**Internet of Things**

**Lab Report 5**

**Hafiz Ahmad**

**19l-1316**

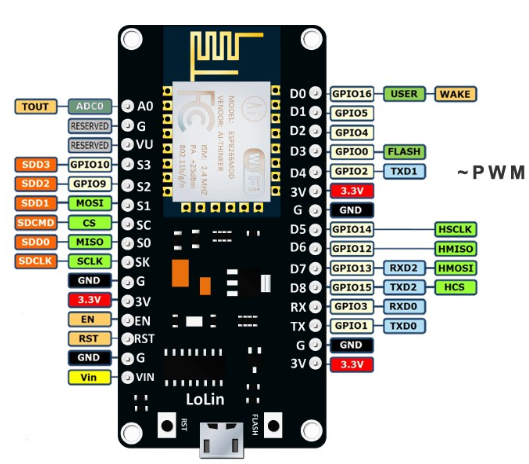
**Section-7A2**

**Introduction to ESP8266 and Wireless Communication**

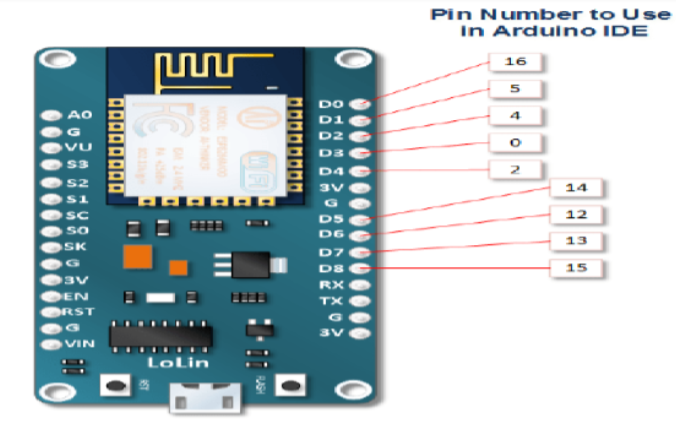
**INTRODUCTION:**

One of the most widely used ESP8266 development boards is the ESP12-E NodeMCU Kit.It has access to 11 GPIO pins, 4 MB of flash memory, and one analog-to-digital converter (ADC) pin with a resolution of 10 bits.The board has a voltage regulator built in, and either the Vin pin or the mini USB socket can be used to power the module.

The board has a built-in usb-to-serial converter, so you don't need an FTDI programmer to upload code to it. Uploading code to the board is the same as uploading code to the Arduino.In our Wi-Fi and Internet of Things projects, this board is used more frequently.It is great for beginners and has a lot of uses.Therefore, this module is an excellent option if this is our first time working with the ESP8266.



For programming this module, we make use of the Arduino IDE software.It is essential to keep in mind that the pin arrangement that appears on the board is distinct from the arrangement that we use to program the board in the software. For instance, if we write code that targets pin 16 in the Arduino IDE, it will actually assist us in arranging the communication with the module's D0 pin.



**OBJECTIVES:**

To learn Node MCU a local web server and then perform tasks.

To learn about Node MCU as Access Point.

**In Lab:**

#include <ESP8266WiFi.h>//We need it or compilers oxygen tank will explode.

const char\* ssid = "ahmad home";

const char\* password = "ahmad@123";

int ledPin = BUILTIN\_LED;

WiFiServer server(80);

void setup() {

pinMode(ledPin,OUTPUT);

digitalWrite(ledPin,LOW);//We start with the led turned off.

Serial.begin(115200);//remember to set the serial monitor to 115200 or you will see a lot of junk text.

Serial.println();

Serial.print("Wifi connecting to ");

Serial.println( ssid );//your ssid

WiFi.begin(ssid,password);//connecting to wifi

Serial.println();

Serial.print("Connecting");//connecting

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

delay(500);

Serial.print(".");

}

Serial.println();

Serial.println("Wifi Connected Success!");//If network connection is success we will see these messages

Serial.print("NodeMCU IP Address : ");

Serial.println(WiFi.localIP() );

server.begin();

Serial.println("NodeMCU Server started");//server has started

// Print the IP address

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

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

Serial.print(WiFi.localIP());

Serial.println("/");

}

void loop() {

// Check if a client has connected

WiFiClient client = server.available();

if (!client) {

return;

}

// Wait until the client sends some data

Serial.println("Hello New client");//Come on, we also have to welcome the client, but the client will not able to see it.

while(!client.available()){

delay(1);

}

// Read the first line of the request

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

Serial.println(request);

client.flush();

// Match the request

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;

}

// Set ledPin according to the request

//digitalWrite(ledPin, value);

// 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("<!DOCTYPE HTML>");

client.println("<html>");

client.println("<body>");

client.println("<h1>LED Control</h1>");

client.println("<p>Made by Qasim Hayee and Umair Raza</p>");

client.println("<br>");

if(value==LOW)

client.println("<a href=\"/LED=ON\"\"><button>Turn Off </button></a>");

else

client.println("<a href=\"/LED=OFF\"\"><button>Turn On </button></a>");

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

delay(1);

Serial.println("Client disonnected");

Serial.println("");

}

#include <ESP8266WiFi.h>//We need it or compilers oxygen tank will explode.

const char\* ssid = "ahmad home";

const char\* password = "ahmad@123";

int ledPin = BUILTIN\_LED;

WiFiServer server(80);

void setup() {

pinMode(ledPin,OUTPUT);

digitalWrite(ledPin,LOW);//We start with the led turned off.

Serial.begin(115200);//remember to set the serial monitor to 115200 or you will see a lot of junk text.

Serial.println();

Serial.print("Wifi connecting to ");

Serial.println( ssid );//your ssid

WiFi.begin(ssid,password);//connecting to wifi

Serial.println();

Serial.print("Connecting");//connecting

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

delay(500);

Serial.print(".");

}

Serial.println();

Serial.println("Wifi Connected Success!");//If network connection is success we will see these messages

Serial.print("NodeMCU IP Address : ");

Serial.println(WiFi.localIP() );

server.begin();

Serial.println("NodeMCU Server started");//server has started

// Print the IP address

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

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

Serial.print(WiFi.localIP());

Serial.println("/");

}

void loop() {

// Check if a client has connected

WiFiClient client = server.available();

if (!client) {

return;

}

// Wait until the client sends some data

Serial.println("Hello New client");//Come on, we also have to welcome the client, but the client will not able to see it.

while(!client.available()){

delay(1);

}

// Read the first line of the request

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

Serial.println(request);

client.flush();

// Match the request

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;

}

// Set ledPin according to the request

//digitalWrite(ledPin, value);

// 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("<!DOCTYPE HTML>");

client.println("<html>");

client.println("<body>");

client.println("<h1>LED Control</h1>");

client.println("<p>Made by Qasim Hayee and Umair Raza</p>");

client.println("<br>");

client.println("<a href=\"/LED=ON\"\"><button>Turn Off </button></a>");

client.println("<a href=\"/LED=OFF\"\"><button>Turn On </button></a>");

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

delay(1);

Serial.println("Client disonnected");

Serial.println("");

}

#include <ESP8266WiFi.h>//We need it or compilers oxygen tank will explode.

const char\* ssid = "FYPEEFAST-3";

const char\* password = "fypeefast";

int ledPin = 5;

WiFiServer server(80);

void setup() {

pinMode(ledPin,OUTPUT);

digitalWrite(ledPin,LOW);//We start with the led turned off.

Serial.begin(115200);//remember to set the serial monitor to 115200 or you will see a lot of junk text.

Serial.println();

Serial.print("Wifi connecting to ");

Serial.println( ssid );//your ssid

WiFi.begin(ssid,password);//connecting to wifi

Serial.println();

Serial.print("Connecting");//connecting

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

delay(500);

Serial.print(".");

}

Serial.println();

Serial.println("Wifi Connected Success!");//If network connection is success we will see these messages

Serial.print("NodeMCU IP Address : ");

Serial.println(WiFi.localIP() );

server.begin();

Serial.println("NodeMCU Server started");//server has started

// Print the IP address

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

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

Serial.print(WiFi.localIP());

Serial.println("/");

}

void loop() {

// Check if a client has connected

WiFiClient client = server.available();

if (!client) {

return;

}

// Wait until the client sends some data

Serial.println("Hello New client");//Come on, we also have to welcome the client, but the client will not able to see it.

while(!client.available()){

delay(1);

}

// Read the first line of the request

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

Serial.println(request);

client.flush();

// Match the request

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

analogWrite(ledPin, 0.2\*255);

}

if (request.indexOf("/LED=2") != -1) {

analogWrite(ledPin, 0.4\*255);

}

if (request.indexOf("/LED=3") != -1) {

analogWrite(ledPin, 0.6\*255);

}

if (request.indexOf("/LED=4") != -1) {

analogWrite(ledPin, 0.8\*255);

}

if (request.indexOf("/LED=5") != -1) {

analogWrite(ledPin, 1.0\*255);

}

// Set ledPin according to the request

//digitalWrite(ledPin, value);

// 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("<!DOCTYPE HTML>");

client.println("<html>");

client.println("<body>");

client.println("<h1>LED Control</h1>");

client.println("<p>Made by Qasim Hayee and Umair Raza</p>");

client.println("<br>");

client.println("<a href=\"/LED=1\"\"><button>Intensity 1 </button></a>");

client.println("<a href=\"/LED=2\"\"><button>Intensity 2</button></a>");

client.println("<a href=\"/LED=3\"\"><button>Intensity 3 </button></a>");

client.println("<a href=\"/LED=4\"\"><button>Intensity 4 </button></a>");

client.println("<a href=\"/LED=5\"\"><button>Intensity 5 </button></a>");

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

delay(1);

Serial.println("Client disonnected");

Serial.println("");

} #include <ESP8266WiFi.h>

const char WiFiPassword[] = "12345678";

const char AP\_NameChar[] = "LEDControl" ;

WiFiServer server(80);

String header = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n";

String html\_1 = "<!DOCTYPE html><html><head><title>LED Control</title></head><body><div id='main'><h2>LED Control</h2>";

String html\_2 = "<form id='F1' action='LEDON'><input class='button' type='submit' value='LED ON' ></form><br>";

String html\_3 = "<form id='F2' action='LEDOFF'><input class='button' type='submit' value='LED OFF' ></form><br>";

String html\_4 = "</div></body></html>";

String request = "";

int LED\_Pin = BUILTIN\_LED;

void setup()

{

pinMode(LED\_Pin, OUTPUT);

boolean conn = WiFi.softAP(AP\_NameChar, WiFiPassword);

server.begin();

} // void setup()

void loop()

{

// Check if a client has connected

WiFiClient client = server.available();

if (!client) { return; }

// Read the first line of the request

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

if ( request.indexOf("LEDON") > 0 ) { digitalWrite(LED\_Pin, HIGH); }

else if ( request.indexOf("LEDOFF") > 0 ) { digitalWrite(LED\_Pin, LOW); }

client.flush();

client.print( header );

client.print( html\_1 );

client.print( html\_2 );

client.print( html\_3 );

client.print( html\_4);

delay(5);

// The client will actually be disconnected when the function returns and 'client' object is detroyed

} // void loop()

**Application:**

Overview. There are designs for open-source prototyping boards and an open source firmware for the NodeMCU.The words "node" and "MCU" are combined in the name "NodeMCU."The firmware, not the associated development kits, are the true meaning of the term "NodeMCU."Home automation users love NodeMCU.It makes it easier for IoT applications thanks to its WiFi capabilities and support for the Arduino IDE.It has a lot of digital I/O pins, is very small, and can communicate with Serial and I2C.Control of the lights.

HVAC.

Management of gardens and lawns.

Smart devices for the home.

enhanced home security and safety

Monitoring of the water and air quality in the home.

Voice assistants based on natural language.

Improved informational presentation.

**Issues:**

we never find any issue regarding this lab but its take time in solving a complex problem.

**Conclusion:**

In this lab we learn to perform following tasks Node MCU a local web server and then perform tasks and Node MCU as Access Point.