Controlling of LED using ESP8266 WiFi module

N Jagadeeswar[1], Gautam Kumar[2],and G V V Sharma[*]

CONTENTS

Abstract—This manual pinovides an introduction to ESP8266 wifi module. This WiFi module acts as a station mode, Access point mode and both at a time.
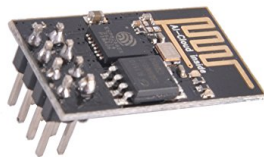
Fig. 1: ESP8266 Module

- The ESP8266 module consists of 8 pins which are arranged in 4x2 order. The labeling of these pins are as shown in below table

1 ESP8266

TX	GND
CH$_P$D	GPIO 2
RST	GPIO 0
Vcc	RX

2 CONNECTIONS WITH ARDUINO

Arduino	ESP8266
Vcc	Vcc & CH-PD
GND	GPIO 0 & GND
TX	TX
RX	RX

- Remove the ATMEGA328 chip from Arduino before uploading the program.
- Connect the GPIO 0 pin to GND during flashing the code and remove the GPIO 0 pin after flashing. We can use GPIO 2 pin for LED.
- Use 3.3 volts for Vcc

3 INSTRUCTIONS BEFORE UPLOADING THE PROGRAM

- Open Arduino IDE software.
- Go to File, At the bottom you will find preferences. Paste the following link and then click on OK.link http://arduino.esp8266.com/stable/package_esp8266com_index.json
- Go to the the tools click on Board at the top you will find Board manager click on it.
- In Board manager you will find lot of Boards but we have to install ESP8266 Board. Search for ESP8266 and install it.
- After installing the ESP8266 Board , Go to tools again and click on Board select the 'Generic ESP8266 Module' Board.
- Since we have to upload the program into the the ESP8266 Board we installed the ESP8266 Board. Here we use Arduino for power supply ,TX and RX pins. We will not use any Digital pins.
- After completion of all above settings upload the following code into the Arduino.
- Before uploading the code Add the required libraries mentioned in the program.

4 PROGRAM

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
```

```cpp
#include <ESP8266WiFiMulti.h>
#include <ESP8266mDNS.h>
#include <ESP8266WebServer.h>

ESP8266WiFiMulti wifiMulti;
ESP8266WebServer server(80);

const int led = 2;

void handleRoot();
void handleLED();
void handleNotFound();

void setup(void){
  Serial.begin(115200);
  delay(10);
  Serial.println('\n');

  pinMode(led, OUTPUT);

  wifiMulti.addAP("motorola", "
      gautam1234");


  Serial.println("Connecting ...")
      ;
  int i = 0;
  while (wifiMulti.run() !=
    WL_CONNECTED) {
    delay(250);
    Serial.print('.');
  }
  Serial.println('\n');
  Serial.print("Connected to ");
  Serial.println(WiFi.SSID());
  Serial.print("IP address:\t");
  Serial.println(WiFi.localIP());

  if (MDNS.begin("esp8266")) {
    Serial.println("mDNS responder
       started");
  } else {
    Serial.println("Error setting
      up MDNS responder!");
  }

  server.on("/", HTTP_GET,
    handleRoot);
  server.on("/LED", HTTP_POST,
    handleLED);
```

```cpp
  server.onNotFound(handleNotFound
      );

  server.begin();
  Serial.println("HTTP server
      started");
}

void loop(void){
  server.handleClient();
}

void handleRoot() {
  server.send(200, "text/html", "<
      form action=\"/LED\" method=\"
      POST\"><input type=\"submit\"
      value=\"Toggle LED\"></form>")
      ;
}

void handleLED() {
  digitalWrite(led,!digitalRead(
      led));
  server.sendHeader("Location","/"
      );
  server.send(303);
}

void handleNotFound(){
  server.send(404, "text/plain", "
      404: Not found");
}
```

- Before uploading the program give your mobile hotspot NAME and PASSWORD. In this program i have given 'motorola' as hotspot name password as 'gautam1234'.
- After uploading the program remove GPIO 0 from GND.
- check serial moniter you will find "connecting............." . After conncting serial moniter will display "mDNS responder started" "HTTP server started" and WiFi IP Address will display like "192.168.43.161".
- Paste the generated IP in your browser and browse you will find "Toggle LED" on the browser.
- click on "Toggle LED" to ON and OFF the LED.

References

[1] program https://tttapa.github.io/ESP8266/Chap10%20-%20Simple%20Web%20Server.html.