

DATA COMMUNICATION

LAB 6: NODE MCU – SERVER MODE (AP MODE)

I. Introduction to Working Mode in ESP8266 Series

The ESP8266 or almost every Wifi modules usually have three working modes: Station, AP (Access Point) and Dual mode. The Station mode is similar to the computers, cell phones or tablets while the AP mode is playing a role of the routers in our network. In a combination, the dual mode allows the Wifi module to work in both Station and AP mode. The ESP8266, ESP8285 and ESP32 module fully support three modes mentioned above.

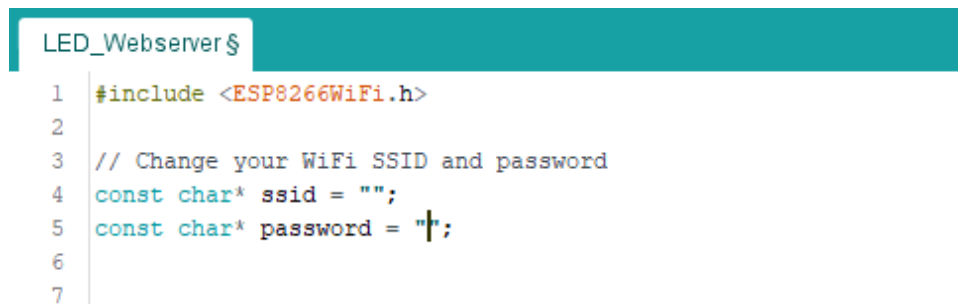
A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Dedicated computers and appliances may be referred to as Web servers as well.

II. Getting Started with Examples

Download an example project from this link

https://www.dropbox.com/s/9od6hox6cn7073f/LED_Webserver.ino?dl=0

Open the LED_Webserver.ino then change the wifi ssid and password according to your wifi network.

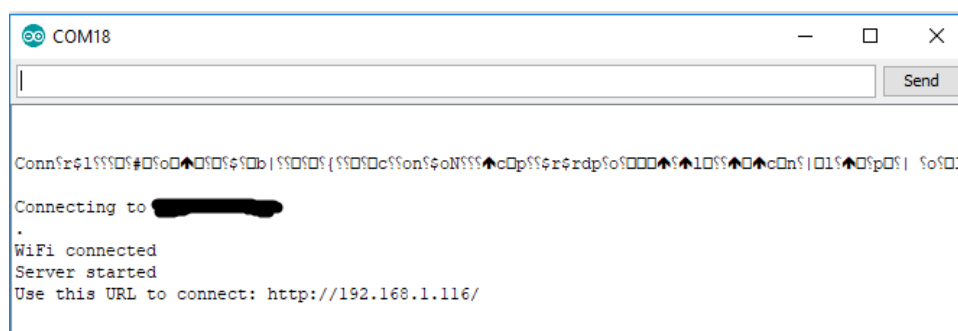


```
LED_Webserver$  
1  #include <ESP8266WiFi.h>  
2  
3  // Change your WiFi SSID and password  
4  const char* ssid = "";  
5  const char* password = "1";  
6  
7
```

Figure 1: Provide the SSID and Password of the wifi

You can also change the LED pin in case you don't want to use the LED on board (GPIO 2)

Then upload the program to your NodeMCU and open the Serial console.



```
COM18  
Connecting to [REDACTED]  
.  
WiFi connected  
Server started  
Use this URL to connect: http://192.168.1.116/
```

Figure 2: Server is running at a local address

Open your browser and enter the URL (local ip address) displayed in the Serial console, you will see the following website (see Figure 3)

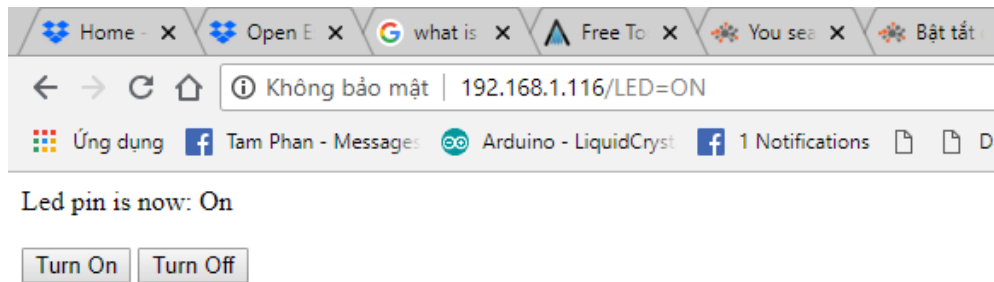


Figure 3: Example of LED demo

If you press the “Turn On” button, the LED on the NodeMCU will turn on, and if you press “Turn Off”, the LED will turn off. You can also look at the following URL for a better web interface: <https://hocarm.org/dieu-khien-den-hoc-quia-website-voi-esp8266/>

III. Exercise

Write a program that update the sensor values (or any kind of value you want) to the NodeMCU webserver. You can use the following URLs for references: <https://www.teachmemicro.com/display-sensor-data-nodemcu-web-server/> and <https://circuits4you.com/2018/02/04/esp8266-ajax-update-part-of-web-page-without-refreshing/>

1. Kết nối node MCU với wifi bằng hàm Wifi.begin, đồng thời kiểm tra kết nối được thiết lập hay chưa.

```
1. WiFi.begin(ssid, password); // ssid = Wifi Name    password = password Wifi
2. while (WiFi.status() != WL_CONNECTED) {
3.     delay(500);
4.     Serial.print(".");
5. }
```

2. Khởi tạo port mà server sẽ nghe. Mặc định là số 80. Khi URL được nhập, nó sẽ “trigger” hàm được thực thi

```
1. ESP8266WebServer server(80);
2. void handleRoot() {
3.     String s = MAIN_page; //Read HTML contents
4.     server.send(200, "text/html", s); //Send web page
5. }
6. server.on("/", handleRoot);
```

Khi HTTP request nhận “/” thì hàm handleRoot sẽ được thực thi. Hàm này sẽ tải lên nội dung trong Mainpage được lưu trong file chứa nội dung định dạng html.

3. Để thực thi việc bật/ tắt LED và cập nhật giá trị sensor, ta thiết lập 2 hàm là handleLED và handleADC

```

1. void handleLED() {
2.   String ledState = "OFF";
3.   String t_state = server.arg("LEDstate"); //Nhận trạng thái gửi từ Webserver
4.   Serial.println(t_state);
5.   if(t_state == "1")
6.   {
7.     digitalWrite(LED,LOW); //LED ON
8.     ledState = "ON";
9.   }
10.  else
11.  {
12.    digitalWrite(LED,HIGH); //LED OFF
13.    ledState = "OFF";
14.  }
15.
16.  server.send(200, "text/plain", ledState); //Gửi trạng thái lên Webserver
17. }

```

Hàm handleLED có chức năng nhận tín hiệu gửi từ webserver về nodeMCU và từ nodeMCU gửi lên Webserver.

```

1. void handleADC() {
2.   int a = analogRead(A0);
3.   String adcValue = String(a);
4.
5.   server.send(200, "text/plain", adcValue); //Gửi giá trị sensor lên Webserver
6. }

```

Hàm handle ADC có chức năng gửi dữ liệu sensor từ NodeMCU lên Webserver để hiển thị.

- Để xử lý các request gửi đến, cần hàm handleClient() đặt trong vòng loop

```

1. void loop(void){
2.   server.handleClient();
3. }

```

- Viết file HTML để nhận và gửi giá trị cho NodeMCU (sử dụng Ajax). Hàm xử lý truyền nhận dữ liệu gồm getData() và sendData()

```

1. function getData() {
2.   var xhttp = new XMLHttpRequest();
3.   xhttp.onreadystatechange = function() {
4.     if (this.readyState == 4 && this.status == 200) {
5.       document.getElementById("ADCValue").innerHTML =
6.         this.responseText;
7.     }
8.   };
9.   xhttp.open("GET", "readADC", true);
10.  xhttp.send();
11. }

```

```

1. function sendData(led) {
2.   var xhttp = new XMLHttpRequest();
3.   xhttp.onreadystatechange = function() {
4.     if (this.readyState == 4 && this.status == 200) {
5.       document.getElementById("LEDState").innerHTML =
6.         this.responseText;
7.     }
8.   };
9.   xhttp.open("GET", "setLED?LEDstate="+led, true);
10.  xhttp.send();
11. }

```

The ESP8266 NodeMCU

ADC Value is : 1
 LED State is : ON

IV. Extra Exercise

Write a program that connect the NodeMCU to an online server (using MQTT or POST/GET request e.g) then display the current time to your webserver (it is expected that you display time of various countries). You can get the time from this URL: <http://www.iforce2d.net/test.php>

1. Sử dụng giao thức HTTP để lấy dữ liệu từ URL: <http://www.iforce2d.net/test.php>

```

1. HTTPClient http;
2. http.begin("http://www.iforce2d.net/test.php");
3. int httpCode = http.GET();
4. if (httpCode > 0) {
5.   payload = http.getString();
6.   Serial.println(payload);
7. }
8. http.end(); //Close connection

```

Dữ liệu nhận được là chuỗi String chứa thông tin ngày tháng năm và thời gian của GIỜ QUỐC TẾ (Ví dụ Current UTC: 16 Nov 2018 2:51:32 PM).

2. Để hiển thị giờ của Việt Nam (Việt Nam ở mốc giờ +7) thì phải cộng thêm 7 vào số Hour của string nhận được

```

1. if(strH.toInt()<=57){           //strH chứa số chỉ của GIỜ
2.   int strInt=strH.toInt()+7;    //strInt chứa số int của GIỜ
3.   strH=String(strInt);         //Chuyển sang String
4.   strTime=strH+temp;           //strTime là chuỗi chứa HH:MM:SS
5. }
6. else{
7.   int strInt=strH.toInt()+7;
8.   strH=String(strInt);
9.   strTime=strH+": "+temp;
10. }

```

3. Tương tự như bài trên, gồm các bước thiết lập server, nội dung hiển thị và cuối cùng là truyền dữ liệu để hiển thị:

```

1. server.on("/timeVN",[](){
2.   server.send(200,"text/plain",strTime);
3. });

```

CURRENT TIME IN:

Viet Nam: 10:09:08