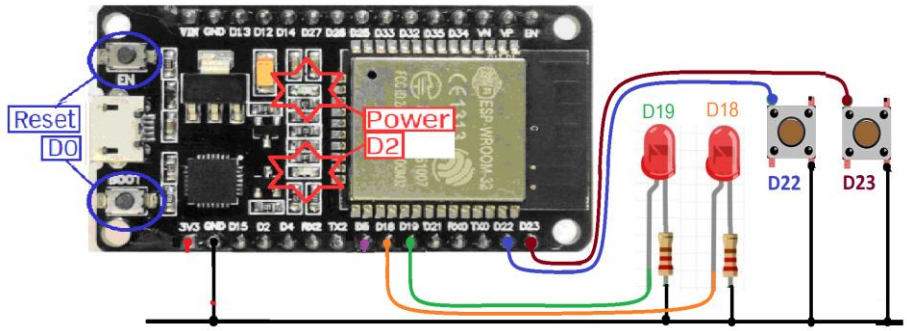
แนวทางการใช้งานอินเทอร์เน็ตของสรรพสิ่งในระบบการผลิต
IoT Approaches to Manufacturing System

ชื่อ-สกุล : วราสิริ ลิ้มประเสริฐ B6214005

4/4. คำถามท้ายบทเพื่อทดสอบความเข้าใจ

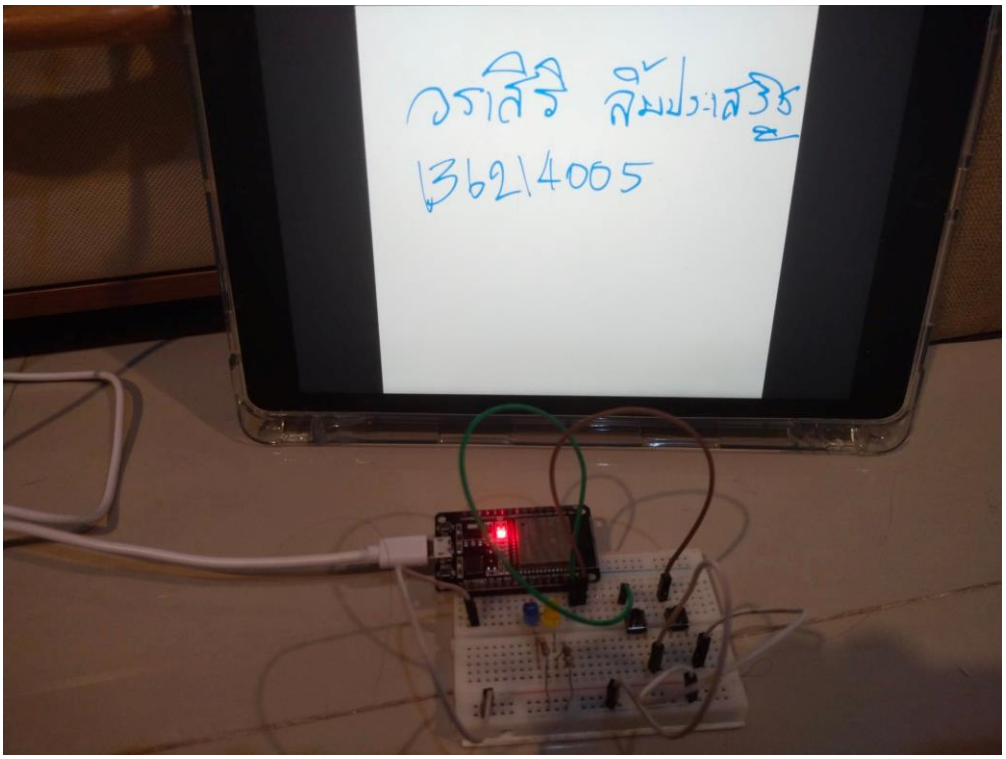
Quiz_101 – กดติด กดดับ 2 ชุด

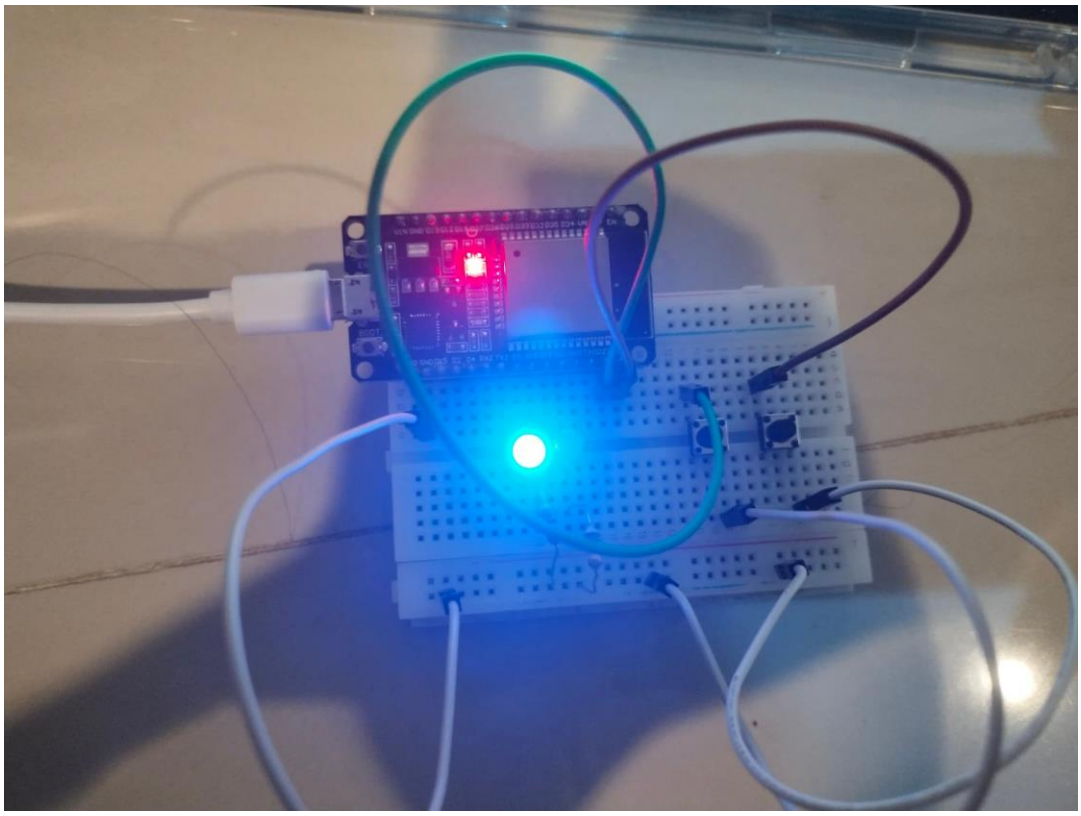
- หากต้องการให้ใช้ 1 สวิตซ์ ควบคุม 1 LED แบบกดติด-กดดับ จำนวน 2 วงจรจะต่อวงจรและเขียนโปรแกรมอย่างไร {SW-D22 -- LED-D19, SW-D23 -- LED-D18}

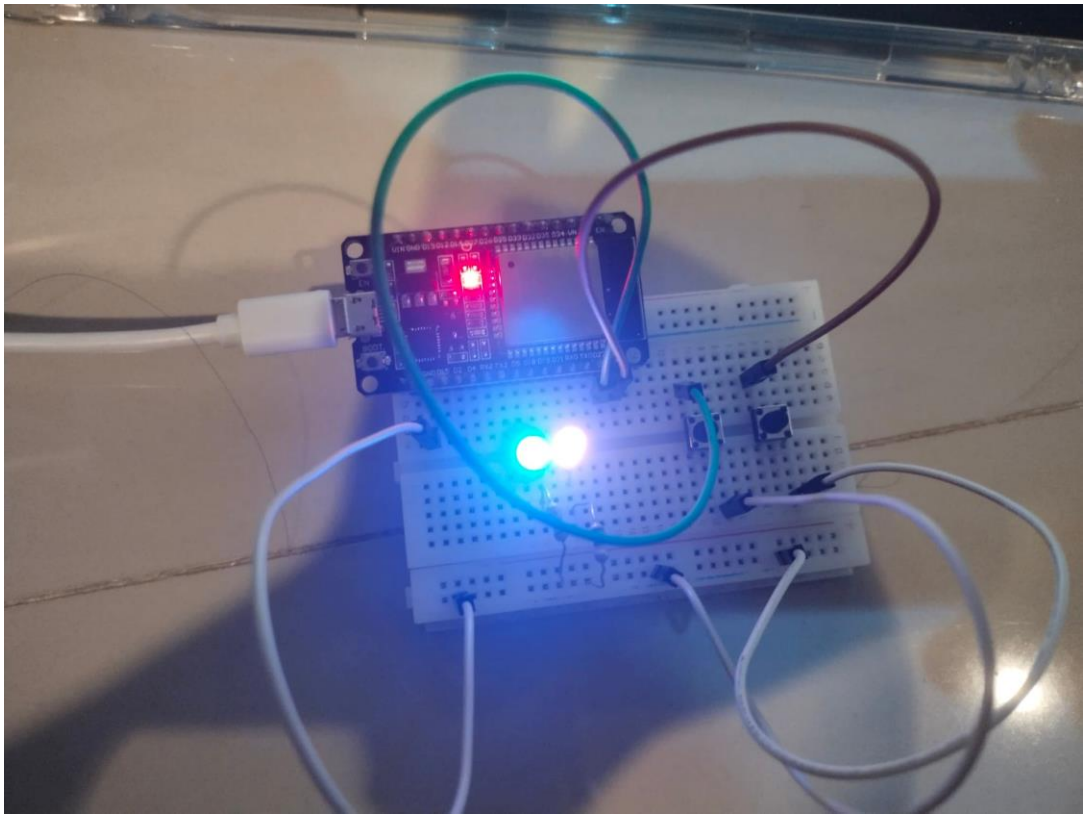



```
#define pushButton1 22
#define pushButton2 23
#define LEDPin1 18
#define LEDPin2 19

int buttonState1 = 0;
int buttonState2 = 0;

void setup() {
  Serial.begin(115200);
  pinMode(pushButton1, INPUT_PULLUP);
  pinMode(pushButton2, INPUT_PULLUP);
  pinMode(LEDPin1, OUTPUT);
  pinMode(LEDPin2, OUTPUT);
}
void loop() {
  if (digitalRead(pushButton1) == LOW) {
    delay(20);
    buttonState1 = 1 - buttonState1;
    digitalWrite(LEDPin1, buttonState1);
```

```
    while (digitalRead(pushButton1) == LOW);
    delay(20);
  }

  if (digitalRead(pushButton2) == LOW) {
    delay(20);
    buttonState2 = 1 - buttonState2;
    digitalWrite(LEDPin2, buttonState2);
    while (digitalRead(pushButton2) == LOW);
    delay(20);
  }
}
```



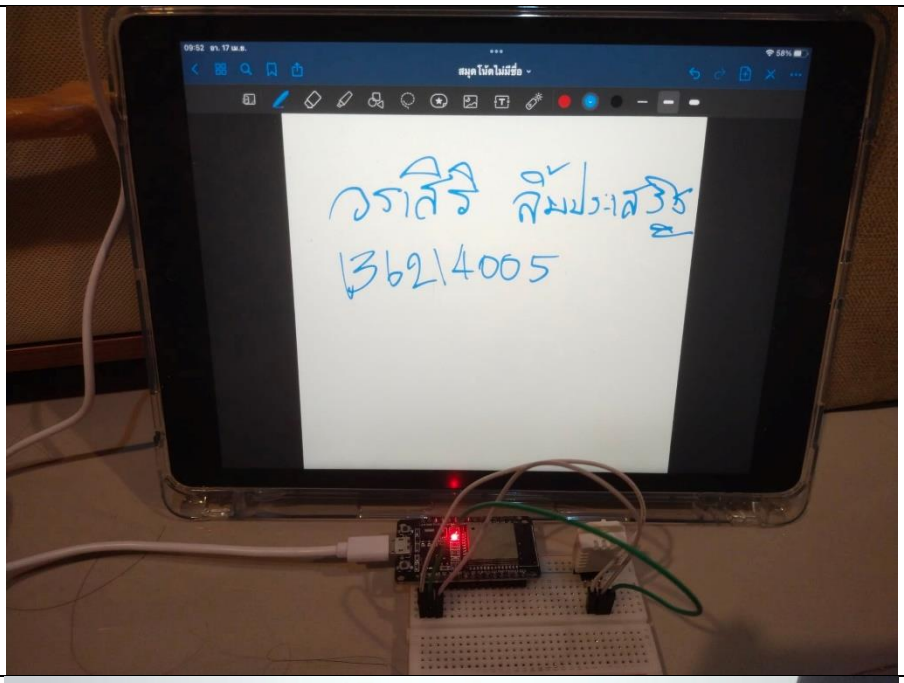


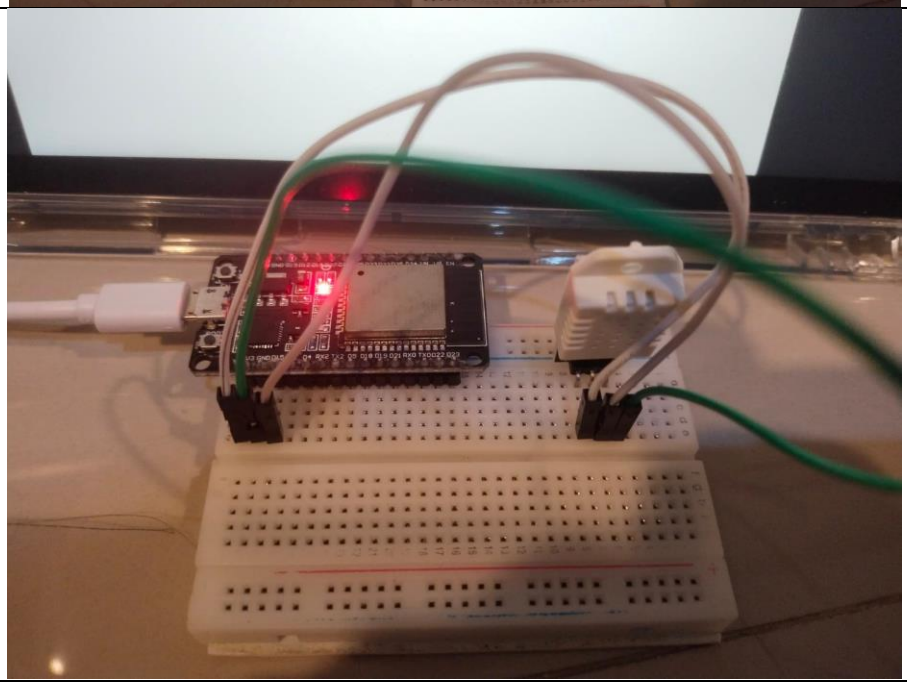


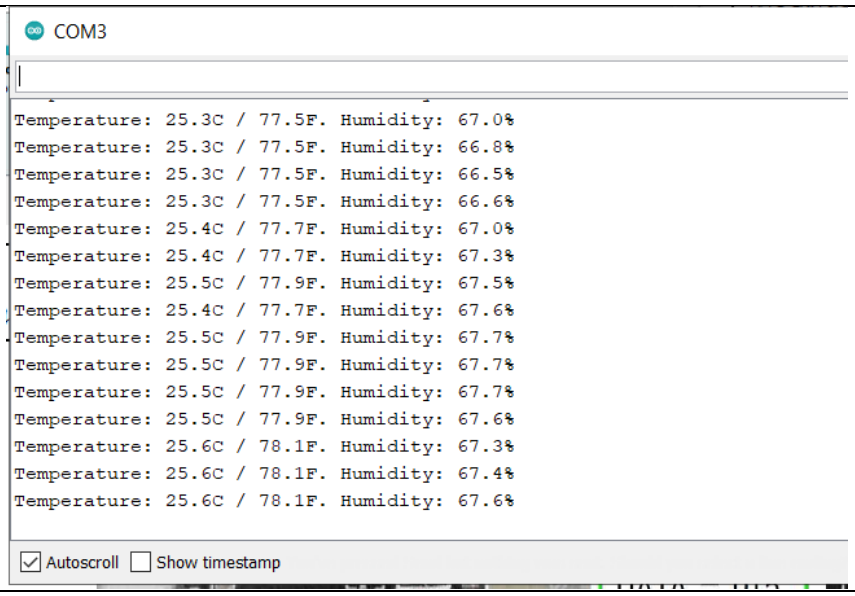
Quiz_102 – ปรับการแสดงผลที่ Serial Monitor เป็นดังนี้

```
Temperature: 23.0C / 74.7F. Humidity: 24.9%
Temperature: 23.0C / 74.7F. Humidity: 24.9%
Temperature: 23.0C / 74.7F. Humidity: 24.9%
```

```
#define DHT22_Pin 15
#include "DHTesp.h"
DHTesp dht;
void setup() {
  Serial.begin(115200);
  Serial.println();
  dht.setup(DHT22_Pin, DHTesp::DHT22); // Connect DHT sensor to GPIO 15
}
void loop() {
  delay(dht.getMinimumSamplingPeriod());
  float humidity = dht.getHumidity();
  float temperature = dht.getTemperature();
  Serial.print("Temperature: ");
  Serial.print(temperature, 1);
  Serial.print("C / ");
  Serial.print(dht.toFahrenheit(temperature), 1);
  Serial.print("F. Humidity: ");
  Serial.print(humidity, 1);
  Serial.print("% \n");
  delay(2000);
}
```

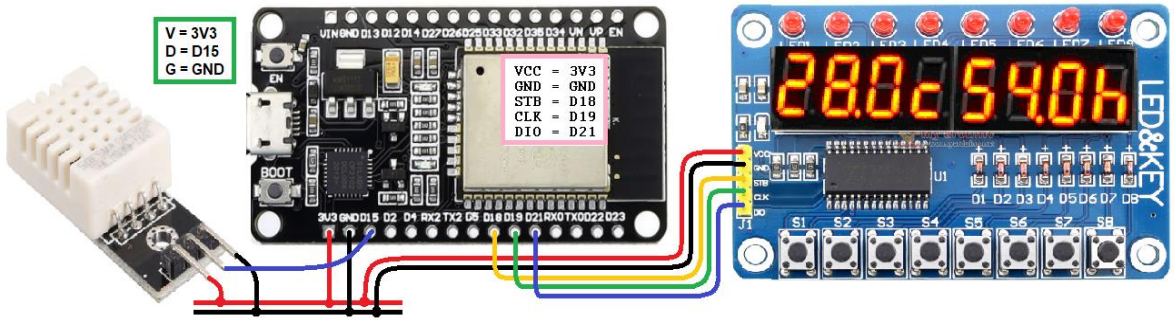

วราสิริ ลิ้มประเสริฐ
13621405




COM3
Temperature: 25.3C / 77.5F. Humidity: 67.0%
Temperature: 25.3C / 77.5F. Humidity: 66.8%
Temperature: 25.3C / 77.5F. Humidity: 66.5%
Temperature: 25.3C / 77.5F. Humidity: 66.6%
Temperature: 25.4C / 77.7F. Humidity: 67.0%
Temperature: 25.4C / 77.7F. Humidity: 67.3%
Temperature: 25.5C / 77.9F. Humidity: 67.5%
Temperature: 25.4C / 77.7F. Humidity: 67.6%
Temperature: 25.5C / 77.9F. Humidity: 67.7%
Temperature: 25.5C / 77.9F. Humidity: 67.7%
Temperature: 25.5C / 77.9F. Humidity: 67.7%
Temperature: 25.5C / 77.9F. Humidity: 67.6%
Temperature: 25.6C / 78.1F. Humidity: 67.3%
Temperature: 25.6C / 78.1F. Humidity: 67.4%
Temperature: 25.6C / 78.1F. Humidity: 67.6%
Autoscroll
Show timestamp

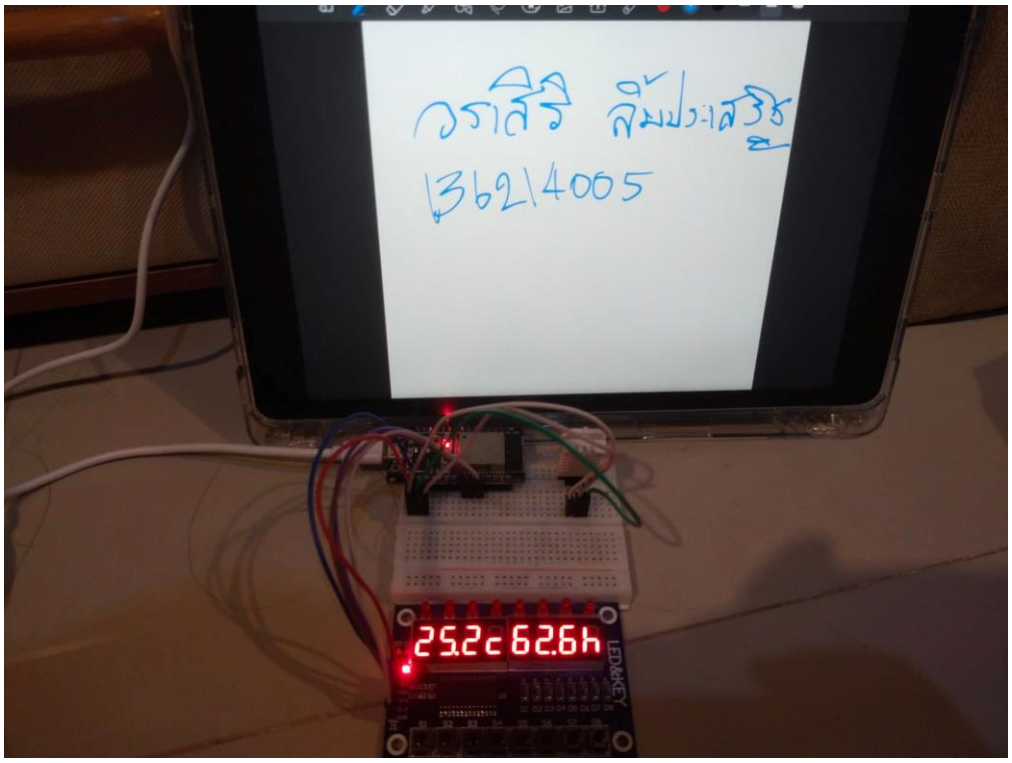
Quiz_103 – Read Sensor and Show

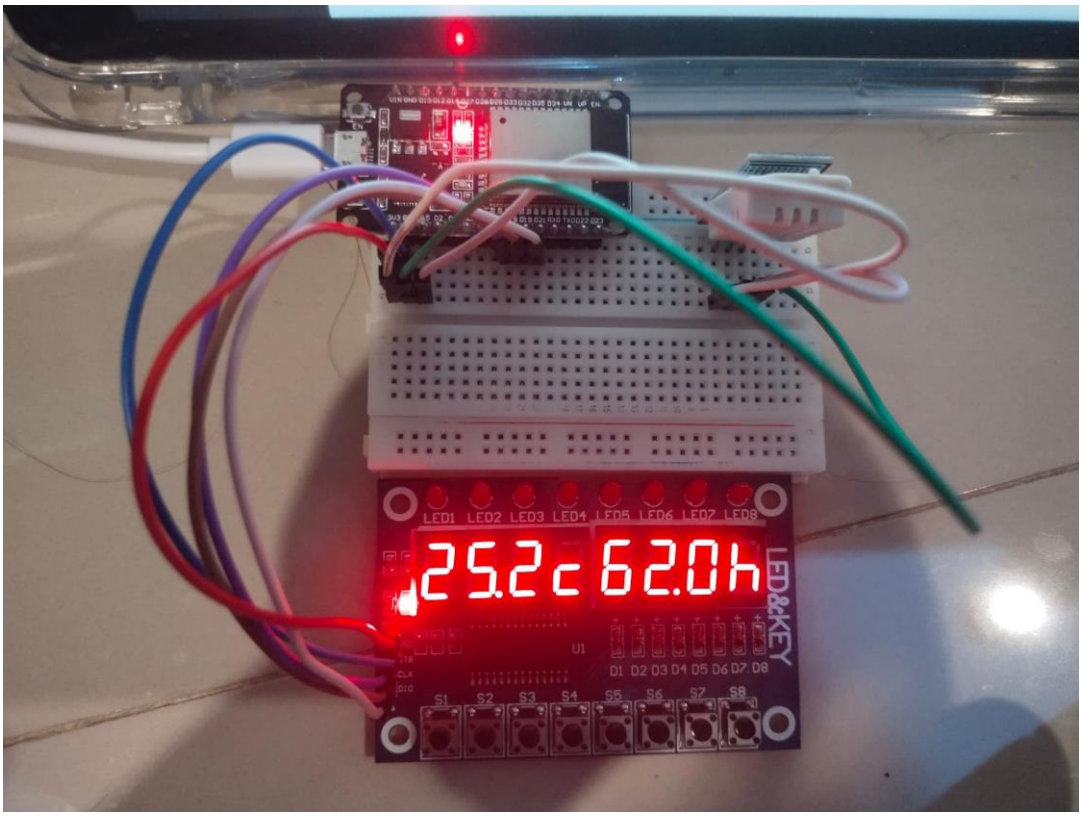
- ต่อวงจรเพิ่มเติม ทดสอบการทำงานด้วยโปรแกรมต่อไปนี้ และปรับแก้ให้ถูกต้อง




```
#include <TM1638plus.h>
#include "DHTesp.h"
DHTesp dht;

#define DHT22_Pin 15
#define Brd_STB 18 // strobe = GPIO connected to strobe line of module
#define Brd_CLK 19 // clock = GPIO connected to clock line of module
#define Brd_DIO 21 // data = GPIO connected to data line of module
bool high_freq = true; //default false,, If using a high freq CPU > ~100 MHZ set to true.
TM1638plus tm(Brd_STB, Brd_CLK , Brd_DIO, high_freq);

void setup() {
  Serial.begin(115200);
  tm.displayBegin();
  dht.setup(DHT22_Pin, DHTesp::DHT22); // Connect DHT sensor to GPIO 15
}

void loop() {

  float humidity = dht.getHumidity();
  float temperature = dht.getTemperature();

  Serial.print("Temperature: ");
  Serial.print(temperature); Serial.print(" *C\t");
  Serial.print("Humidity: ");
```

```
  Serial.print(humidity); Serial.print(" %\n");

  float Tempp_1 = temperature / 10;
  float Tempp_2 = (int)temperature % 10;
  int Tempp_3 = (int)(temperature * 10) % 10;
  int Humi_1 = humidity / 10;
  int Humi_2 = (int)humidity % 10;
  int Humi_3 = int(humidity * 10) % 10;

  tm.displayHex(0, Tempp_1);
  tm.displayASCIIwDot(1, Tempp_2 + '0'); // turn on dot
  tm.displayHex(2, Tempp_3);
  tm.display7Seg(3, B01011000); // Code=tgfedcba
  tm.displayHex(4, Humi_1);
  tm.displayASCIIwDot(5, Humi_2 + '0'); // turn on dot
  tm.displayHex(6, Humi_3);
  tm.display7Seg(7, B01110100); // Code=tgfedcba
  delay(2000);
}
```




LED1 LED2 LED3 LED4 LED5 LED6 LED7 LED8
25.2c 62.0h
LED&KEY
U1
D1 D2 D3 D4 D5 D6 D7 D8
S1 S2 S3 S4 S5 S6 S7 S8

แนวทางการใช้งานอินเทอร์เน็ตของสรรพสิ่งในระบบการผลิต
IoT Approaches to Manufacturing System

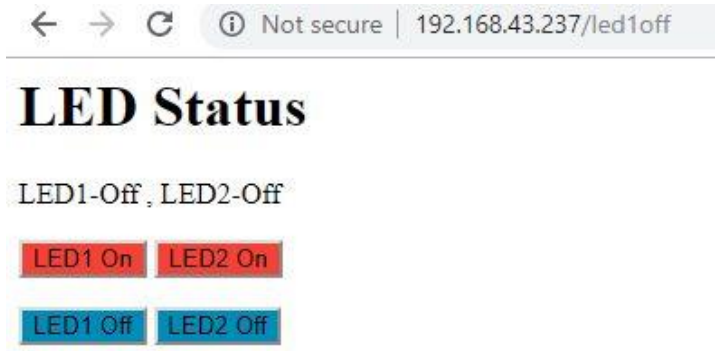
ชื่อ-สกุล : วราสิริ ลิ้มประเสริฐ B6214005

4/4. คำถามท้ายบทเพื่อทดสอบความเข้าใจ

Quiz_201 – Web Control 2 LED

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

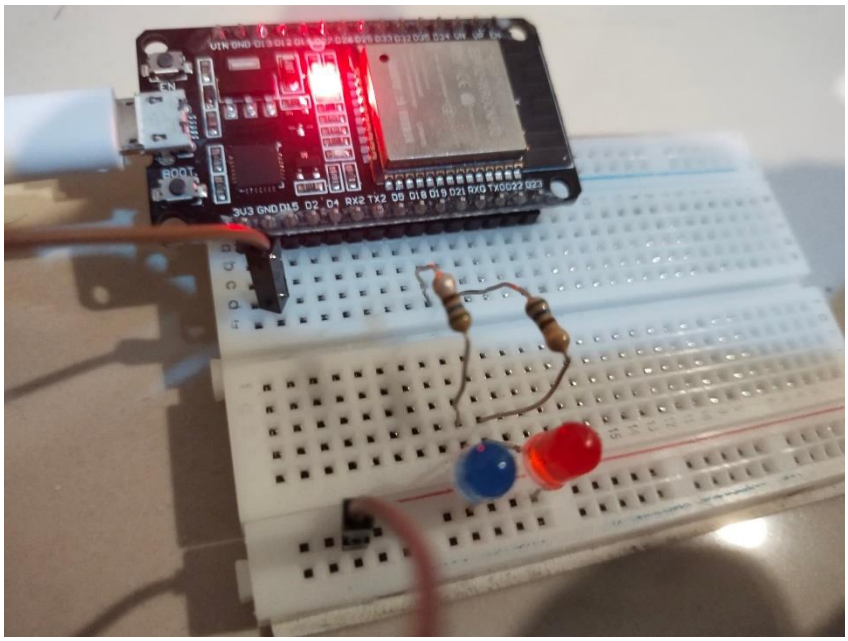
Note : Mac Address 3C:61:05:13:11:58

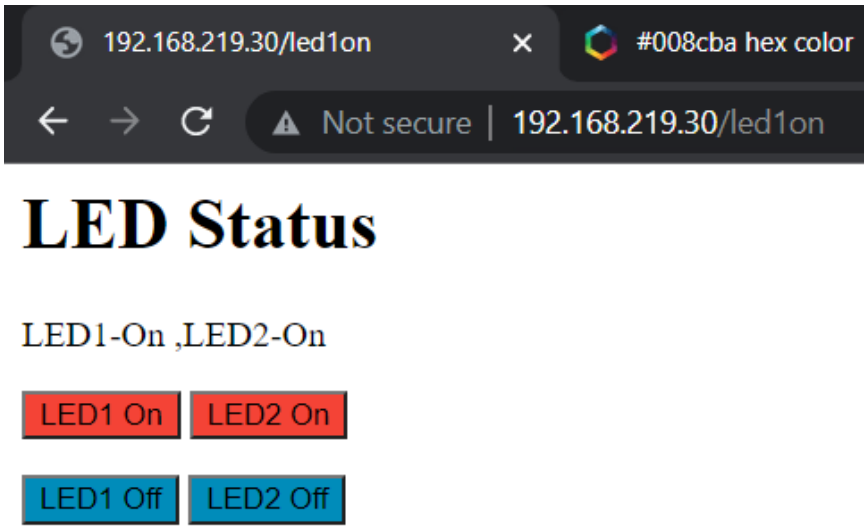


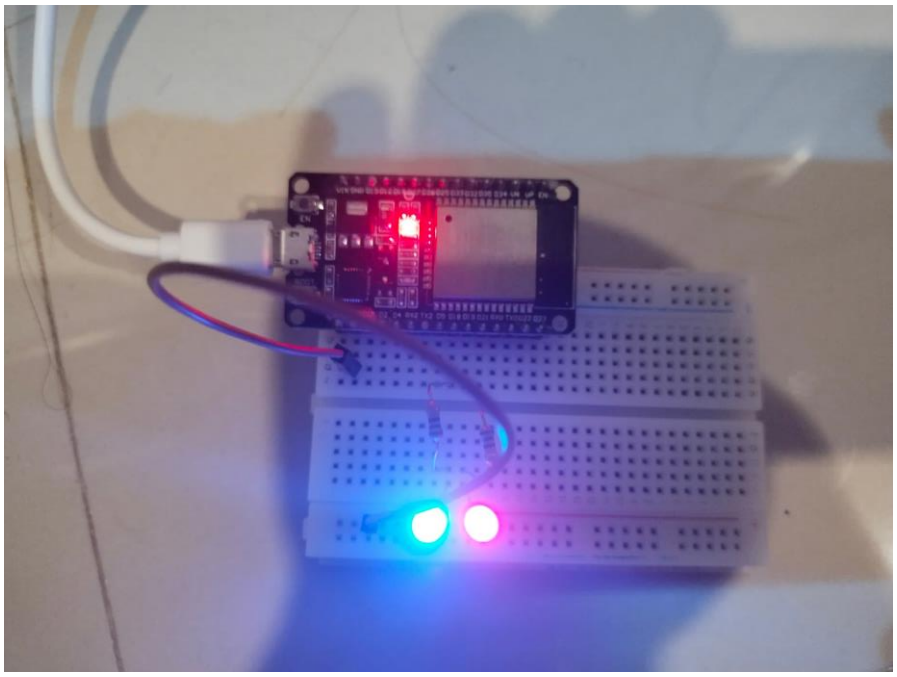

```
#include <WiFi.h>
const char* ssid = "V2036";
const char* password = "fnafchica";

int pinTest1 = 18;
int pinTest2 = 19;

WiFiServer server(80);

void setup() {
  Serial.begin(115200);
  pinMode(pinTest1, OUTPUT); // set the LED pin mode
  pinMode(pinTest2, OUTPUT); // set the LED pin mode
  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 LED_Status1 = LOW;
bool LED_Status2 = LOW;
```

```
void loop() {
  digitalWrite(pinTest1, LED_Status1);
  digitalWrite(pinTest2, LED_Status2);
  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 (LED_Status1 == HIGH)
              client.println("LED1-On");
            else
              client.println("LED1-Off");

            if (LED_Status2 == HIGH)
              client.println(",LED2-On");
            else
              client.println(",LED2-Off");

            client.println("<p>");
            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=\"/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("<html>");
            break;
          } else {
            currentLine = "";
          }
        } else if (c != '\r') {
          currentLine += c;
        }
        if (currentLine.endsWith("GET /led1on")) LED_Status1 = HIGH;
        if (currentLine.endsWith("GET /led1off")) LED_Status1 = LOW;
        if (currentLine.endsWith("GET /led2on")) LED_Status2 = HIGH;
        if (currentLine.endsWith("GET /led2off")) LED_Status2 = LOW;
```

```
        }
      }
      client.stop(); // close the connection:
      Serial.println("Client Disconnected.");
    }
}
```

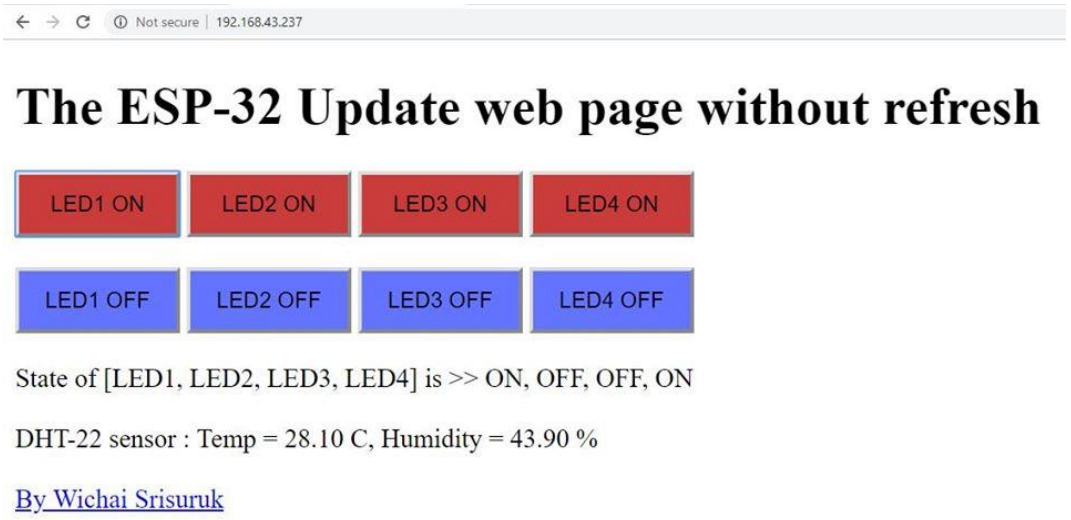






Quiz_202 – Web Control 4 LED and Monitor Humid/Temperature

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



SourceCode

```
#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 = "V2036";
const char* password = "fnafchica";
WebServer server(80); //Server on port 80
DHTesp dht;
String ledState1 = "OFF";
String ledState2 = "OFF";
String ledState3 = "OFF";
String ledState4 = "OFF";
//==============================================================
// 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();
```

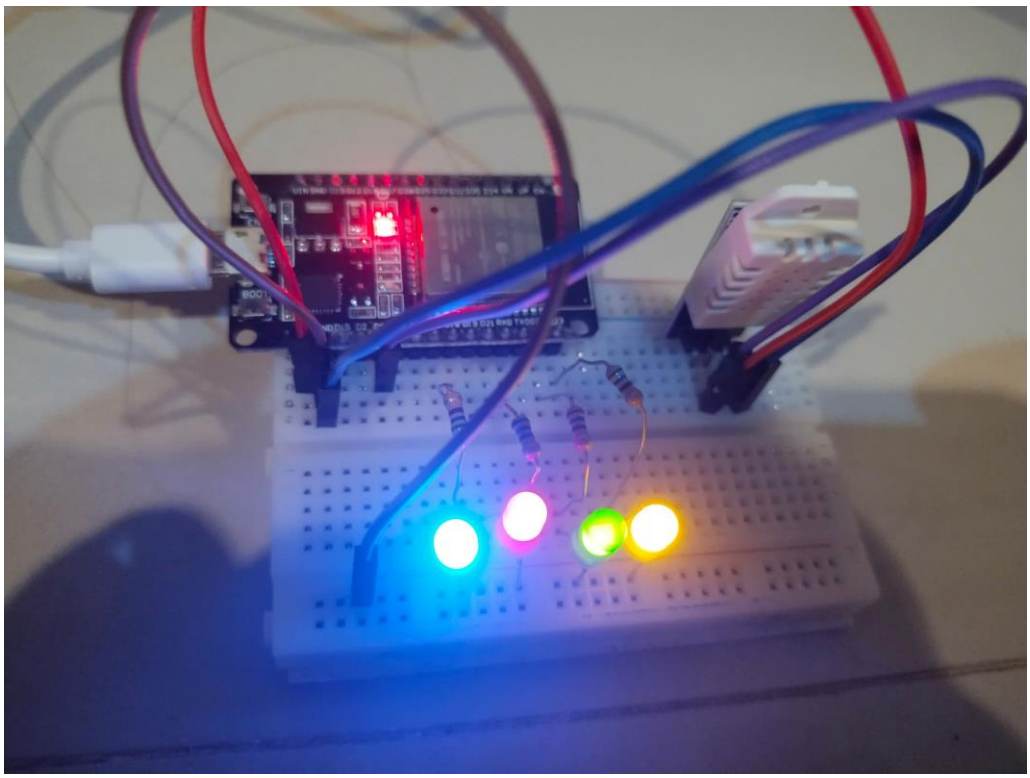
```
  float t = dht.getTemperature();
  String tmpValue = "Temp = ";
  tmpValue += String(t) + " C, Humidity = ";
  tmpValue += String(h) + " %";
  server.send(200, "text/plane", tmpValue); //Send value to client ajax request
}
void handleLED() {
  String t_state = server.arg("LEDstate"); //Refer xhttp.open("GET",
"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);
```

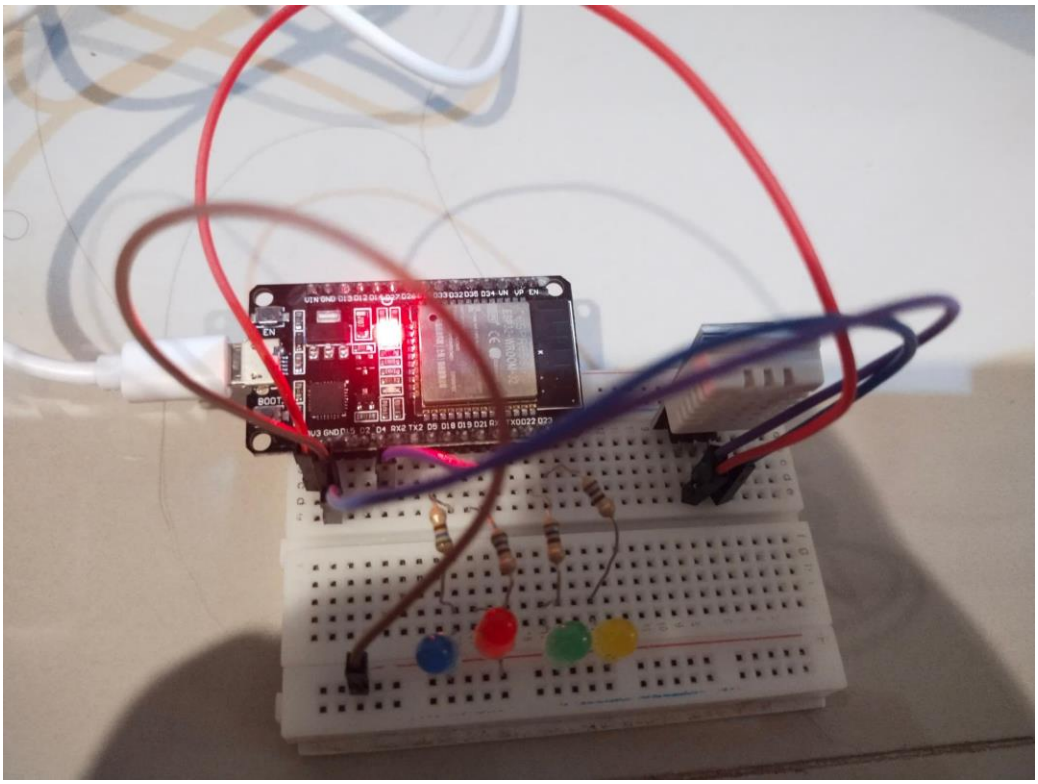
```
  while (WiFi.status() != WL_CONNECTED) {
    delay(500); Serial.print(".");
  }
  Serial.print("\nConnected "); Serial.println(ssid);
  Serial.print("IP address: "); Serial.println(WiFi.localIP());
  server.on("/", handleRoot);
  server.on("/setLED", handleLED);
  server.on("/readADC", handleADC);
  server.begin();
  Serial.println("HTTP server started");
}
void loop(void) {
  server.handleClient(); //Handle client requests
}
```

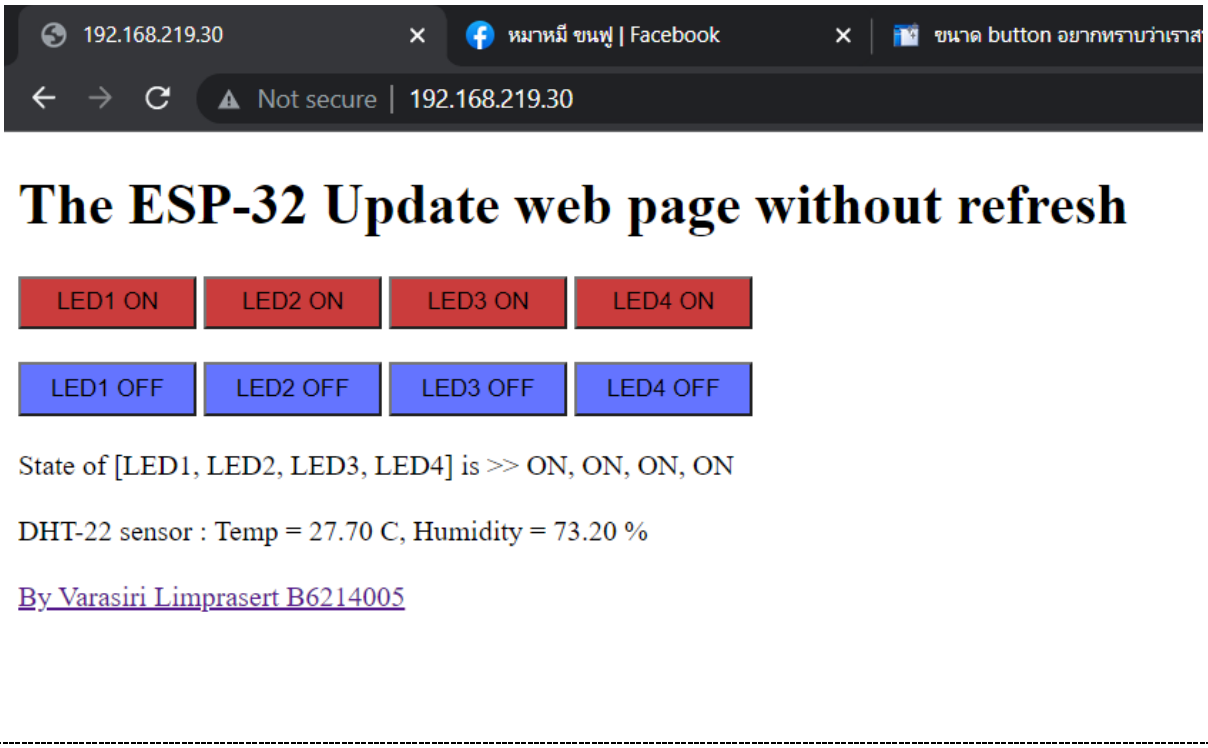
Index.h

```
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);width:100px;height:30px">LED1 ON</button>
<button type="button" onclick="sendData(21)" style="background: rgb(202, 60,
60);width:100px;height:30px">LED2 ON</button>
<button type="button" onclick="sendData(31)" style="background: rgb(202, 60,
60);width:100px;height:30px">LED3 ON</button>
<button type="button" onclick="sendData(41)" style="background: rgb(202, 60,
60);width:100px;height:30px">LED4 ON</button><br><br>
<button type="button" onclick="sendData(10)" style="background:
rgb(100,116,255);width:100px;height:30px">LED1 OFF</button>
<button type="button" onclick="sendData(20)" style="background:
rgb(100,116,255);width:100px;height:30px">LED2 OFF</button>
<button type="button" onclick="sendData(30)" style="background:
rgb(100,116,255);width:100px;height:30px">LED3 OFF</button>
<button type="button" onclick="sendData(40)" style="background:
rgb(100,116,255);width:100px;height:30px">LED4 OFF</button><br><br>
State of [LED1, LED2, LED3, LED4] is >> <span id="LEDState">/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/chi.sweethome.50/">By Varasiri Limprasert
B6214005</a>
</body>
</html>
)=====";
```



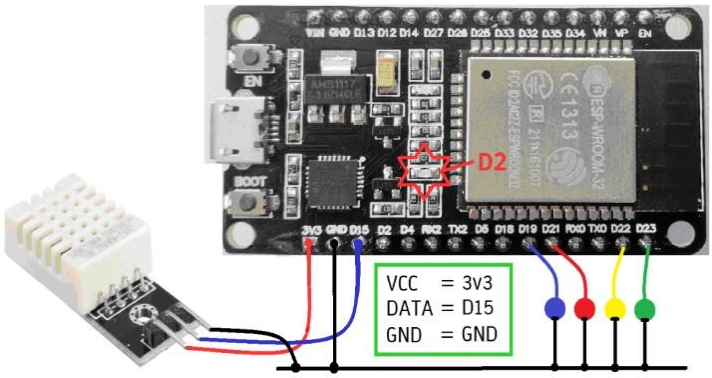



192.168.219.30
หมาหมี ขนฟู | Facebook
ขนาด button อยากทราบว่าเราส
Not secure | 192.168.219.30
The ESP-32 Update web page without refresh
LED1 ON
LED2 ON
LED3 ON
LED4 ON
LED1 OFF
LED2 OFF
LED3 OFF
LED4 OFF
State of [LED1, LED2, LED3, LED4] is >> ON, ON, ON, ON
DHT-22 sensor : Temp = 27.70 C, Humidity = 73.20 %
By Varasiri Limprasert B6214005

Quiz_203 – Publish

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

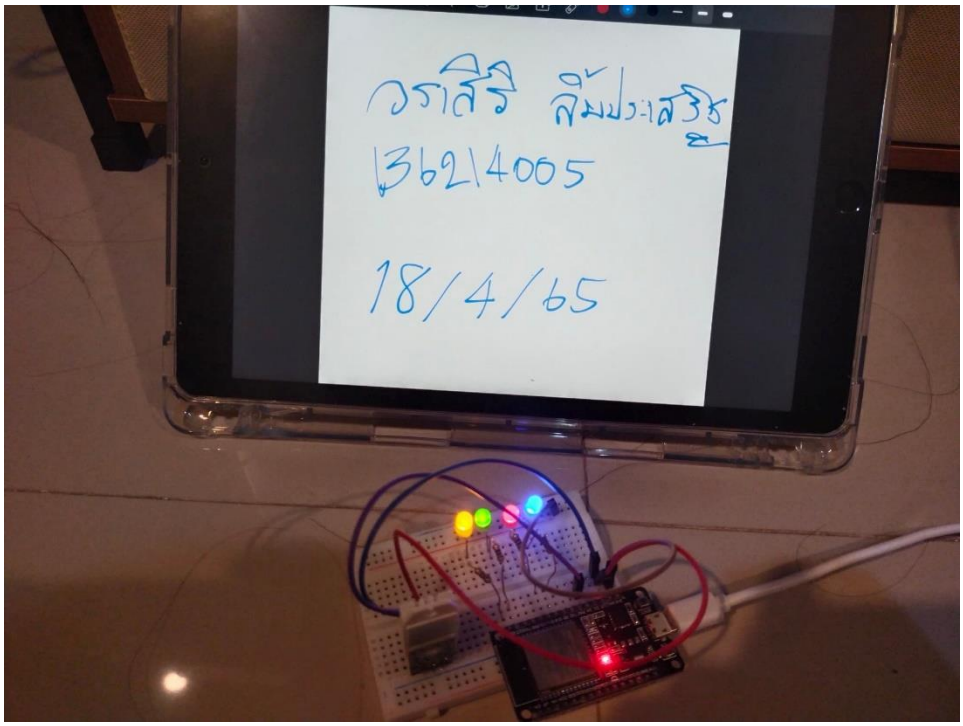
✱○○○(Blink)	หากการอ่านค่าแล้วเป็น null, หรือไม่มีเซ็นเซอร์
●○○○	ช่วงของอุณหภูมิ (-∞, 24)
●●○○	ช่วงของอุณหภูมิ [24,26)
●●●○	ช่วงของอุณหภูมิ [26,28)
●●●●	ช่วงของอุณหภูมิ [28,30)
✱✱✱✱(Blink)	ช่วงของอุณหภูมิ [30,∞)

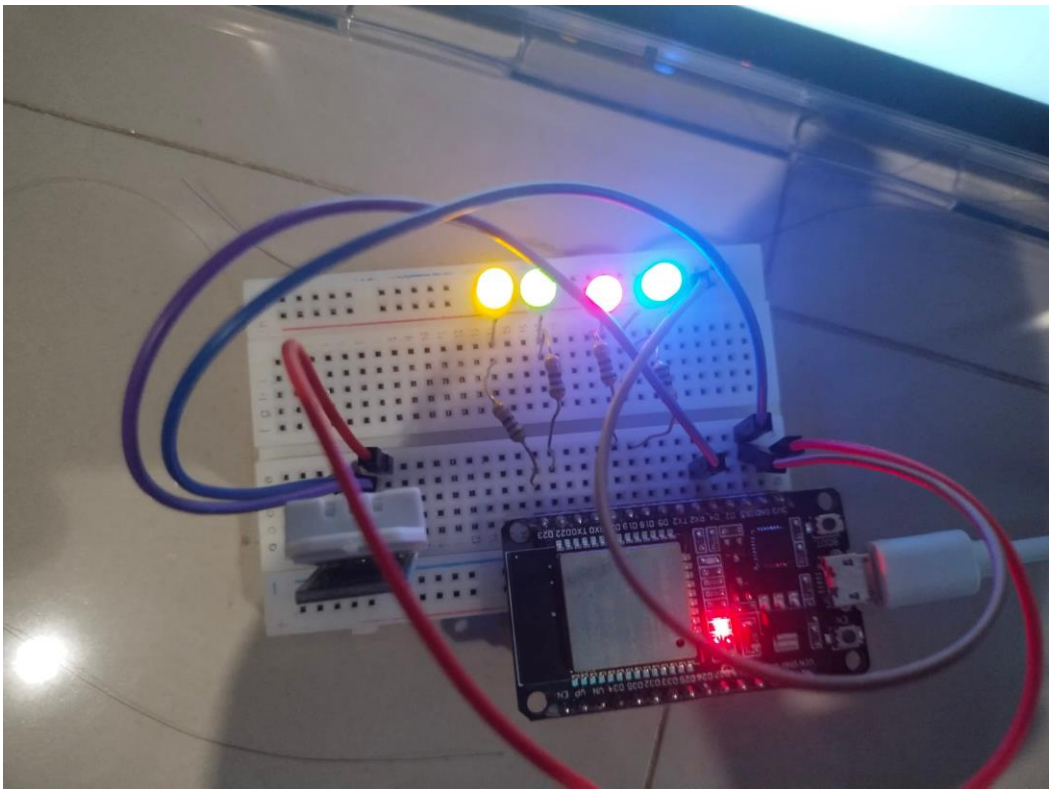


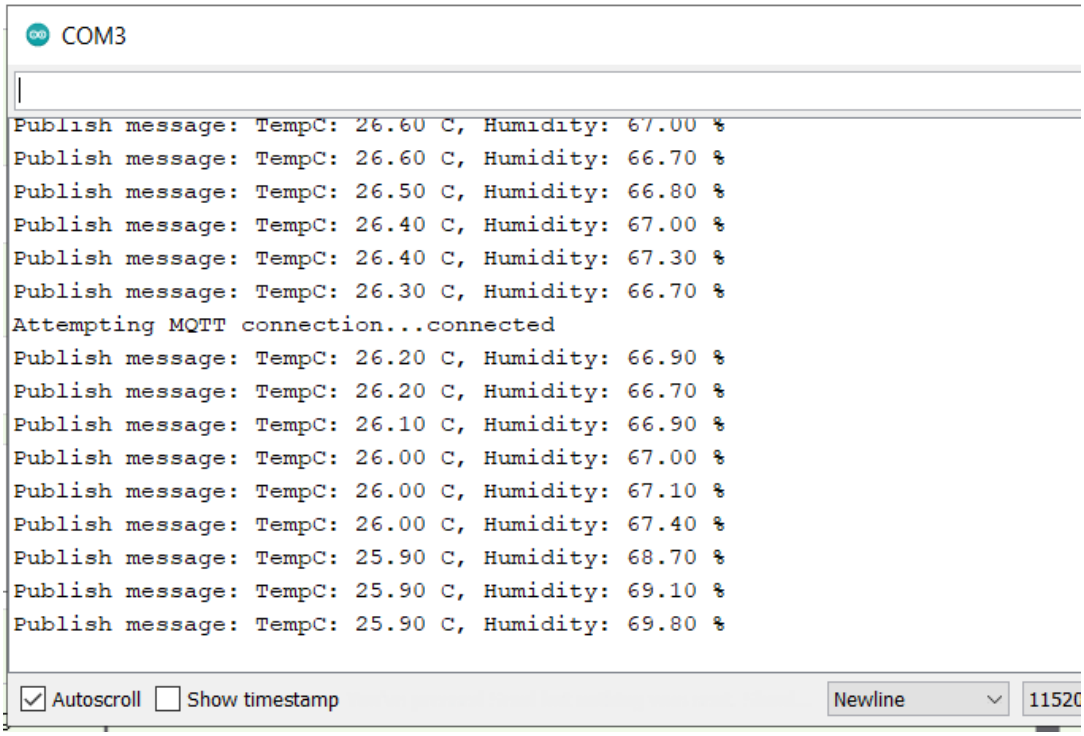

```
#include <WiFi.h>
#include <Wire.h>
#include <PubSubClient.h>
#include "DHTesp.h"

DHTesp dht;
#define PinLED1 18
#define PinLED2 19
#define PinLED3 22
#define PinLED4 23
#define DHT22_Pin 4

float h, t;
int blinkStatus = 1;
int LED_PinArray[] = {PinLED1, PinLED2, PinLED3, PinLED4};
int LED_StsArray[] = {0, 0, 0, 0};

const char* ssid = "V2036";
const char* password = "fnafchica";
const char* mqtt_server = "test.mosquitto.org";
const char* topic1 = "bearish";
String ledState1 = "NA";
WiFiClient espClient;
PubSubClient client(espClient);
long lastMsg = 0;
char msg[50];
int value = 0;
void setup_wifi() {
  delay(10);
```

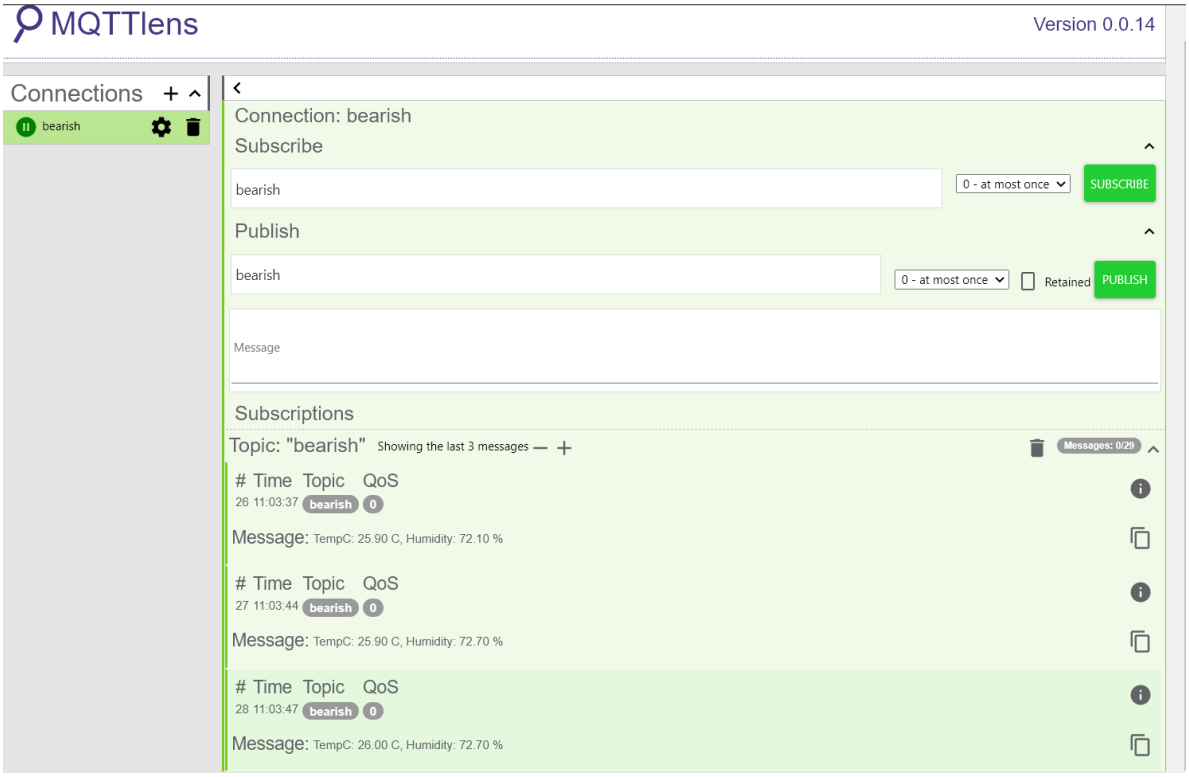
```
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500); Serial.print(".");
  }
  randomSeed(micros());
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  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);
}
```


วราสิริ สิมมาเสริฐ
13621405
18/4/65

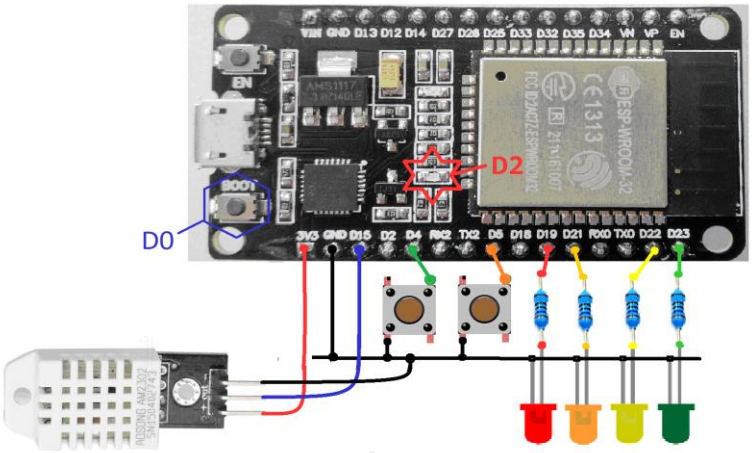



COM3
Publish message: TempC: 26.60 C, Humidity: 67.00 %
Publish message: TempC: 26.60 C, Humidity: 66.70 %
Publish message: TempC: 26.50 C, Humidity: 66.80 %
Publish message: TempC: 26.40 C, Humidity: 67.00 %
Publish message: TempC: 26.40 C, Humidity: 67.30 %
Publish message: TempC: 26.30 C, Humidity: 66.70 %
Attempting MQTT connection...connected
Publish message: TempC: 26.20 C, Humidity: 66.90 %
Publish message: TempC: 26.20 C, Humidity: 66.70 %
Publish message: TempC: 26.10 C, Humidity: 66.90 %
Publish message: TempC: 26.00 C, Humidity: 67.00 %
Publish message: TempC: 26.00 C, Humidity: 67.10 %
Publish message: TempC: 26.00 C, Humidity: 67.40 %
Publish message: TempC: 25.90 C, Humidity: 68.70 %
Publish message: TempC: 25.90 C, Humidity: 69.10 %
Publish message: TempC: 25.90 C, Humidity: 69.80 %
Autoscroll
Show timestamp
Newline
11520


MQTTlens
Version 0.0.14
Connections
bearish
Connection: bearish
Subscribe
bearish
0 - at most once
SUBSCRIBE
Publish
bearish
0 - at most once
Retained
PUBLISH
Message
Subscriptions
Topic: "bearish"
Showing the last 3 messages
Messages: 0/29
Time Topic QoS
26 11:03:37 bearish 0
Message: TempC: 25.90 C, Humidity: 72.10 %
Time Topic QoS
27 11:03:44 bearish 0
Message: TempC: 25.90 C, Humidity: 72.70 %
Time Topic QoS
28 11:03:47 bearish 0
Message: TempC: 26.00 C, Humidity: 72.70 %

Quiz_204 – Publish and Subscribe

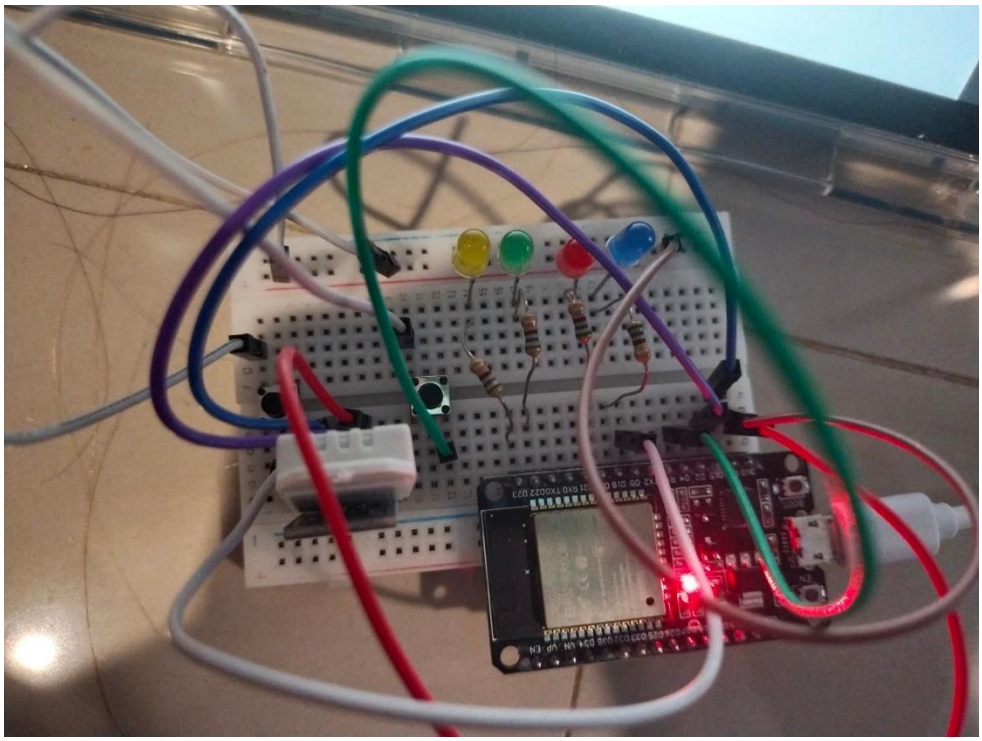
- อ่านค่า DHT-22 แล้วส่งไปยัง MQTT Broker ทุกๆ 5 วินาที
- ควบคุมการปิดเปิด 4 LED
- รับค่าสวิตซ์กำหนด SW1 แจ้ง Overheat Alarm, SW2 แจ้ง Intruders Alarm

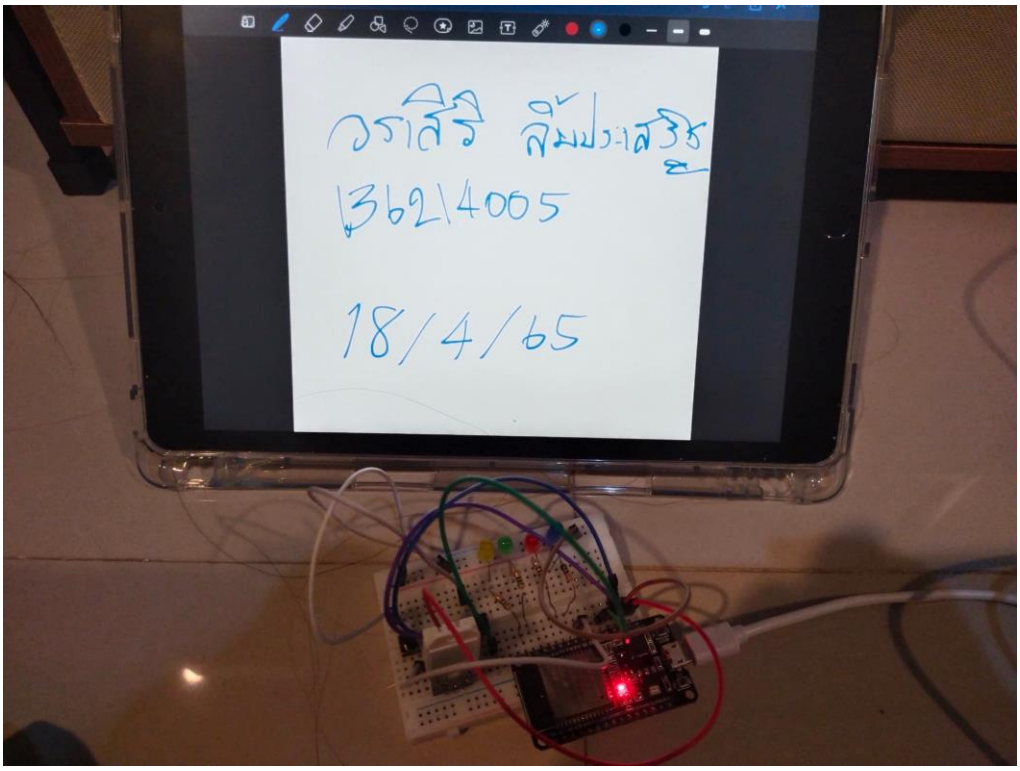


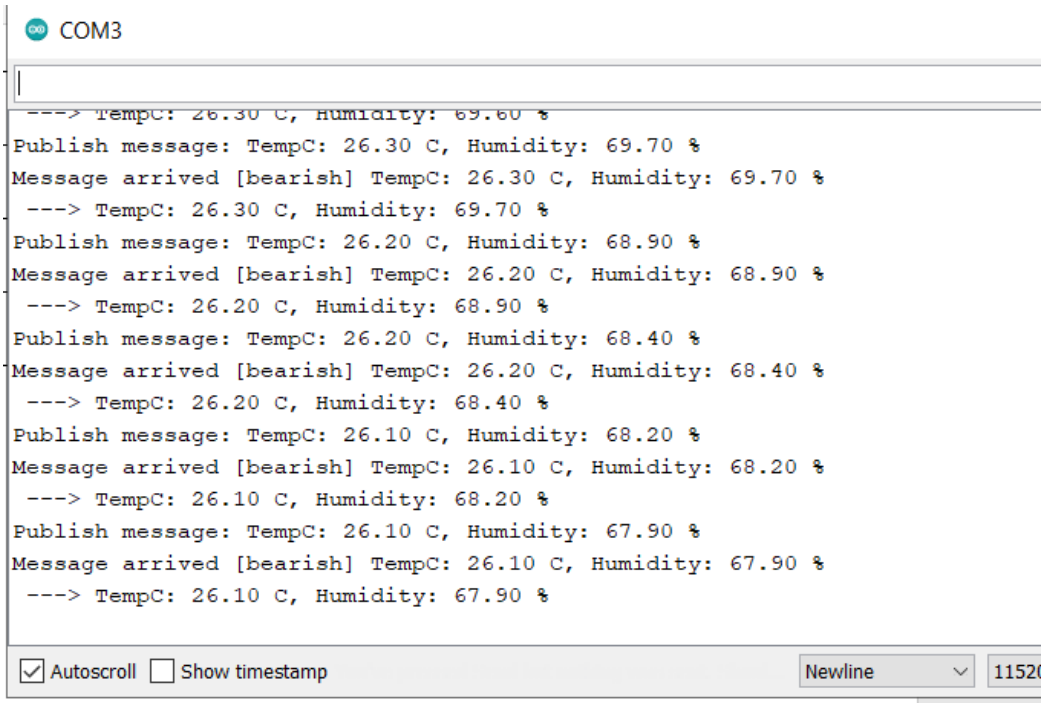

```
#include <WiFi.h>
#include <Wire.h>
#include <PubSubClient.h>
#include "DHTesp.h"

DHTesp dht;
#define testLED1 18
#define testLED2 19
#define testLED3 22
#define testLED4 23
#define DHT22_Pin 15

const char* ssid = "V2036";
const char* password = "fnafchica";
const char* mqtt_server = "test.mosquitto.org";
const char* topic1 = "bearish";
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);
```

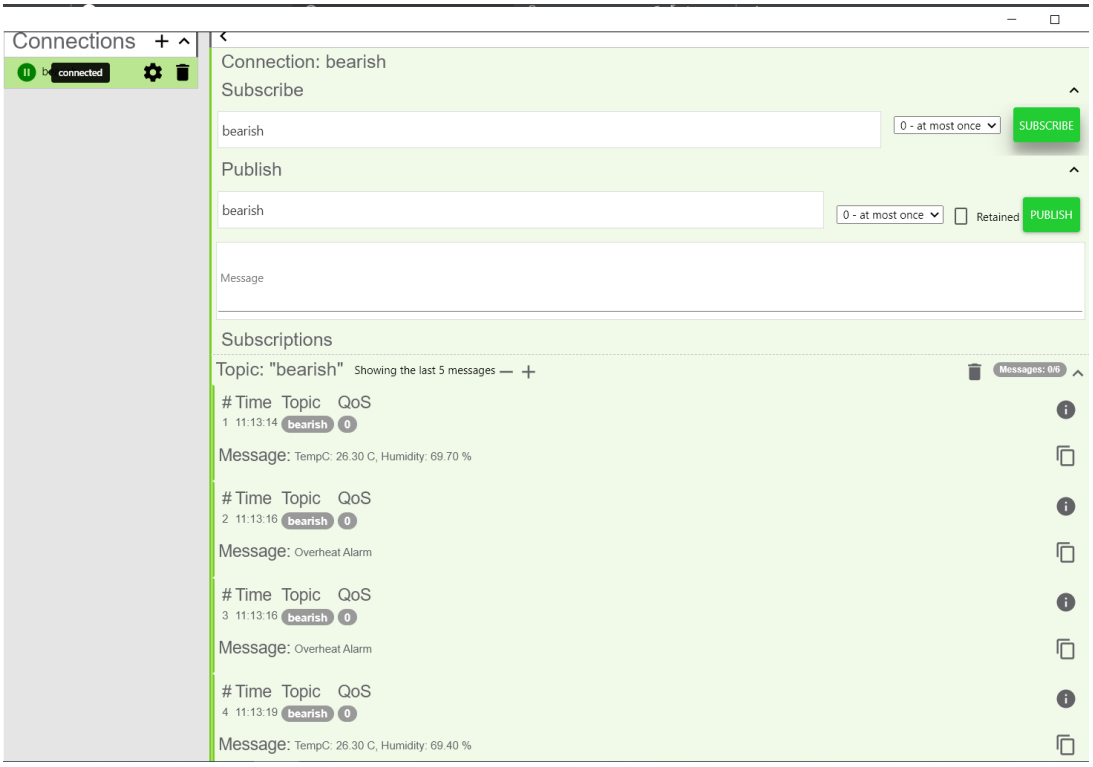
```
  while (WiFi.status() != WL_CONNECTED) {
    delay(500); Serial.print(".");
  }
  randomSeed(micros());
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  pinMode(testLED1, OUTPUT);
  pinMode(testLED2, OUTPUT);
  pinMode(testLED3, 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);
  }
}
```




วราสิริ ลิ้มประเสริฐ
6214005
18/4/65


COM3
---> TempC: 26.30 C, Humidity: 69.60 %
Publish message: TempC: 26.30 C, Humidity: 69.70 %
Message arrived [bearish] TempC: 26.30 C, Humidity: 69.70 %
---> TempC: 26.30 C, Humidity: 69.70 %
Publish message: TempC: 26.20 C, Humidity: 68.90 %
Message arrived [bearish] TempC: 26.20 C, Humidity: 68.90 %
---> TempC: 26.20 C, Humidity: 68.90 %
Publish message: TempC: 26.20 C, Humidity: 68.40 %
Message arrived [bearish] TempC: 26.20 C, Humidity: 68.40 %
---> TempC: 26.20 C, Humidity: 68.40 %
Publish message: TempC: 26.10 C, Humidity: 68.20 %
Message arrived [bearish] TempC: 26.10 C, Humidity: 68.20 %
---> TempC: 26.10 C, Humidity: 68.20 %
Publish message: TempC: 26.10 C, Humidity: 67.90 %
Message arrived [bearish] TempC: 26.10 C, Humidity: 67.90 %
---> TempC: 26.10 C, Humidity: 67.90 %
Autoscroll
Show timestamp
Newline


Connections
connected
Connection: bearish
Subscribe
bearish
0 - at most once
SUBSCRIBE
Publish
bearish
0 - at most once
Retained
PUBLISH
Message
Subscriptions
Topic: "bearish"
Showing the last 5 messages
Messages: 0/6
Time Topic QoS
1 11:13:14 bearish 0
Message: TempC: 26.30 C, Humidity: 69.70 %
Time Topic QoS
2 11:13:16 bearish 0
Message: Overheat Alarm
Time Topic QoS
3 11:13:16 bearish 0
Message: Overheat Alarm
Time Topic QoS
4 11:13:19 bearish 0
Message: TempC: 26.30 C, Humidity: 69.40 %