แนวทางการใช้งานอินเทอร์เน็ตของสรรพสิ่งในระบบการผลิต **IoT Approaches to Manufacturing System**	
ชื่อ-สกุล : นายปราชญา ธนพิบูลผล	**รหัสนักศึกษา : B6323059**

4/4. คำถามท้ายบทเพื่อทดสอบความเข้าใจ

Quiz_201 – Web Control 2 LED

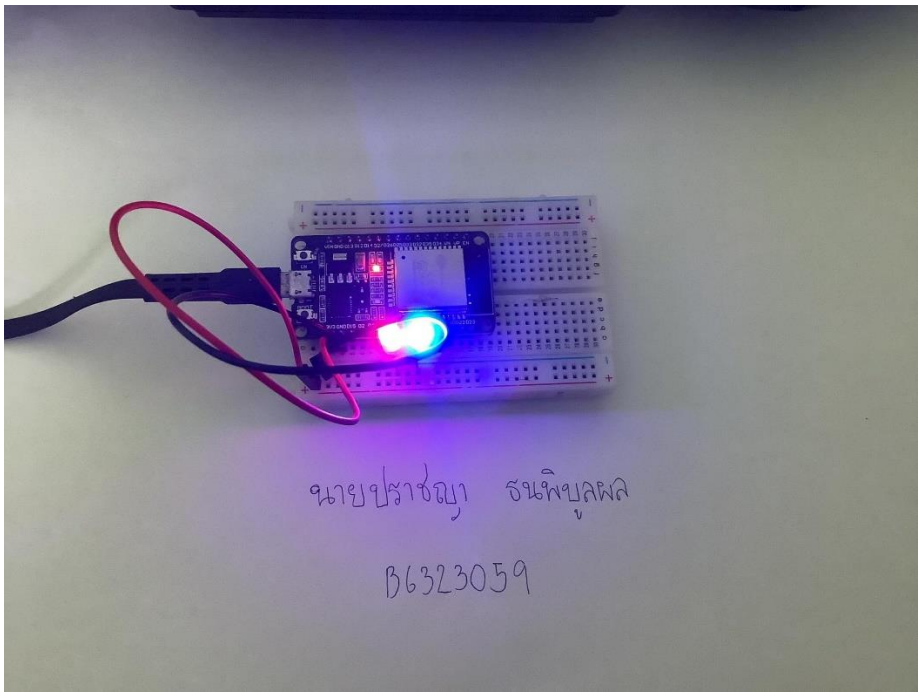
- อยากได้ปุ่มสำหรับคุมปิด-เปิด หลอดไฟ LED 2 ดวง
- https://www.colorhexa.com/008cba?fbclid=IwAR3dIZ_gRgDWmREmnzuknLbMxV3pOHy4YIPuLEz8-ZzTOX2VhWxcH2QjLGk


```
#include <WiFi.h>

const char* ssid = "It's bad day not a bad life";

const char* password = "0641453596";

int pin5Test = 5;

int pin18Test = 18;

WiFiServer server(80);

void setup() {

  Serial.begin(115200);

  pinMode(pin5Test, OUTPUT); // set the LED pin mode

  pinMode(pin18Test, OUTPUT);

  delay(10);

  Serial.print("\n\nConnecting to "); Serial.println(ssid);
```

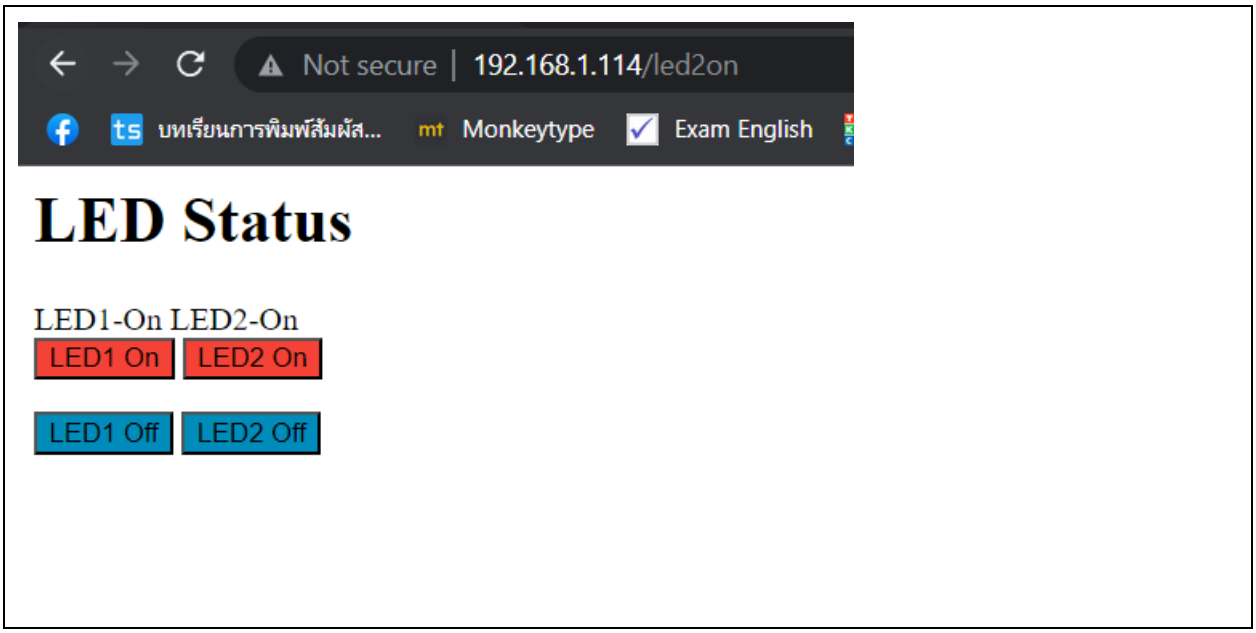
```
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500); Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected."); Serial.println("IP address: ");
  Serial.println(WiFi.localIP()); server.begin();
}
int value = 0;
bool LED1_Status = LOW;
bool LED2_Status = LOW;
void loop() {
  digitalWrite(pin5Test, LED1_Status);
  digitalWrite(pin18Test, LED2_Status);
  WiFiClient client = server.available(); // listen for incoming clients
  if (client) { // if you get a client,
    Serial.println("New Client."); // print a message out the serial port
    String currentLine = ""; // make a String to hold incoming data from the client
    while (client.connected()) { // loop while the client's connected
      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
        if (c == '\n') { // if the byte is a newline character
          if (currentLine.length() == 0) {
```

```
            client.println("HTTP/1.1 200 OK");
            client.println("Content-type:text/html");
            client.println();
            client.println("<html>");
            client.println("<body>");
            client.println("<h1>LED Status</h1>");
            client.println("<p>");
            if (LED1_Status == HIGH)
              client.println("LED1-On");
            else
              client.println("LED1-Off");
            if (LED2_Status == HIGH)
              client.println("LED2-On");
            else
            client.println("LED2-Off");
            //client.println("<a href = \"/ledon\"><button>LED On</button></a>");
            client.println("<br />");
            client.println("<a href=\"/led1on\"><button style = \"background-color: #f44336;\">LED1
On</button></a>");
            client.println("<a href=\"/led2on\"><button style = \"background-color: #f44336;\">LED2
On</button></a>");
            client.println("</p>");
            //client.println("<a href = \"/ledoff\"><button>LED Off</button></a>");
            client.println("<a href=\"/led1off\"><button style = \"background-color: #008CBA;\">LED1
Off</button></a>");
```

```
            client.println("<a href=\"/led2off\"><button style = \"background-color: #008CBA;\">LED2
Off</button></a>");
            client.println("<body>");
            client.println("<br />");
            client.println("<html>");
            break;
          } else {
            currentLine = "";
          }
        } else if (c != '\r') {
          currentLine += c;
        }
        //Led1
        if (currentLine.endsWith("GET /led1on")) LED1_Status = HIGH;
        if (currentLine.endsWith("GET /led1off")) LED1_Status = LOW;
        //Led2
        if (currentLine.endsWith("GET /led2on")) LED2_Status = HIGH;
        if (currentLine.endsWith("GET /led2off")) LED2_Status = LOW;
      }
    }
    client.stop(); // close the connection:
    Serial.println("Client Disconnected.");
  }
}
```

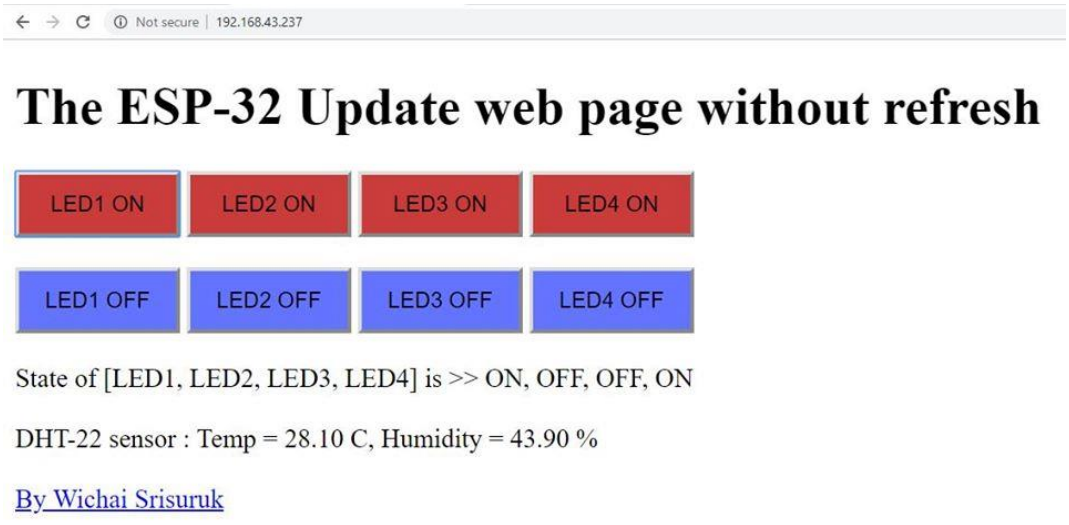

นายปราชญา ธนพิบูลผล
B6323059


นายปราชญา ธนพิบูลผล
B6323059


Not secure | 192.168.1.114/led2on
บทเรียนการพิมพ์สัมผัส...
Monkeytype
Exam English
LED Status
LED1-On LED2-On
LED1 On
LED2 On
LED1 Off
LED2 Off

Quiz_202 – Web Control 4 LED and Monitor Humid/Temperature

- เพิ่มเติมจาก Q202 อยากได้ปุ่มสำหรับคุมปิด-เปิด หลอดไฟ LED 4 ดวง
- อยากมีกด Link ไปที่หน้า FB ของตัวเอง




```
#include <WiFi.h>

#include <WiFiClient.h>

#include <WebServer.h>

#include "DHTesp.h"

#include "index.h" //Our HTML webpage contents with javascripts

#define DHT_Pin 4

#define testLED1 18

#define testLED2 19

#define testLED3 22

#define testLED4 23

//SSID and Password of your WiFi router

const char* ssid = "It's bad day not a bad life";

const char* password = "0641453596";

WebServer server(80); //Server on port 80
```

```
DHTesp dht;

String ledState1 = "NA";

String ledState2 = "NA";

String ledState3 = "NA";

String ledState4 = "NA";

//=========================================================

// This routine is executed when you open its IP in browser

//=========================================================

void handleRoot() {

  String s = MAIN_page; //Read HTML contents

  server.send(200, "text/html", s); //Send web page

}

void handleADC() {

  float h = dht.getHumidity();

  float t = dht.getTemperature();

  String tmpValue = "Temp = ";

  tmpValue += String(t) + " C, Humidity = ";

  tmpValue += String(h) + " %";

  server.send(200, "text/plane", tmpValue); //Send value to client ajax request

}

void handleLED() {

  String t_state = server.arg("LEDstate"); //Refer xhttp.open("GET","setLED?LEDstate=" +led,

true);

  Serial.println(t_state);

  if (t_state == "11") {
```

```
digitalWrite(testLED1, HIGH); //Feedback parameter
  ledState1 = "ON";
}
if (t_state == "10") {
digitalWrite(testLED1, LOW); //Feedback parameter
  ledState1 = "OFF";
}
if (t_state == "21") {
digitalWrite(testLED2, HIGH); //Feedback parameter
  ledState2 = "ON";
}
if (t_state == "20") {
digitalWrite(testLED2, LOW); //Feedback parameter
  ledState2 = "OFF";
}
if (t_state == "31") {
digitalWrite(testLED3, HIGH); //Feedback parameter
  ledState3 = "ON";
}
if (t_state == "30") {
digitalWrite(testLED3, LOW); //Feedback parameter
  ledState3 = "OFF";
}
if (t_state == "41") {
```

```
  digitalWrite(testLED4, HIGH); //Feedback parameter
    ledState4 = "ON";
  }
  if (t_state == "40") {
  digitalWrite(testLED4, LOW); //Feedback parameter
    ledState4 = "OFF";
  }
  server.send(200, "text/plane", ledState1 + ", " + ledState2 + ", " + ledState3 + ", " + ledState4);
  //Send web page
}
void setup(void) {
  Serial.begin(115200);
  dht.setup(DHT_Pin, DHTesp::DHT22); // DHT_Pin D4, DHT22
  pinMode(testLED1, OUTPUT);
  pinMode(testLED2, OUTPUT);
  pinMode(testLED3, OUTPUT);
  pinMode(testLED4, OUTPUT);
  Serial.print("\n\nConnect to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500); Serial.print(".");
  }
  Serial.print("\nConnected "); Serial.println(ssid);
```

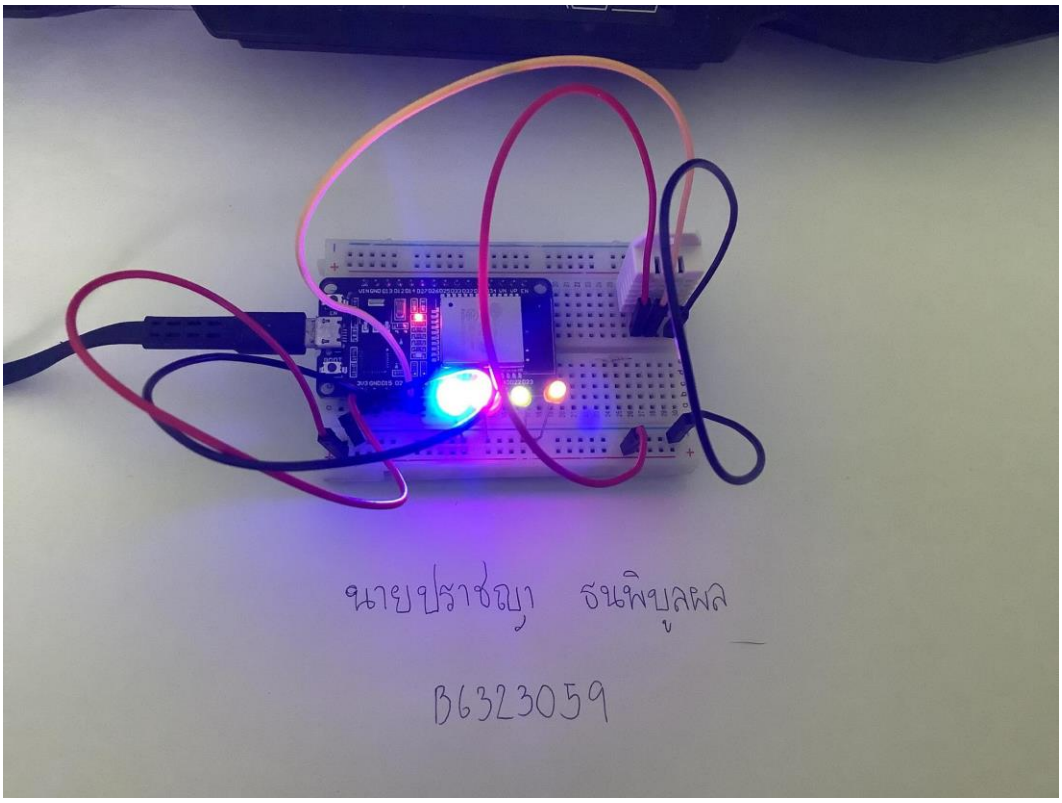
```
  Serial.print("IP address: "); Serial.println(WiFi.localIP());
  server.on("/", handleRoot);
  server.on("/setLED", handleLED);
  server.on("/readADC", handleADC);
  server.begin();
  Serial.println("HTTP server started");
}
void loop(void) {
  server.handleClient(); //Handle client requests
}
```

Index.h

```
const char MAIN_page[] PROGMEM = R"=====(
<!DOCTYPE html><html><body><div id="demo">
<h1>The ESP-32 Update web page without refresh</h1>
<button type="button" onclick="sendData(11)" style="background: rgb(202,60,60);">LED1 ON</button>
<button type="button" onclick="sendData(21)" style="background: rgb(202,60,60);">LED2 ON</button>
<button type="button" onclick="sendData(31)" style="background: rgb(202,60,60);">LED3 ON</button>
<button type="button" onclick="sendData(41)" style="background: rgb(202,60,60);">LED4 ON</button><br><br>
<button type="button" onclick="sendData(10)" style="background: rgb(100,116,255);">LED1 OFF</button>
```

```
<button type="button" onclick="sendData(20)" style="background: rgb(100,116,255);">LED2
OFF</button>
<button type="button" onclick="sendData(30)" style="background: rgb(100,116,255);">LED3
OFF</button>
<button type="button" onclick="sendData(40)" style="background: rgb(100,116,255);">LED4
OFF</button><br><br>
State of [LED1, LED2,LED3,LED4] is >> <span id="LEDState">NA</span><br></div><div><br>
DHT-22 sensor : <span
id="ADCValue">0</span><br></div><script>
function sendData(led) {
var xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
if (this.readyState == 4 && this.status == 200) {
document.getElementById("LEDState").innerHTML =
this.responseText;
}
};
xhttp.open("GET", "setLED?LEDstate="+led, true);
xhttp.send();
}
setInterval(function() { // Call a function repetatively with 2 Second interval
getData();
}, 2000); //2000mSeconds update rate
function getData() {
var xhttp = new XMLHttpRequest();
```

```
xhttp.onreadystatechange = function() {
if (this.readyState == 4 && this.status == 200) {
document.getElementById("ADCValue").innerHTML =
this.responseText;
}
};
xhttp.open("GET", "readADC", true);
xhttp.send();
}
</script><br><a href="https://www.facebook.com/BenzPratchaya">Benz
Pratchaya</a></body></html>)=====";
```



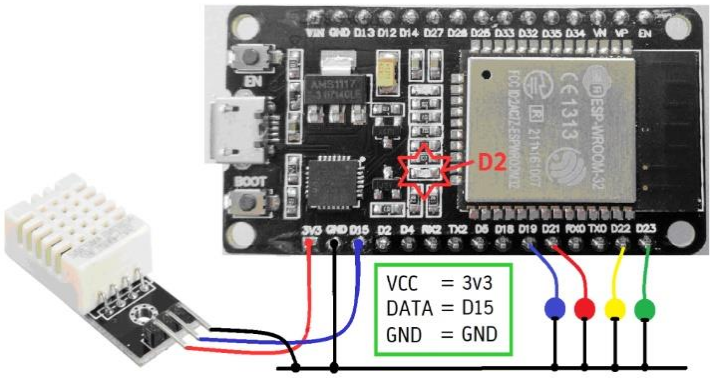

นายปราชญา ธนพิบูลผล
B6323059


Not secure | 192.168.1.114
บทเรียนการพิมพ์สัมผัส...
Monkeytype
Exam English
Tinkercad
REG มทส
SUT e-Lea
The ESP-32 Update web page without refresh
LED1 ON
LED2 ON
LED3 ON
LED4 ON
LED1 OFF
LED2 OFF
LED3 OFF
LED4 OFF
State of [LED1, LED2,LED3,LED4] is >> ON, ON, ON, ON
DHT-22 sensor : Temp = 36.20 C, Humidity = 1.40 %
Benz Pratchaya

Quiz_203 – Publish

- อ่านค่า DHT-22 แล้วส่งไปยัง MQTT Broker ทุกๆ 5 วินาที
- ควบคุมการแสดงผลให้ 4 LED แสดงผลตามข้อกำหนดดังนี้

✱○○○(Blink)	หากการอ่านค่าแล้วเป็น null, หรือไม่มีเซ็นเซอร์
●○○○	ช่วงของอุณหภูมิ (-∞, 24)
●●○○	ช่วงของอุณหภูมิ [24,26)
●●●○	ช่วงของอุณหภูมิ [26,28)
●●●●	ช่วงของอุณหภูมิ [28,30)
✱✱✱✱(Blink)	ช่วงของอุณหภูมิ [30,∞)




```
#include <WiFi.h>

#include <Wire.h>

#include <PubSubClient.h>

#include "DHTesp.h"

DHTesp dht;

#define PinLED0 19

#define PinLED1 21

#define PinLED2 22

#define PinLED3 23

#define DHT22_Pin 15

float h, t;

int blinkStatus = 1;
```

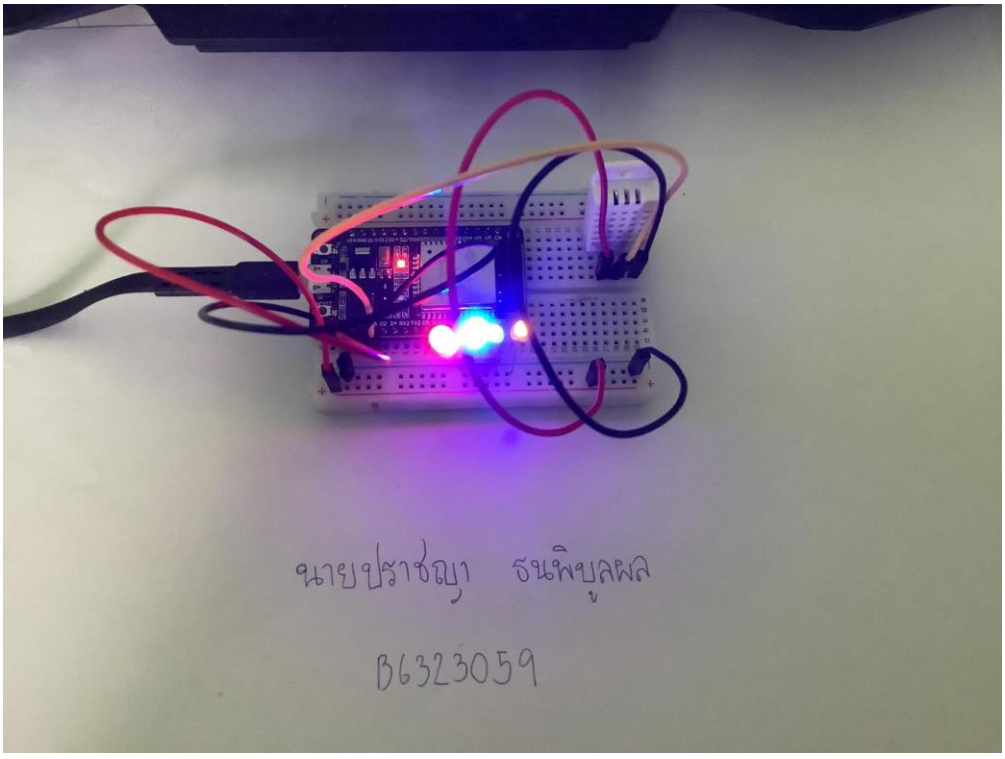
```
int LED_PinArray[] = {PinLED0, PinLED1, PinLED2, PinLED3};

int LED_StsArray[] = {0, 0, 0, 0};

const char* ssid = "It's bad day not a bad life";

const char* password = "0641453596";

const char* mqtt_server = "test.mosquitto.org";

const char* topic1 = "test1";

String ledState1 = "NA";

WiFiClient espClient;

PubSubClient client(espClient);

long lastMsg = 0;

char msg[50];

int value = 0;

void setup_wifi() {

  delay(10);

  Serial.println();

  Serial.print("Connecting to ");

  Serial.println(ssid);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {

    delay(500); Serial.print(".");

  }

  randomSeed(micros());

  Serial.println("");

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

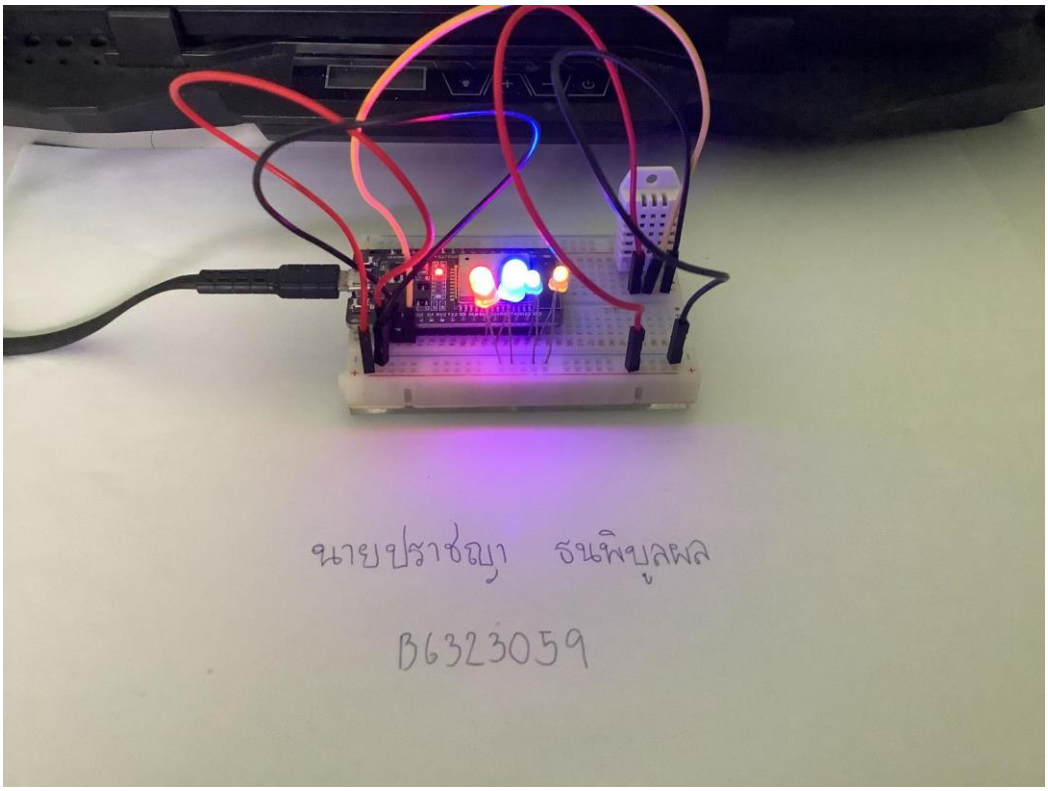
```
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
void reconnect()
{ while (!client.connected()) // Loop until we're reconnected
  { Serial.print("Attempting MQTT connection...");
    String clientId = "ESP8266Client-";
    clientId += String(random(0xffff), HEX); // Create a random client ID
    if (client.connect(clientId.c_str())) // Attempt to connect
    { Serial.println("connected"); // Once connected, publish an announcement...
      client.publish(topic1, "Hello World Pk007"); // ... and resubscribe
      client.subscribe(topic1);
    } else
    { Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      delay(5000);
    }
  }
}
void LEDShowStatus(void) {
  if (isnan(t)) {
    blinkStatus = 1 - blinkStatus;
    LED_StsArray[0] = 1;
```

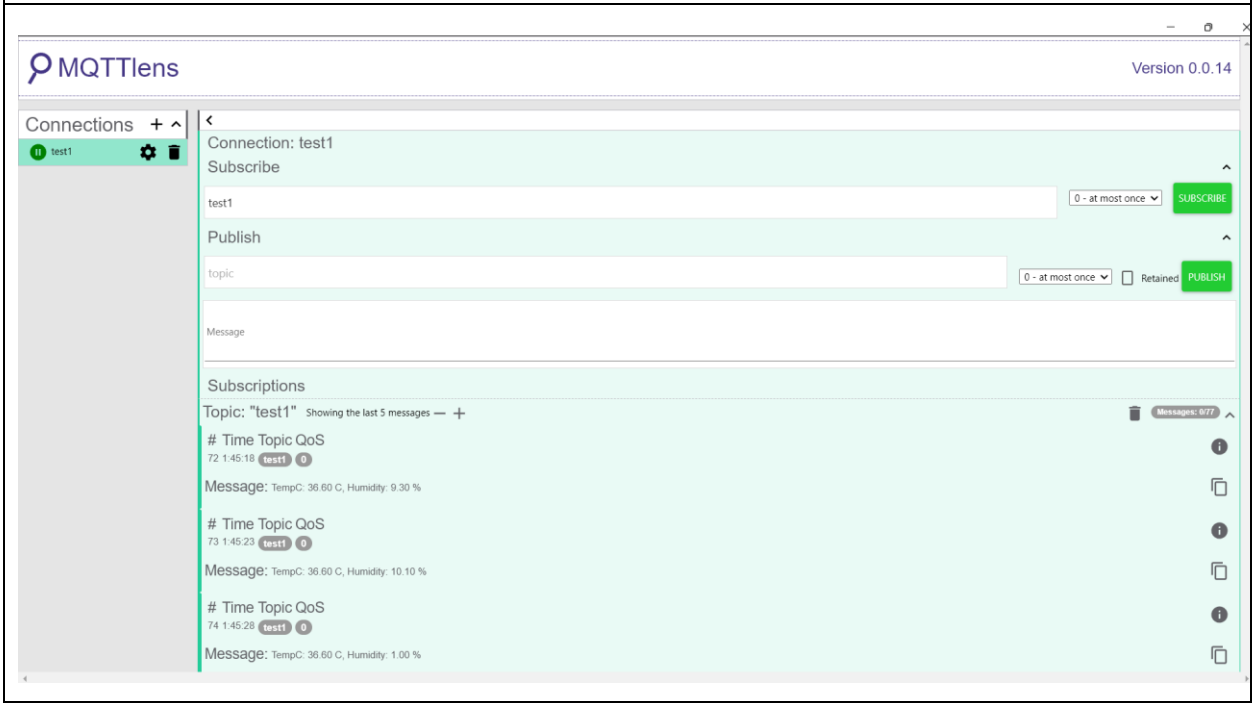
```
    LED_StsArray[1] = 0;
    LED_StsArray[2] = 0;
    LED_StsArray[3] = 0;
  }
  if (t < 27) {
    blinkStatus = 1;
    LED_StsArray[0] = 1;
    LED_StsArray[1] = 0;
    LED_StsArray[2] = 0;
    LED_StsArray[3] = 0;
  }
  if (t >= 27) {
    blinkStatus = 1 - blinkStatus;
    LED_StsArray[0] = 1;
    LED_StsArray[1] = 1;
    LED_StsArray[2] = 1;
    LED_StsArray[3] = 1;
  }
  LED_StsArray[1] = 1;
  LED_StsArray[2] = 1;
  LED_StsArray[3] = 1;
  for (int i = 0; i < 4; i++)
    digitalWrite(LED_PinArray[i], LED_StsArray[i] & blinkStatus);
}
```

```
void setup()
{ Serial.begin(115200);
  setup_wifi();
  //Wire.begin(22, 23);
  client.setServer(mqtt_server, 1883);
  dht.setup(DHT22_Pin, DHTesp::DHT22);
  for (int i = 0; i < 4; i++) {
    pinMode(LED_PinArray[i], OUTPUT);
  }
}
void loop()
{
  if (!client.connected()) reconnect();
  client.loop();
  long now = millis();
  if (now - lastMsg > 5000)
  { lastMsg = now;
    ++value;
    //float t = s.readTempC();
    //float h = s.readHumidity();
    delay(dht.getMinimumSamplingPeriod());
    h = dht.getHumidity();
    t = dht.getTemperature();
    sprintf (msg, "TempC: %.2f C, Humidity: %.2f %%", t, h);
```

```
    Serial.print("Publish message: ");
    Serial.println(msg);
    client.publish(topic1, msg);
  }
  LEDShowStatus(); delay(250);
  LEDShowStatus(); delay(250);
  LEDShowStatus(); delay(250);
  LEDShowStatus(); delay(250);
  LEDShowStatus(); delay(250);
  LEDShowStatus(); delay(250);
}
```

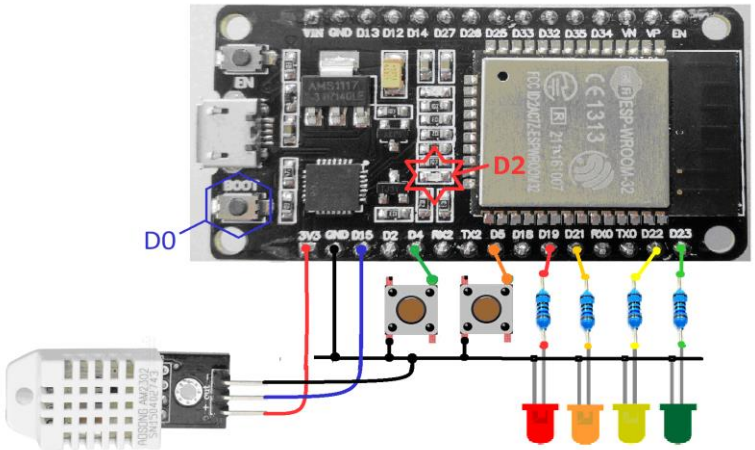



นายปราชญา ธนพิบูลผล
B6323059


MQTTlens
Version 0.0.14
Connections
test1
Connection: test1
Subscribe
test1
0 - at most once
SUBSCRIBE
Publish
topic
0 - at most once
Retained
PUBLISH
Message
Subscriptions
Topic: "test1"
Showing the last 5 messages
Messages: 0/77
Time Topic QoS
72 1:45:18 test1 0
Message: TempC: 36.60 C, Humidity: 9.30 %
Time Topic QoS
73 1:45:23 test1 0
Message: TempC: 36.60 C, Humidity: 10.10 %
Time Topic QoS
74 1:45:28 test1 0
Message: TempC: 36.60 C, Humidity: 1.00 %

Quiz_204 – Publish and Subscribe

- อ่านค่า DHT-22 แล้วส่งไปยัง MQTT Broker ทุกๆ 5 วินาที
- ควบคุมการปิดเปิด 4 LED
- รับค่าสวิตซ์กำหนด SW1 แจ้ง Overheat Alarm, SW2 แจ้ง Intruders Alarm




```
#include <WiFi.h>

#include <Wire.h>

#include <PubSubClient.h>

#include "DHTesp.h"

DHTesp dht;

#define testLED1 19

#define testLED2 21

#define testLED3 22

#define testLED4 23

#define DHT22_Pin 15

const char* ssid = "It's bad day not a bad life";

const char* password = "0641453596";

const char* mqtt_server = "test.mosquitto.org";

const char* topic1 = "test2";
```

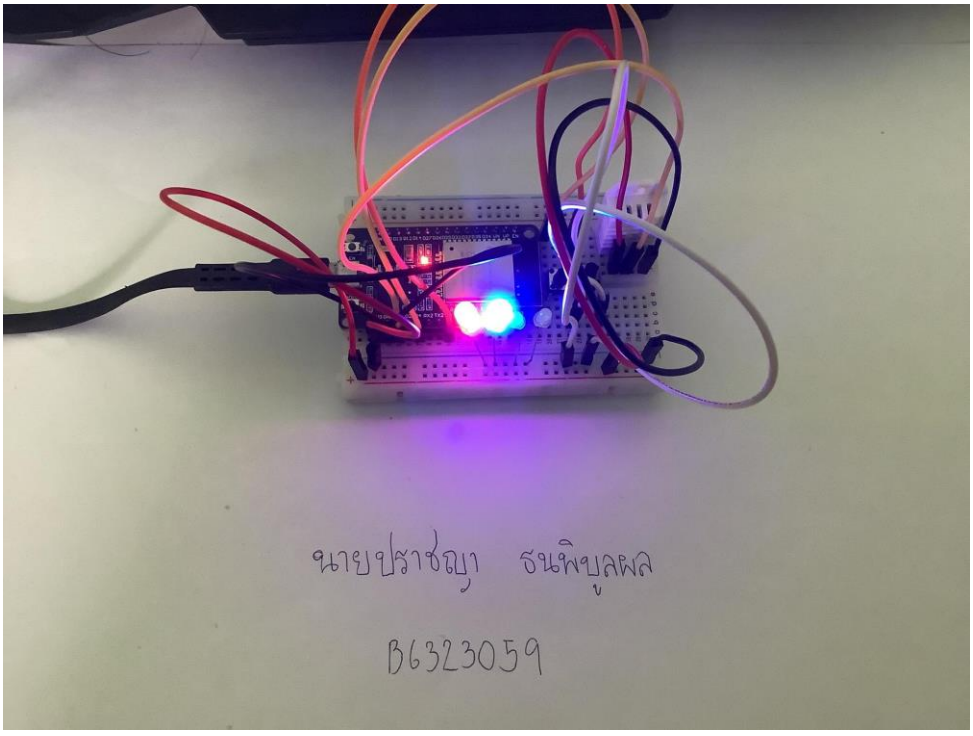
```
String ledState1 = "NA";
int pushButton1 = 4;
int pushButton2 = 5;
WiFiClient espClient;
PubSubClient client(espClient);
long lastMsg = 0;
char msg[50];
int value = 0;
void setup_wifi() {
  delay(10);
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500); Serial.print(".");
  }
  randomSeed(micros());
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  pinMode(testLED1, OUTPUT);
  pinMode(testLED2, OUTPUT);
```

```
  pinMode(testLED3, OUTPUT);
  pinMode(testLED4, OUTPUT);
}
void callback(char* topic, byte* payload, unsigned int length)
{ char myPayLoad[50];
  Serial.print("Message arrived [");
  Serial.print(topic1);
  Serial.print("] ");
  for (int i = 0; i < length; i++)
  { Serial.print((char)payload[i]);
    myPayLoad[i] = payload[i];
    myPayLoad[i + 1] = '\0'; // End of String
  }
  Serial.print("\n ---> "); Serial.println(myPayLoad);
  myPayLoad[4] = '\0'; // String lessthan 4 Charector
  if ((String)myPayLoad == "ON1") digitalWrite(testLED1, HIGH);
  if ((String)myPayLoad == "OFF1") digitalWrite(testLED1, LOW);
  if ((String)myPayLoad == "ON2") digitalWrite(testLED2, HIGH);
  if ((String)myPayLoad == "OFF2") digitalWrite(testLED2, LOW);
  if ((String)myPayLoad == "ON3") digitalWrite(testLED3, HIGH);
  if ((String)myPayLoad == "OFF3") digitalWrite(testLED3, LOW);
  if ((String)myPayLoad == "ON4") digitalWrite(testLED4, HIGH);
  if ((String)myPayLoad == "OFF4") digitalWrite(testLED4, LOW);
}
```

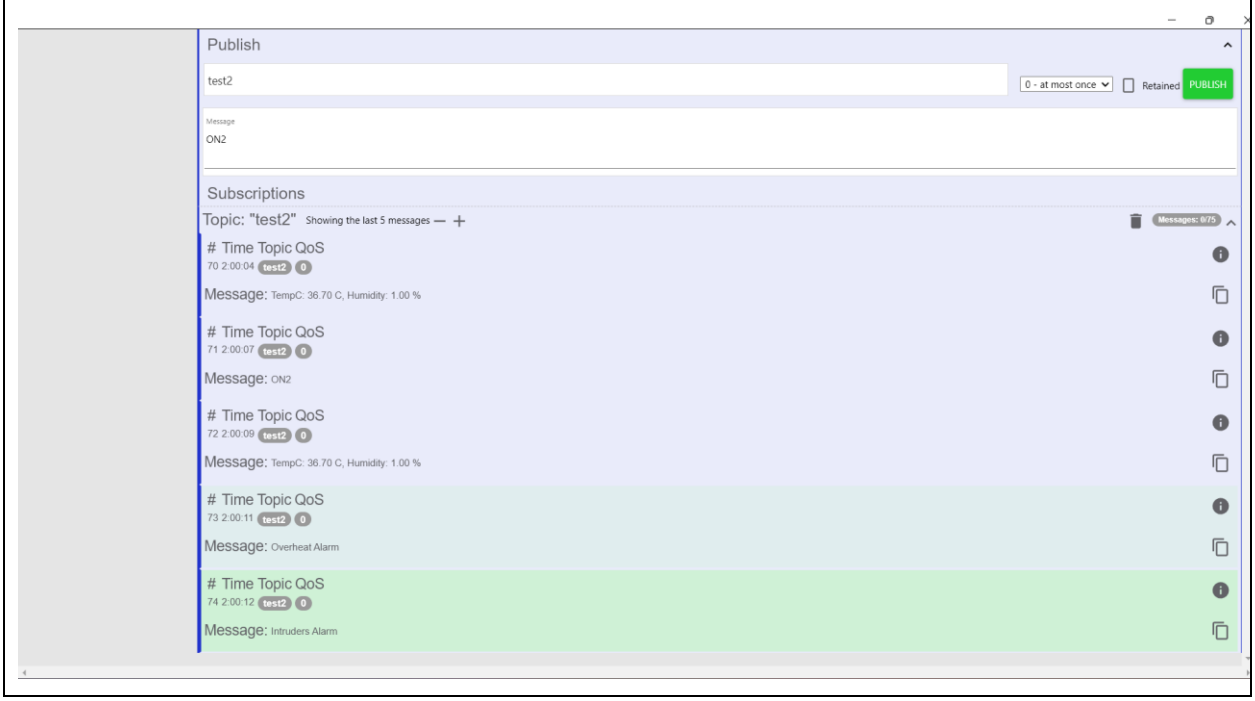
```
void reconnect()
{ while (!client.connected()) // Loop until we're reconnected
  { Serial.print("Attempting MQTT connection...");
    String clientId = "ESP8266Client-";
    clientId += String(random(0xffff), HEX); // Create a random client ID
    if (client.connect(clientId.c_str())) // Attempt to connect
    { Serial.println("connected"); // Once connected, publish an announcement...
      client.publish(topic1, "Hello World Pk007"); // ... and resubscribe
      client.subscribe(topic1);
    } else
    { Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      delay(5000);
    }
  }
}
void setup()
{ Serial.begin(115200);
  setup_wifi();
  dht.setup(DHT22_Pin, DHTesp::DHT22);
  pinMode(pushButton1, INPUT_PULLUP);
  pinMode(pushButton2, INPUT_PULLUP);
  client.setServer(mqtt_server, 1883);
```

```
  client.setCallback(callback);
  pinMode(testLED1, OUTPUT);
  pinMode(testLED2, OUTPUT);
  pinMode(testLED3, OUTPUT);
  pinMode(testLED4, OUTPUT);
}
void loop()
{
  if (!client.connected()) reconnect();
  client.loop();
  long now = millis();
  if (now - lastMsg > 5000)
  { lastMsg = now;
    ++value;
    float h = dht.getHumidity();
    float t = dht.getTemperature();
    sprintf (msg, "TempC: %.2f C, Humidity: %.2f %%", t, h);
    Serial.print("Publish message: ");
    Serial.println(msg);
    client.publish(topic1, msg);
  }
  if (digitalRead(pushButton1) == 0) {
    sprintf (msg, "Overheat Alarm");
    Serial.println(msg);
```

```
    client.publish(topic1, msg);
    delay(500);
  }
  if (digitalRead(pushButton2) == 0) {
    sprintf (msg, "Intruders Alarm");
    Serial.println(msg);
    client.publish(topic1, msg);
    delay(500);
  }
}
```




นายปราชญา ธนพิบูลผล
B6323059


Publish
test2
0 - at most once
Retained
PUBLISH
Message
ON2
Subscriptions
Topic: "test2"
Showing the last 5 messages
Messages: 0/75
Time Topic QoS
70 2:00:04 test2 0
Message: TempC: 36.70 C, Humidity: 1.00 %
Time Topic QoS
71 2:00:07 test2 0
Message: ON2
Time Topic QoS
72 2:00:09 test2 0
Message: TempC: 36.70 C, Humidity: 1.00 %
Time Topic QoS
73 2:00:11 test2 0
Message: Overheat Alarm
Time Topic QoS
74 2:00:12 test2 0
Message: Intruders Alarm