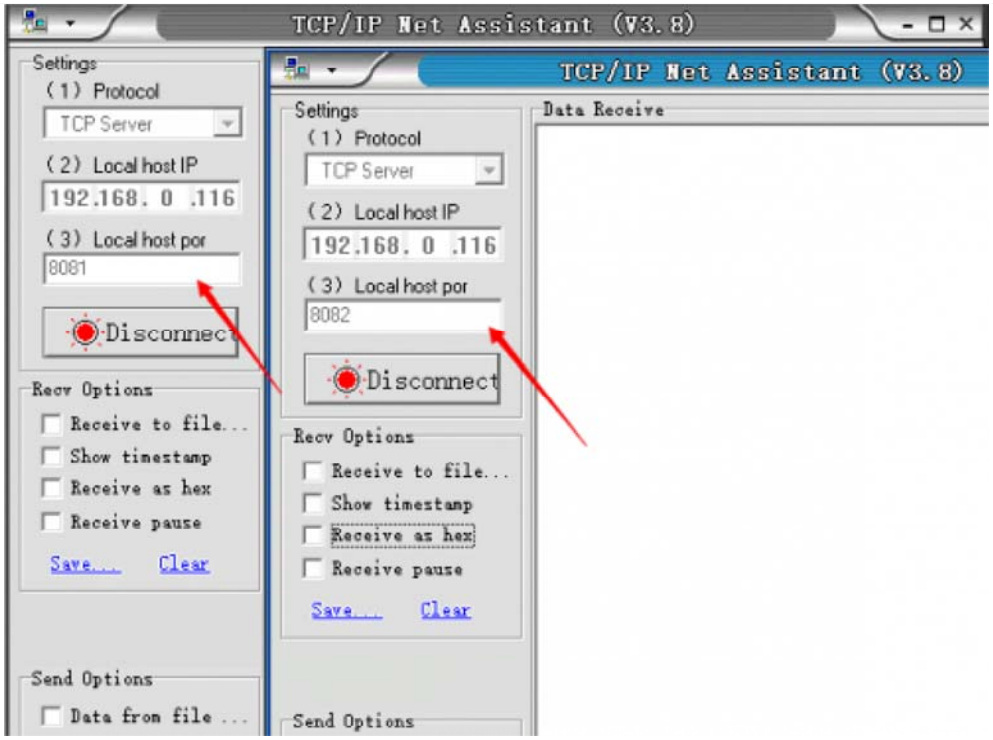
3 TCP_Client_multi

1 Pull the switch"RUN/Prog" to "Prog";
2 Open sample sketch **"TCP_Client_multi"**, write your **wifi's ssid, password , Server Ip address, port**;



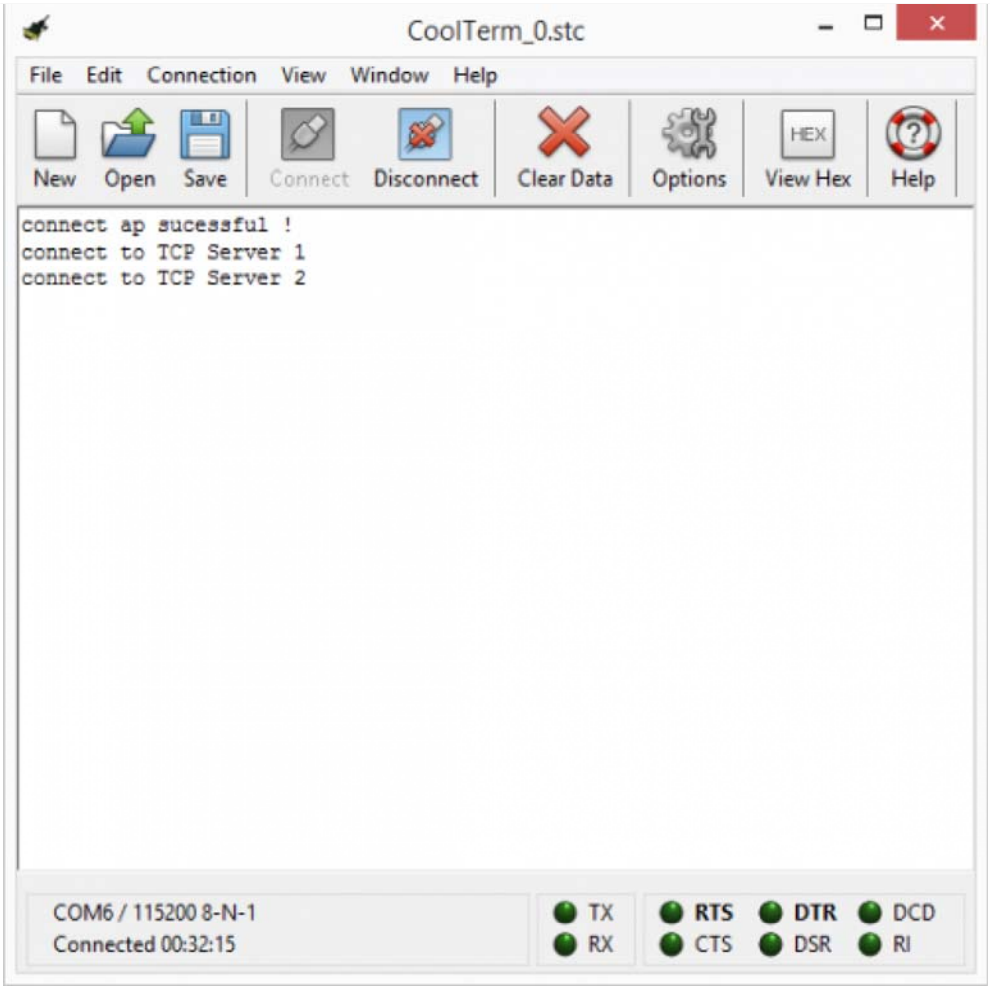
(/wiki/index.php/File:TEL0092TCP_Client_multi_modif.png)

3 Open another "TCP/IP Net Assistant V3.8", configure as follow, and click **Connect**;



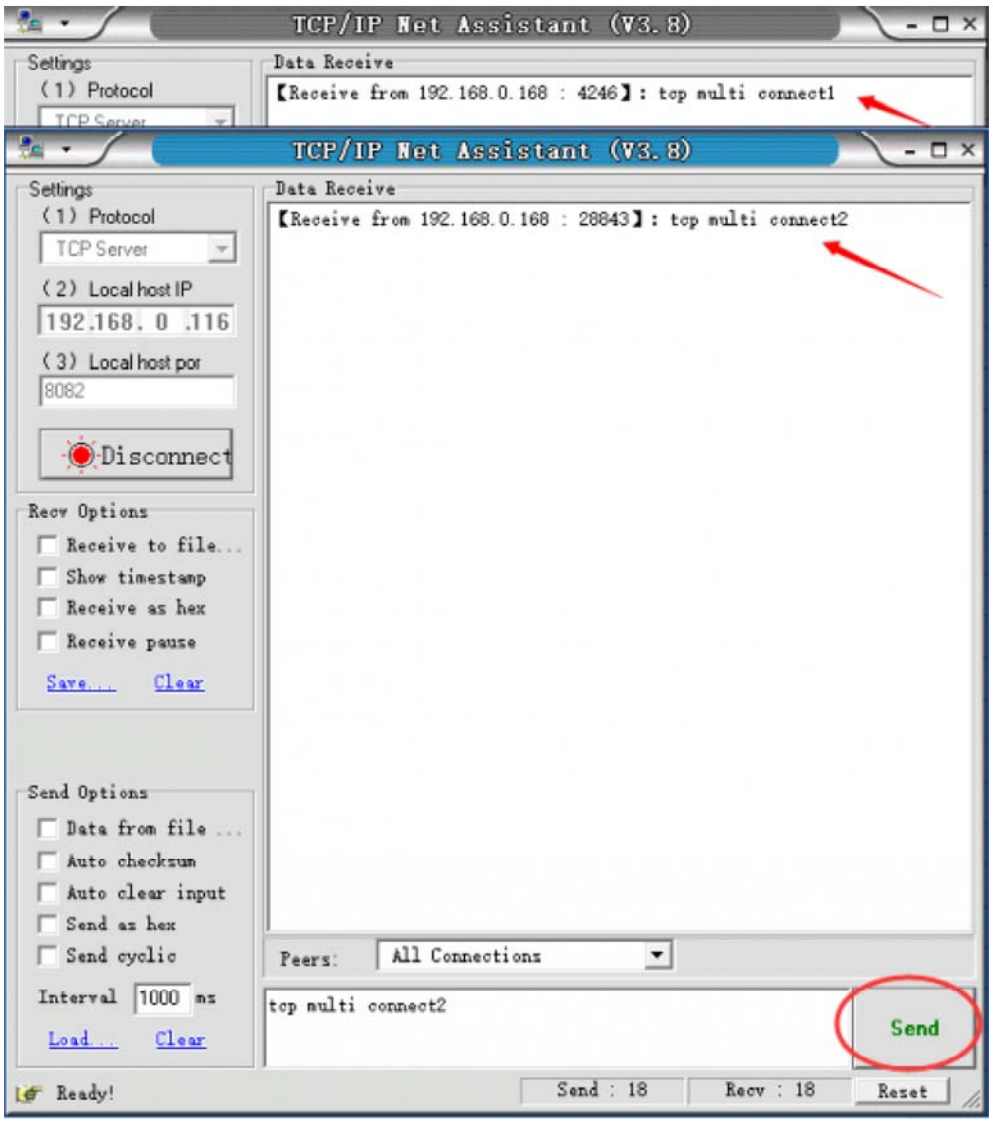
(/wiki/index.php/File:TEL0092TCP_Client_multi_tcpConfig.png)

4 Upload the modified sketch, and then pull the switch"RUN/Prog" to "RUN";
6 Watch**"CoolTerm"** to see if the AP connection was good;

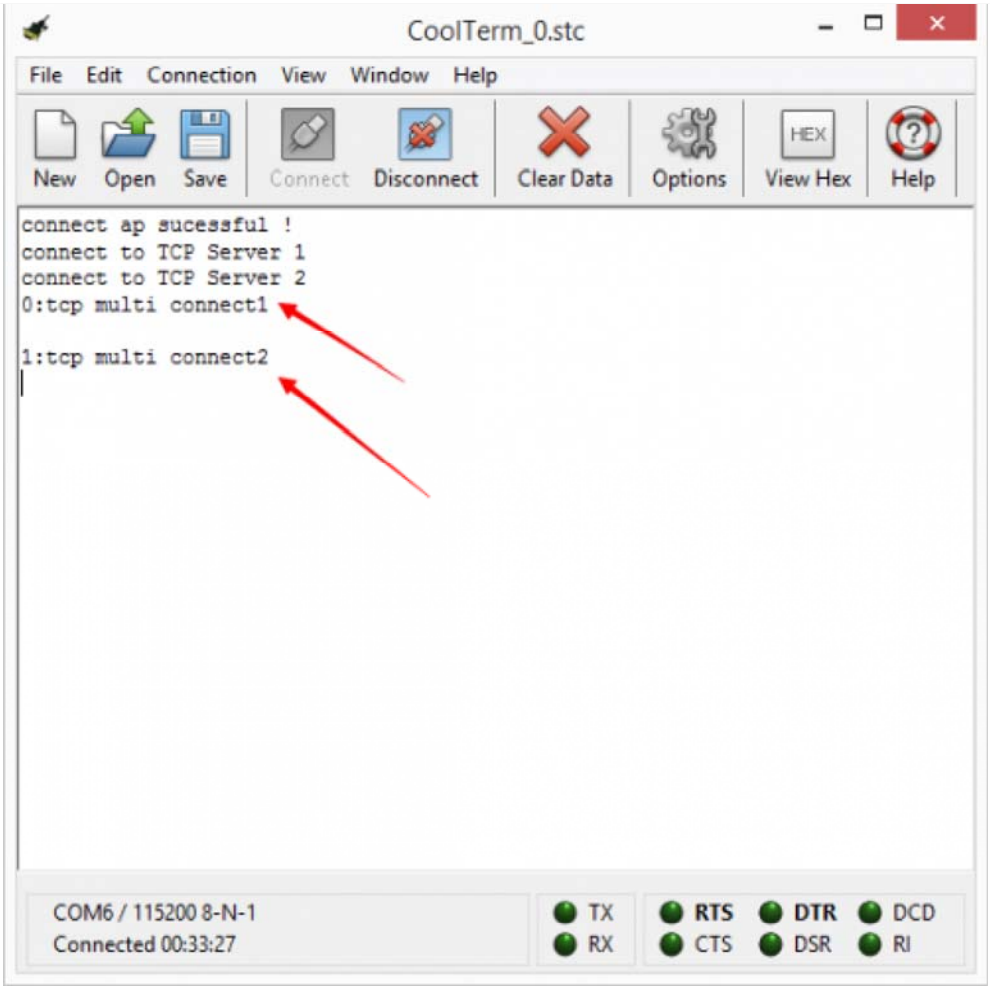


(/wiki/index.php/File:TEL0092TCP_Client_multi_ctemif.png)

7 Send a message from different **"TCP/IP Net Assistant V3.8"**, you could see New message appear on **"CoolTerm"** from different server .



(/wiki/index.php/File:TEL0092TCP_Client_moti_Send.png)

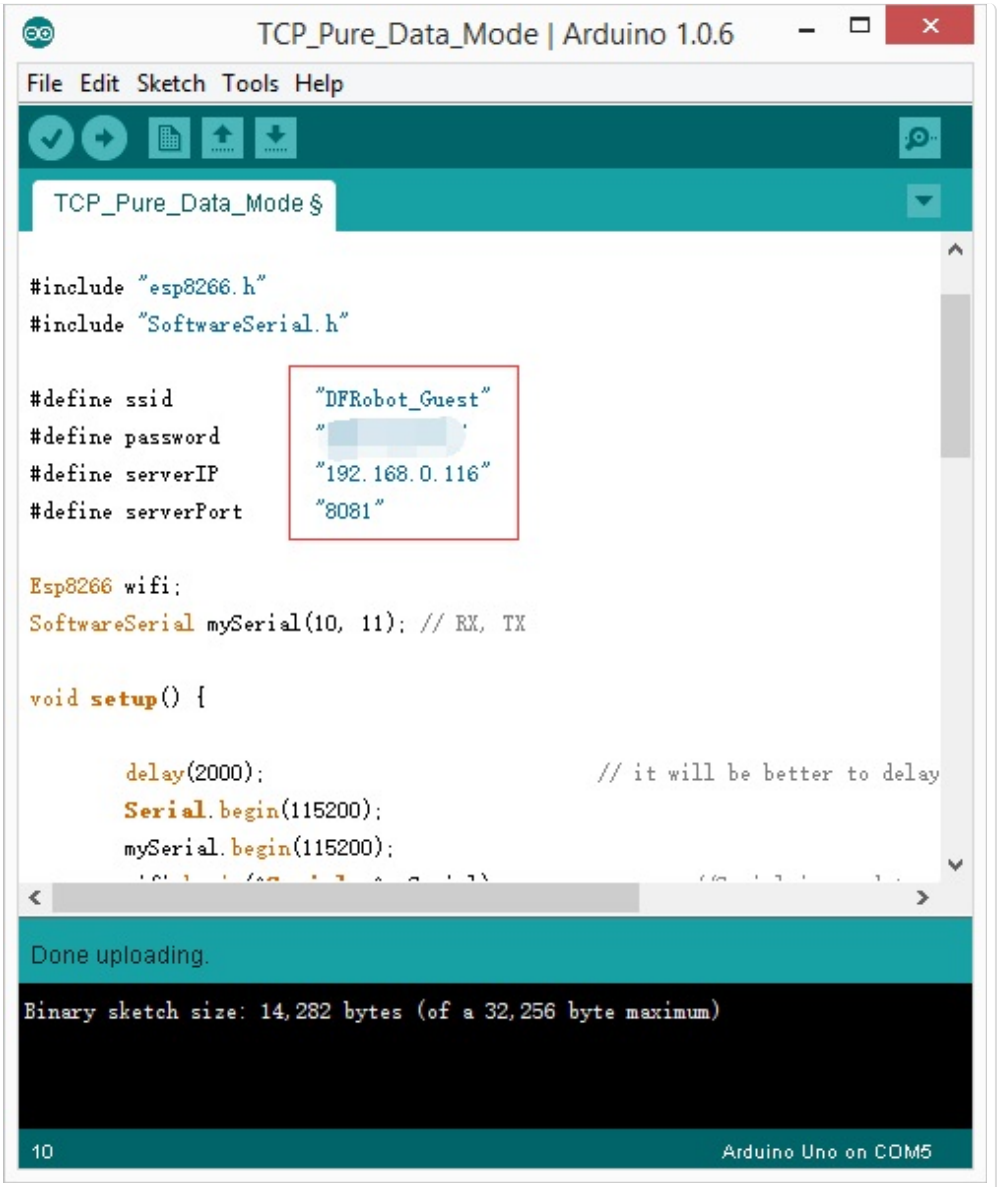


(/wiki/index.php/File:TEL0092TCP_Client_multi_recv.png)

Note:ESP8266 can connect to 5 servers simultaneously.

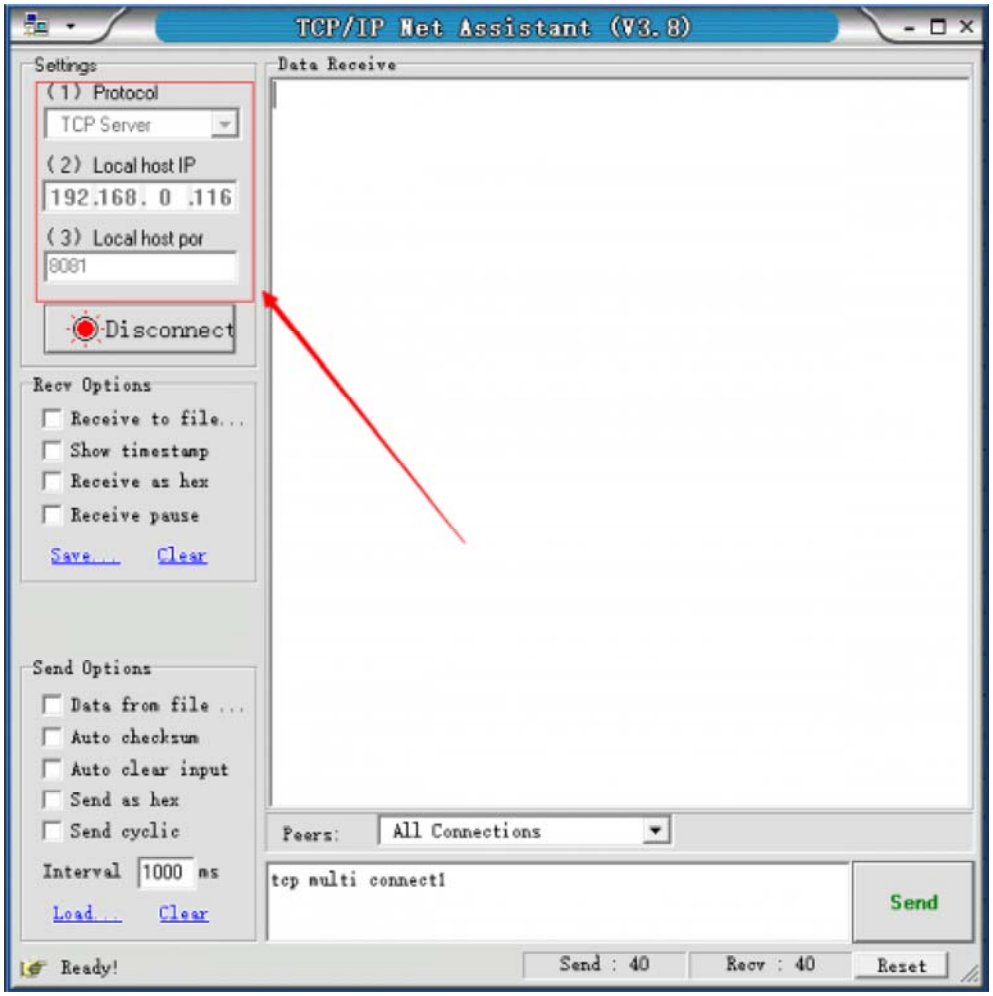
4 TCP_Pure_Data_Mode

1 Pull the switch"RUN/Prog" to "Prog";
2 Open sample sketch **"TCP_Pure_Data_Mode"**, write your **wifi's ssid, password , Server Ip address, port**;



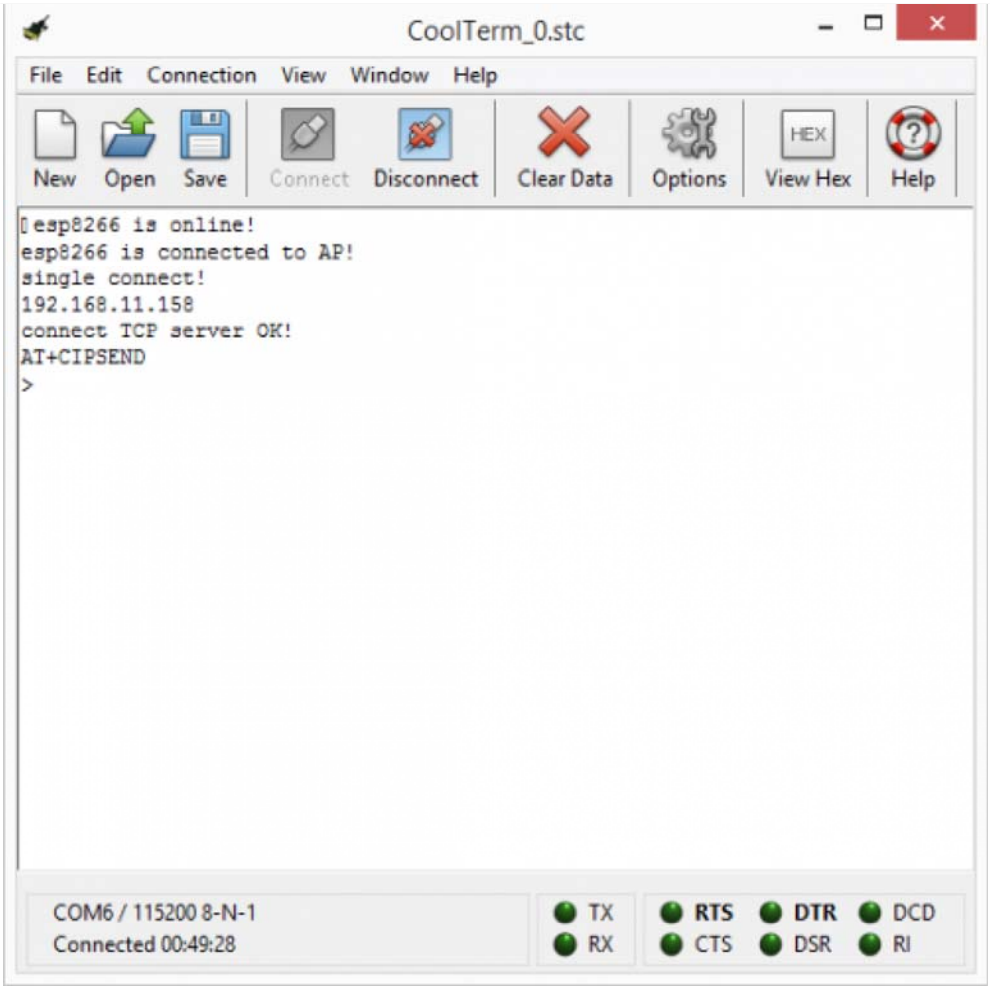
(/wiki/index.php/File:TEL0092TCP_Pure_Data_Modemodifioh.jpg)

3 Open the software"TCP/IP Net Assistant V3.8", configure as follow, and click **Connect** to monitor the PC doing as Server;



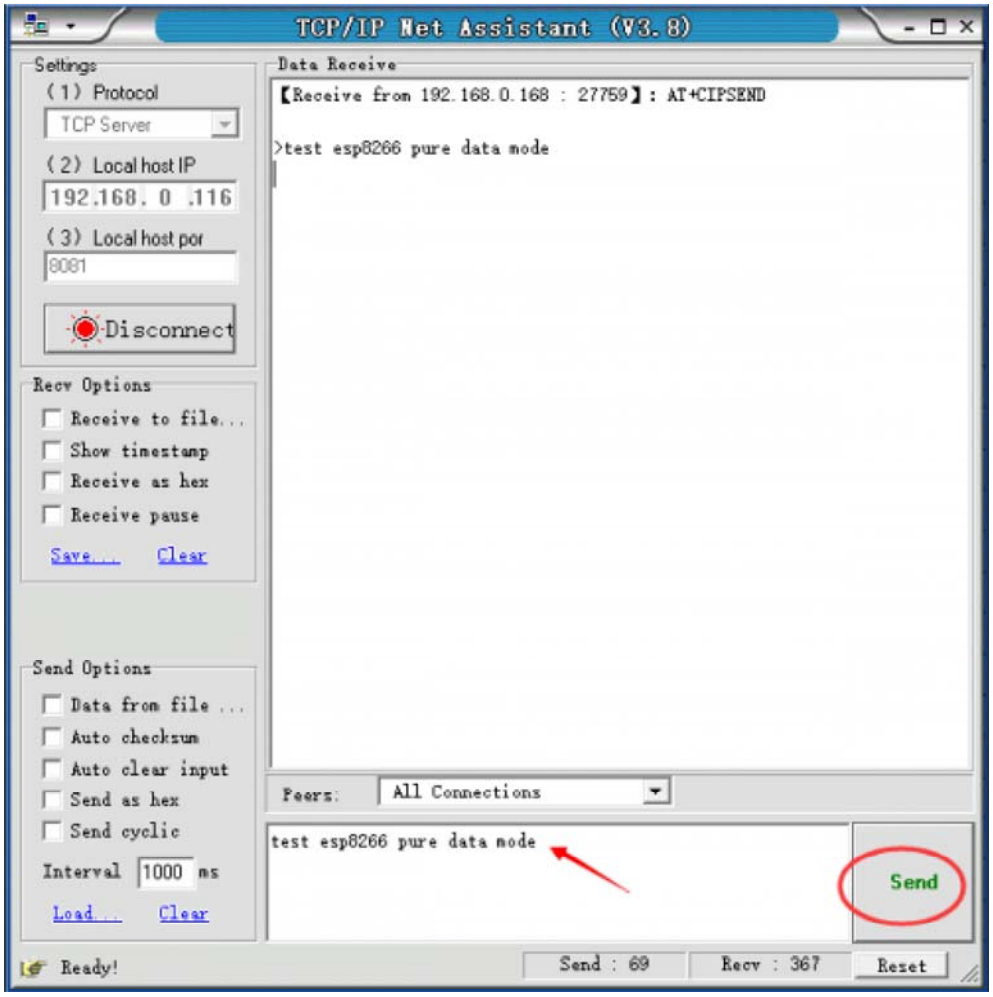
(/wiki/index.php/File:TEL0092TCP_Pure_Data_ModeTCPCFI.png)

4 Upload the modified sketch, and then pull the switch"RUN/Prog" to "RUN";
5 Open **"CoolTerm"** and monitor the if the AP connection was good;

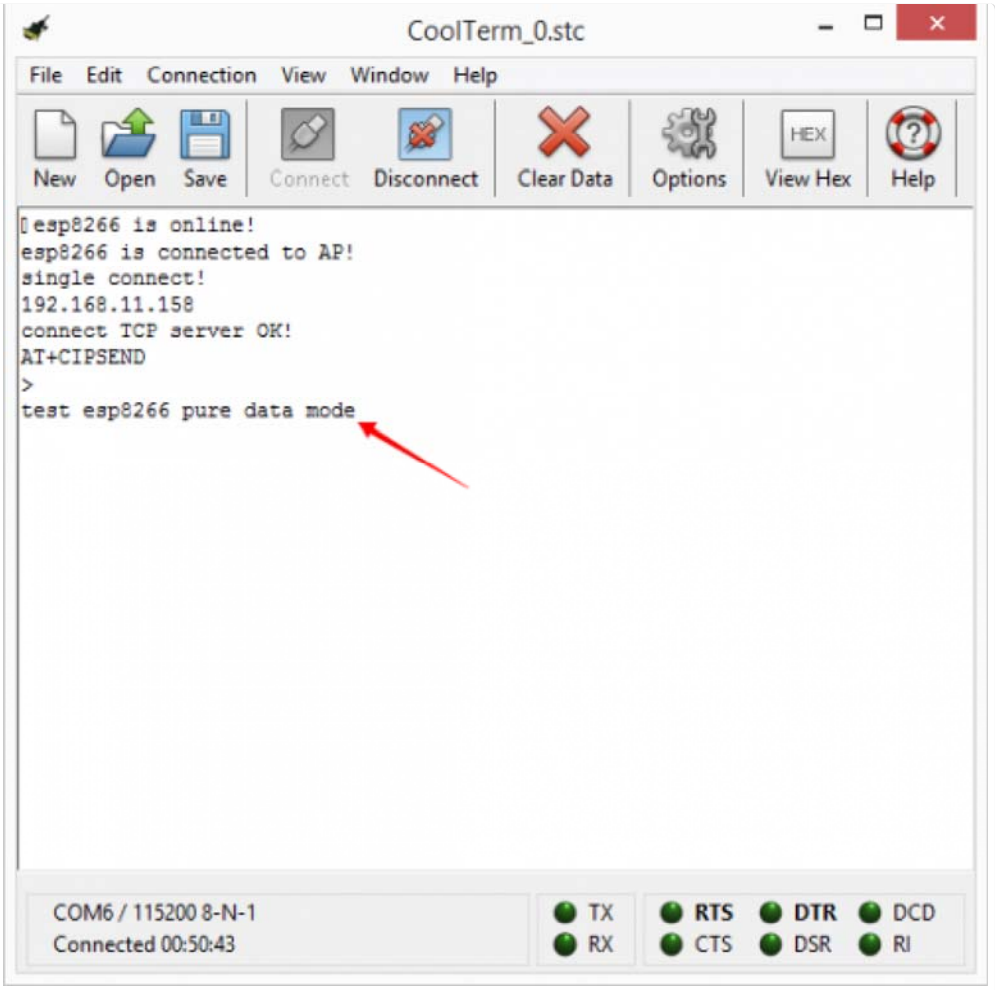


(/wiki/index.php/File:TEL0092TCP_Pure_Data_ModeCOOLIF.png)

6 Send a message from **"TCP/IP Net Assistant V3.8"**, you could see New message appear on **"CoolTerm"**.



(/wiki/index.php/File:TEL0092TCP_Pure_Data_ModeSEN.png)



(/wiki/index.php/File:TEL0092TCP_Pure_Data_ModeREC.png)

Note: When the symbol">" appeared, it means ESP8266 entered into **Transparent Mode** which transfer data faster than normal mode.

5 Server mode

```
Note:Since now, the STA mode which support the Server mode is not stable
, we are working on that you could refer to the ESP8266 manual book to t
ry.
```

* Problem Shooting

If the monitor print"**Connect failed!**" Please try/check these steps:

- Unplug the USB cable from Arduino to power off, and plug it again to restart module.
- The wifi you are using is good, and the code **ssid and password** is correct.
- Whether the button on expansion shield was push to the left side"RUN" but not"PROG".
- Whether the botton on the ESP8266 is also on the left side"BOOT".
- It's might for the wrong wire.
- The wifiBee socket on expansion shield has a loose connection to the wifi Bee.

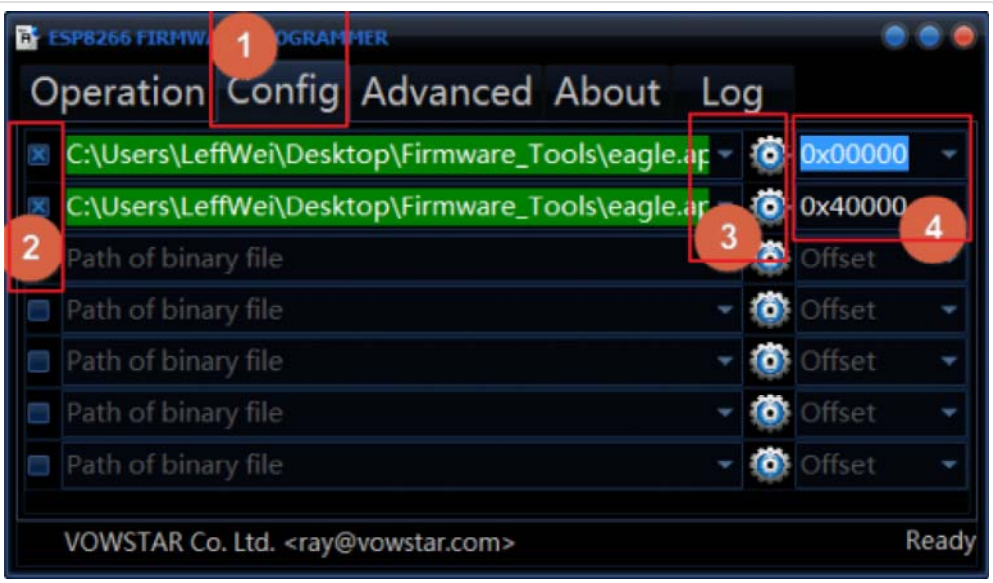
If everything is ok, but still failed connection. You may have a try "Update Firmware"

Update Firmware

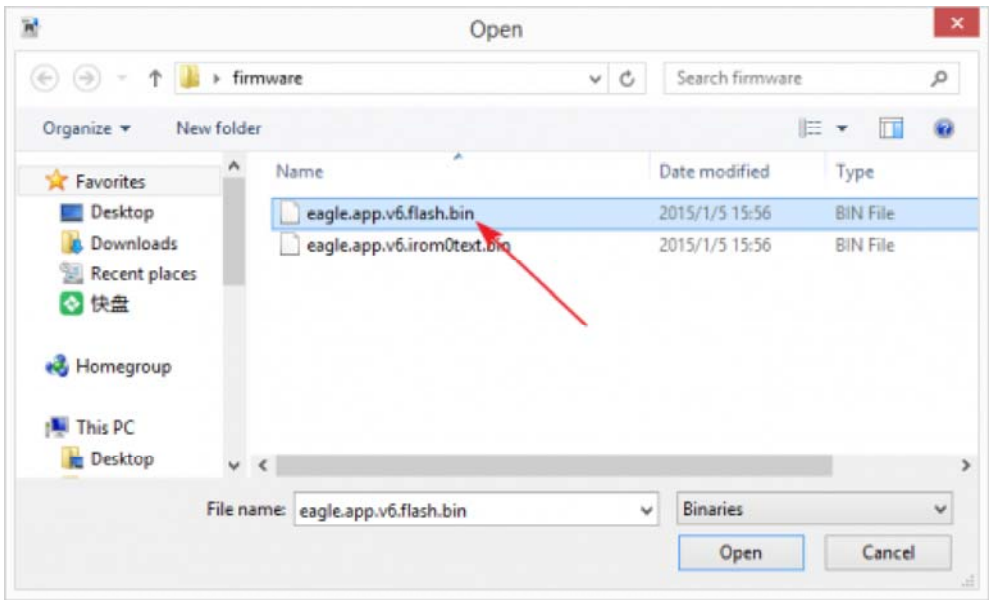
Please download the Firmware and Tools (https://github.com/Arduinolibrary/DFRobot_Wifi_Bee_ESP8266/blob/master/ESP8266Flasher_en.zip?raw=true) first. then Like in **AT mode**, but pull the swith of ESP8266 to "UART" side.Don't forget to pull it back to "BOOT" after updating firmware.

1 Open software **"ESP_Flasher"**;

2 Choose Firmware;

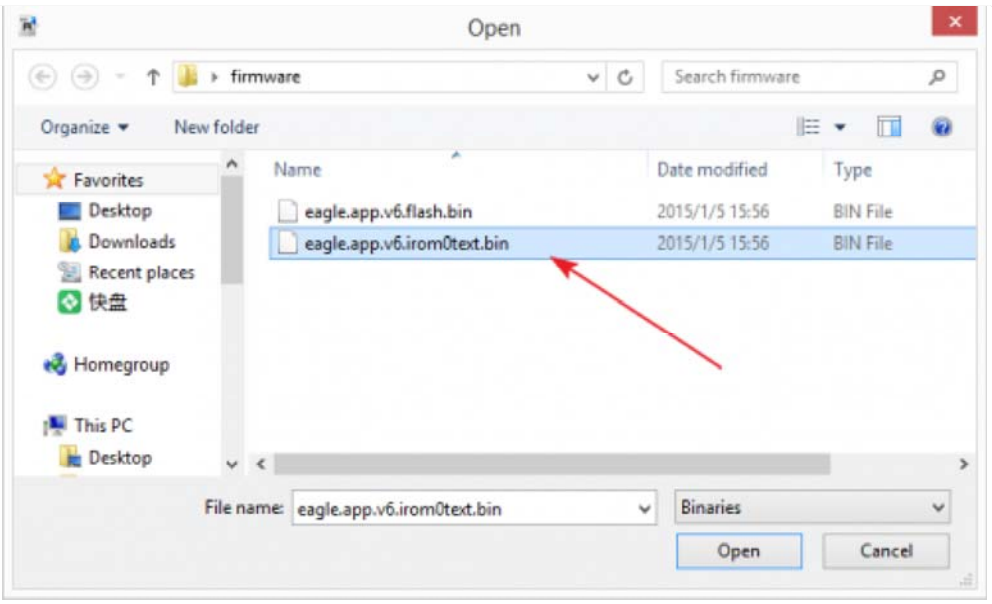


(/wiki/index.php/File:TEL0092Firmware2en.png)



(/wiki/index.php/File:TEL0092Firmware3.png)

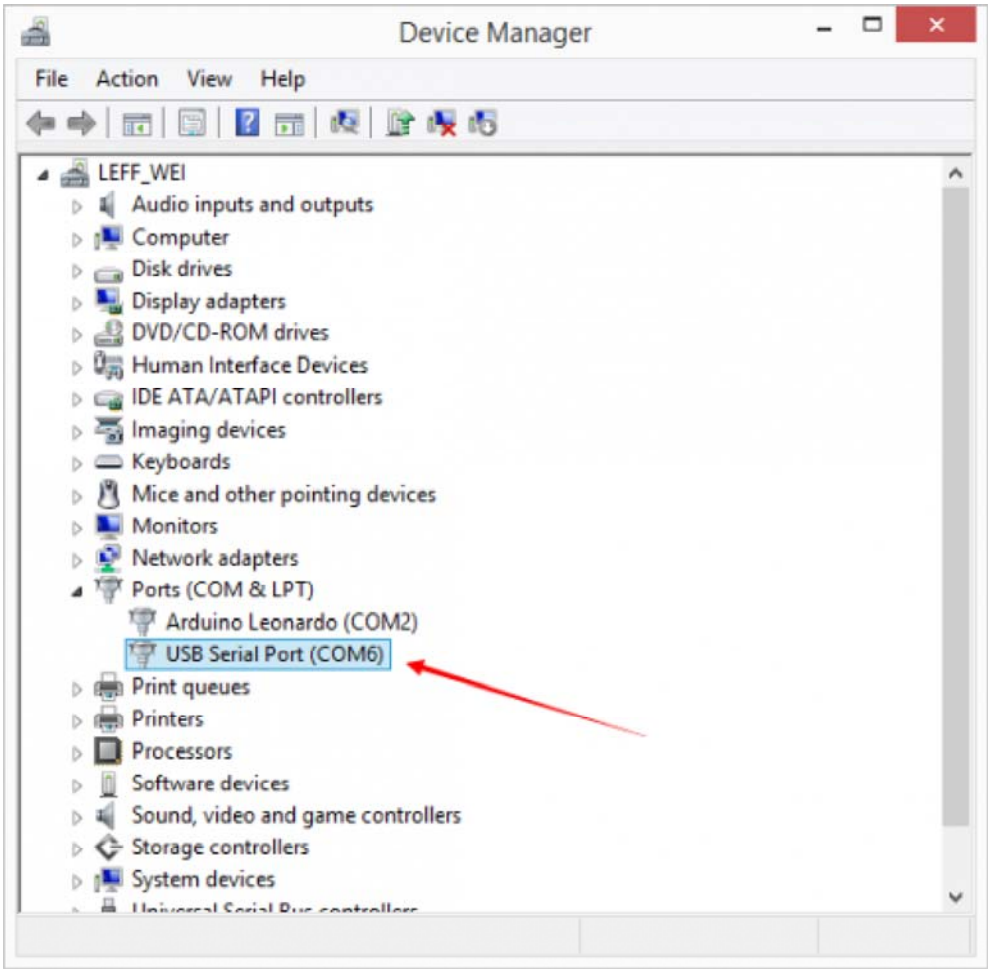
Address at 0x00000



(/wiki/index.php/File:TEL0092Firmware4.png)

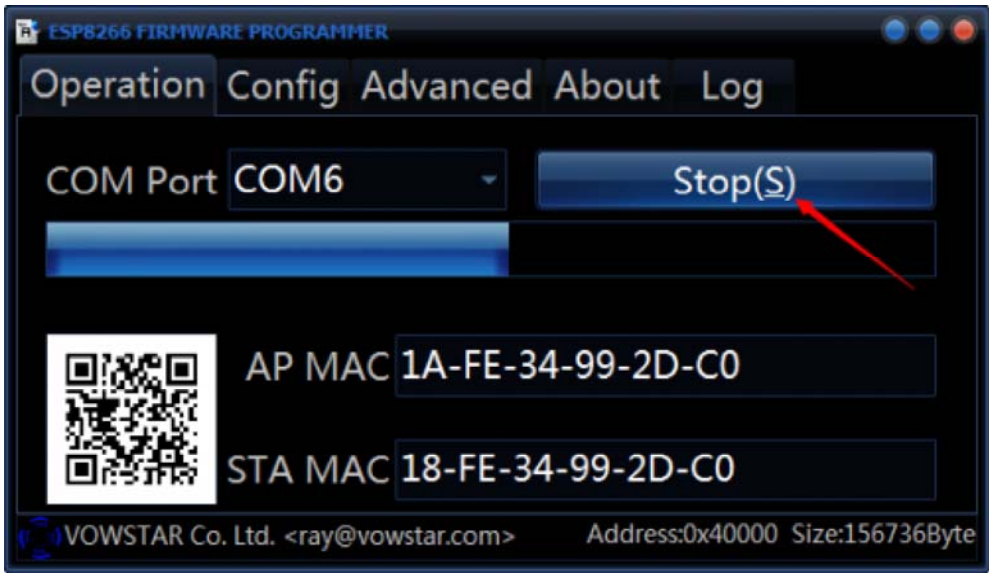
Address at 0x40000

3 Choose Serial Port of ESP8266, click **Flash** to burn firmware ;



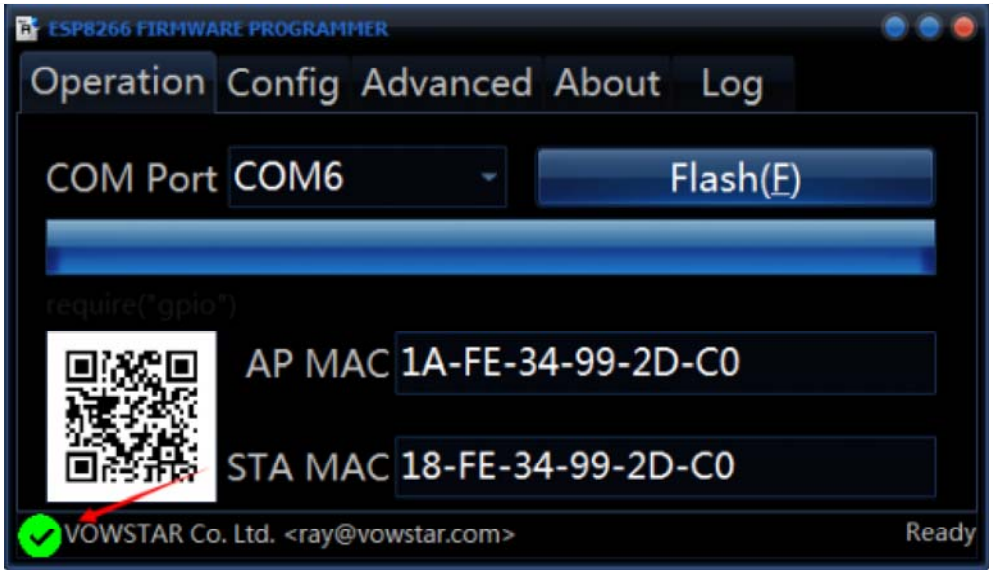
(/wiki/index.php/File:TEL0092Firmware5en_check.png)

(/wiki/index.php/File:TEL0092Firmware5en.png)



(/wiki/index.php/File:TEL0092Firmware6en.png)

4 Wait for done.



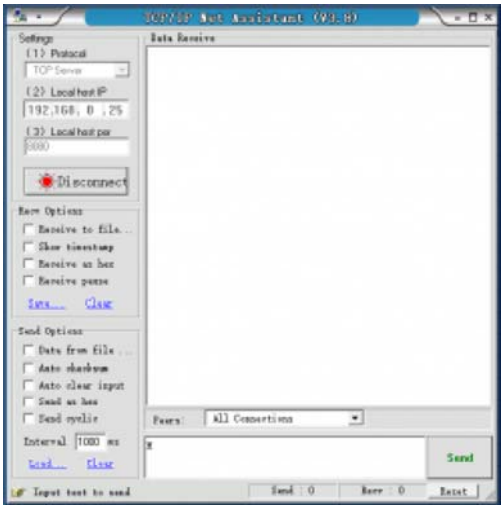
(/wiki/index.php/File:TEL0092Firmware7en.png)

Application

This is a little application which is built on the **4.1.2 2 TCP_Client_Single** trail, if you have finished that part, you could upload the sketch below, and send commands "H","L" to open up or turn off it on your computer. And since that almost every Arduino card has a LED built on board, connected with D13, so in the sketch, I will use it as the target LED.

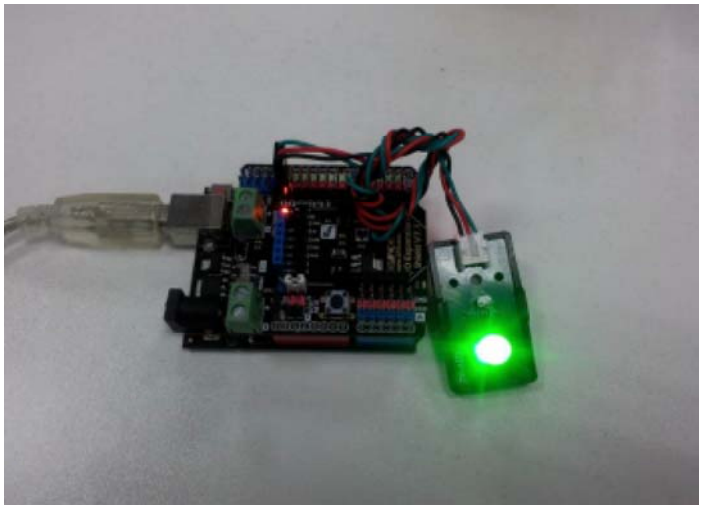
```
// this example use esp8266 to connect to an Access Point and connect to SINGLE TCP Server which is at the same subnet
// such as the esp8266 is is 192.168.1.3, and the server ip is 192.168.1.1 ,then esp8266 can connect to the server

//Then connect a LED on Digital pin13, and open the software on PC_TCP server, send command to control the LED state:
//send "H" to turn ON LED; send "L" to turn OFF LED

#include "esp8266.h"
#include "SoftwareSerial.h"

#define ssid            "test"          // you need to change it
#define password        "12345678"

#define serverIP        "192.168.1.1"
#define serverPort      "8081"
int ledPin = 13;
String incomingData = "";

Esp8266 wifi;
SoftwareSerial mySerial(10, 11); // RX, TX

void setup() {
  pinMode(ledPin, OUTPUT);
  delay(2000);                          // it will be better to delay 2s to wait esp8266 module OK
  Serial.begin(115200);
  mySerial.begin(115200);
  wifi.begin(&Serial, &mySerial);               //Serial is used to communicate with esp8266 module, mySerial is used to debug
  if (wifi.connectAP(ssid, password)) {
    wifi.debugPrintln("connect ap sucessful !");
  } else {
    while (true);
  }
  wifi.setSingleConnect();
  if (wifi.connectTCPServer(serverIP, serverPort)) {
    wifi.debugPrintln("connect to TCP server successful !");
  }
  String ip_addr;
  ip_addr = wifi.getIP();
  wifi.debugPrintln("esp8266 ip:" + ip_addr);
}

void loop() {
  int state = wifi.getState();
  switch (state) {
```

```
44      case WIFI_NEW_MESSAGE:
45        wifi.debugPrintln("new message!");
46        incomingData = wifi.getMessage();
47        wifi.sendMessage(incomingData);           //send the message to
 TCP server what it has received
48        wifi.setState(WIFI_IDLE);
49        break;
50      case WIFI_CLOSED :
         //reconnet to the TCP server
51        wifi.debugPrintln("server is closed! and trying to reconnect it
!");
52        if (wifi.connectTCPServer(serverIP, serverPort)) {
53          wifi.debugPrintln("reconnect OK!");
54          wifi.setState(WIFI_IDLE);
55        }
56        else {
57          wifi.debugPrintln("reconnect fail");
58          wifi.setState(WIFI_CLOSED);
59        }
60        break;
61      case WIFI_IDLE :
62        int sta = wifi.checkMessage();
63        wifi.setState(sta);
64        break;
65    }
66    if (incomingData == "H")  {
67      digitalWrite(13, HIGH);
68      incomingData = "";
69    }
70    else if (incomingData == "L")  {
71      digitalWrite(13, LOW);
72      incomingData = "";
73    }
74  }
```



(/wiki/index.php/File:TEL0092_control_LED13.png)

Send command H



(/wiki/index.php/File:TEL0092_control_LED13PIC.png)

The LED turn ON

AT command

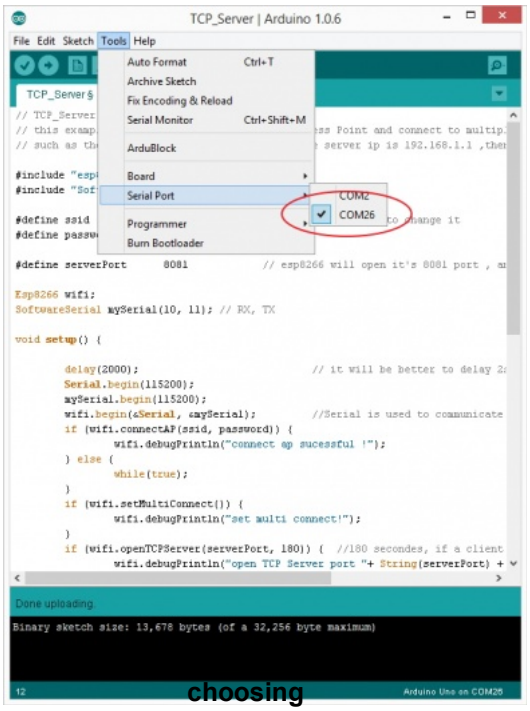
How to enter AT mode

You can setup and control the module completely with AT command through Serial.

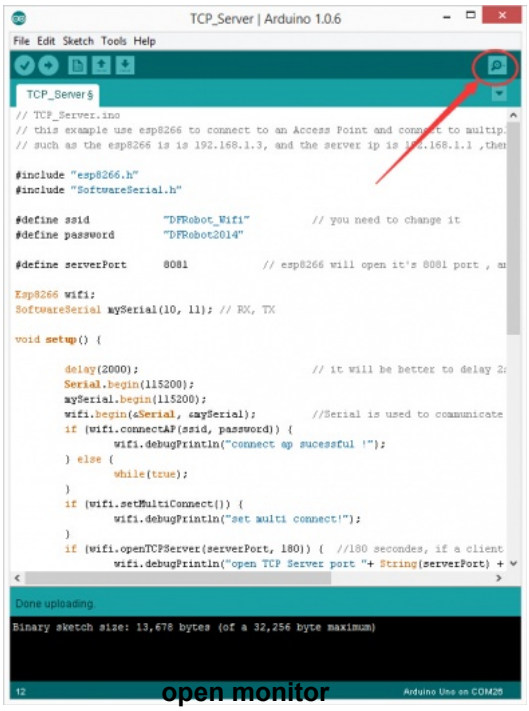
1 Insert ESP8266 onto the USB-Serial adapter (http://www.dfrobot.com/index.php?route=product/product&product_id=72&search=FTDI&description=true#.Vp3NdVlPrzY)

(/wiki/index.php/File:TEL0092_atmode_choo1.jp

2 Open the Monitor in Arduino IDE. Choosing "Both NL & CR" "115200"

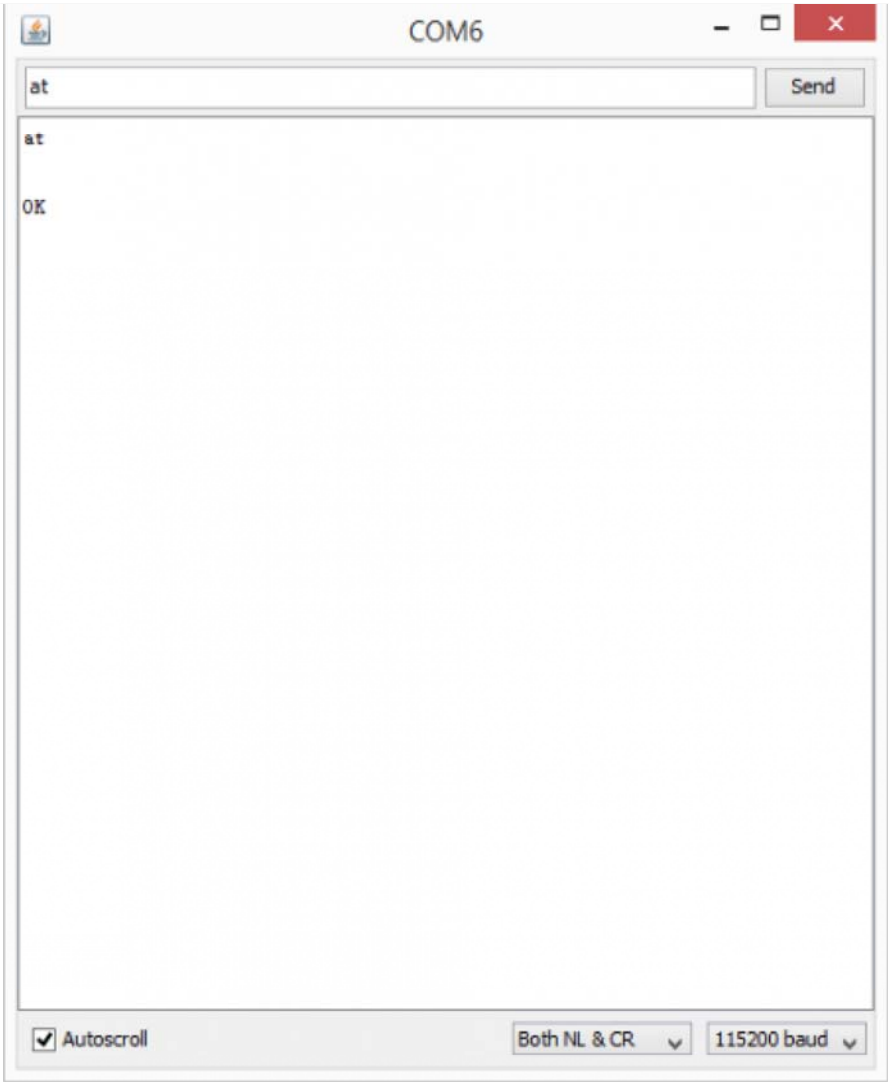


choosing (/wiki/index.php/File:ESP8266_atMode_1.jpg)



open monitor (/wiki/index.php/File:ESP8266_atMode_2.jpg)

3 Send "AT" to enter into the AT mode once reveived OK.



(/wiki/index.php/File:TEL0092_ctmode_chodum.png)