



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of "Quality Parts,Customers Priority,Honest Operation,and Considerate Service",our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



SKU:TEL0092 WiFi Bee-ESP8266 Wireless module

Contents

- 1 Introduction
- 2 Specifications
- 3 Pin out
- 4 Tutorials
 - 4.1 How to Use ?
 - 4.1.1 1 Connect AP
 - 4.1.2 2



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

WiFi Bee-ESP8266 SKU:TEL0092

- TCP_Client_Single
 - 4.1.3 3 TCP_Client_multi
 - 4.1.4 4 TCP_Pure_Data_Mode
 - 4.1.5 5 Server mode
 - 4.1.6 * Problem Shooting
- 4.2 Update Firmware
- 5 Application
- 6 AT command
 - 6.1 How to enter AT mode
- 7 Related link
- 8 More

Introduction

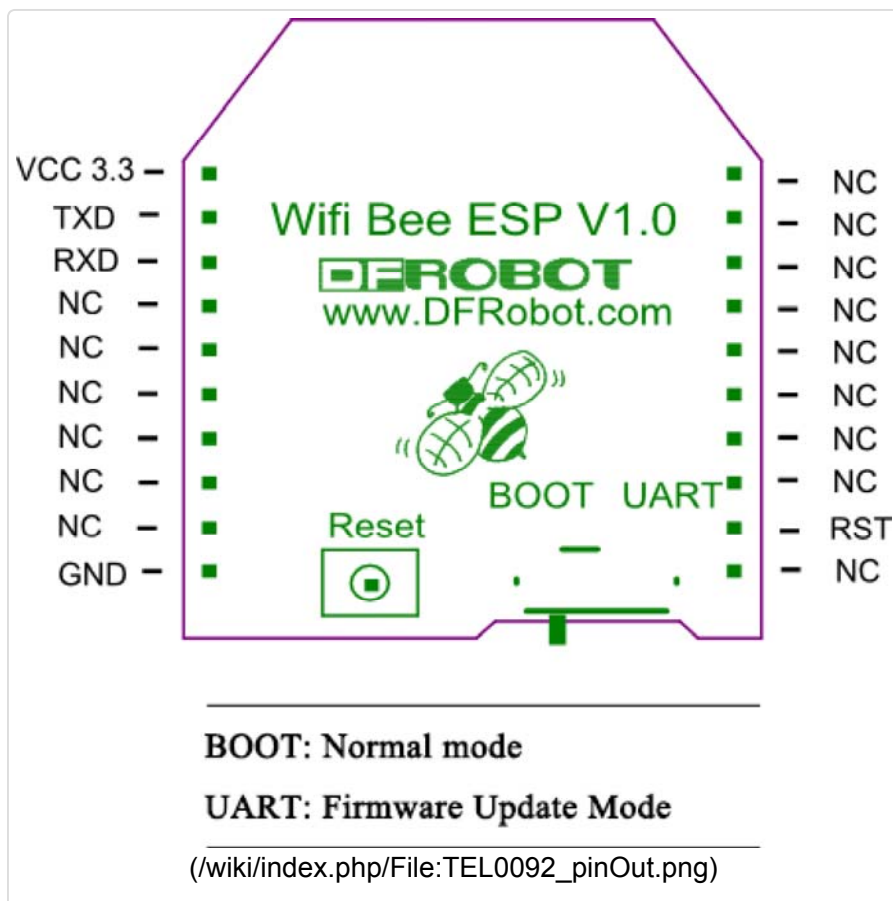
Wifi Bee-ESP8266 is a Serial-to-WIFI module using XBEE design in a compact size, compatible with XBEE expansion base, applicable to a variety of 3.3V single-chip system. It can be used for Arduino, wireless data transfer, remote control. On-board switch can be used to easily select the Startup module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



Tutorials

These stuffs are needed:

1 Software

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

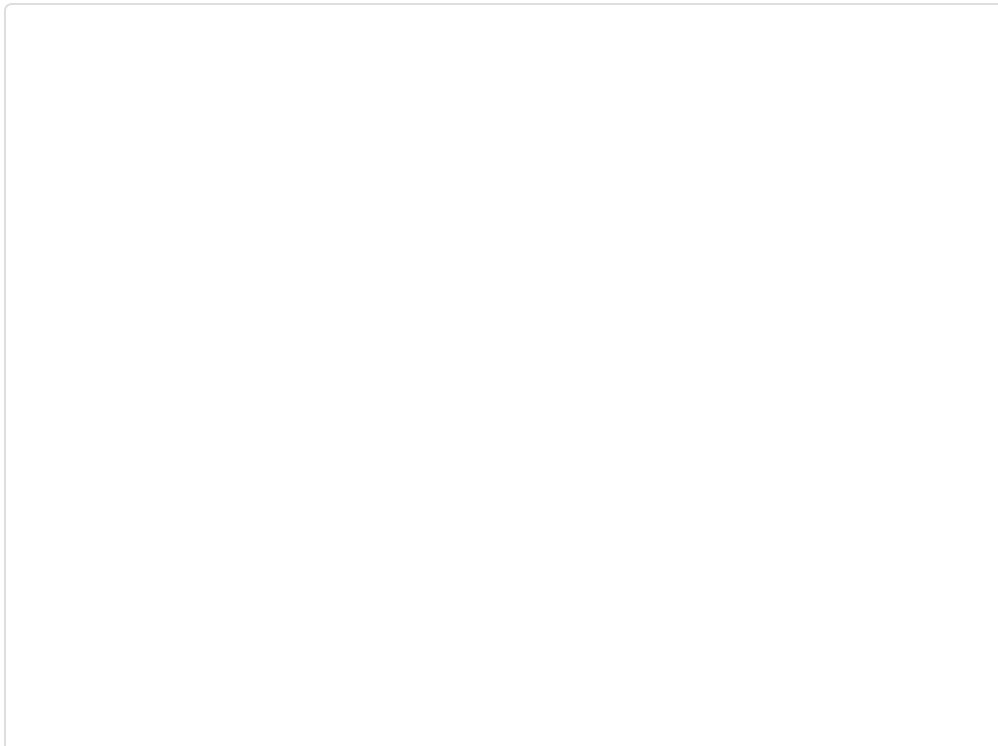
2 Hardware

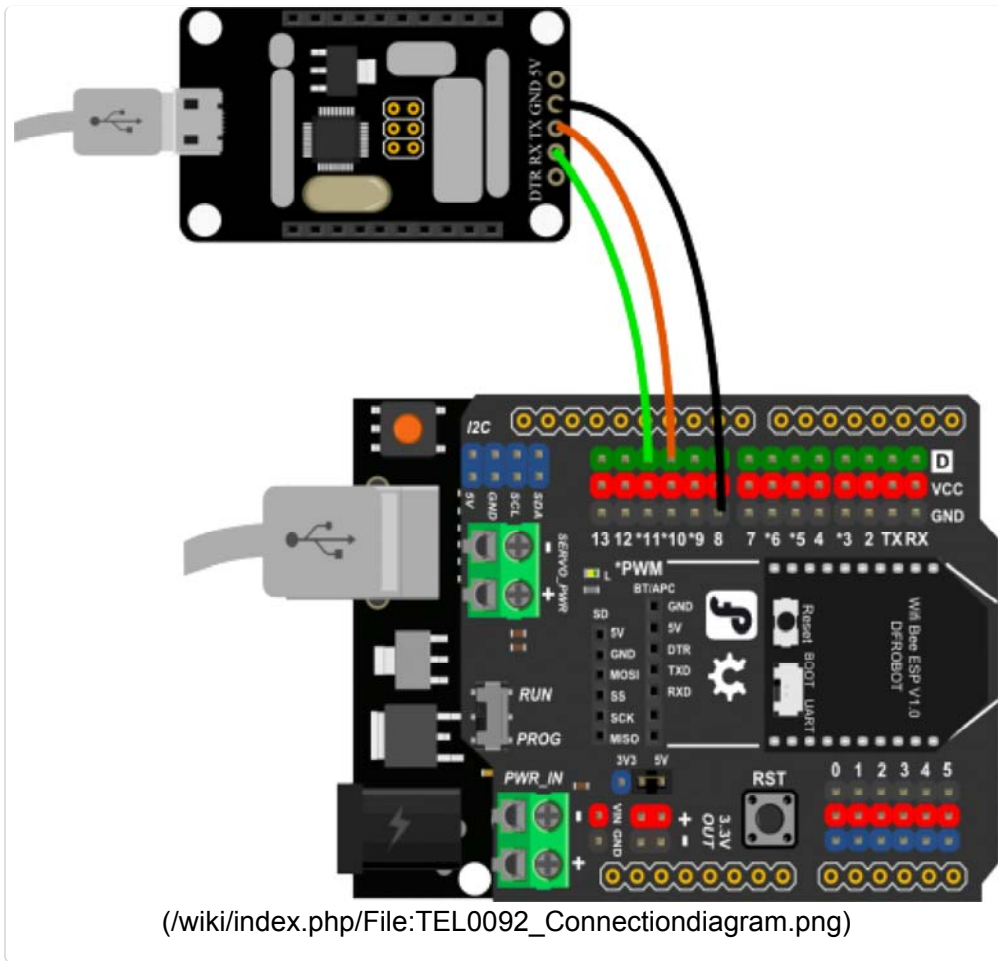
1. DFRduino UNO R3 (http://www.dfrobot.com/index.php?route=product/product&product_id=838&search=uno&description=true)
2. Xbee USB adapter (FTDI ready) (http://www.dfrobot.com/index.php?route=product/product&product_id=72&search=USB&description=true&page=2)
3. IO Expansion Shield for Arduino (V6) (http://www.dfrobot.com/index.php?route=product/product&product_id=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 ?

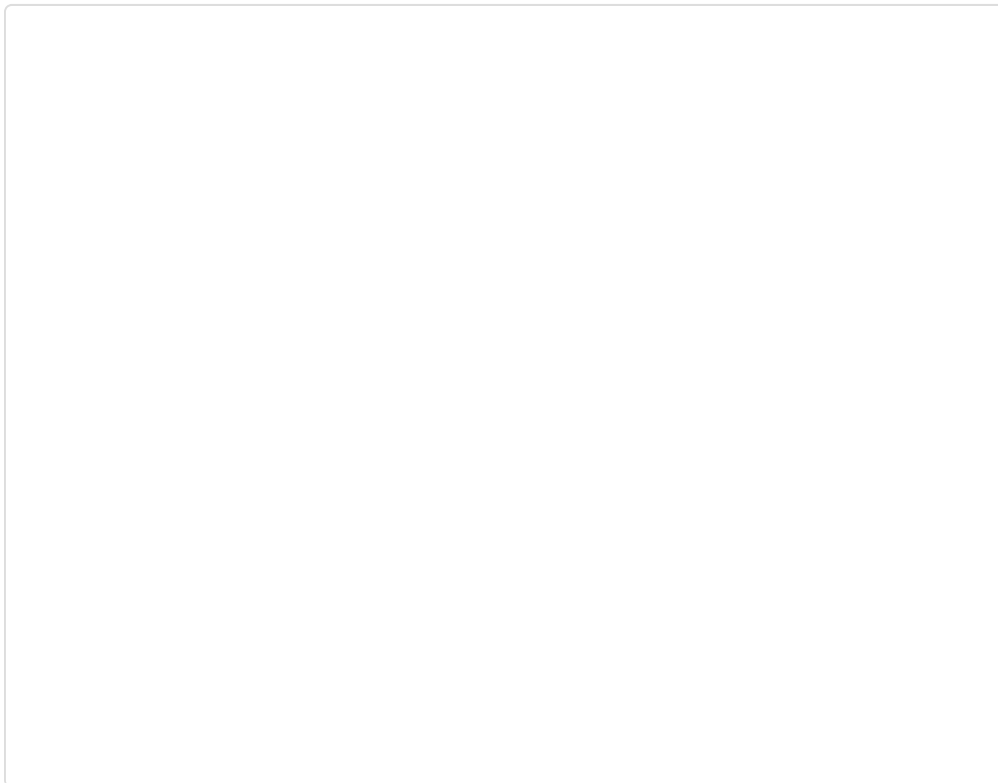
1 Connect AP

- 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;**



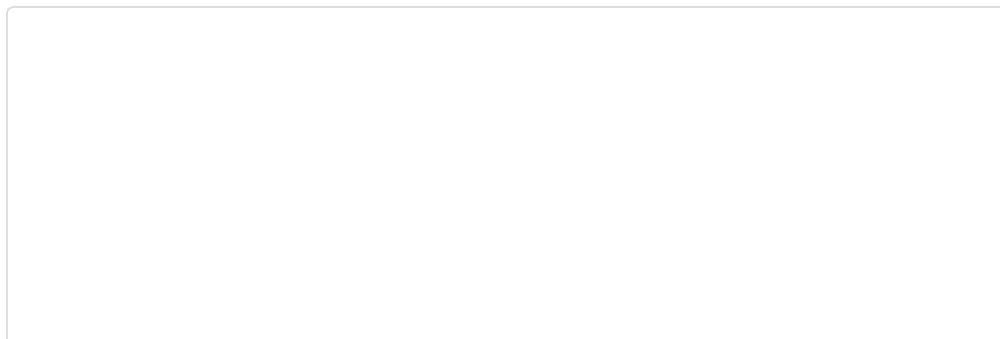


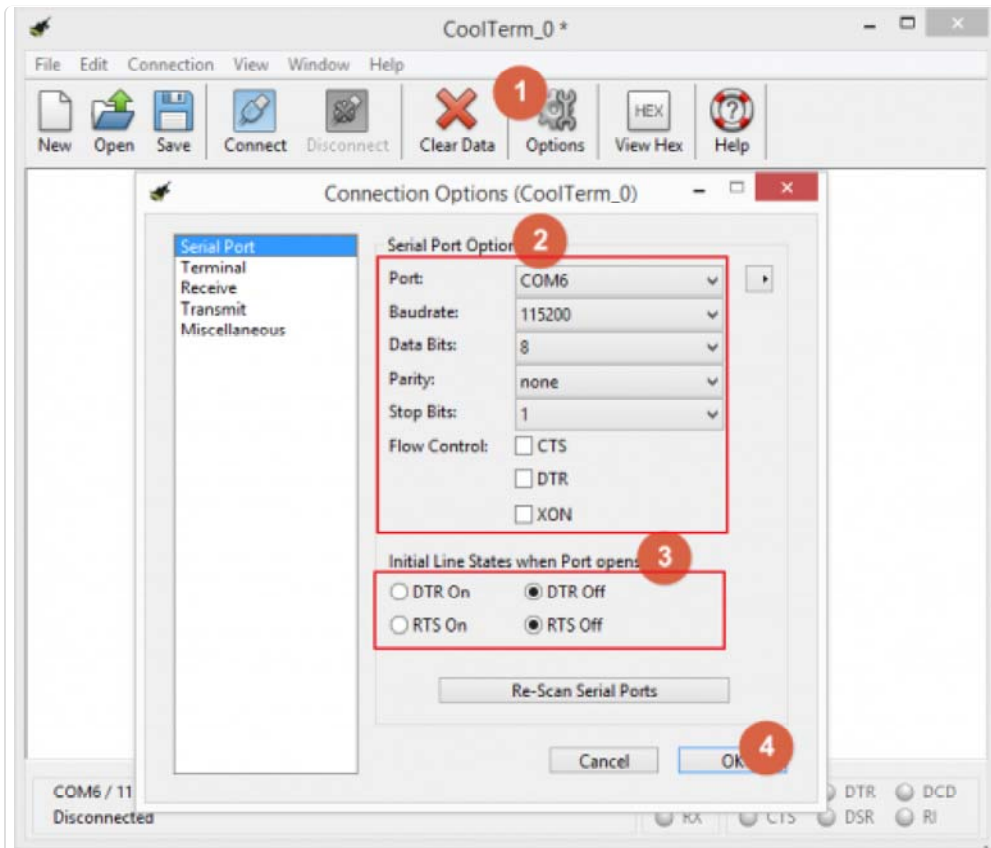
5 Open the sketch "Connect AP", and modify the wifi AP ssid and password of yours;



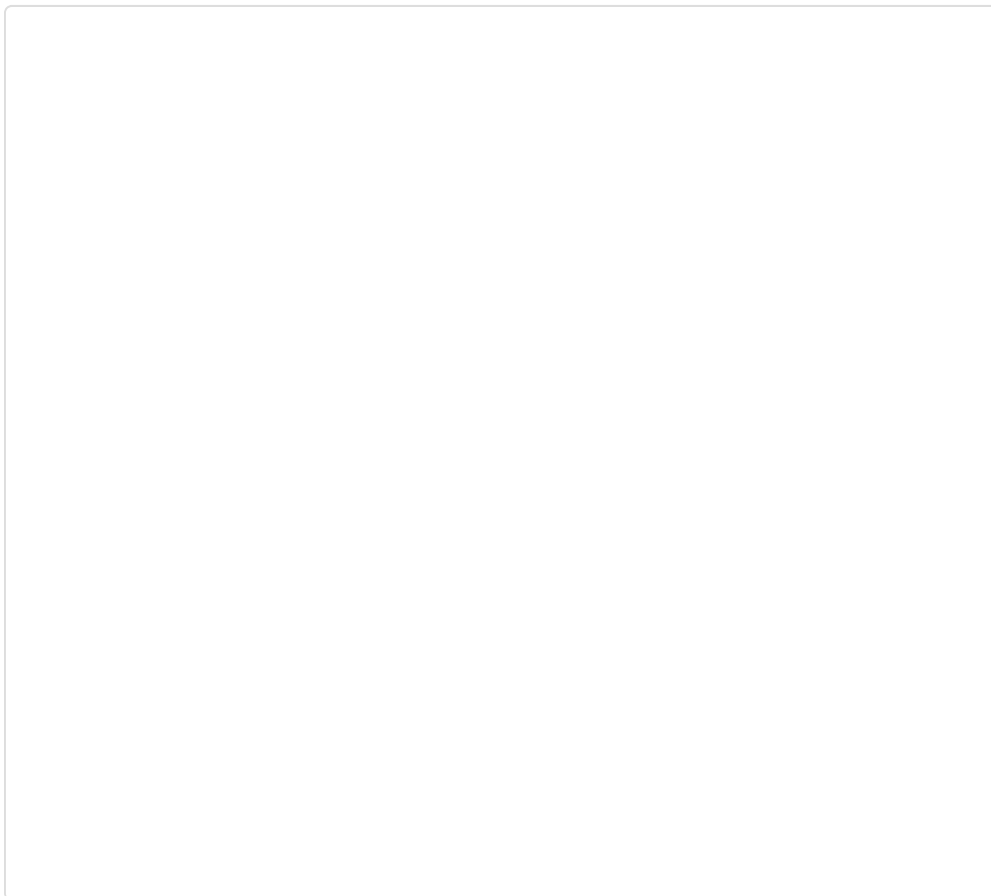


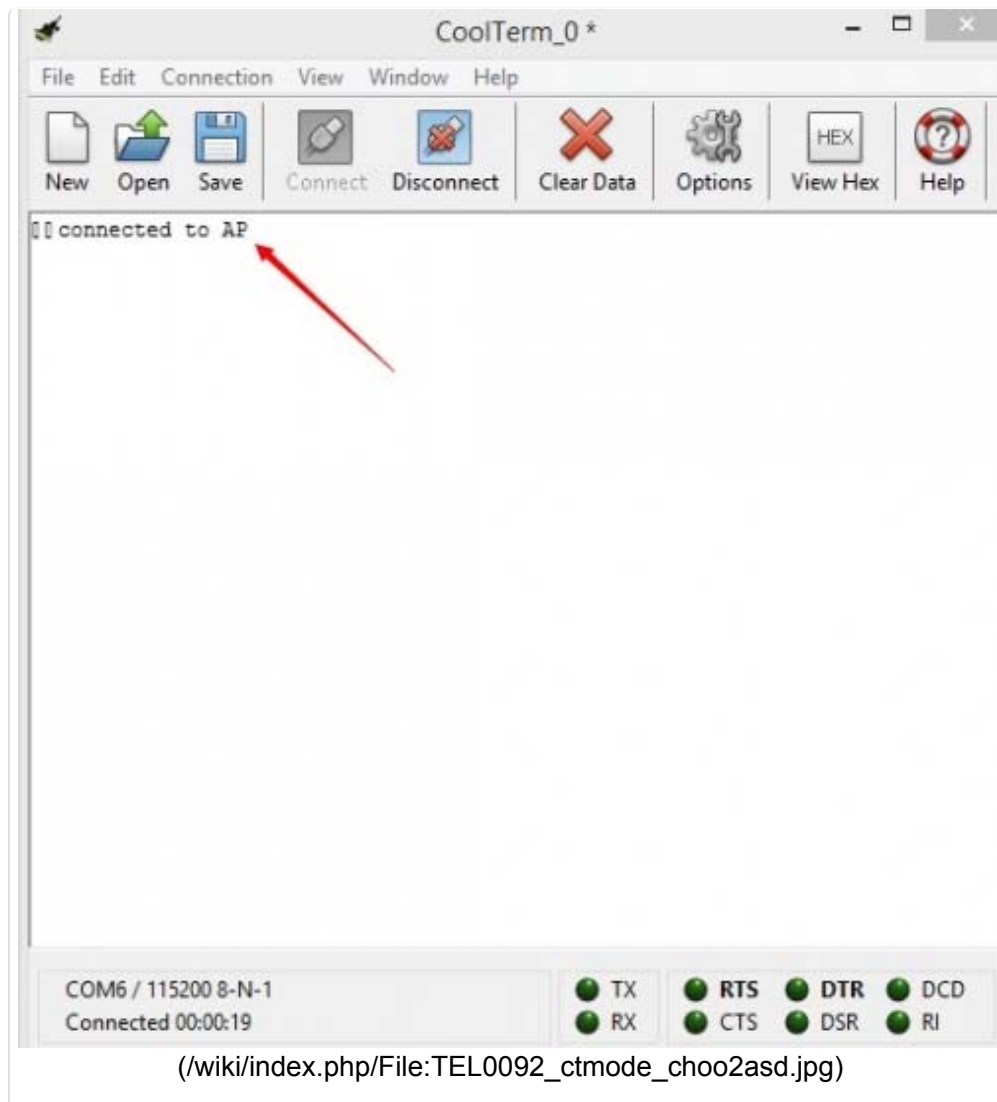
- 6 Upload the sketch;
- 7 Turn the switch of the expansion shield "RUN/Prog" to "Run" side;
- 8 Use software "CoolTerm" to monitor if the AP connection is done.(Configure as the follow picture)





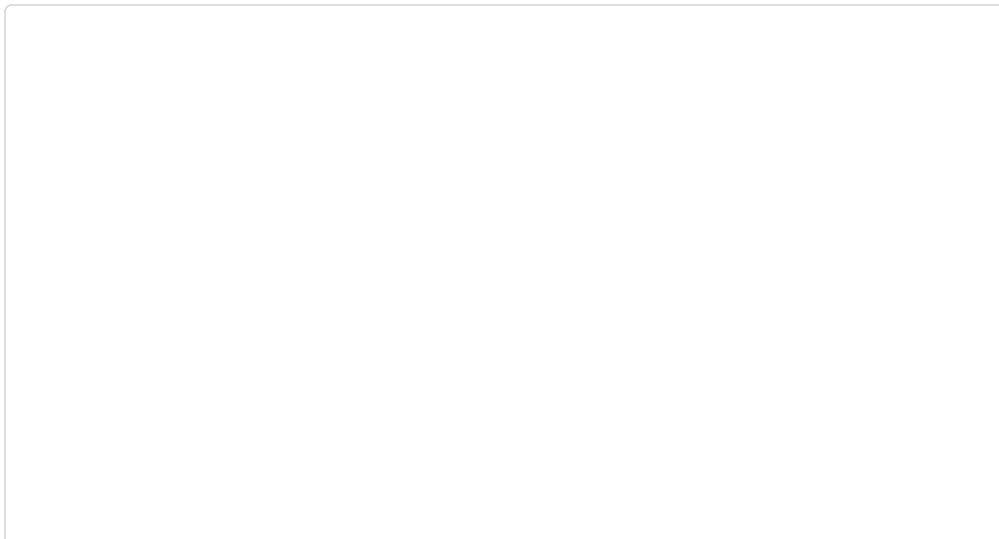
(/wiki/index.php/File:TEL0092_Connect_AP_para.png)

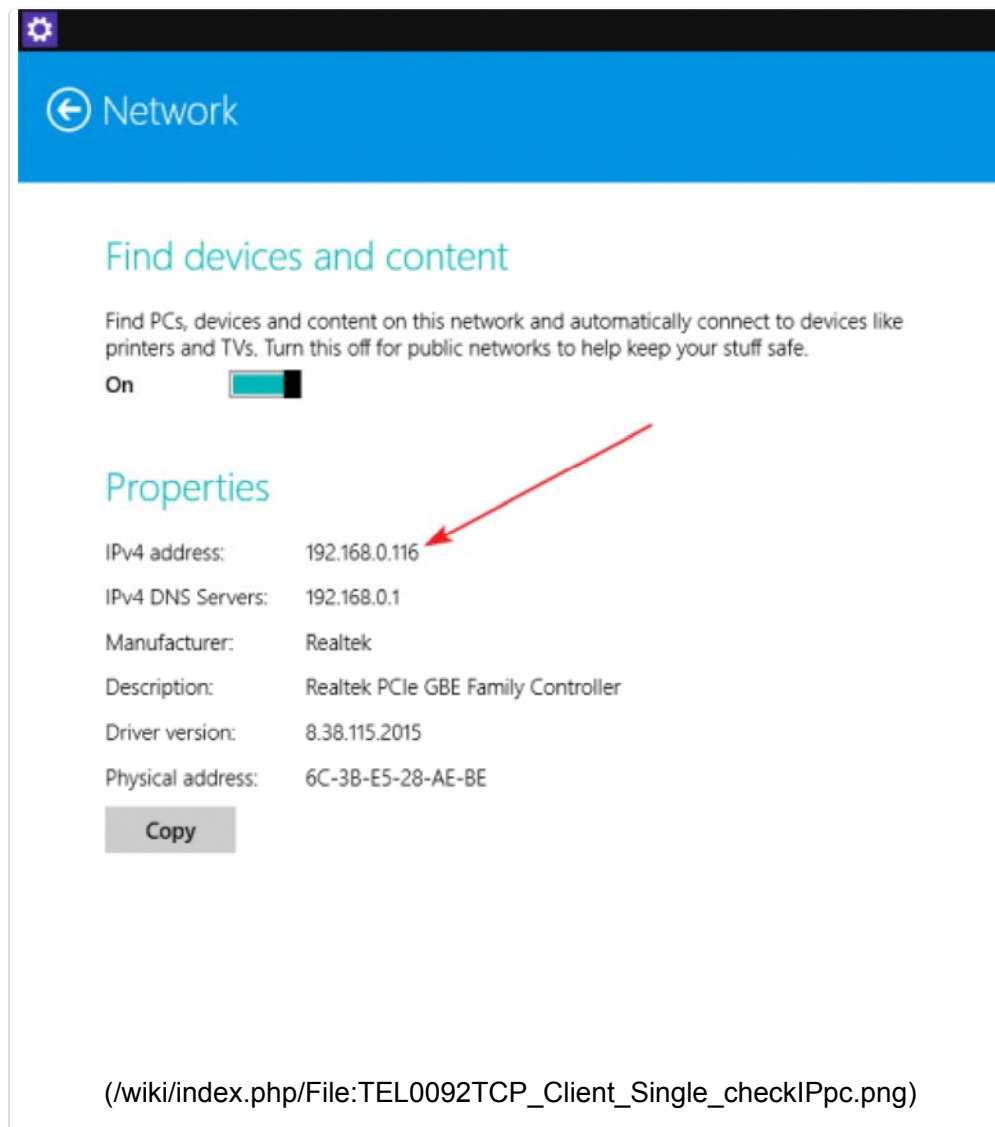




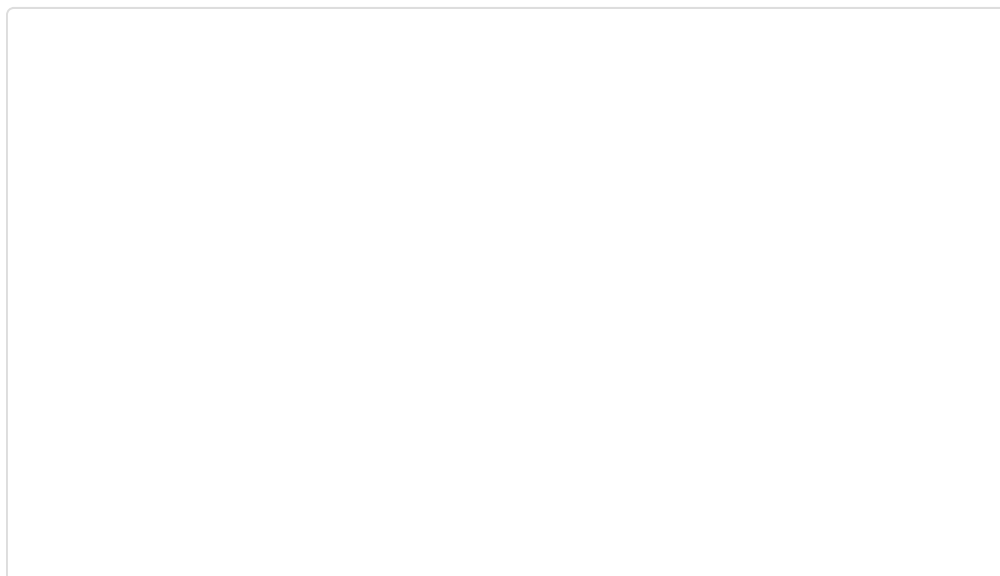
2 TCP_Client_Single

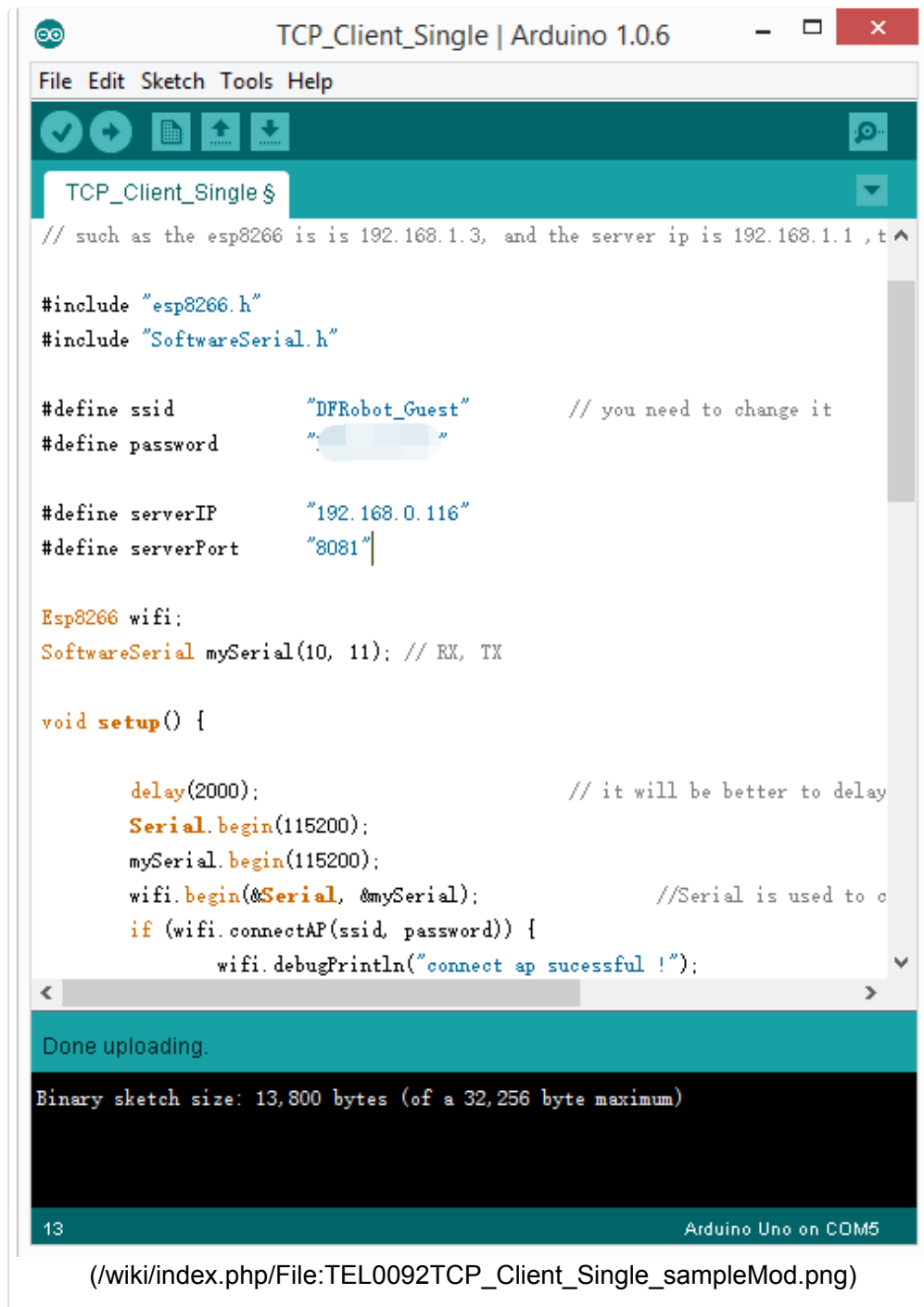
- 1 Pull the switch "RUN/Prog" to "Prog";
- 2 Check the Ip address of your computer, we will use it as the Server Ip address;





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





```
File Edit Sketch Tools Help
TCP_Client_Single $
// such as the esp8266 is is 192.168.1.3, and the server ip is 192.168.1.1 , t ^
#include "esp8266.h"
#include "SoftwareSerial.h"

#define ssid      "DFRobot_Guest"      // you need to change it
#define password  " "

#define serverIP  "192.168.0.116"
#define serverPort "8081"

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

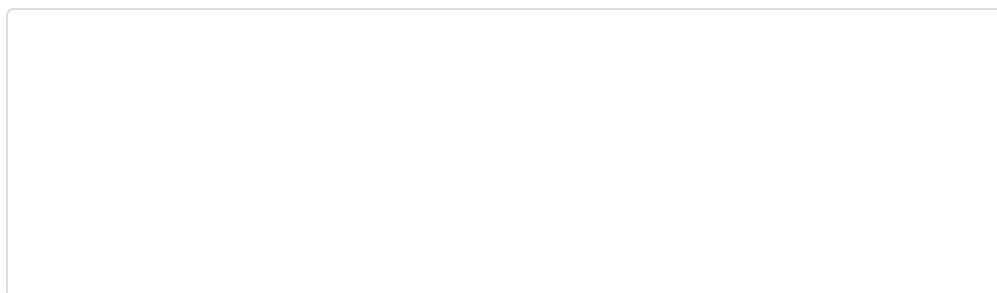
void setup() {

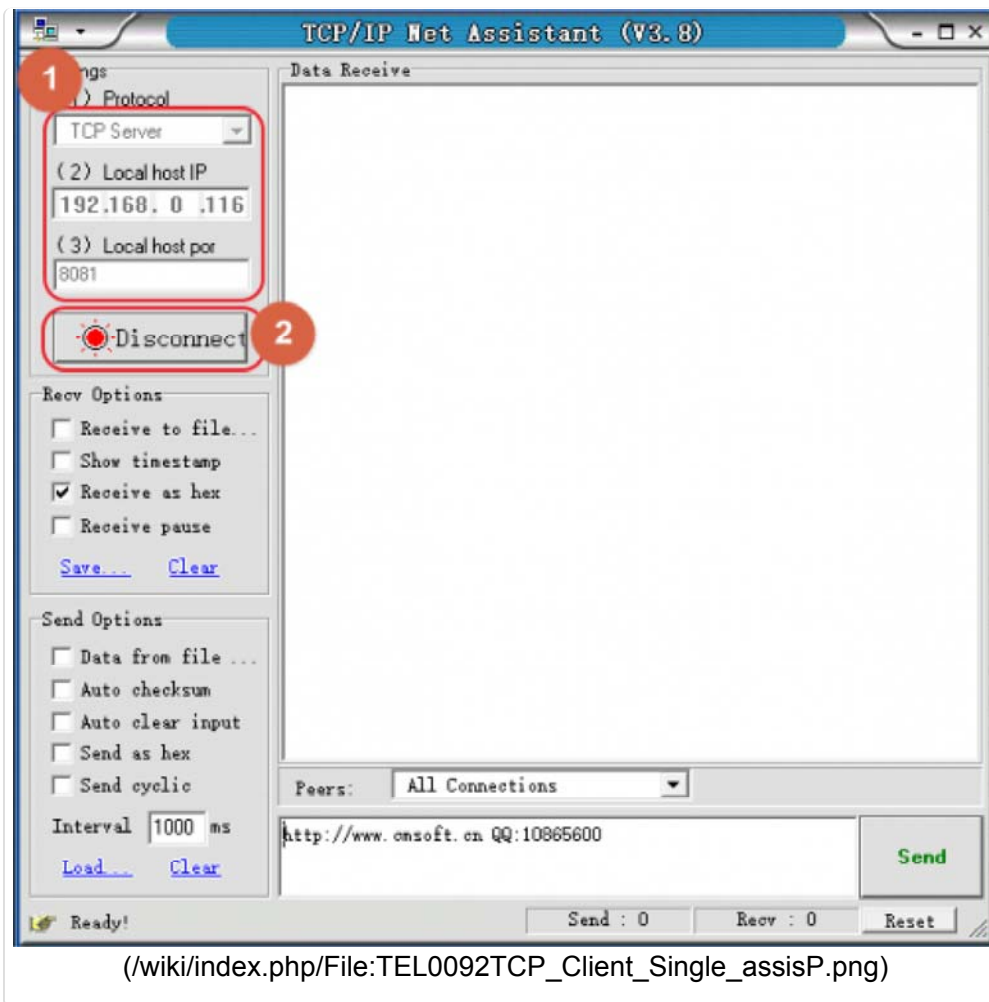
    delay(2000);                // it will be better to delay
    Serial.begin(115200);
    mySerial.begin(115200);
    wifi.begin(@Serial, @mySerial);    //Serial is used to c
    if (wifi.connectAP(ssid, password)) {
        wifi.debugPrintln("connect ap sucessful !");
    }
}

Done uploading.
Binary sketch size: 13,800 bytes (of a 32,256 byte maximum)
13 Arduino Uno on COM5
```

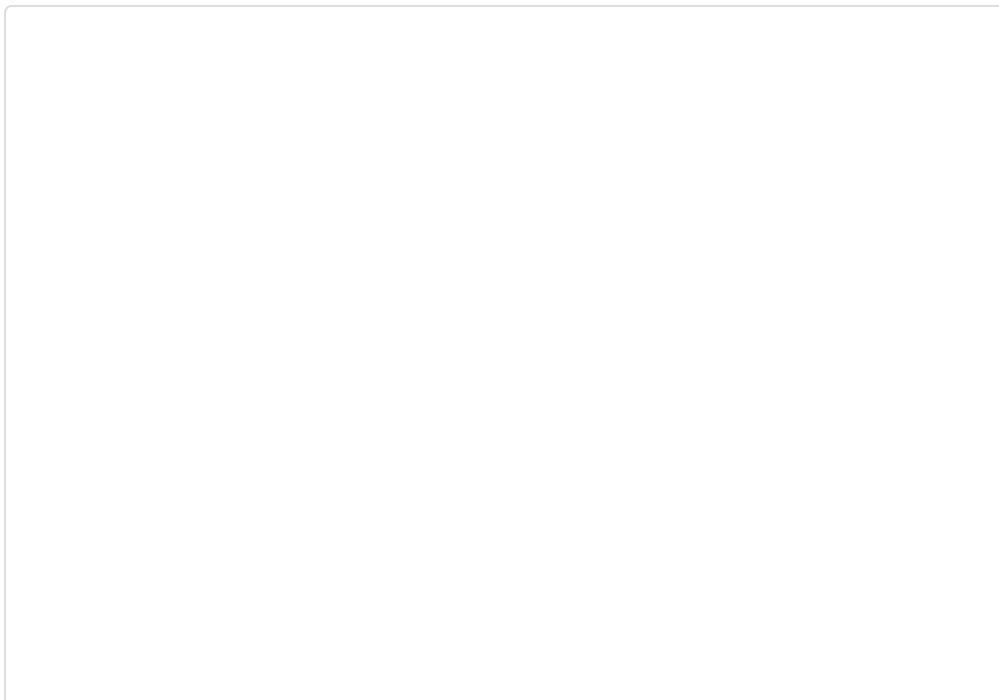
(/wiki/index.php/File:TEL0092TCP_Client_Single_sampleMod.png)

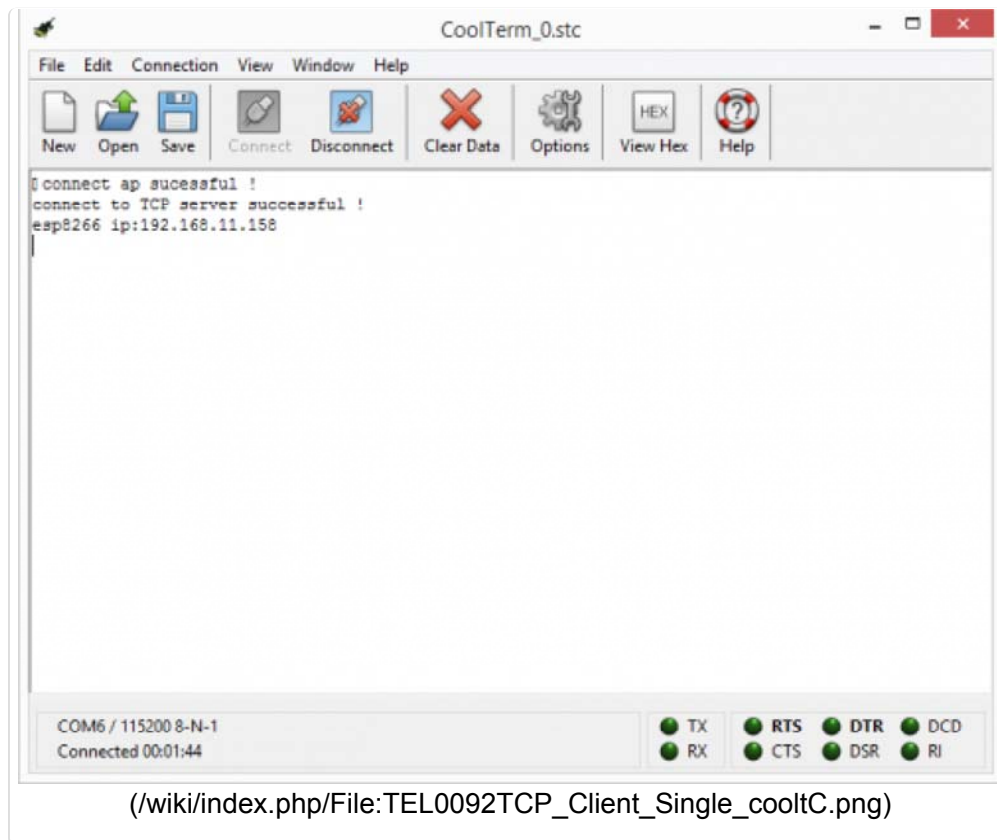
4 Open the software "TCP/IP Net Assistant V3.8", configure as follow, and click **Connect** to monitor the PC doing as Server;



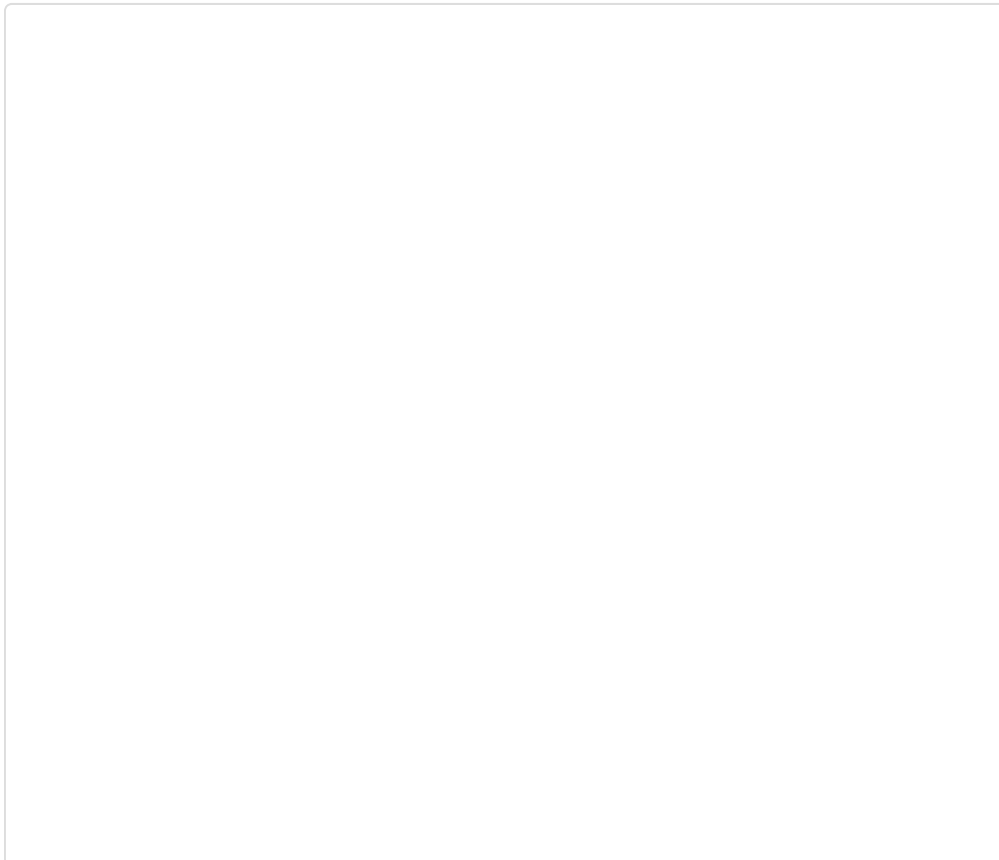


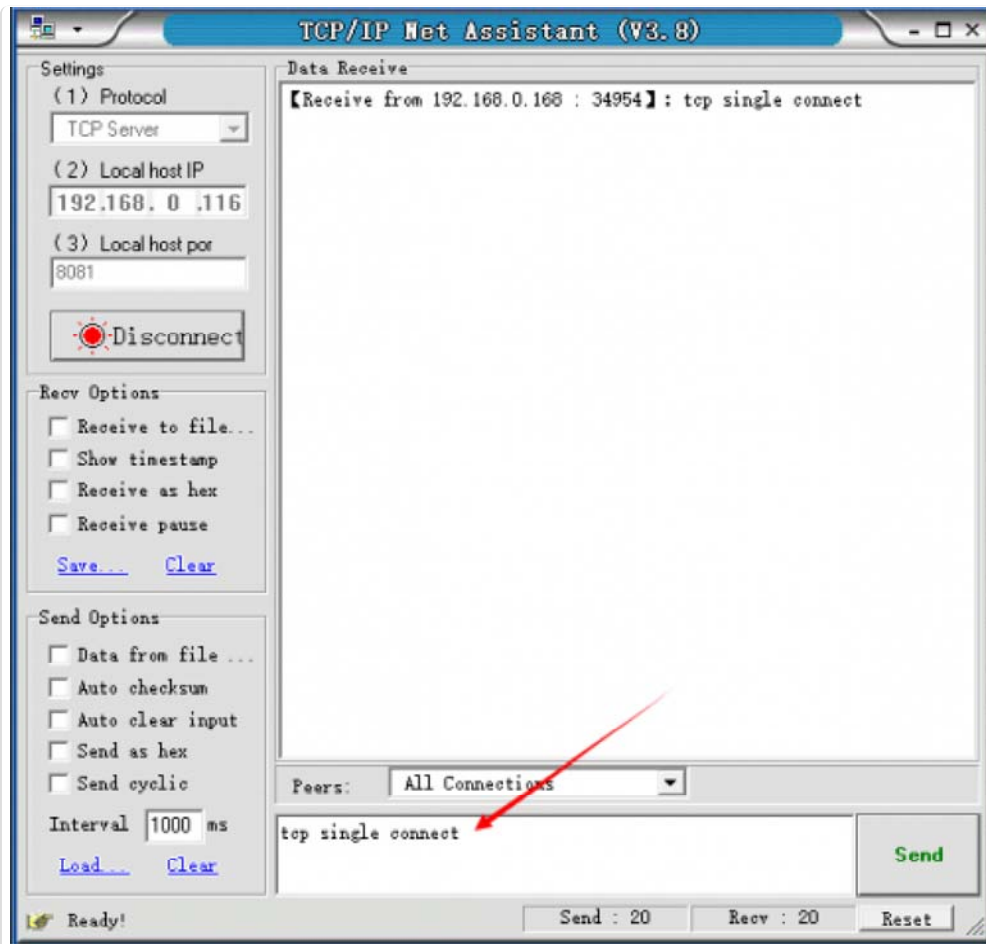
- 5 Upload the modified sketch, and then pull the switch "RUN/Prog" to "RUN";
- 6 Open "**CoolTerm**" and monitor the if the AP connection was good;



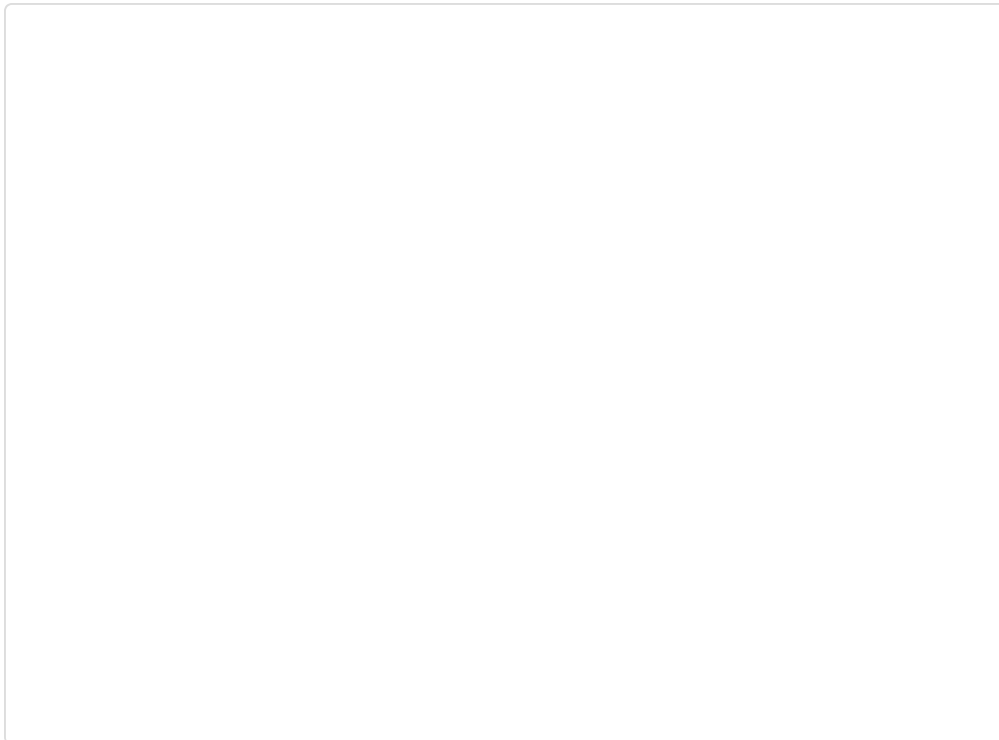


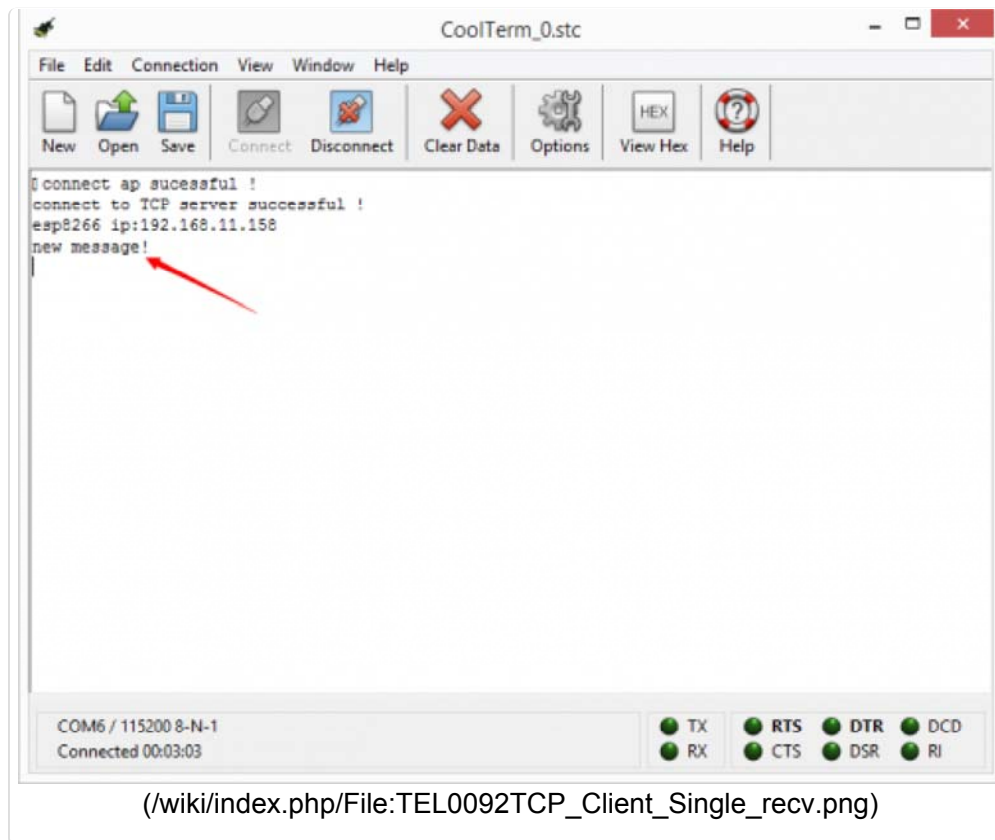
7 Send a message from "**TCP/IP Net Assistant V3.8**", you could see "New message" appear on "**CoolTerm**".





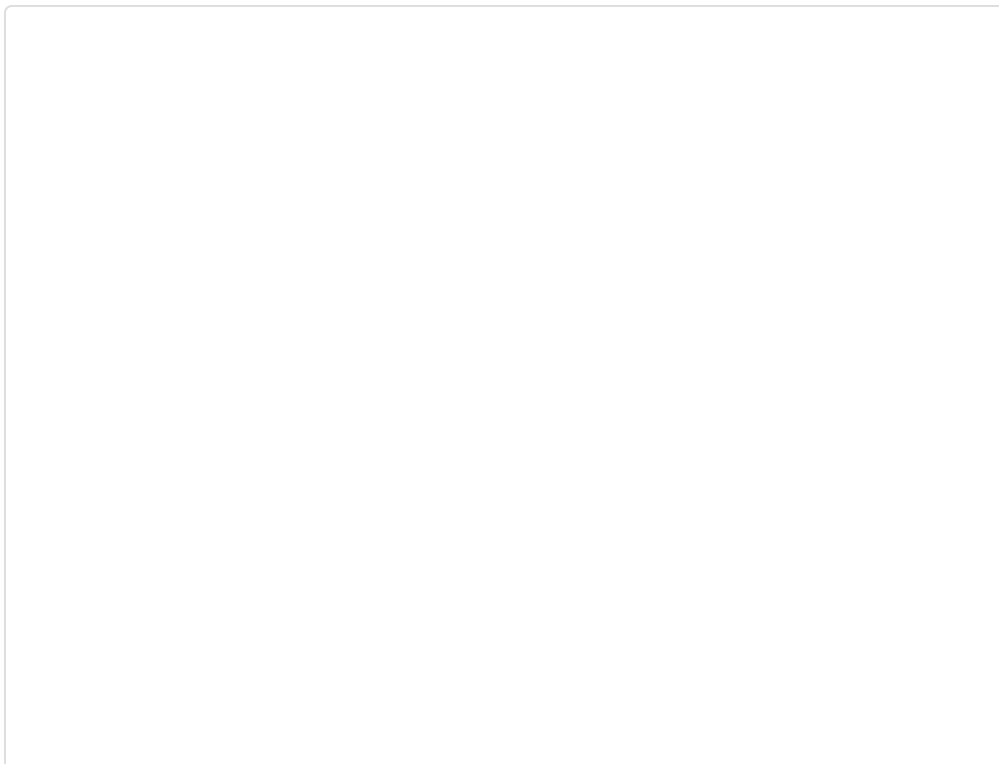
(/wiki/index.php/File:TEL0092TCP_Client_Single_Send.png)





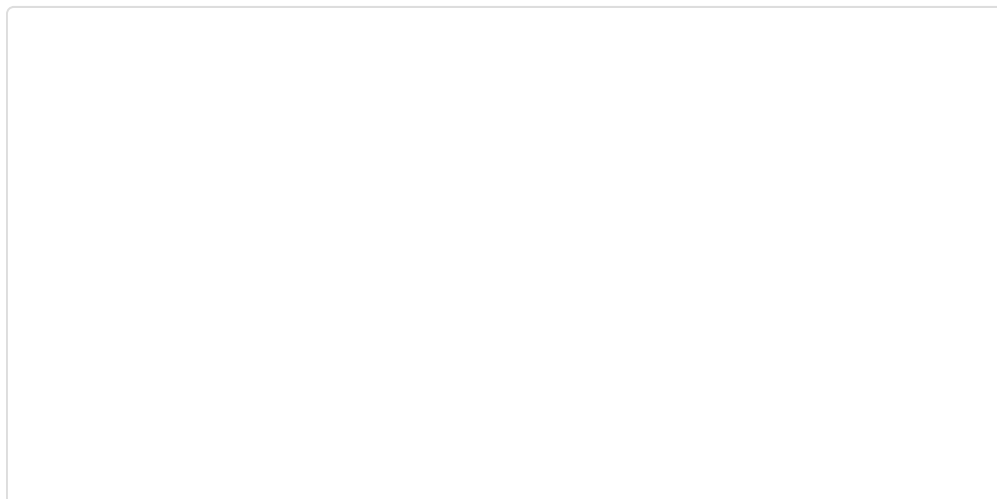
3 TCP_Client_multi

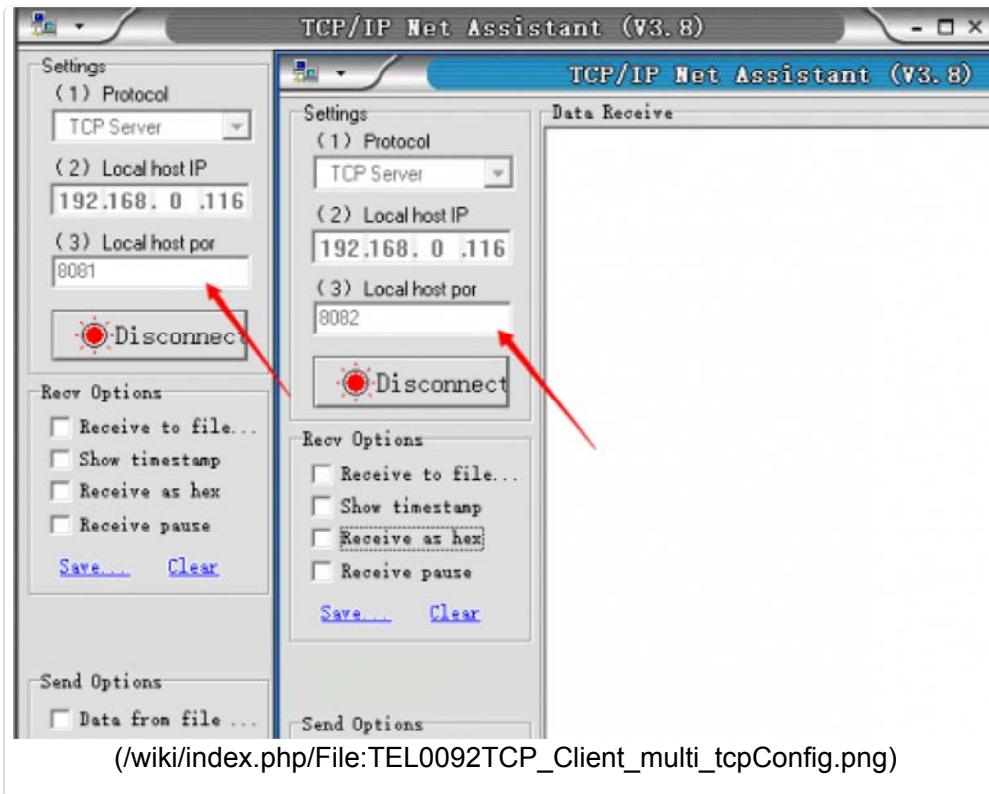
- 1 Pull the switch "RUN/Prog" to "Prog";
- 2 Open sample sketch "TCP_Client_multi", write your **wifi's ssid, password , Server Ip address, port**;





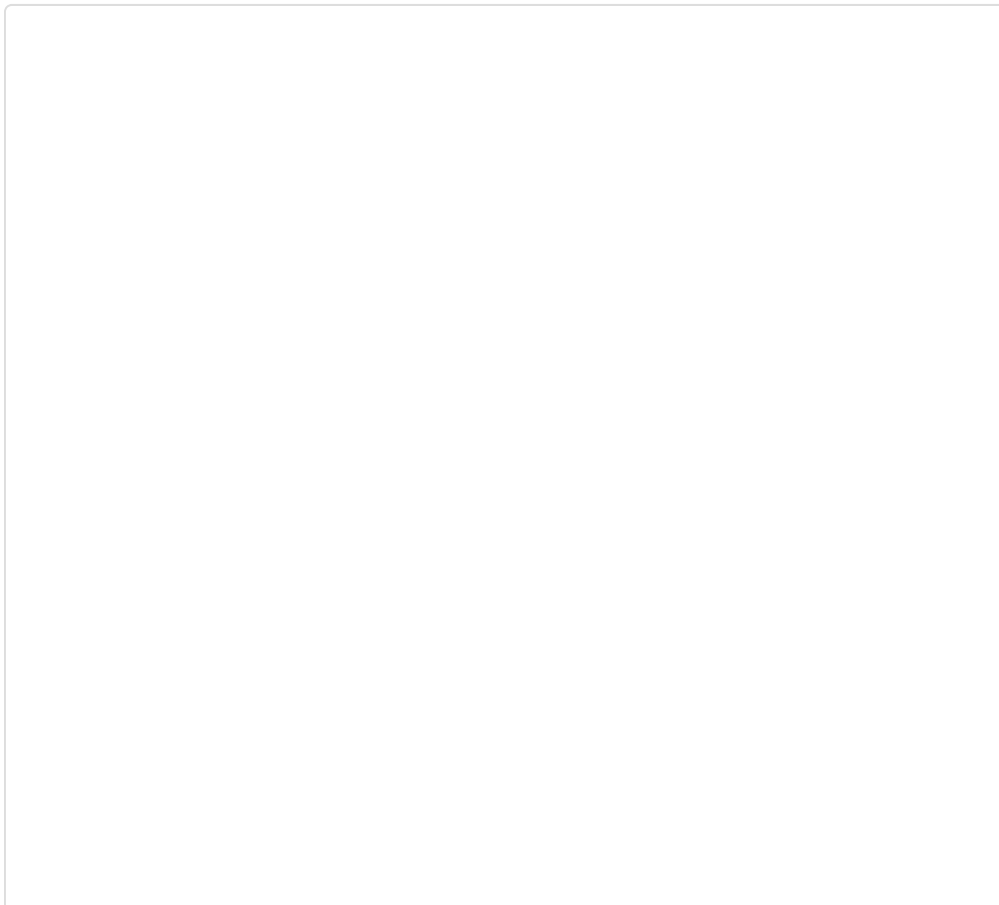
3 Open another "TCP/IP Net Assistant V3.8", configure as follow, and click **Connect**;

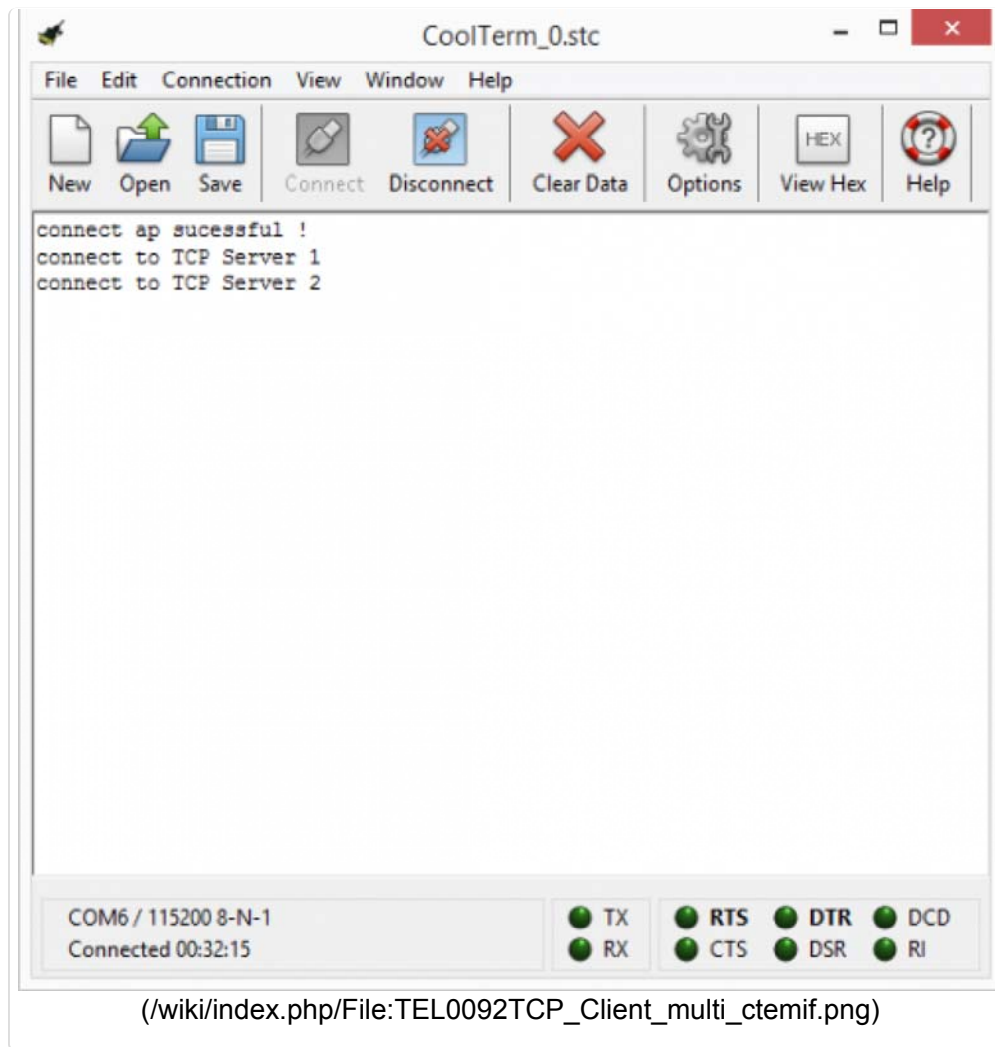




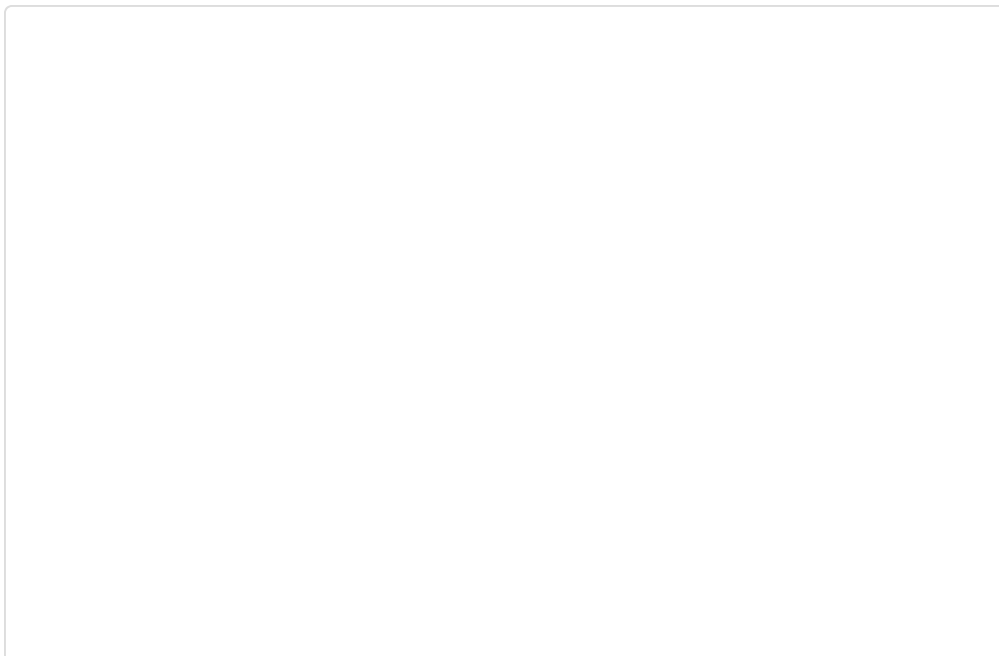
4 Upload the modified sketch, and then pull the switch "RUN/Prog" to "RUN";

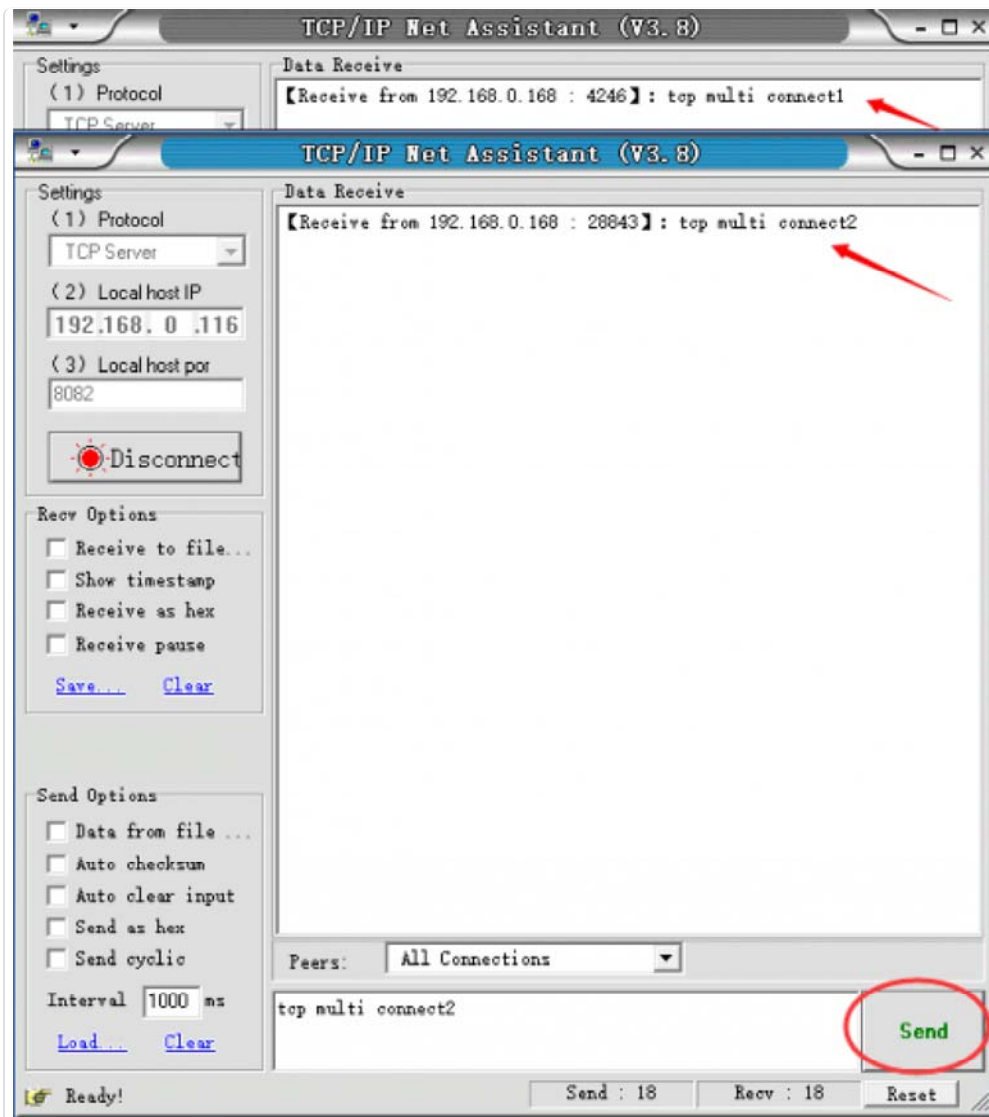
6 Watch "CoolTerm" to see if the AP connection was good;



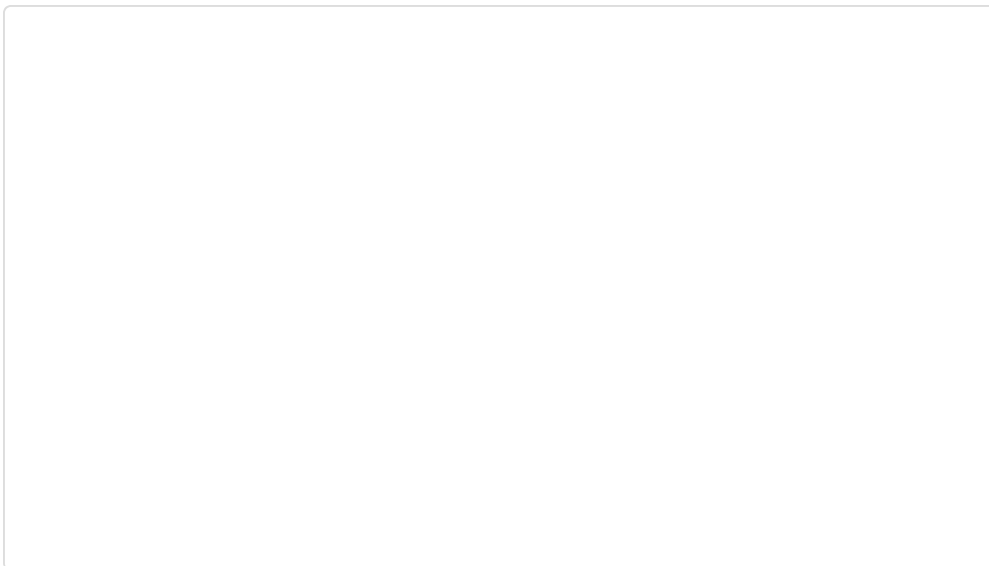


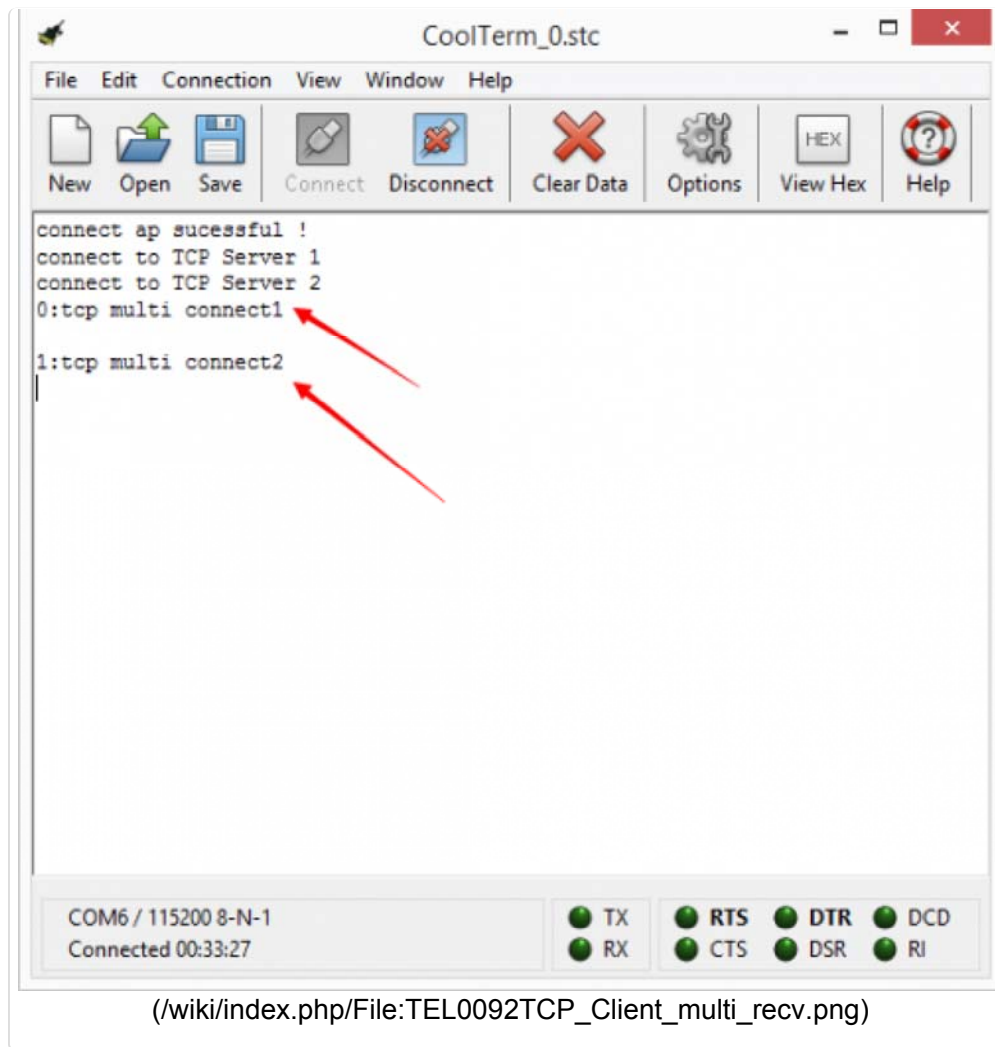
7 Send a message from different "**TCP/IP Net Assistant V3.8**", you could see New message appear on "**CoolTerm**" from different server .





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



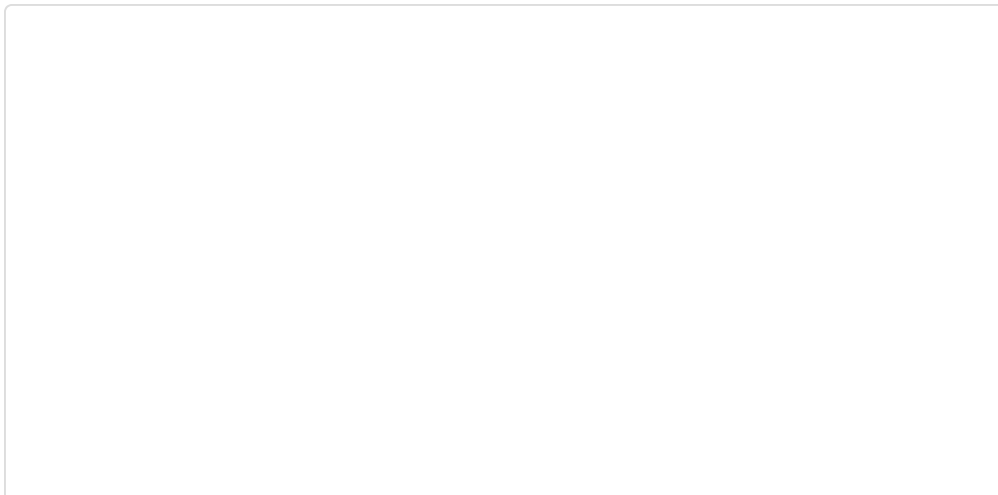


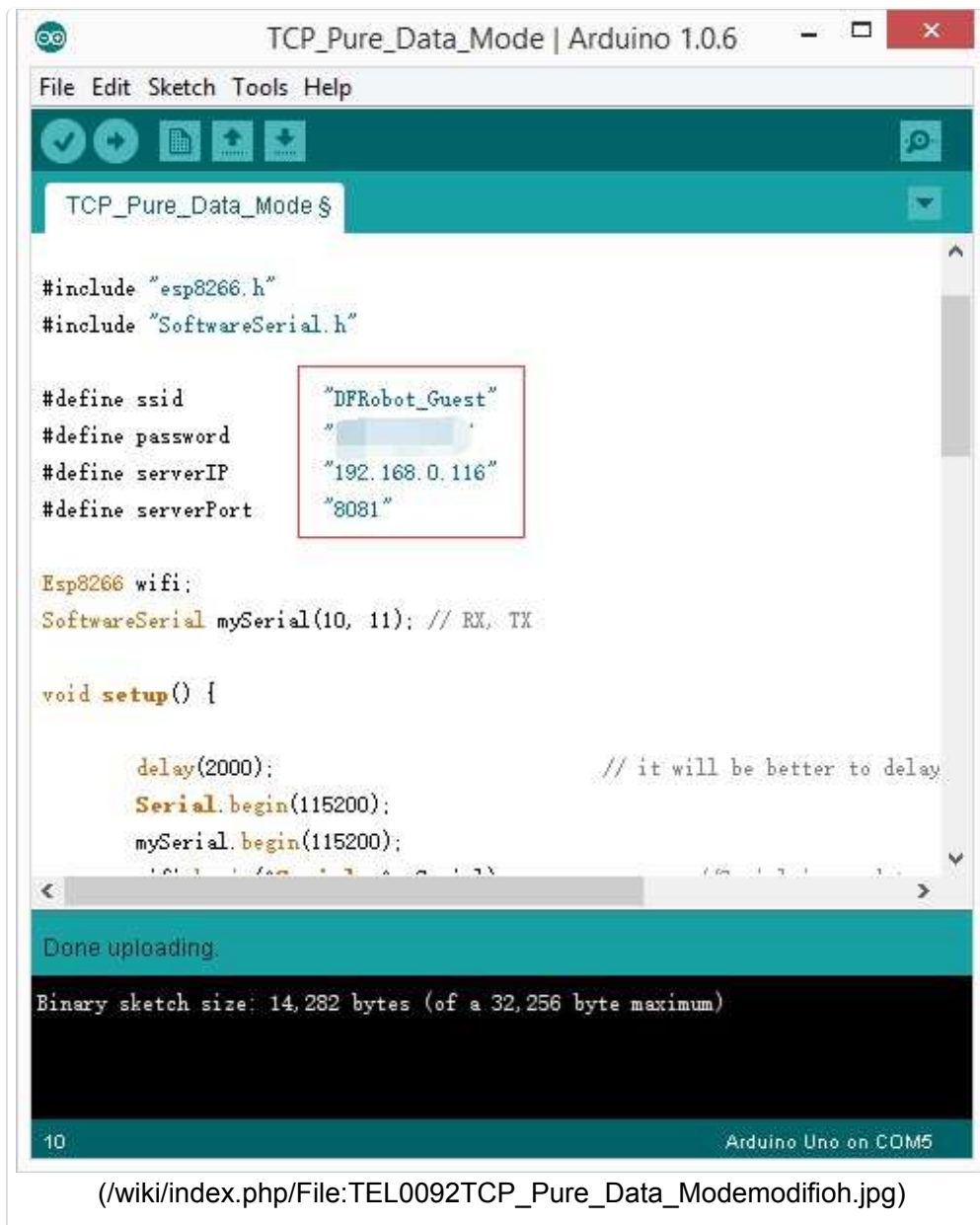
Note:ESP8266 can connect to 5 servers simultaneously.

4 TCP_Pure_Data_Mode

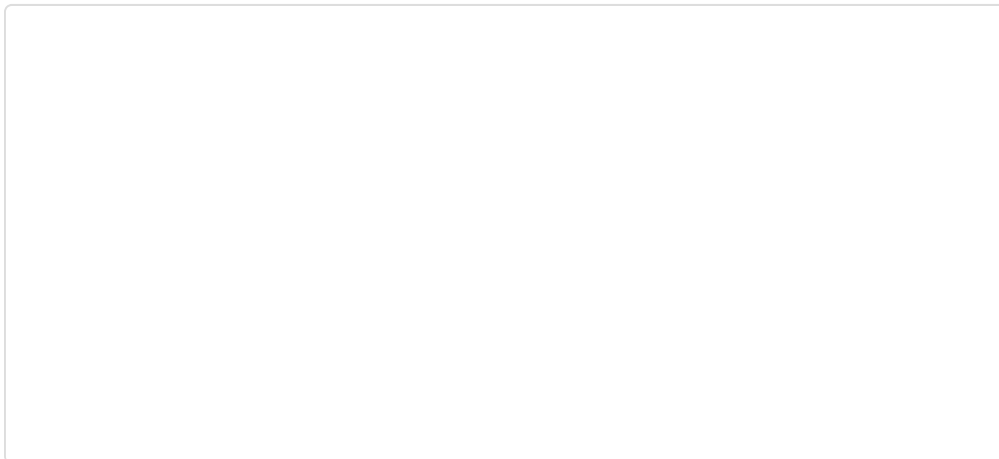
1 Pull the switch "RUN/Prog" to "Prog";

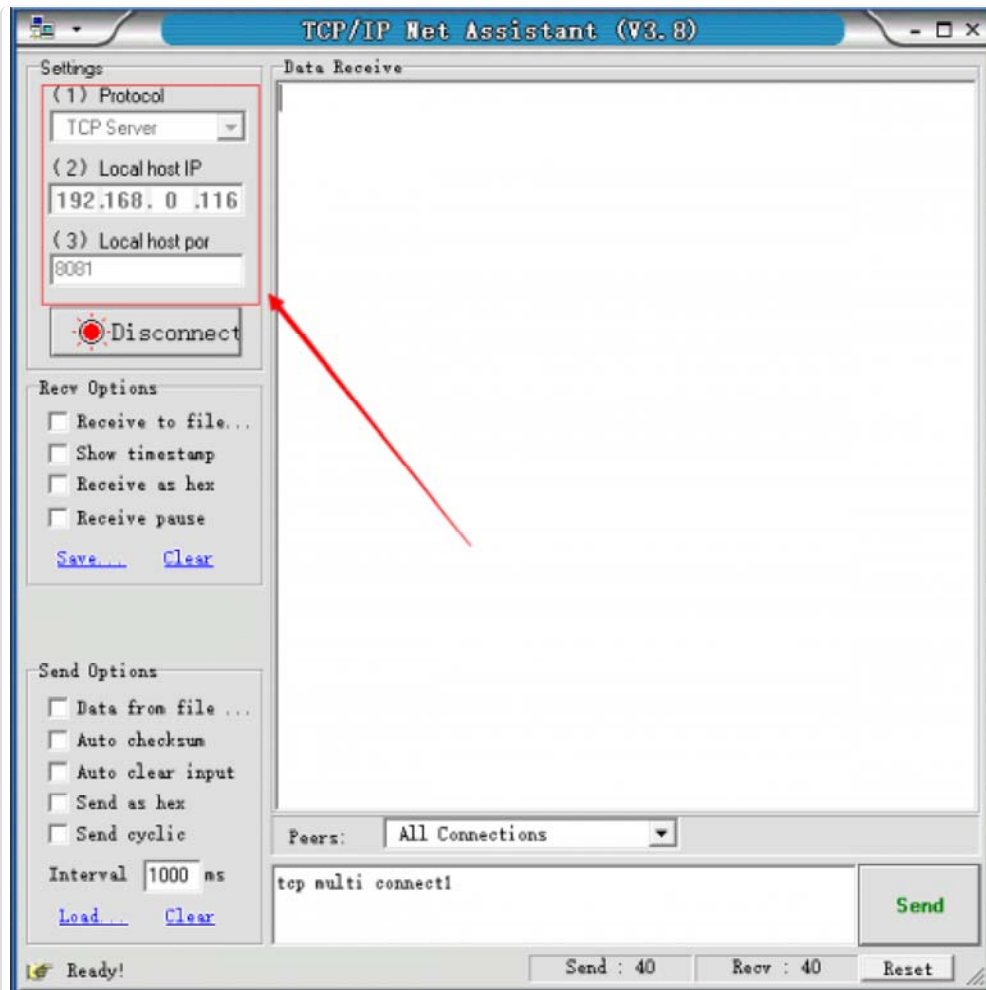
2 Open sample sketch "**TCP_Pure_Data_Mode**", write your **wifi's ssid, password , Server Ip address, port**;





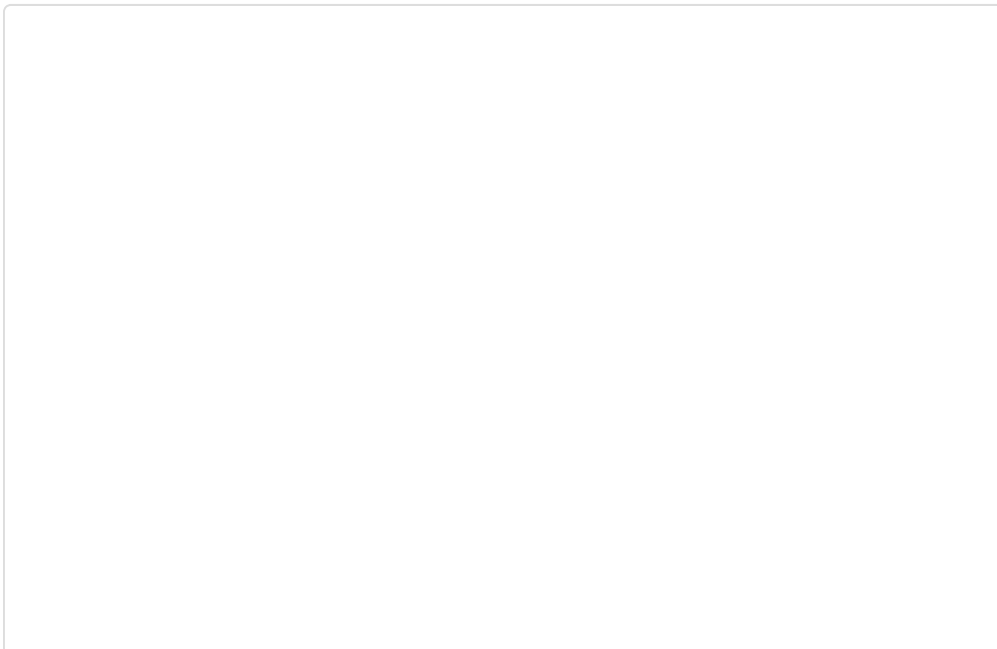
3 Open the software "TCP/IP Net Assistant V3.8", configure as follow, and click **Connect** to monitor the PC doing as Server;

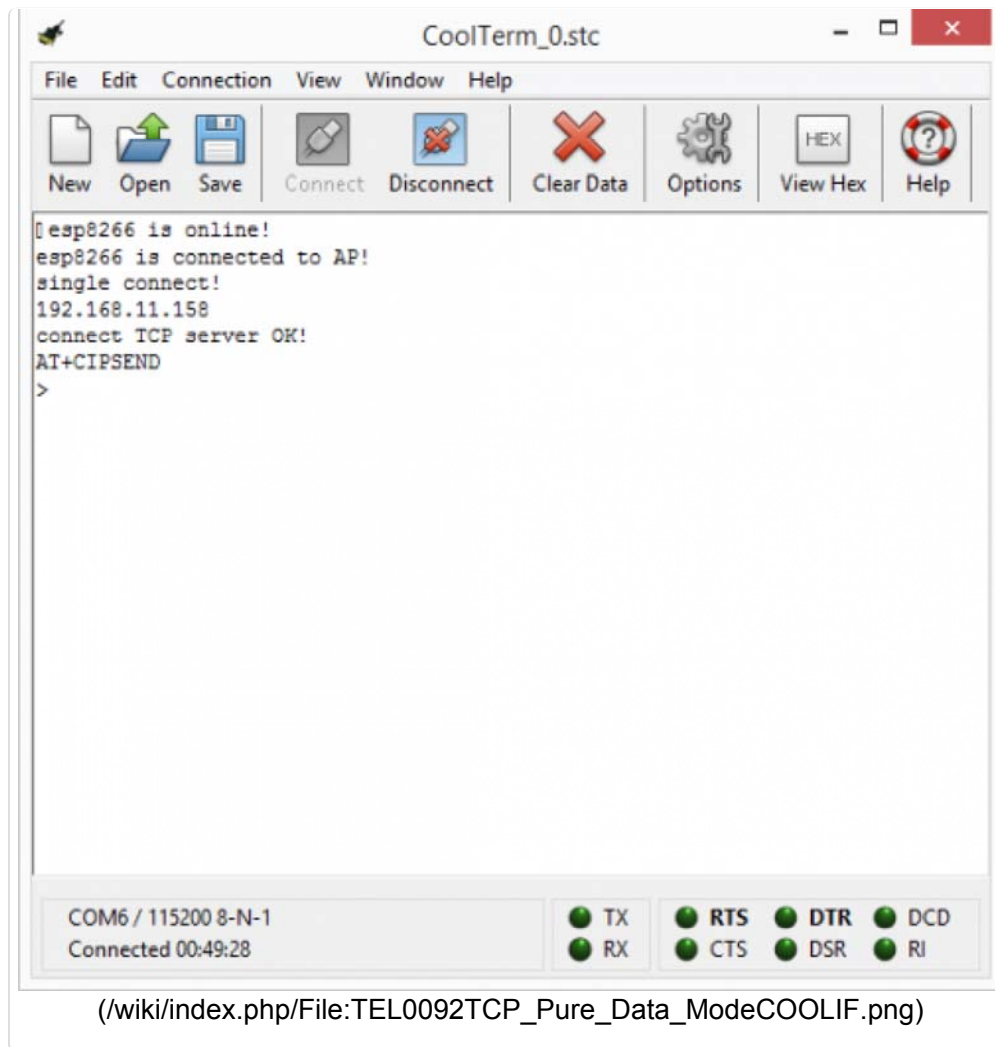




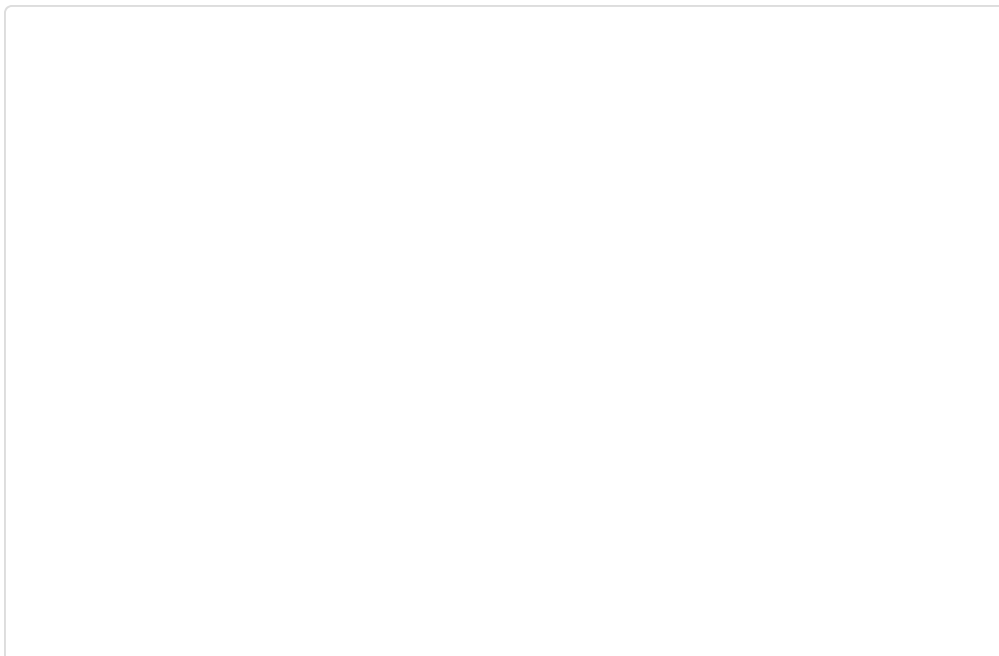
(/wiki/index.php/File:TEL0092TCP_Pure_Data_ModeTCPCFI.png)

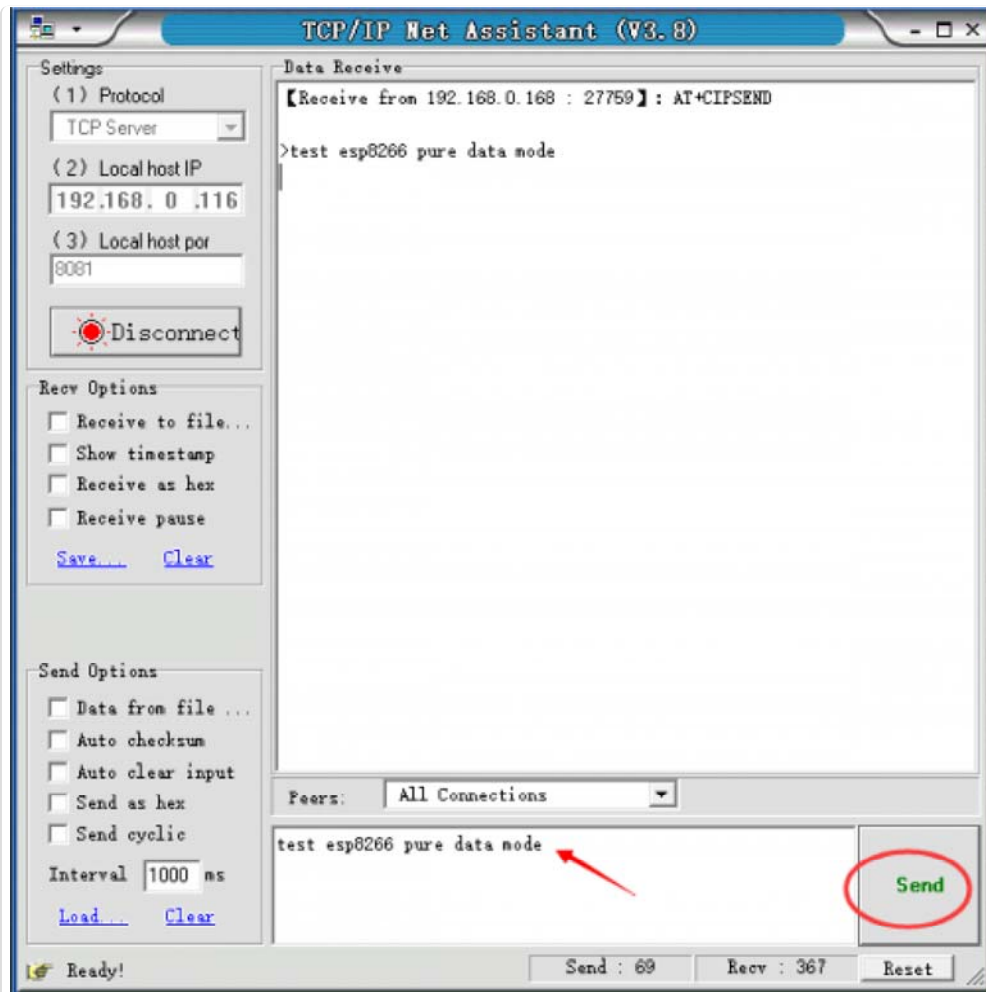
- 4 Upload the modified sketch, and then pull the switch "RUN/Prog" to "RUN";
- 5 Open "**CoolTerm**" and monitor the if the AP connection was good;



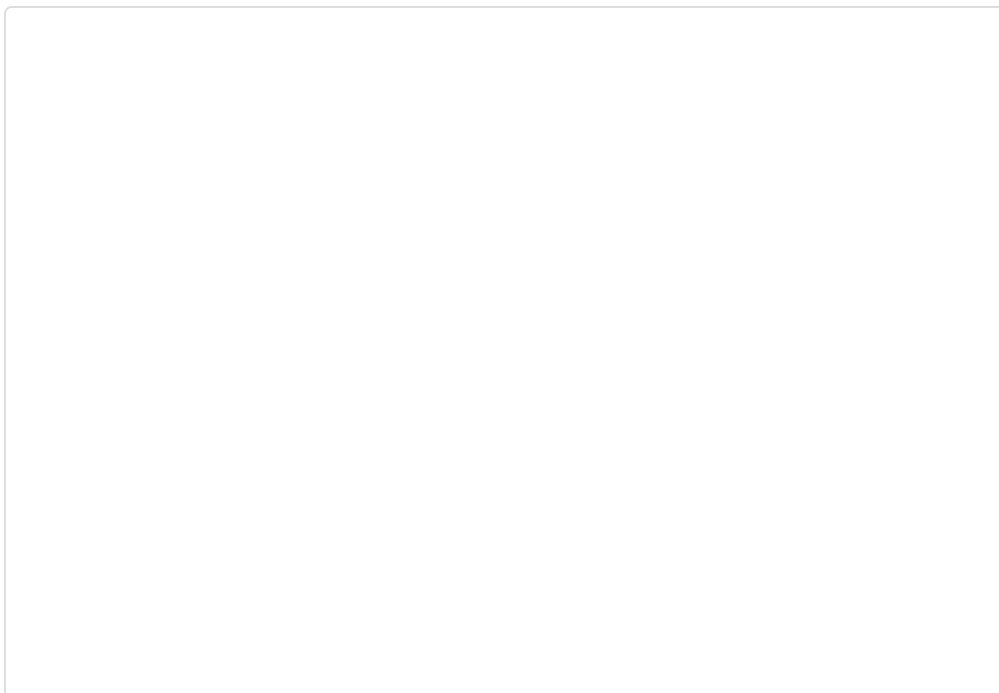


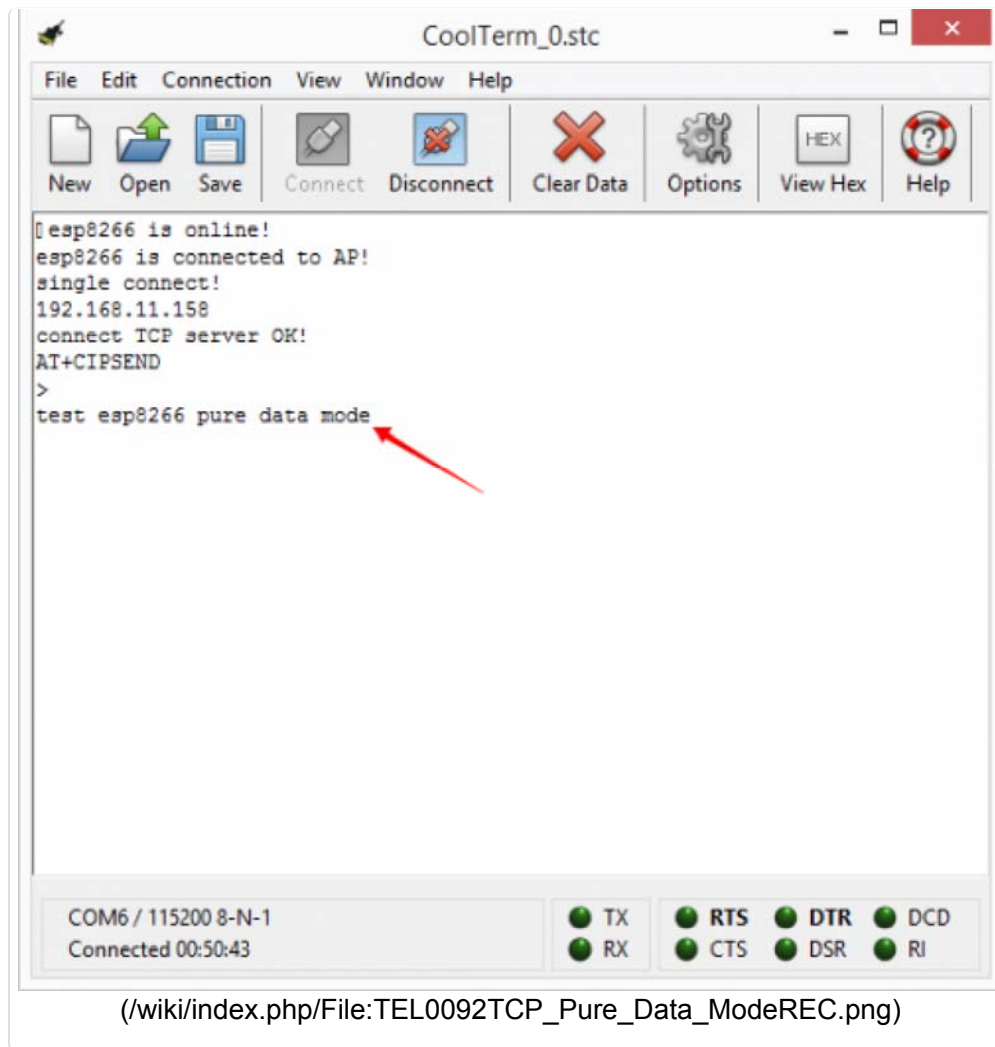
6 Send a message from **"TCP/IP Net Assistant V3.8"**, you could see New message appear on **"CoolTerm"**.





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





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 try.

* 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"

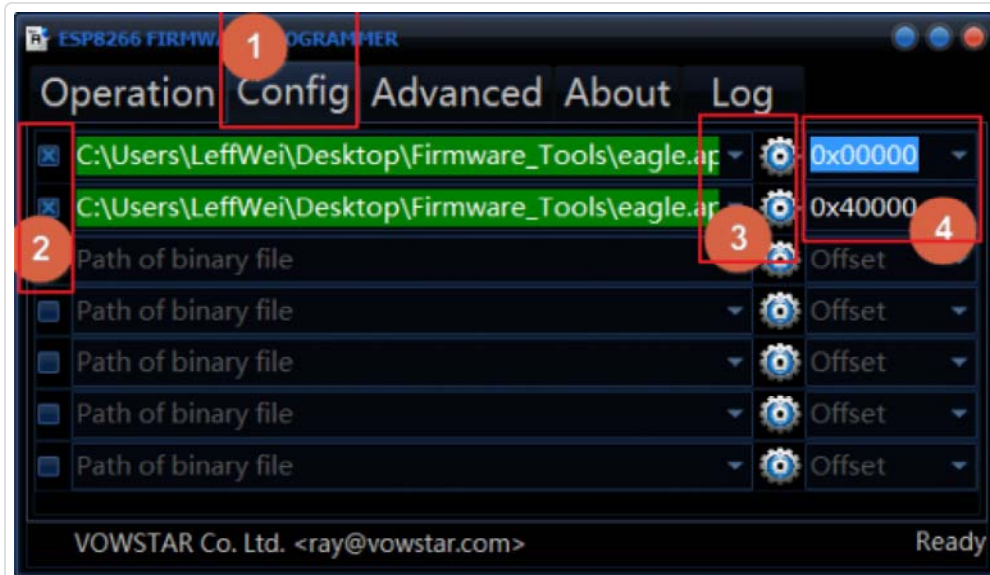
Update Firmware

Please download the Firmware and Tools

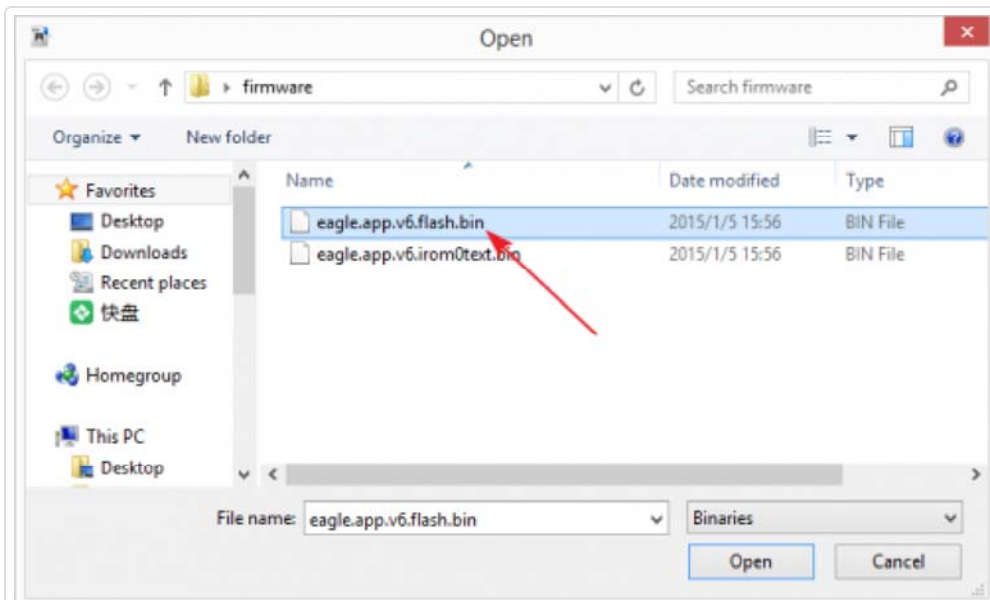
(https://github.com/Arduinolibary/DFRobot_Wifi_Bee_ESP8266/blob/master/ESP8266Flasher_en.zip?raw=true) first. then Like in **AT mode**, but pull the swith of ESP8266 to "UART" side. Don't forget to pull it back to "BOOT" after updating firmware.

1 Open software "**ESP_Flasher**";

2 Choose Firmware;

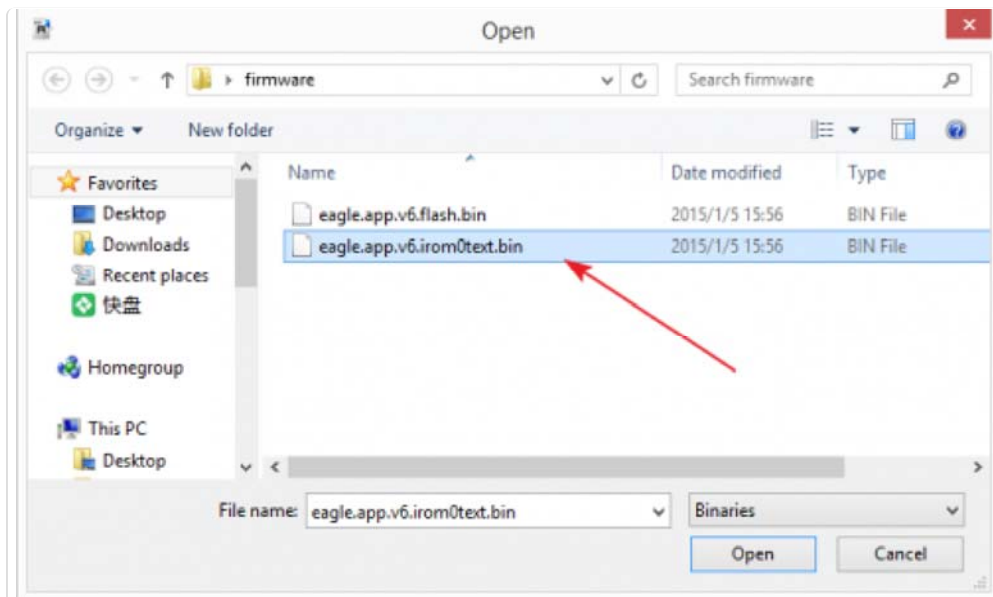


(/wiki/index.php/File:TEL0092Firmware2en.png)



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

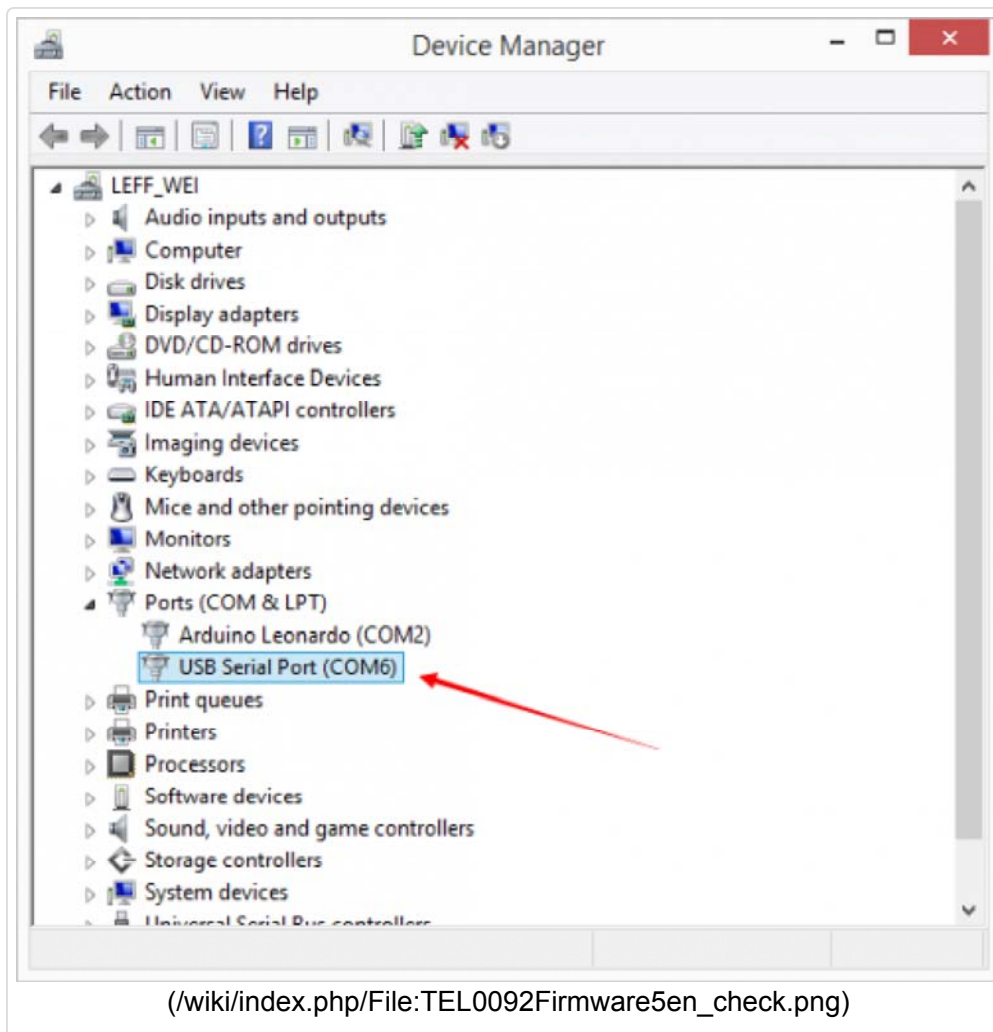
Address at 0x00000



(/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)