

**PINS NOTATIONS:**

GPIO-General Purpose Input Output

SDIO-Secure Digital Input Output

SPI-Serial Periphery Interphase

UART-Universal Asynchronous Receiver Transmitter

PWM-Pulse Width Modulation

ADC-Analog to Digital Converter

**SPECIFICATION**:

• Voltage:3.3V.

• Wi-Fi Direct (P2P), soft-AP.

• Current consumption: 10uA~170mA.

• Flash memory attachable: 16MB max (512K normal).

• Integrated TCP/IP protocol stack.

• Processor: Tensilica L106 32-bit.

• Processor speed: 80~160MHz.

• RAM: 32K + 80K.

• GPIOs: 17 (multiplexed with other functions).

• Analog to Digital: 1 input with 1024 step resolution.

• +19.5dBm output power in 802.11b mode

• 802.11 support: b/g/n.

• Maximum concurrent TCP connections: 5.

| **Commands** | **Description** | **Set/Execute** | **Parameters** |
| --- | --- | --- | --- |
| AT+RST | restart the module | – | – |
| AT+CWMODE | wifi mode | AT+CWMODE=<mode> | 1= Sta, 2= AP, 3=both |
| AT+CWJAP | join the AP | AT+ CWJAP =<ssid>,< pwd > | ssid = ssid, pwd = wifi password |
| AT+CWLAP | list the AP | AT+CWLAP |  |
| AT+CWQAP | quit the AP | AT+CWQAP |  |
| AT+ CWSAP | set the parameters of AP | AT+ CWSAP= <ssid>,<pwd>,<chl>, <ecn> | ssid, pwd, chl = channel, ecn = encryption |
| AT+ CIPSTATUS | get the connection status | AT+ CIPSTATUS |  |
| AT+CIPSTART | set up TCP or UDP connection | 1)single connection (+CIPMUX=0) AT+CIPSTART= <type>,<addr>,<port>; 2) multiple connection (+CIPMUX=1) AT+CIPSTART= <id><type>,<addr>, <port> | id = 0-4, type = TCP/UDP, addr = IP address, port= port |
| AT+CIPSEND | send data | 1)single connection(+CIPMUX=0) AT+CIPSEND=<length>; 2) multiple connection (+CIPMUX=1) AT+CIPSEND= <id>,<length> |  |
| AT+CIPCLOSE | close TCP or UDP connection | AT+CIPCLOSE=<id> or AT+CIPCLOSE |  |
| AT+CIFSR | Get IP address | AT+CIFSR |  |
| AT+ CIPMUX | set mutiple connection | AT+ CIPMUX=<mode> | 0 for single connection 1 for mutiple connection |
| AT+ CIPSERVER | set as server | AT+ CIPSERVER= <mode>[,<port> ] | mode 0 to close server mode, mode 1 to open; port = port |
| +IPD | received data |  |  |

\*it operates on 3.3v thus to convert the 5volts from the usb jack there is 3.3v voltage regulator

**NODEMCU:**

**NodeMCU** is an open source IoT platform.It includes firmware which runs on the ESP8266 [Wi-fi](https://en.wikipedia.org/wiki/Wi-Fi)SoC from [Espressif Systems](https://en.wikipedia.org/w/index.php?title=Espressif_Systems&action=edit&redlink=1" \o "Espressif Systems (page does not exist)), and hardware which is based on the ESP-12 module. The term "NodeMCU" by default refers to the firmware rather than the dev kits. The firmware uses the [Lua](https://en.wikipedia.org/wiki/Lua_(programming_language)) scripting language. It is based on the eLua project,and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson, and spiffs.

It has a library which is the library of esp module “ #include<ESP8266WiFi.h>”

**Used Functions:**

***Serial.begin(\_baud\_rate):***

This is usually to set a communication bridge between Arduino and computer.it is the rate at which the data is circulated between the Arduino and computer.In simple words parameter passed in the function is in bits per seconds.

Serial monitor on Arduino IDE has a set of baud rates which Arduino can handel. If both the Arduino and serial monitor have same operating baud rate data transmission takes place properly and data is displayed.

***WiFi.begin(pointer1, pointer2):***

2 pointers are declared in the global as character type like ssid and password. These are then passed to the function.

This ssid and password are the name of the wifi and the password of the wifi.

WiFi.begin(ssid,password) checks the ssid and the password and sends the true(1) or flase(0) to the WiFi.status() which detects and declares the status of the Nodemcu as connected or disconnected as the output from Begin function.

***WiFi.status():***

It receives the values from the WiFi.begin(ssid,password),as 1 and 0 and send the output as WL\_CONNECTED and vica versa.

It is basically the status of nodemcu connection with the wifi.