

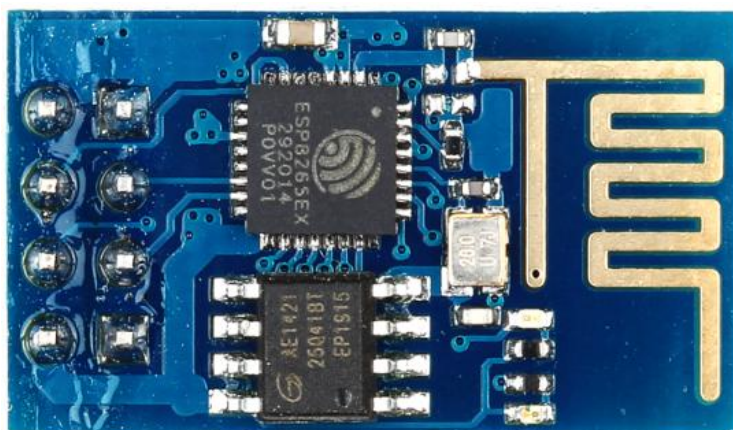
## ESP8266 Serial WiFi Module (ESP-01)

Hari Krishnan

August 10, 2015

Wireless

With the popularity of Wifi IoT devices, there is an increasing demand for low-cost and easy-to-use WiFi modules. ESP8266 is a new player in this field: it's tiny (25mm x 15mm), with simple pin connections (standard 2x4 pin headers), using serial TX/RX to send and receive Ethernet buffers, and similarly, using serial commands to query and change configurations of the WiFi module. This is quite convenient as it only requires two wires (TX/RX) to communicate between a micro-controller and WiFi, but more importantly, it offloads WiFi-related tasks to the module, allowing the microcontroller code to be very light-weighted.



ESP8266 ESP-01 Serial WIFI Transceiver Module is a cheap and easy way to connect any small microcontroller platform, like Arduino, wirelessly to Internet. ESP8266 has powerful on-board processing and storage capabilities that allow it to be integrated with the sensors and other application specific devices through its GPIOs with minimal development up-front and minimal loading during runtime. Its high degree of on-chip integration allows for minimal external circuitry, and the entire solution, including front-end module, is designed to occupy minimal PCB area.

[Arduino](#)[Development Boards](#)[Displays](#)[GPS](#)[GSM/GPRS](#)[Interfacing Modules](#)[Mini Development Boards](#)[Miscellaneous](#)[Programmeters](#)[Quick Start Boards](#)[Raspberry Pi](#)[RFID Reader & Tag](#)[Robotics](#)[Sensors](#)[Wireless](#)

ESP-01 WIFI Transceiver Module is addressable over SPI and UART, making this an exceptionally easy choice for anyone wanting to build an Internet of Things thing. You can use AT commands to connect to WiFi networks and open TCP connections without need to have TCP/IP stack running in your own microcontroller: You can simply connect any microcontroller to this module and start pushing data up to the Internet.

## Features

---

- Serial UART Interface
- It run LWIP
- 802.11 bgn
- WIFI Direct (P2P),SOFT-AP
- Built-in TCP/IP
- The AT command is perfect,efficient,concise
- Support three modes: AP, STA and AP+STA coexistence mode
- Onboard PCB Antenna

## Specifications

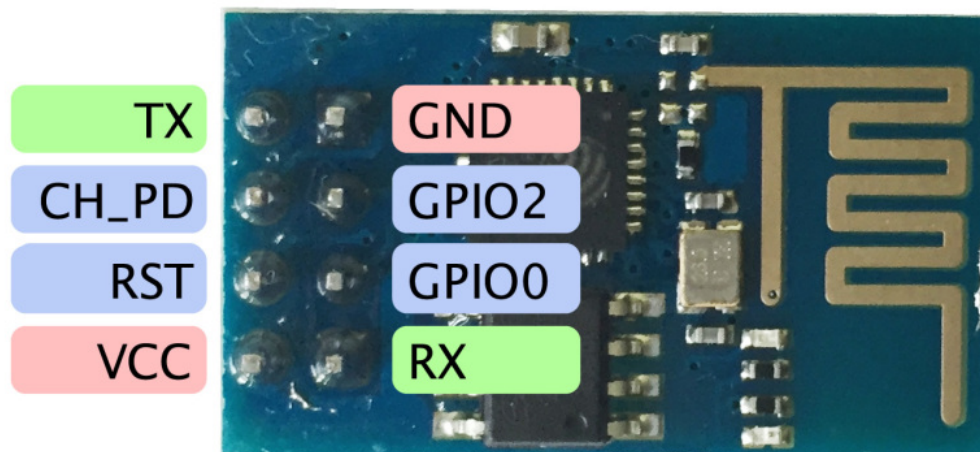
---

- WIFI Direct (P2P), SOFT-AP
- Controlled via UART interface at 115200 baud with a simple set of AT commands
- Only 2 microcontroller pins needed for communication (RXD/TXD)
- Support three modes: AP, STA and AP+STA coexistence mode the TCP/IP protocol suit
- Integrated TCP/IP protocol stack
- Integrated TR switch, balun, LNA, power amplifier and matching network
- Integrated PLLs, regulators, DCXO and power management units
- +19.5dBm output power in 802.11b mode
- Power down leakage current of <10uA
- Integrated low power 32-bit CPU could be used as application processor
- SDIO 1.1/2.0, SPI, UART
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU & A-MSDU aggregation & 0.4ms guard interval
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW (DTIM3)

- ESP-01 PCB Antenna , after matching the distance to achieve open 400Meters.

## Pin Description

---



**ESP8266 is 3.3v device & cannot tolerate 5v levels.**

Name	Type	Description
VCC	-	Voltage DC input (3.3 V)
RST	-	Reset
CH-PD	Chip Enable	High: On, chip works properly Low: Off, small current
TXD	Output	Serial Port Transmit Data (TTL level)
GND	-	Ground
GPIO 2		UART Tx during flash programming
GPIO 0		SPICS2
RXD	Input	Serial Port Receive Data (TTL level)

Connect the CH\_PD pin to 3.3V to activate the chip.

## ESP8266 Communication with other WiFi modules: ESP8266 as a Client

---

ESP 8266 works in 3.3V TTL so we need a level converter to communicate with our PC. You can use an FTDI converter for this purpose. **Make sure that your FTDI is working in 3.3V.** Connect your ESP module with the FTDI Driver and plug the FTDI to the USB port of your PC.

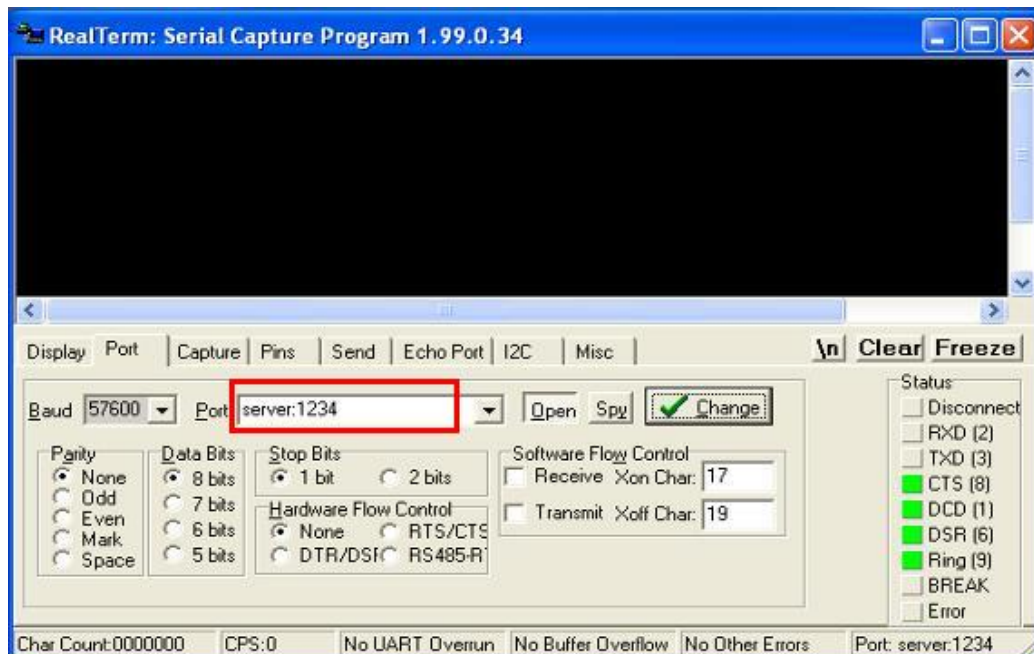
**WIFI Module****USB-TTL****Vcc****3.3v****Gnd****Gnd****Tx****Rx****Rx****Tx****CH\_PD****connected to 3.3v to enable chip firmware****boot.****RST****connected to 3.3v through 3k3 resistor**

For communication you must need another computer or a phone which has WiFi access. Connect the computer or phone with our ESP 8266 module through WiFi. You will have to configure the PC or mobile as server, use terminal software to set the PC as server. In realterm software you can change it in the port settings.

Set the port as server and give a port number.

Example: Server:1234

Here 1234 is the port number.



You can use the [AT commands](#) to set the parameters of your ESP module. All AT commands must end with a "\r\n" – a carriage return & a New Line. Use any terminal software for giving AT commands to the ESP module.

- Initialize the ESP module by sending AT command, will get an 'OK' response.
- Query the ESP 8266 as client or as a server, here it is client.

**AT+CIPMUX?**

- 0 = Client , 1 = Server

- ESP8266 WIFI module has 3 modes of operation. **AT+CWMODE?** returns the Mode of operation of the module. CWMODE returns an

integer designating the mode of operation either 1, 2, or 3.

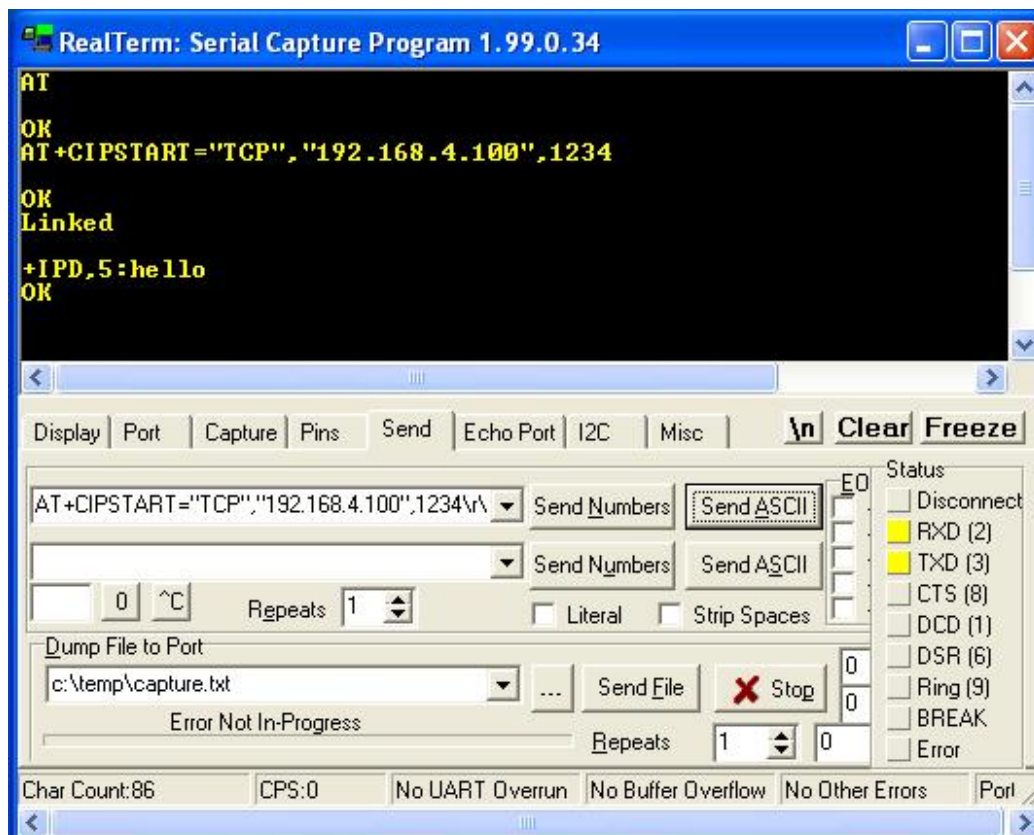
- 1 = Station mode (client).
- 2 = AP mode (host).
- 3 = AP + Station mode (Yes, ESP8266 has a dual mode!)

Note: To send a data from ESP module we must need to know, **Type of connection(TCP/UDP), IP address and port number** to assigned with command AT+CIPSTART. (First three sections of IP address will be same last section only will have some change)

Let's take a break from networking class to see if your little 8266 module is able to communicate with your network.

For example,

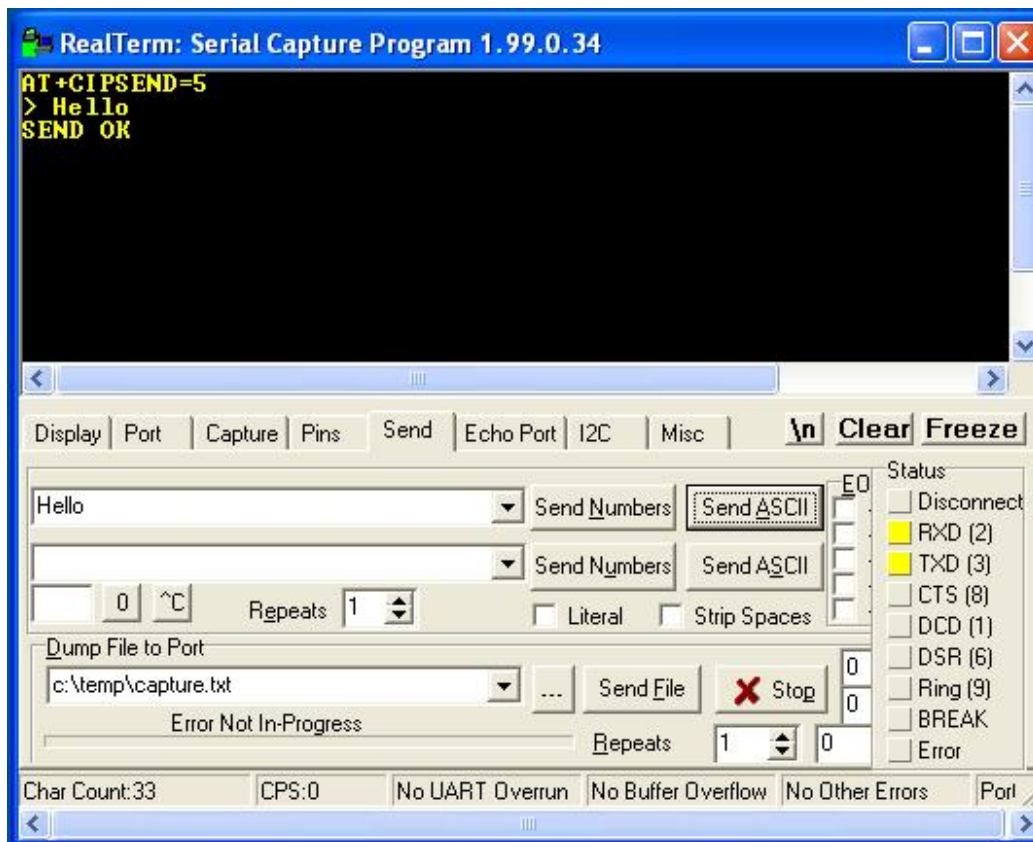
1. Click on Start, Run, and type CMD and press enter.
  2. Type IPCONFIG and press enter.
  3. Thus, IP Address of your module get identified
- After sending the command **AT+CIPSTART**, connection get established thus get response as **OK Linked**. Now communication is possible, from the communicating PC we received a data in the format **+IPD, size of the data : data** with 'OK' response as shown below.



- From ESP module you can send data after receiving >symbol. Send



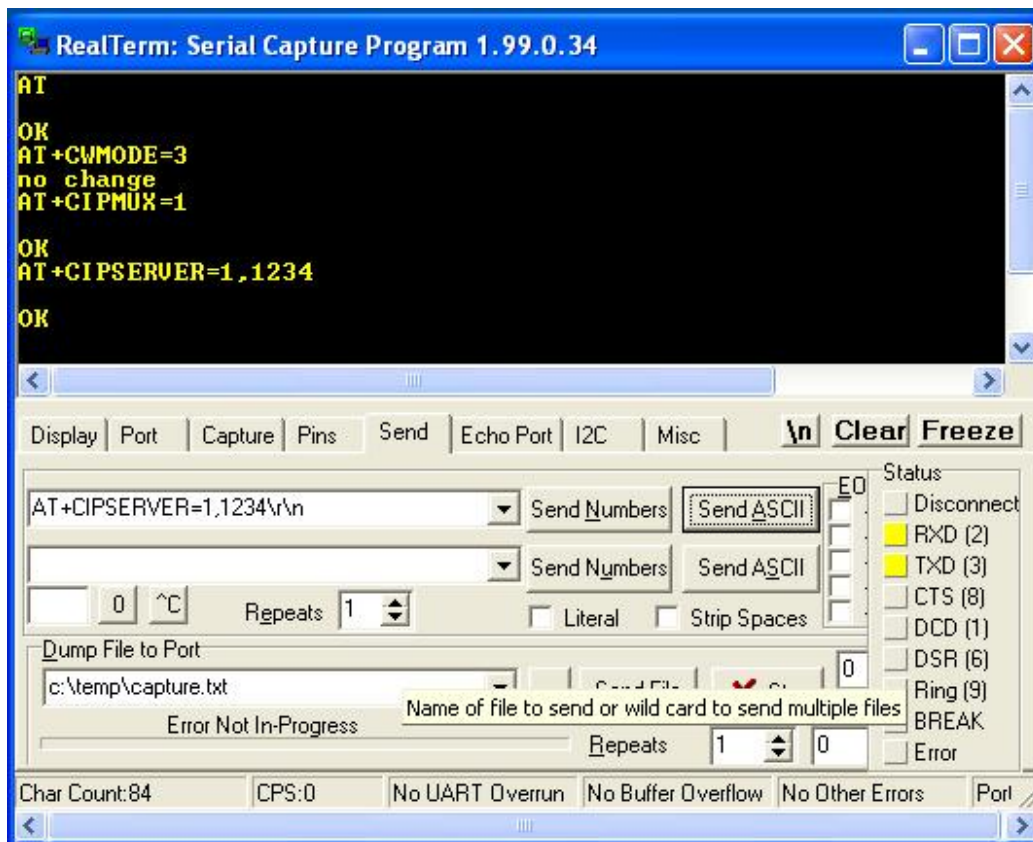
the exact data with data length which is mentioned in the **AT+CIPSEND=<len>** command. You will get a SEND OK after sending the data as shown below.



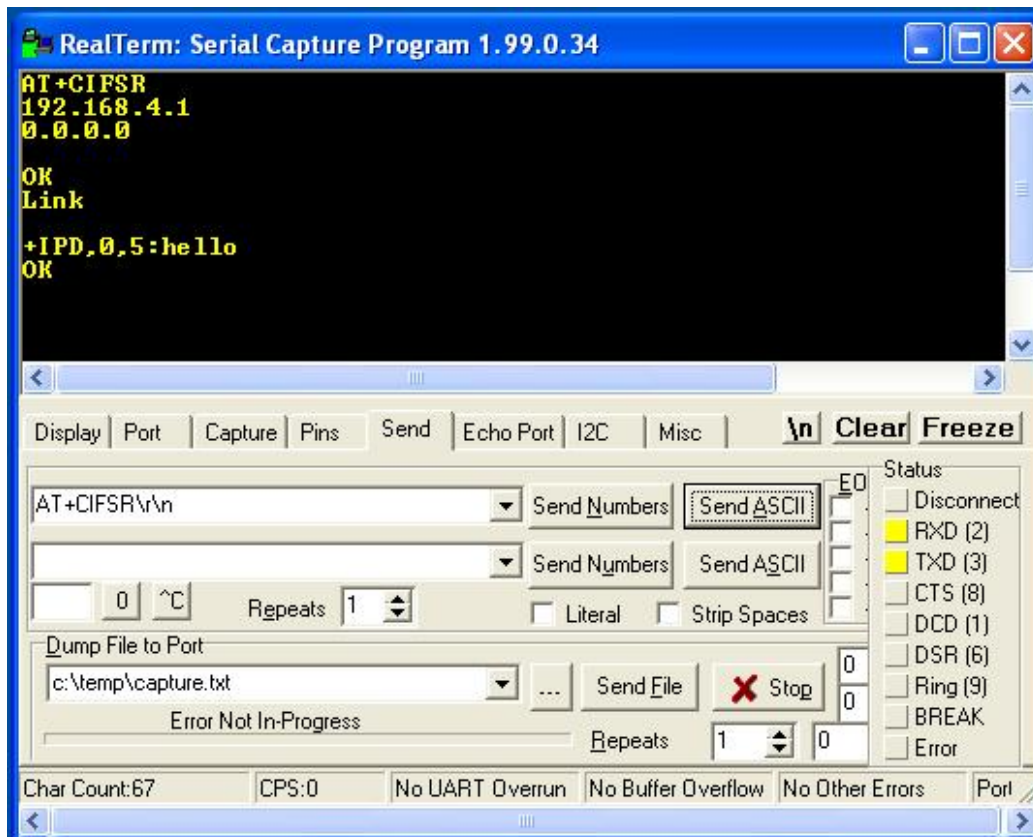
## ESP8266 Communication with other WiFi modules: ESP8266 as a Server

The user can configure the ESP 8266 as server by following commands.

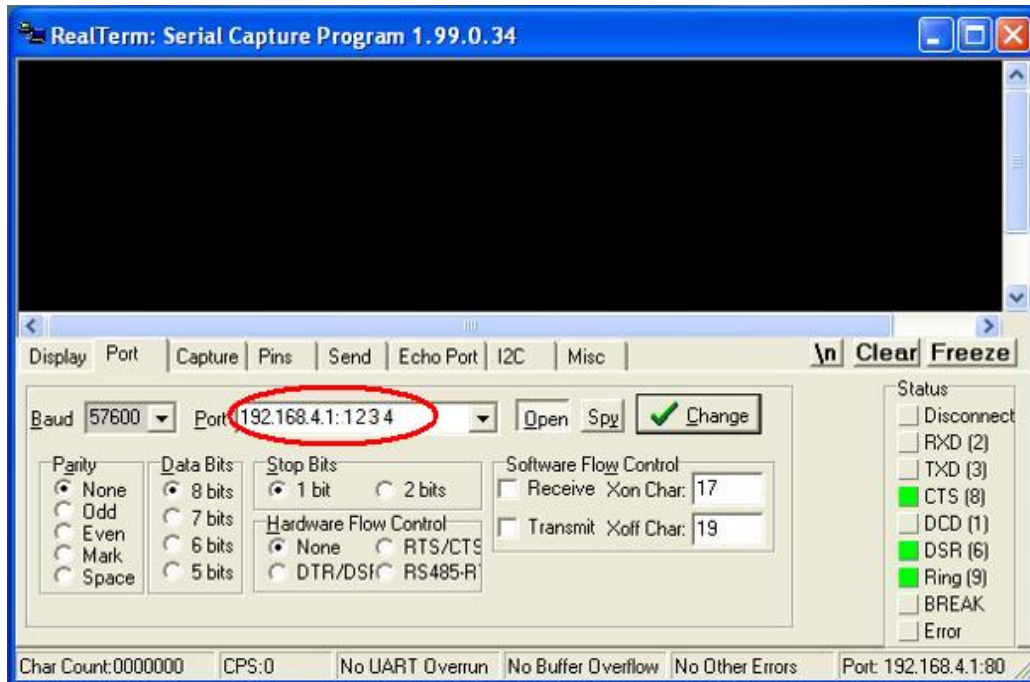
- In order to configure the ESP as server you should enable multiple connections to ESP(**CIPMUX=1**).
- Then configure the ESP module as server by command **AT+CIPSERVER=<par>,<port>** ,we get 'OK' as response. And use this IP for communication from other PC.
  - par: 1 – open server, 0 – close server
  - port: port number



- You will get the IP address by sending **AT+CIFSR** command. After connection get established with the IP, communication is possible from both sides. We may use **AT+CIPSEND=<len>** for sending a data. Note: If CWMODE is 3 (Both) you will get two IP address. Choose the station IP Address which is the first one.



Now take another PC which has WiFi access connect it with our ESP 8266 module. Use terminal software to communicate with ESP 8266. In realterm you can connect the ESP module by setting the port as module IP as shown below, where 1234 is the port number.



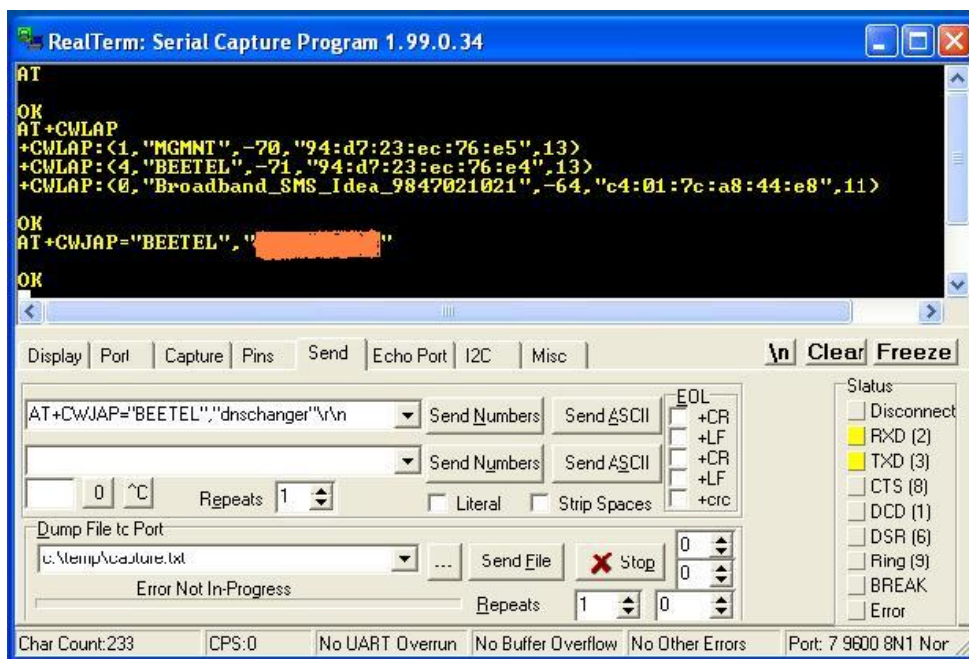
This method can be used to communicate only when two WiFi modules are in range.

## WiFi Module ESP 8266 – TCP Client /Server

Now we will discuss about Internet of Things (IoT). So by using IoT the ESP module can be accessed from anywhere. In order to access the ESP 8266 from anywhere, you should first connect it with an Access Point. The ESP 8266 can be connected to an Access Point by using the following AT commands. The common commands for TCP client and server are explained below, let's go through it ..

- Initialize the ESP module by sending **AT** command, will get an 'OK' response.
- Lists available APs with **AT+CWLAP**, after listing we get 'OK' as response.
- After listing, Connect to AP with the command **AT+CWJAP=ssid,pwd**. And thus Commands ESP8266 to connect a SSID with supplied password. Get 'OK' as response as shown below.

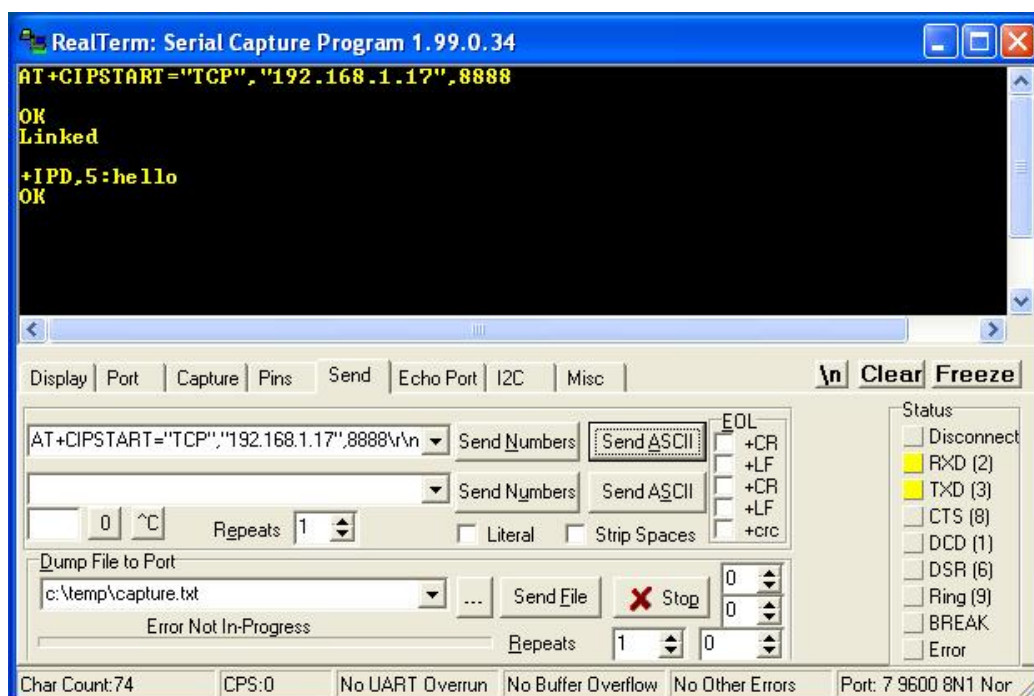




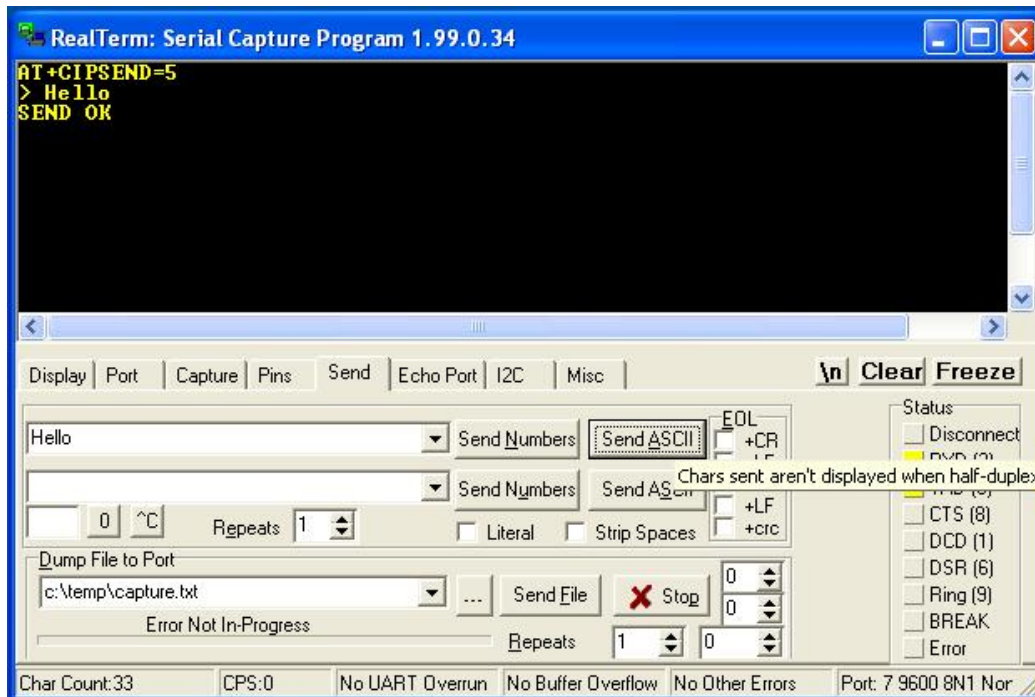
## WiFi Module ESP 8266 – TCP client

Configure one of the ESP 8266 module as server and other as client.  
Client section is explained below.

- Now start a connection as client. (Single connection mode) with **AT+CIPSTART=type,addr,port** as shown below ,(First three sections of IP address will be same last section only will have some change where 192.168.1.17 is the IP of the router).
- Thus connection get established receives a response as OK Linked
- And then go for sending and receiving data as shown below.



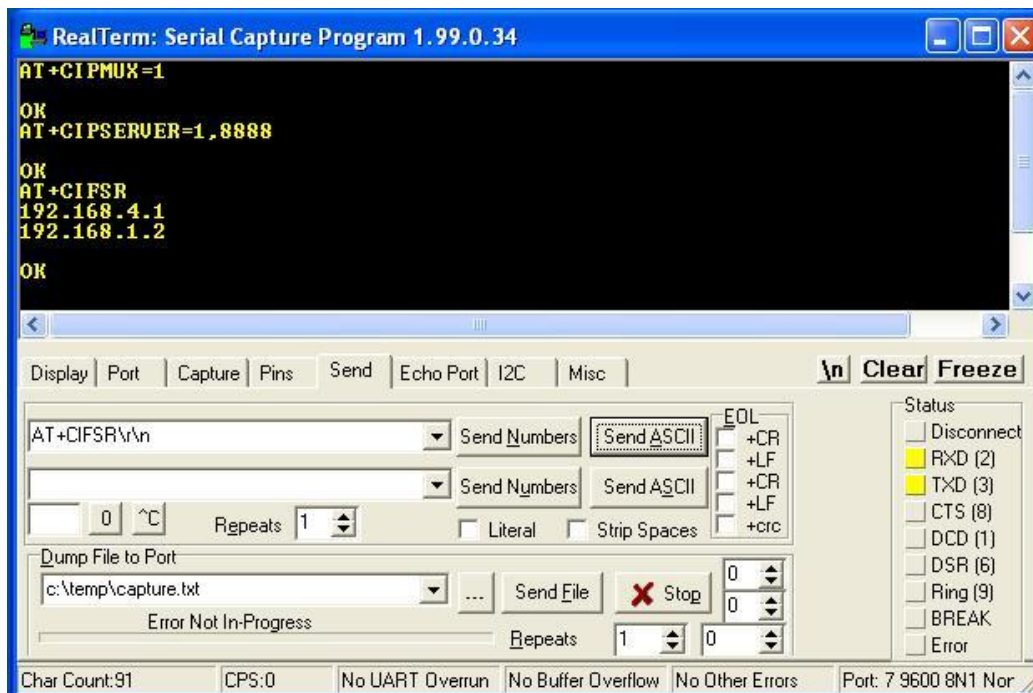
- Send the exact data with data length which is mentioned in the **AT+CIPSEND=<len>** command. You will get a SEND OK after sending the data. The data from the server is received in format as shown below



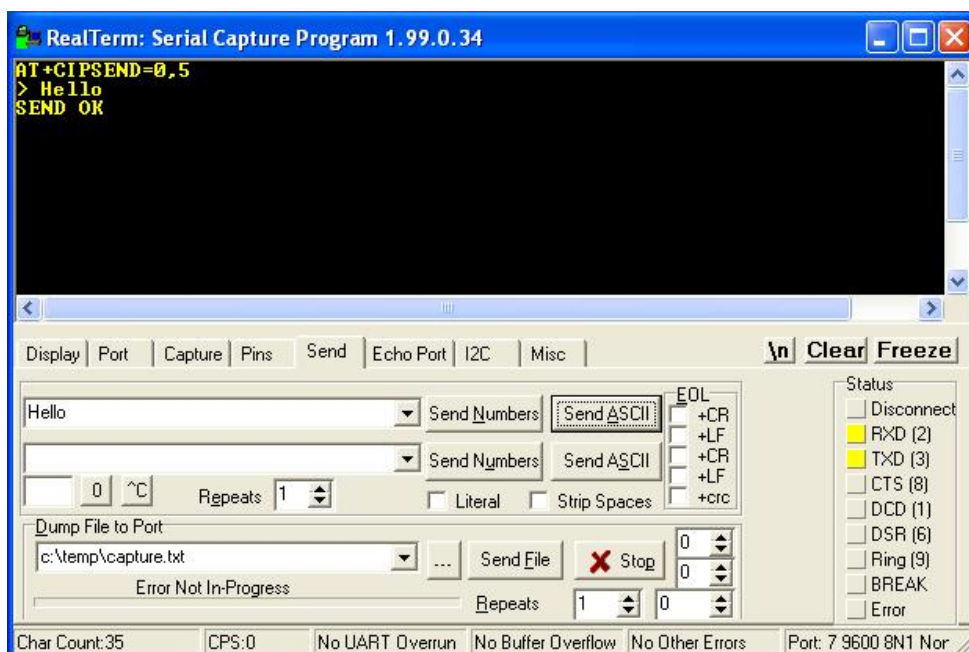
## WiFi Module ESP 8266 – TCP SERVER

Configure one of the ESP 8266 module as server and other as client. Server section is explained below.

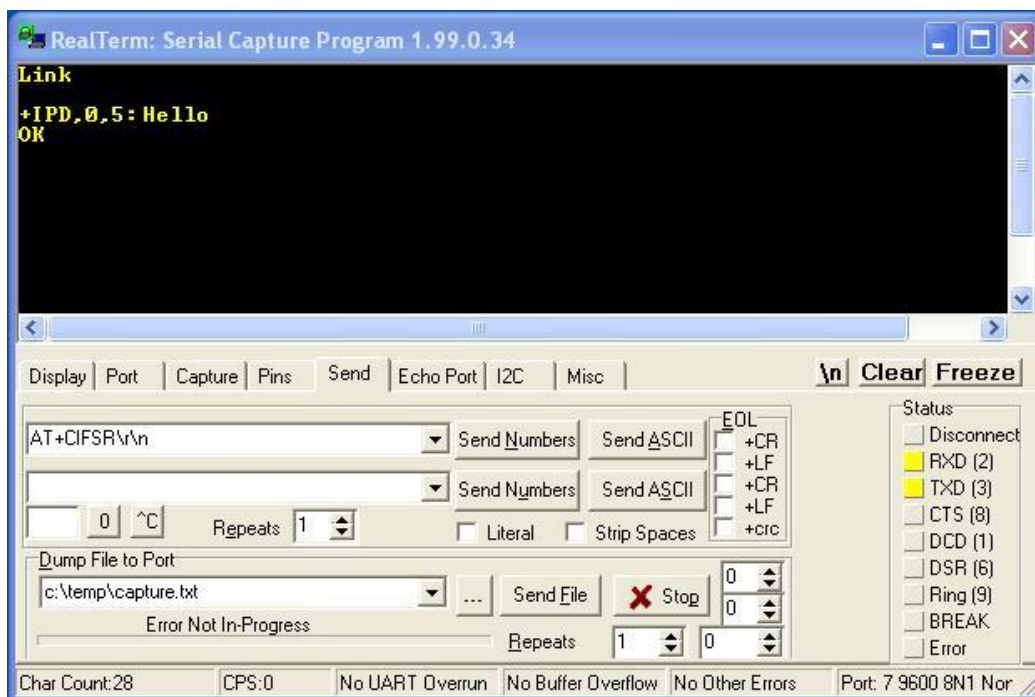
- To communicate with a webpage, we need to start a server with ESP module. You need to set multiple connections using command `AT+CIPMUX=1`. This initiates the server for a Webserver.
- Next we start the server at HTTP port 8888 using command, `AT+CIPSERVER=1,8888`
- To close the server the command is `AT+CIPSERVER=0` followed by a Reset `AT+RST`(Depends the module)
- To get the STATION IP address, send `AT+CIFSR`, the module returns the Station IP address, 192.168.1.2 . If the CWMODE is set to 3, you'll get 2 IP address, one for AP & another for Station mode. We need the station IP address for Client mode operation.



- Now type the address 192.168.1.2. The browser sends a HTTP GET request which is displayed on the terminal window where ESP module is connected. You will get 'OK' as response.
- From within ESP window type AT+CIPSEND=0,5. ESP will wait till you type 5 characters.



- On terminal window the characters are received. You get these inside ESP window +IPD,0,5:HELLO. 0 is the channel of communication & 5 is the number of characters received. Use the channel number which is displayed as CONNECT while a request is received from the browser.



## Communication with a web page

To communicate with web page the ESP must be a server. Now Open Web Browser and type the station IP address of ESP module

**OK**

**Link**

**+IPD,0,278:GET / HTTP/1.1**

**Host: 192.168.1.4**

**User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:39.0)**

**Gecko/20100101 Firefox/39.0**

**Accept:**

**text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8**

**Accept-Language: en-US,en;q=0.5**

**Accept-Encoding: gzip, deflate**

**Connection: keep-alive**

**OK**

After getting > Response send message. Now the characters are sent to the Web browser. But the browser is still waiting for the channel to be closed. To view output the channel should be closed. AT+CIPCLOSE=0 command is used to close the channel. Here 0 is the channel ID. Now, you can communicate from your web browser, your phone, laptop, or other Internet capable device to control your projects. But all connections should be from the same network (ie, from same router). So how can you access your WiFi from outside world, from other network. In order to do so we should forward our ESP port to the internet which is completely a routers task.

[ESP-01](#)[ESP8266](#)[ESP8266 Serial WiFi Module](#)[WiFi](#)

Did you like this article? Share it with your friends!

Like 0

Tweet G+1 0

## 5 Responses to "ESP8266 Serial WiFi Module (ESP-01)"

**Ahmed Osman** says:

December 18, 2015 at 7:41 pm

Your wifi password is "dnschanger" xD 😊  
Edit the 7th photo :v



Reply

**Ahmed Osman** says:

December 18, 2015 at 7:44 pm

Nice tutorial, it was very useful for me  
Thank you 😊



Reply

**Hari Krishnan** says:

December 19, 2015 at 9:07 am

Thank you for your valuable feedback! 😊 More of these  
are available here.



Reply



---

**The Emperor** says:

December 29, 2015 at 5:35 am



Cool stuff,  
Think I should buy one.  
How can I get hold of one?

Reply

---

**Keshav** says:

April 22, 2016 at 6:02 pm



After Interfacing Arduino UNO with ESP8266, i am able to send the characters from ESP8266 module to webpage by using AT commnads i.e AT+CIPSEND=(channel number),(number of characters). Now i want to turn ON/OFF an LED from webpage . I heard we need to connect to server, So i installed WAMP server software in my windows operating system and tried connecting with AT command AT+CIPSTART="TCP","my local sytem IP",80,but it is not connecting. Can you suggest me the correct procedure.

Thanks

Reply

---

## Leave a Reply

Your email address will not be published. Required fields are marked \*



Comment

You may use these HTML tags and attributes: <a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite=""> <cite> <code class="" title="" data-url=""> <del datetime=""> <em> <i> <q cite=""> <strike> <strong> <pre class="" title="" data-url=""> <span class="" title="" data-url="">

Post Comment

---

© 2016 rhydoLABZ-wiki