TO CONTROL LIGHTS USING Wi-Fi MODULE

The ESP8266 WiFi Module is a self contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. The ESP8266 is capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor. Each ESP8266 module comes pre-programmed with an AT command set firmware, meaning, you can simply hook this up to your Arduino device and get about as much WiFi-ability as a WiFi Shield offers

Thing we will need

Esp 8266

* Perfboard
* Arms1117
* FTDI
* Arduino uno
* Headers
* Breadboard

**ESP8266 Pinouts**

Vcc -power supply 3.3 v

Gnd - Ground

RST- reset

CH-PD-Chip power Down- Connected to 3.3V

Rx-Recieve

Tx - Transmit

GPIO 0- Normal mode for HIGH Programming Mode for LOW

GPIO 2 General purpose 2

**Connections**

Vcc and ch\_pd to vcc. Ground to ground. Rst to button . GPIO0 to button 2. The led is connected to the GPIO 2.

Tx is connected to Rx.

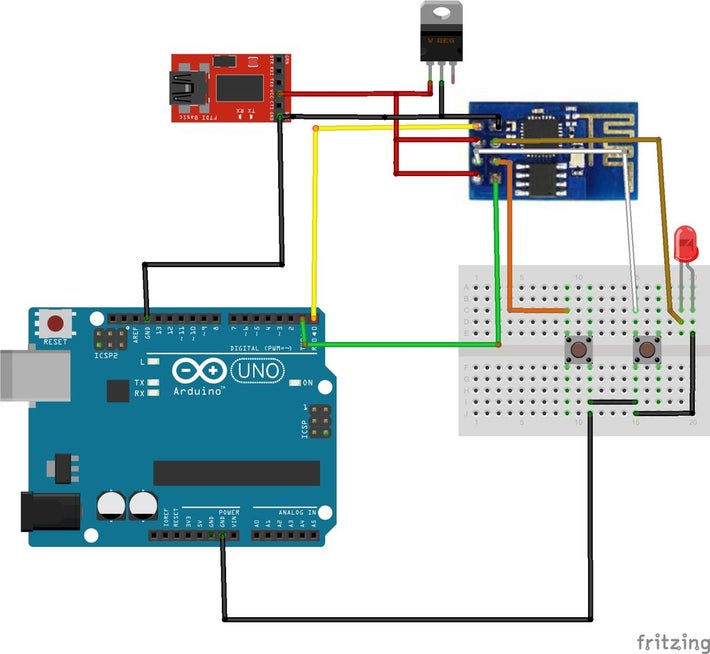
Rx is connected to Tx.

Vcc to 3.3v.

ground to ground.

Ch\_pd to 3.3v.

GPIO 0 to gnd while programming.



**Making a Web Controlled Home Automation**

**Connections**

The led is connected to the GPIO 2

Tx is connected to Rx  
Rx is connected to Tx

Vcc to 3.3v gnd to gnd

Ch\_pd to 3.3v

GPIO 0 to gnd while programming

CODE :

<p>#include <ESP8266WiFi.h><br></p><p>const char\* ssid = "dlink";

const char\* password = "ilovechips";</p><p>int ledPin = 2; // GPIO2

WiFiServer server(80);</p><p>void setup() {

Serial.begin(115200);

delay(10);</p><p>pinMode(ledPin, OUTPUT);

digitalWrite(ledPin, LOW);</p><p>// Connect to WiFi network

Serial.println();

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);</p><p>WiFi.begin(ssid, password);</p><p>while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");</p><p>// Start the server

server.begin();

Serial.println("Server started");</p><p>// Print the IP address

Serial.print("Use this URL to connect: ");

Serial.print("http://");

Serial.print(WiFi.localIP());

Serial.println("/");</p><p>}</p><p>void loop() {

// Check if a client has connected

WiFiClient client = server.available();

if (!client) {

return;

}</p><p>// Wait until the client sends some data

Serial.println("new client");

while(!client.available()){

delay(1);

}</p><p>// Read the first line of the request

String request = client.readStringUntil('\r');

Serial.println(request);

client.flush();</p><p>// Match the request</p><p>int value = LOW;

if (request.indexOf("/LED=ON") != -1) {

digitalWrite(ledPin, HIGH);

value = HIGH;

}

if (request.indexOf("/LED=OFF") != -1) {

digitalWrite(ledPin, LOW);

value = LOW;

}</p><p>// Set ledPin according to the request

//digitalWrite(ledPin, value);</p><p>// Return the response

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println(""); // do not forget this one

client.println("");

client.println("

");</p><p>client.print("Led pin is now: ");</p><p>if(value == HIGH) {

client.print("On");

} else {

client.print("Off");

}

client.println("<br><br>");

client.println("Click <a>here</a> turn the LED on pin 2 ON<br>");

client.println("Click <a>here</a> turn the LED on pin 2 OFF<br>");

client.println("</p><p>");</p><p>delay(1);

Serial.println("Client disonnected");

Serial.println("");</p><p>}</p>