**Để coding cho ESP 8226 cần : xây dựng IDE**

1.tải Arduino IDE :

Tải tệp này **arduino-1.8.13-windows.zip tại**

[**https://taimienphi.vn/download-arduino-21997/1813-phien-ban**](https://taimienphi.vn/download-arduino-21997/1813-phien-ban)

**2. Cài đặt tại đây :**

[**ESP8266 kết nối Internet - Phần 1: Cài đặt ESP8266 làm một socket client kết nối tới socket server trong mạng LAN | Cộng đồng Arduino Việt Nam**](http://arduino.vn/bai-viet/1496-esp8266-ket-noi-internet-phan-1-cai-dat-esp8266-lam-mot-socket-client-ket-noi-toi)

[**http://arduino.esp8266.com/stable/package\_esp8266com\_index.json**](http://arduino.esp8266.com/stable/package_esp8266com_index.json)

**3.Tạo WEB server http với code sau :**

**Các bước :**

1. **Đăng kí để kết nối wifi tới router**
2. **Tạo một http server trên bo mạch ESP8266**
3. **Tạo trang WEB bằng HTML tương tác để nhận client và trả lời**
4. **Dịch code và tải (nạp) vào bo mạch, chạy luôn. Dùng monitor để giám sát (vào monitor :Tools->Serial monitor. Từ đây sẽ nhận được địa chỉ IP của WEB server tạo trên ESP8266.**

**Code :**

**//Tạo một WEB server trên ESP8266, kết nối không dây**

**// Load Wi-Fi library**

**#include <ESP8266WiFi.h>**

**// Replace with your network credentials-ĐĂNG KÍ ROUTER gia đình**

**const char\* ssid     = "htc";   //SSID của router để ESP8266 WEB server kết nối**

**const char\* password = "ducngoc-15386";**

**// Set web server port number to 80**

**WiFiServer server(80);                //hàm tạo server, cổng chuẩn 80**

**// Variable to store the HTTP request**

**String header;**

**// Auxiliar variables to store the current output state**

**String output5State = "off";**

**String output4State = "off";**

**// Assign output variables to GPIO pins**

**const int output5 = 5;**

**const int output4 = 4;**

**// Current time**

**unsigned long currentTime = millis();**

**// Previous time**

**unsigned long previousTime = 0;**

**// Define timeout time in milliseconds (example: 2000ms = 2s)**

**const long timeoutTime = 2000;**

**//Hàm setup() chạy 1 lần duy nhất**

**void setup()**

**{**

**//set baurate của công ghép nối với máy chủ**

**Serial.begin(115200);**

**// Initialize the output variables as mode outputs**

**pinMode(output5, OUTPUT);**

**pinMode(output4, OUTPUT);**

**// Set outputs to LOW**

**digitalWrite(output5, LOW);**

**digitalWrite(output4, LOW);**

**// Connect to Wi-Fi network with SSID and password**

**Serial.print("Connecting to ");**

**Serial.println(ssid);**

**WiFi.begin(ssid, password); //kêt nối tới router**

**while (WiFi.status() != WL\_CONNECTED)**

**{**

**delay(500);**

**Serial.print(".");**

**}**

**// Print local IP address and start web server ra Monitor**

**Serial.println("");**

**Serial.println("WiFi connected.");**

**Serial.println("IP address: "); //thông báo IP cho user để vào WEB**

**Serial.println(WiFi.localIP());**

**server.begin();**

**}**

**//Trong loop() chúng ta lập trình sự kiện xảy ra khi có một kết nối được tạo**

**void loop()**

**{**

**WiFiClient client = server.available();   // ESP server nghe request-//Listen for incoming clients**

**//Khi có một kết nối tới, ESP sẽ lưu các thông tin lại.**

**//Các lệnh thực hiện trong while() sẽ tiếp tục đến khi Client mất kết nối.**

**if (client)**

**{                             // If a new client connects,**

**Serial.println("New Client.");          // print a message out in the serial port**

**String currentLine = "";                // make a String to hold //incoming data from the client**

**currentTime = millis();**

**previousTime = currentTime;**

**while (client.connected() && currentTime - previousTime <= timeoutTime)**

**{**

**// loop while the client's connected**

**currentTime = millis();**

**if (client.available()) {             // if there's bytes to read from the client,**

**char c = client.read();             // read a byte, then**

**Serial.write(c);                    // print it out the serial monitor**

**header += c;**

**if (c == '\n') {                    // if the byte is a newline character**

**// if the current line is blank, you got two newline characters in a row.**

**// that's the end of the client HTTP request, so send a response:**

**if (currentLine.length() == 0) {**

**// HTTP headers always start with a response code (e.g. HTTP/1.1 //200 OK)**

**// and a content-type so the client knows what's coming, then a blank line:**

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

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

**client.println("Connection: close");**

**client.println();**

**//Next điều khiển chân GPIO tưng ứng với nút được nhấn.**

**// turns the GPIOs on and off**

**if (header.indexOf("GET /5/on") >= 0) {**

**Serial.println("GPIO 5 on");**

**output5State = "on";**

**digitalWrite(output5, HIGH);**

**} else if (header.indexOf("GET /5/off") >= 0) {**

**Serial.println("GPIO 5 off");**

**output5State = "off";**

**digitalWrite(output5, LOW);**

**} else if (header.indexOf("GET /4/on") >= 0) {**

**Serial.println("GPIO 4 on");**

**output4State = "on";**

**digitalWrite(output4, HIGH);**

**} else if (header.indexOf("GET /4/off") >= 0) {**

**Serial.println("GPIO 4 off");**

**output4State = "off";**

**digitalWrite(output4, LOW);**

**}**

**//Và sau đó lưu lại trạng thái nút nhấn để phản hồi với Client.**

**//Now server page HTML**

**//Để hiên thị trên trình duyệt ESP8266 sẽ gửi các Response dưới dạng chỗi HTML.**

**//Trang web sẽ được gửi tới Client dùng hàm client.println().**

**// Display the HTML web page**

**//Trang web luôn bắt đầu với thẻ: <!DOCTYPE html><html>**

**client.println("<!DOCTYPE html><html>");**

**client.println("<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1\">");**

**client.println("<link rel=\"icon\" href=\"data:,\">");**

**//Sử dụng Style trên Web Server. Khai báo Style cho button: màu sắc, kích thước...**

**// CSS to style the on/off buttons**

**// Feel free to change the background-color and font-size attributes to fit your preferences**

**client.println("<style>html { font-family: Helvetica; display: inline-block; margin: 0px auto; text-align: center;}");**

**client.println(".button { background-color: #195B6A; border: none; color: white; padding: 16px 40px;");**

**client.println("text-decoration: none; font-size: 30px; margin: 2px; cursor: pointer;}");**

**client.println(".button2 {background-color: #77878A;}</style></head>");**

**// Web Page Heading**

**client.println("<body><h1>ESP8266 Web Server</h1>");**

**//Sau khi nhấn, trạng thái các nút nhấn thay đổi sẽ được lưu lại và được...**

**// Display current state, and ON/OFF buttons for GPIO 5**

**client.println("<p>GPIO 5 - State " + output5State + "</p>");**

**//…phản hồi tới Client**

**// If the output5State is off, it displays the ON button**

**if (output5State=="off") {**

**client.println("<p><a href=\"/5/on\"><button class=\"button\">ON</button></a></p>");**

**} else {**

**client.println("<p><a href=\"/5/off\"><button class=\"button button2\">OFF</button></a></p>");**

**}**

**//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**//đây là đoạn code tạo ra 1 giao diện nút nhấn,**

**//để tạo thêm nút chỉ cần copy và thay đổi tên nút mới.**

**//thêm các biến cho nút thứ đó như trên !**

**//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**// Display current state, and ON/OFF buttons for GPIO 4**

**client.println("<p>GPIO 4 - State " + output4State + "</p>");**

**// If the output4State is off, it displays the ON button**

**if (output4State=="off") {**

**client.println("<p><a href=\"/4/on\"><button class=\"button\">ON</button></a></p>");**

**} else {**

**client.println("<p><a href=\"/4/off\"><button class=\"button button2\">OFF</button></a></p>");**

**}**

**client.println("</body></html>");**

**//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**// The HTTP response ends with another blank line**

**client.println();**

**// Break out of the while loop**

**break;**

**} else { // if you got a newline, then clear currentLine**

**currentLine = "";**

**}**

**} else if (c != '\r') {  // if you got anything else but a carriage return character,**

**currentLine += c;      // add it to the end of the currentLine**

**}**

**}**

**}**

**//Ngắt kết nối đến Web Server. Khi Respone kết thúc,**

**//ESP ngắt kết nối và xóa các biến lưu trữ.**

**// Clear the header variable**

**header = "";**

**// Close the connection**

**client.stop();**

**Serial.println("Client disconnected.");**

**Serial.println("");**

**}**

**}**

**Kết quả chạy :**

1. **Trên Monitor của IDE phát triển sẽ thấy :**

**..............**

**WiFi connected.**

**IP address:**

**172.16.10.109**