

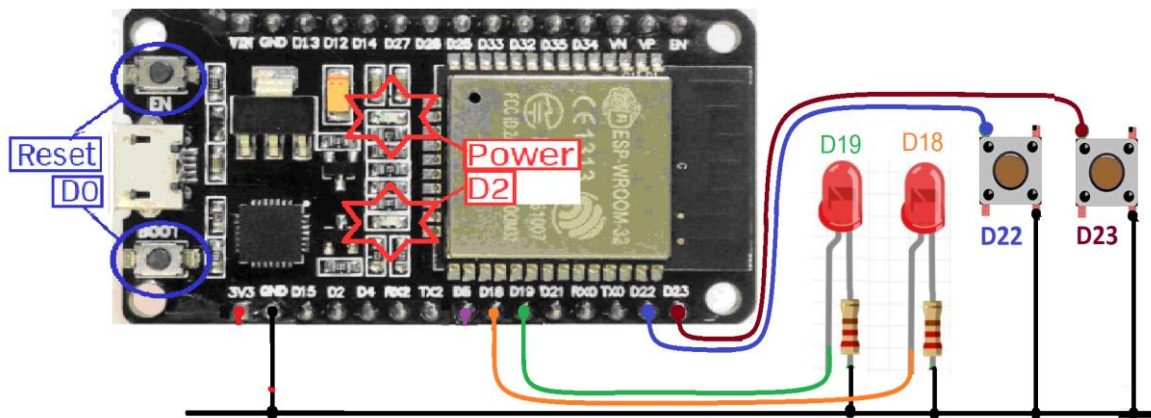
## การใช้งาน ThingsBoard IoTs Platform เพื่อสร้างและจัดการระบบอัจฉริยะ ThingsBoard IoTs Platform for smart system

ชื่อ-สกุล : นางสาวณัฐชยา ผ่องกุล B6226718

### 6/6 -- คำถามท้ายบทเพื่อทดสอบความเข้าใจ

#### Quiz\_101 – กดติด กดดับ 2 ชุด

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



#### Arduino

```

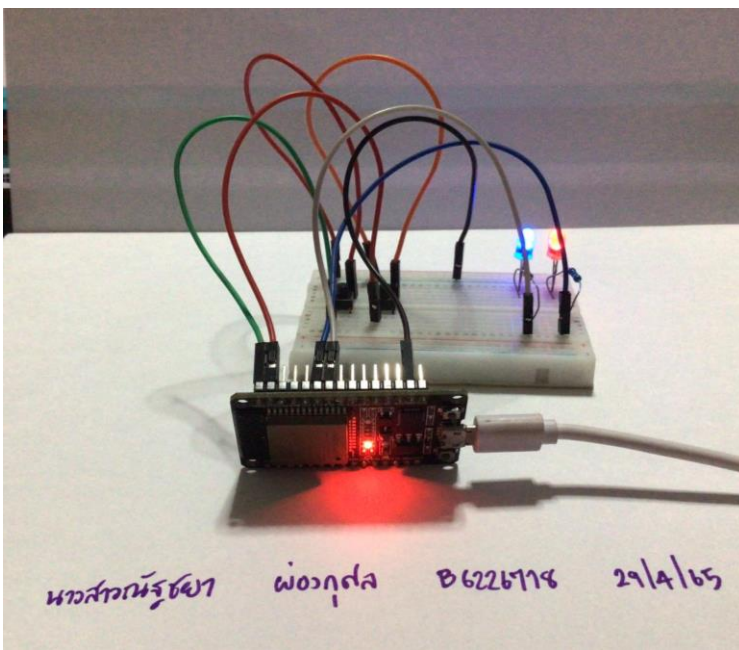
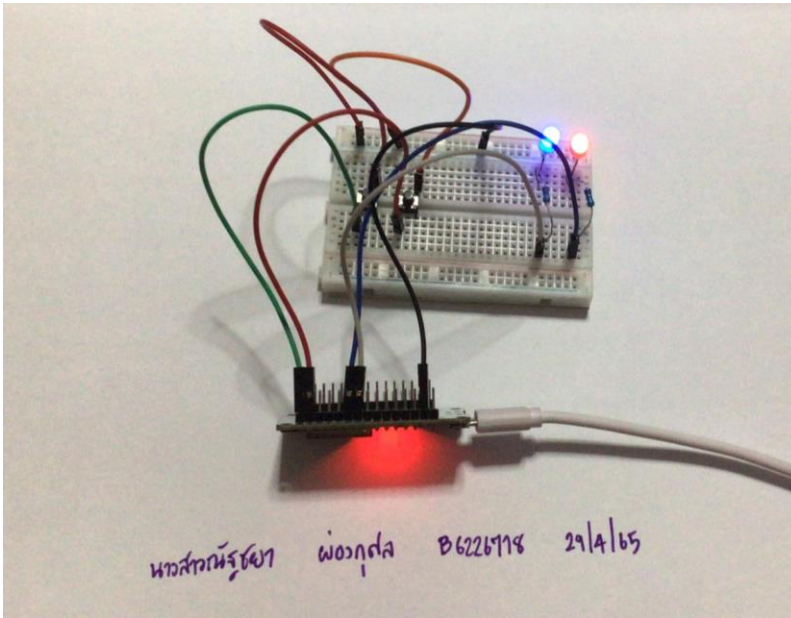
D1 | Arduino 1.8.13
File Edit Sketch Tools Help
Serial Monitor
D1 _ConnectWifi.h _ThingBoardRPC.h
1 #define pushButton1 22
2 #define LEDPin1 18
3 #define pushButton2 23
4 #define LEDPin2 19
5 int buttonState1 = 0;
6 int buttonState2 = 0;
7 void setup() {
8   Serial.begin(115200);
9   pinMode(pushButton1, INPUT_PULLUP);
10  pinMode(LEDPin1, OUTPUT);
11  pinMode(pushButton2, INPUT_PULLUP);
12  pinMode(LEDPin2, OUTPUT);
13 }
14 void loop() {
15   if (digitalRead(pushButton1) == LOW) {
16     delay(20);
17     buttonState1 = 1 - buttonState1;
18     digitalWrite(LEDPin1, buttonState1);
19     while (digitalRead(pushButton1) == LOW);
20     delay(20);
21   }
22   if (digitalRead(pushButton2) == LOW) {
23     delay(20);
24     buttonState2 = 1 - buttonState2;
25     digitalWrite(LEDPin2, buttonState2);
26   }
27 }
Leaving...
Hard resetting via RTS pin...
ESP8266 Pico K4 on COM3
27°C มีฝนฟ้า... ENG 19:00

```

```
#define pushButton1 22
```

```
#define LEDPin1 18
#define pushButton2 23
#define LEDPin2 19
int buttonState1 = 0;
int buttonState2 = 0;
void setup() {
  Serial.begin(115200);
  pinMode(pushButton1, INPUT_PULLUP);
  pinMode(LEDPin1, OUTPUT);
  pinMode(pushButton2, INPUT_PULLUP);
  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 – Web Control 4 LED and Monitor Humid/Temperature

- เพิ่มเติมจาก Q202 อยากได้ปุ่มสำหรับคุมปิด-เปิด หลอดไฟ LED 4 ดวง
- อยากมีกิด Link ไปที่หน้า FB ของตัวเอง
- [https://www.colorhexa.com/008cba?fbclid=IwAR3dIZ\\_gRgDWmREmnzukuLbMxV3pOHY4YIPuLEz8-ZzTOX2VhWxcH2QjLGk](https://www.colorhexa.com/008cba?fbclid=IwAR3dIZ_gRgDWmREmnzukuLbMxV3pOHY4YIPuLEz8-ZzTOX2VhWxcH2QjLGk)

**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, OFF, OFF, ON

DHT-22 sensor : Temp = 28.10 C, Humidity = 43.90 %

By [Wichai Srisuruk](#)

## Arduino

```

1 const char MAIN_page[] PROGMEM = R"=====
2 <!DOCTYPE html><html><body><div id="demo">
3 <h1>The ESP-32 Update web page without refresh</h1>
4 <button type="button" onclick="sendData(11)" style="background: rgb(202, 60, 60);">LED1 ON</button>
5 <button type="button" onclick="sendData(21)" style="background: rgb(202, 60, 60);">LED2 ON</button>
6 <button type="button" onclick="sendData(31)" style="background: rgb(202, 60, 60);">LED3 ON</button>
7 <button type="button" onclick="sendData(41)" style="background: rgb(202, 60, 60);">LED4 ON</button>
8 <button type="button" onclick="sendData(10)" style="background: rgb(100, 116, 255);">LED1 OFF</button>
9 <button type="button" onclick="sendData(20)" style="background: rgb(100, 116, 255);">LED2 OFF</button>
10 <button type="button" onclick="sendData(30)" style="background: rgb(100, 116, 255);">LED3 OFF</button>
11 <button type="button" onclick="sendData(40)" style="background: rgb(100, 116, 255);">LED4 OFF</button>
12 State of [LED1, LED2, LED3, LED4] is >> <span
13 id="LEDState">NA</span><br></div></div><br>
14 DHT-22 sensor : <span id="ADCValue">0</span><br></div></div><script>
15 function sendData(led) {
16   var xhttp = new XMLHttpRequest();
17   xhttp.onreadystatechange = function() {
18     if (this.readyState == 4 && this.status == 200) {
19       document.getElementById("LEDState").innerHTML =
20       this.responseText;
21     }
22   };
23   xhttp.open("GET", "setLED?LEDState="+led, true);
24   xhttp.send();
25 }
26 setInterval(function() { // Call a function repetatively with 2 Second interval

```

Serial Monitor Output:

```

Brownout detector was triggered
ets Jun  8 2016 00:22:57
rst:0xc (SW_CPU_RESET),boot:0x17 (SPI_FAST_FLASH_BOOT)
config: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:8896
load:0x40080400,len:5816
entry 0x400806ac

Connect to BOOK
Connected BOOK
IP address: 192.168.1.148
HTTP server started
11
21
31
41

```

```
#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 = "BOOK";
const char* password = "book1017";
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/plain", 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/plain", 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>
(The challenge didn't tell me to do it.)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/profile.php?id=100007563972020">Natchaya
Phongkuson</a></body></html>
)=====";
```

← → ↻ ▲ ไม่ปลอดภัย | 192.168.1.148

## The ESP-32 Update web page without refresh

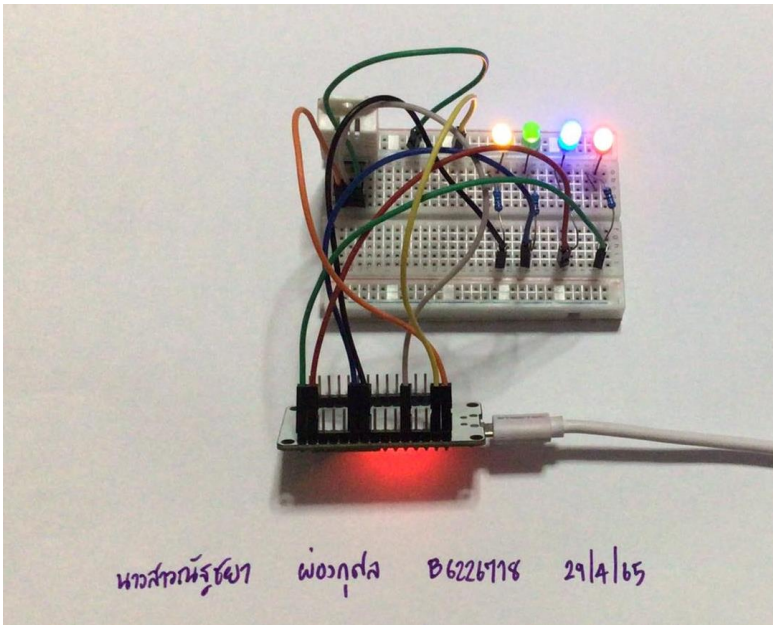
LED1 ON LED2 ON LED3 ON LED4 ON

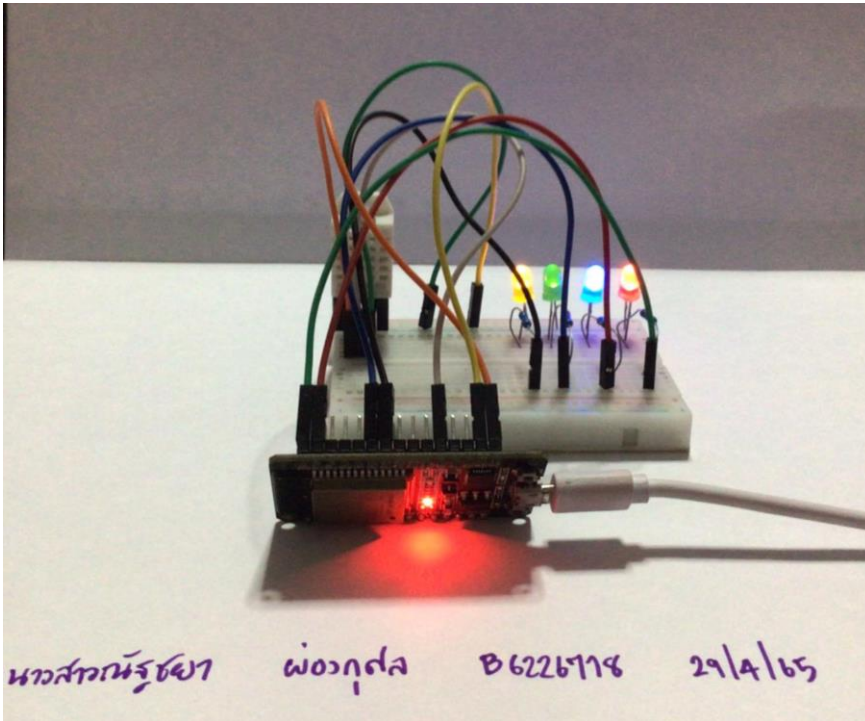
LED1 OFF LED2 OFF LED3 OFF LED4 OFF

State of [LED1, LED2, LED3, LED4] is >> ON, ON, ON, ON

DHT-22 sensor : Temp = 29.30 C, Humidity = 99.90 %

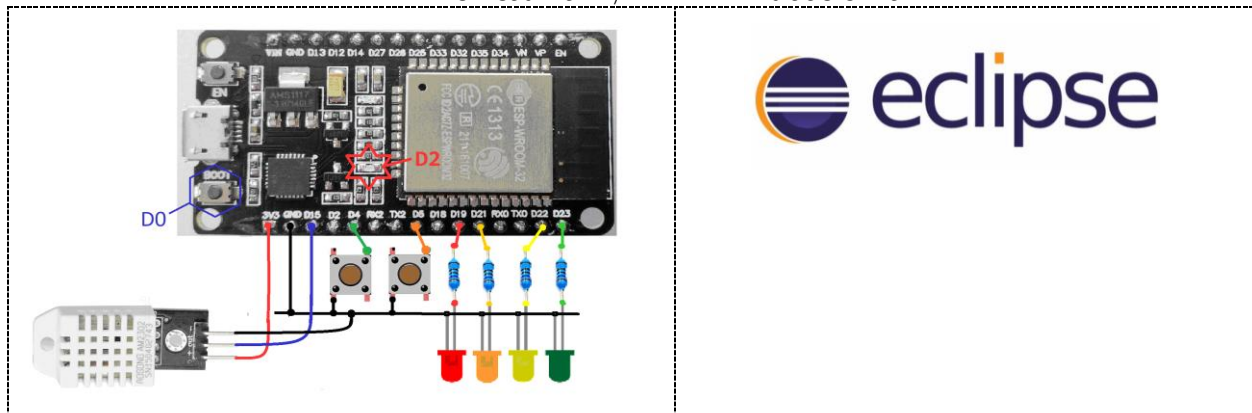
[Natchaya Phongkuson](#)



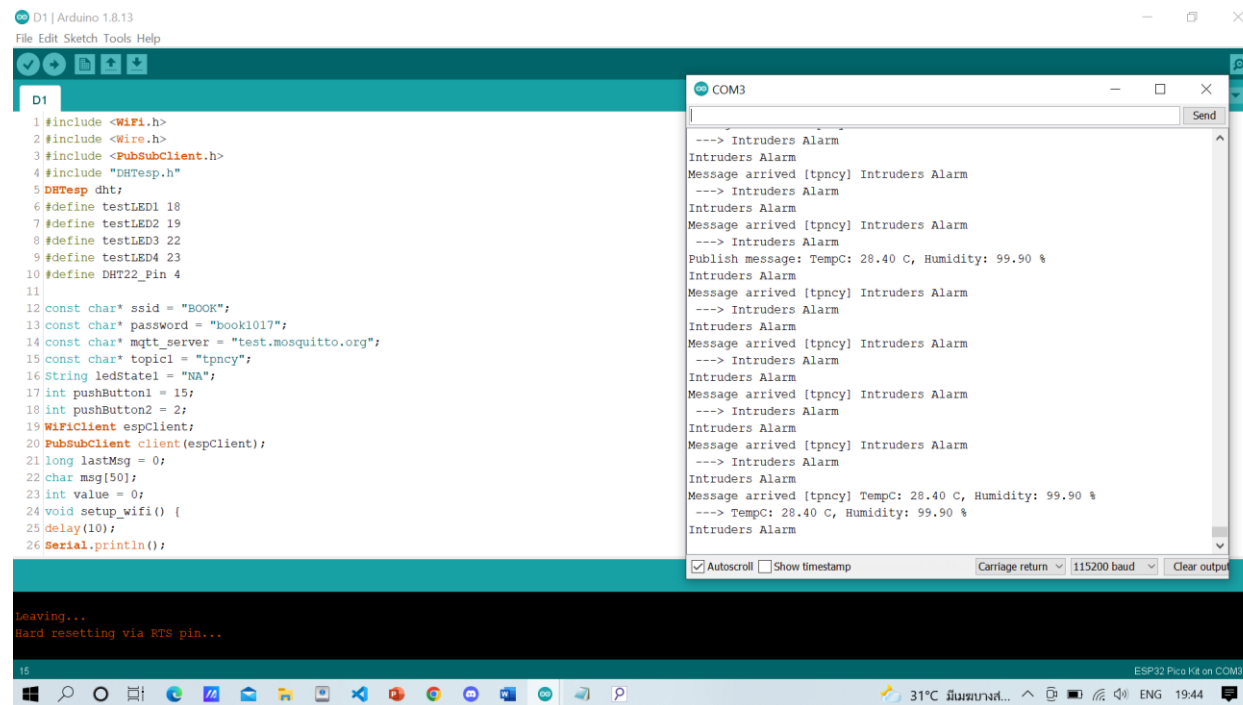


## Quiz\_103 – Pub/Sub Data from (DHT22 + 4 LED + 2 Switch)

- อ่านค่า DHT-22 แล้วส่งไปยัง MQTT Broker ทุกๆ 5 วินาที
- กำหนดให้ใช้ mqtt.eclipse.org เป็น Broker
- ควบคุมการปิดเปิด 4 LED
- รับค่าสวิตช์กำหนด SW1 แจ้ง Overheat Alarm, SW2 แจ้ง Intruders Alarm



## Arduino



#include &lt;WiFi.h&gt;

#include &lt;Wire.h&gt;

#include &lt;PubSubClient.h&gt;

#include "DHTesp.h"

```
DHTesp dht;
#define testLED1 18
#define testLED2 19
#define testLED3 22
#define testLED4 23
#define DHT22_Pin 4

const char* ssid = "BOOK";
const char* password = "book1017";
const char* mqtt_server = "test.mosquitto.org";
const char* topic1 = "tpncy";
String ledState1 = "NA";
int pushButton1 = 15;
int pushButton2 = 2;
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(topic);
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 less than 4 characters
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);
  }

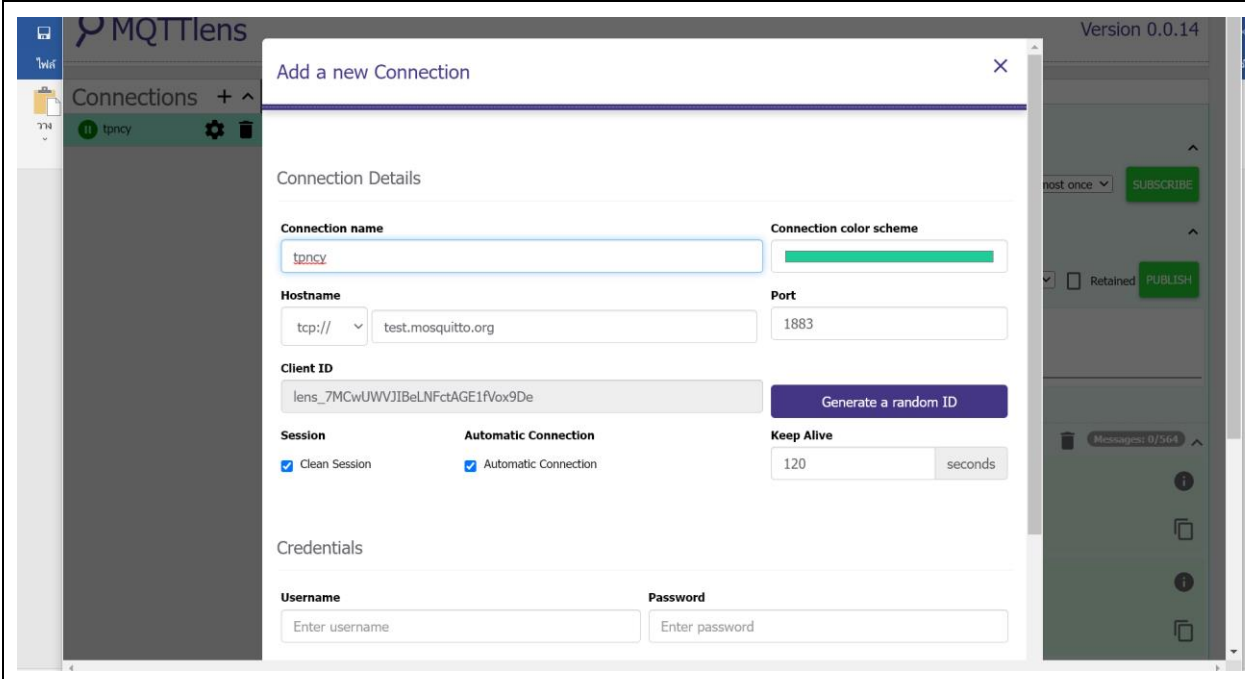
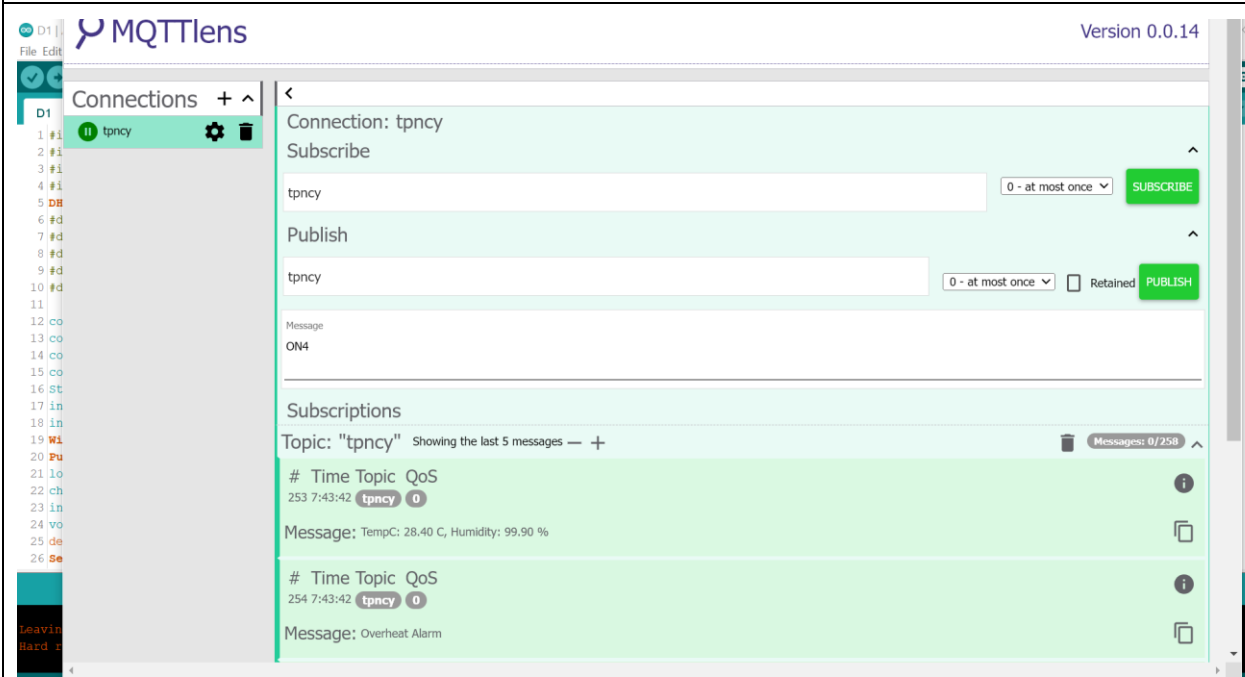
  if (digitalRead(pushButton2) == 0) {
    sprintf (msg, "Intruders Alarm");
```

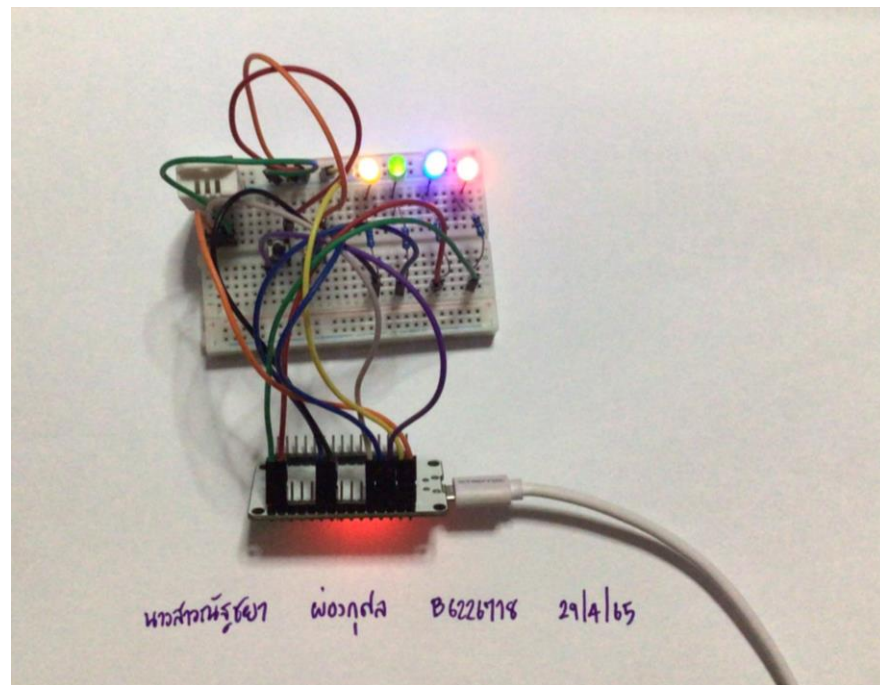
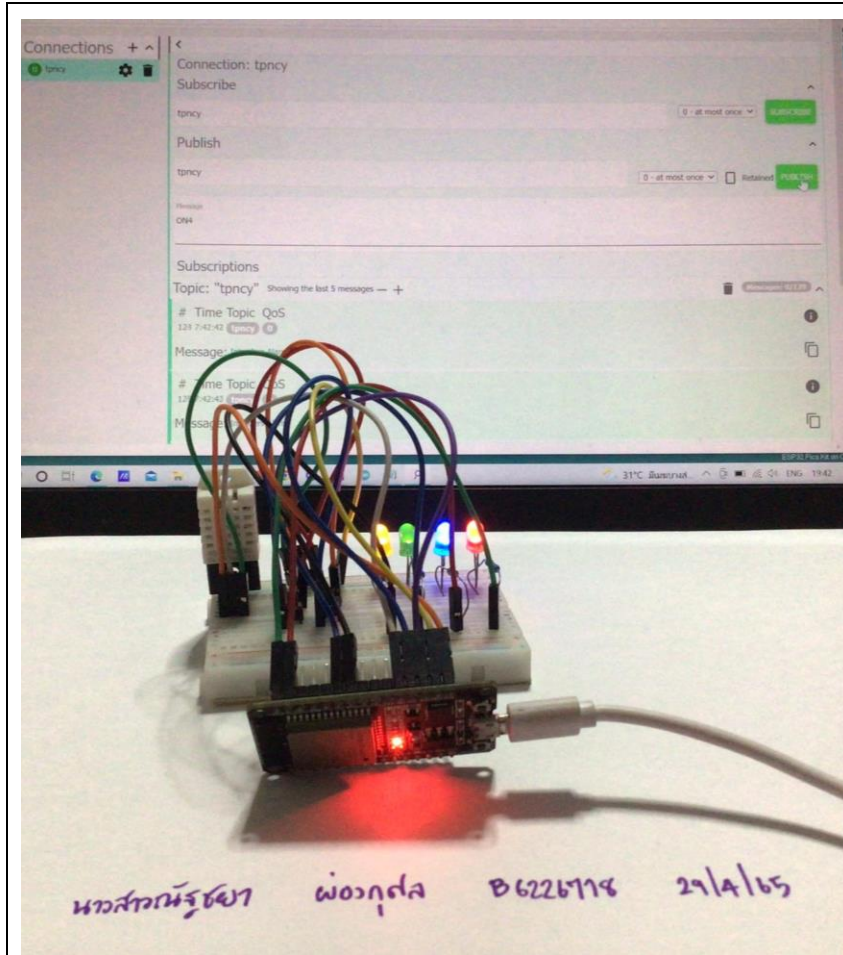


```

Serial.println(msg);
client.publish(topic1, msg);
delay(500);
}
}

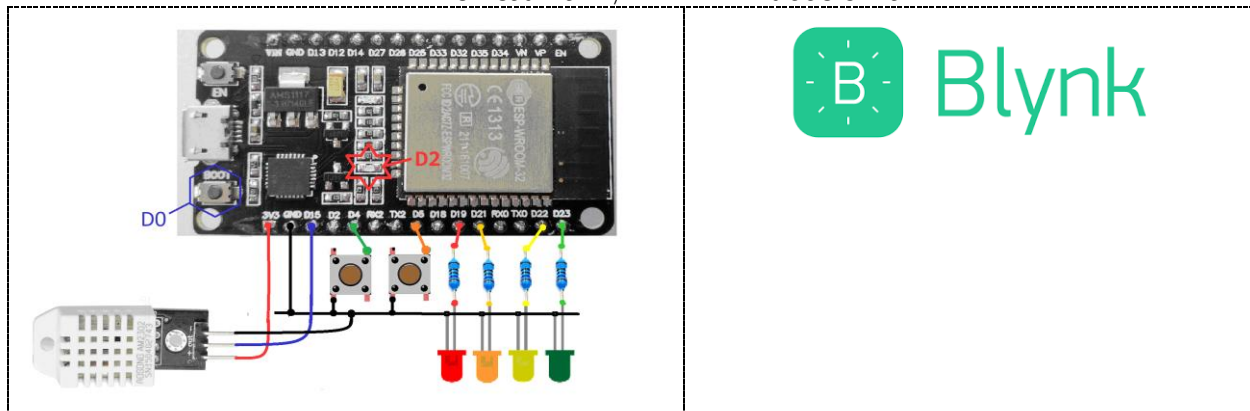
```





## Quiz\_104 – Blynk and LINE from (DHT22 + 4 LED + 2 Switch)

- ควบคุมการปิดเปิด 4 LED
- อ่านค่า DHT-22 แล้วส่งไปยัง Blynk ทุกๆ 5 วินาที
- บันทึกค่าไปยัง Google Sheet
- หากอุณหภูมิเกิน 28°C ให้แจ้งไปยัง LINE
- รับคำสั่งวัดที่กำหนด SW1 แจ้ง Overheat Alarm, SW2 แจ้ง Intruders Alarm ไปยัง LINE



## Arduino

```

D1 | Arduino 1.8.13
File Edit Sketch Tools Help

D1 |
1 #define BLYNK_PRINT Serial
2 #include <WiFi.h>
3 #include <WiFiClient.h>
4 #include <BlynkSimpleEsp32.h>
5 #include <HTTPEClient.h>
6 #include "DHTesp.h"
7
8 #define DHT22_Pin 4
9 #define sw1 15
10 #define sw2 2
11
12 char auth[] = "WPa2mtFT_6qi9RGfh3Nxc8AXs-yifnoe";
13 char ssid[] = "BOOK";
14 char pass[] = "book1017";
15 #define WebHooksKey "y2TRnq2xw1XjxELy7-ro-"
16 #define WebHooksEventName "test_GSheet"
17 #define WebHooksEventName_line "Test_Key"
18 #define My_NAME "B6226718 Natchaya Phongkuson"
19
20
21 DHTesp dht;
22
23 WidgetLED LED1(V2);
24 WidgetLED LED2(V3);
25 BlynkTimer timer;
26
Done uploading.
Leaving...
Hard resetting via RTS pin...
COM3
HTTP Response code: 200
[Line] --> Successfully sent
[13634] Connecting to blynk-cloud.com:80
Temp('C') >> 32.5, Humidity(%) >> 99.9
HTTP Response code: 200
[Google sheet] --> Successfully sent
Server Name >> http://maker.ifttt.com/trigger/Test_Key/with/key/y2TRnq2xw
json httpRequestBody >> value1=32.50
HTTP Response code: 200
[Line] --> Successfully sent
Server Name >> http://maker.ifttt.com/trigger/Test_Key/with/key/y2TRnq2xw
json httpRequestBody >> value2=Intruders Alarm
HTTP Response code: 200
[Line] --> Successfully sent
Server Name >> http://maker.ifttt.com/trigger/Test_Key/with/key/y2TRnq2xw
json httpRequestBody >> value2=Overheat Alarm
HTTP Response code: 200
[Line] --> Successfully sent
[20075] Ready (ping: 6427ms).
Temp('C') >> 32.5, Humidity(%) >> 99.9
Autoscroll Show timestamp Carriage return 115200 baud Clear output

```

```

#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

```

```
#include <HTTPClient.h>
#include "DHTesp.h"

#define DHT22_Pin 4
#define sw1 15
#define sw2 2

char auth[] = "WPa2mtFT_6qi9RGfh3Nxc8AXs-y1fnoe";
char ssid[] = "BOOK";
char pass[] = "book1017";
#define WebHooksKey "y2TRnqZxW1XjxELy7-rO-"
#define WebHooksEventName "test_GSheet"
#define WebHooksEventName_line "Test_Key"
#define My_NAME "B6226718 Natchaya Phongkuson"


DHTesp dht;


WidgetLED LED1(V2);
WidgetLED LED2(V3);
BlynkTimer timer;

void myTimerEvent() {
  float humidity = dht.getHumidity();
  float temperature = dht.getTemperature();
  Blynk.virtualWrite(V0, temperature);
  Blynk.virtualWrite(V1, humidity);
  if (digitalRead(sw1)) LED1.on();
```

```

else LED1.off();
if (digitalRead(sw2)) LED2.on();
else LED2.off();
Serial.print(" Temp('C) >> "); Serial.print(temperature, 1);
Serial.print(", Humidity(%) >> "); Serial.println(humidity, 1);
}

void setup() {
  Serial.begin(115200);
  dht.setup(DHT22_Pin, DHTesp::DHT22); // Connect DHT sensor to GPIO 15
  pinMode(sw1, INPUT_PULLDOWN);
  pinMode(sw2, INPUT_PULLDOWN);
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(1000L, myTimerEvent);
  WiFi.begin(ssid, pass);
  Serial.println("Connecting");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
}

void loop(){
  delay(5000);
  float humidity = dht.getHumidity();
  float temperature = dht.getTemperature();

  String serverName = "http://maker.ifttt.com/trigger/" +

```

```

String(WebHooksEventName) + "/with/key/" + String(WebHooksKey);
String httpRequestData = "value1=" + String(My_NAME) + "&value2=" +
String(temperature) + "&value3=" +
String(humidity);
if (WiFi.status() == WL_CONNECTED) {
  HTTPClient http;
  http.begin(serverName);
  http.addHeader("Content-Type", "application/x-www-form-urlencoded");
  int httpResponseCode = http.POST(httpRequestData);
  Serial.print("HTTP Response code: ");
  Serial.println(httpResponseCode);
  http.end();
  if (httpResponseCode == 200)
    Serial.println("[Google sheet] --> Successfully sent");
  else
    Serial.println("[Google sheet] --> Failed!");
}
else {
  Serial.println("WiFi Disconnected");
}

/// if temp > 28 C send notifications >> line
if (temperature > 28) {
  String serverName = "http://maker.ifttt.com/trigger/" +
String(WebHooksEventName_line) + "/with/key/" + String(WebHooksKey);
  String httpRequestData = "value1=" + String(temperature);
  Serial.println();
  Serial.println("Server Name >> " + serverName);
  Serial.println("json httpRequestData >> " + httpRequestData);
}

```

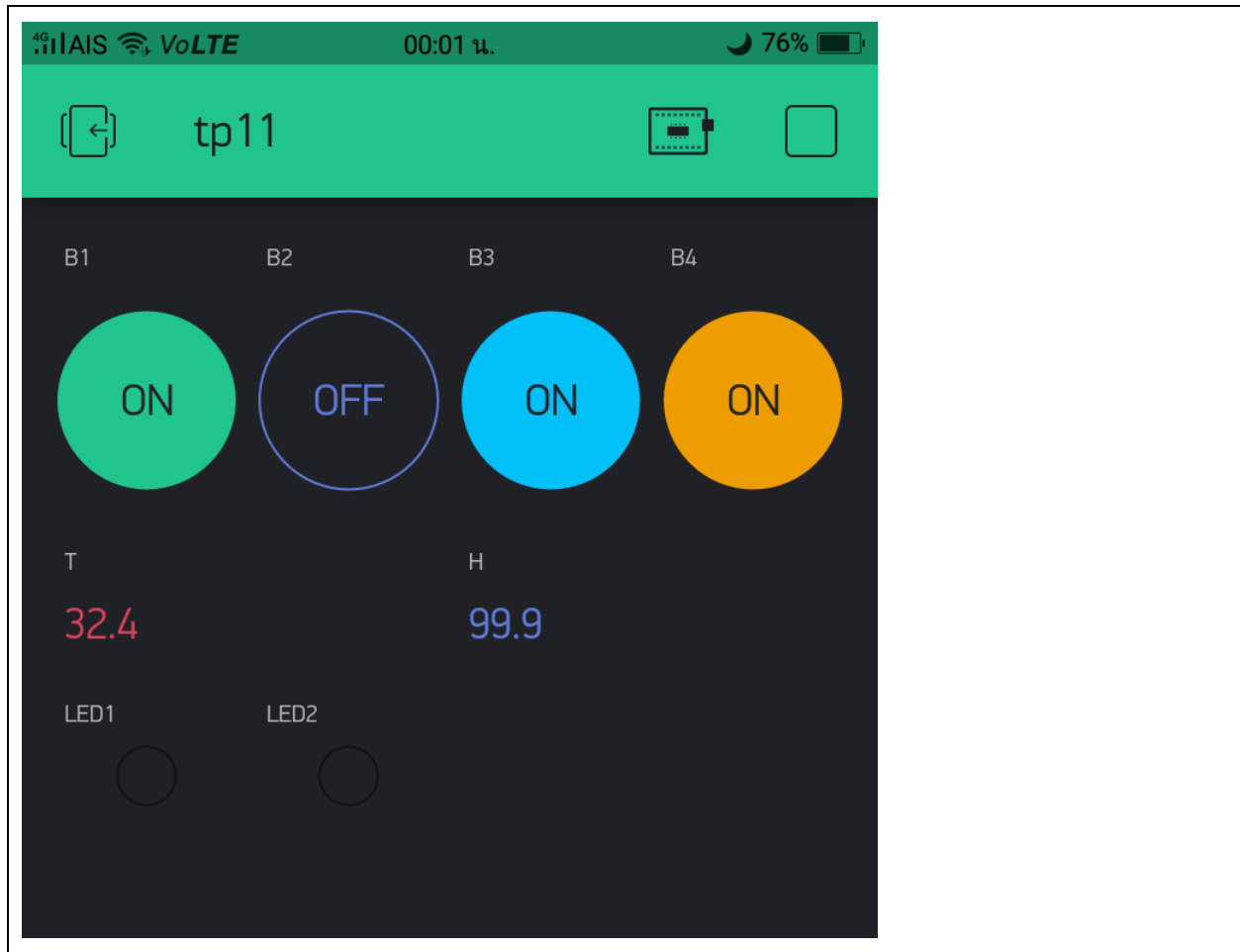
```
if (WiFi.status() == WL_CONNECTED) {  
    HTTPClient http;  
    http.begin(serverName);  
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");  
    int httpResponseCode = http.POST(httpRequestData);  
    Serial.print("HTTP Response code: ");  
    Serial.println(httpResponseCode);  
    http.end();  
    if (httpResponseCode == 200)  
        Serial.println("[Line] --> Successfully sent");  
    else  
        Serial.println("[Line] --> Failed!");  
    }  
    else {  
        Serial.println("WiFi Disconnected");  
    }  
  
}  
  
if (digitalRead(sw1) == LOW) {  
    String serverName = "http://maker.ifttt.com/trigger/" +  
    String(WebHooksEventName_line) + "/with/key/" + String(WebHooksKey);  
    String httpRequestData = "value3=" + String("Intruders Alarm");  
    Serial.println();  
    Serial.println("Server Name >> " + serverName);  
    Serial.println("json httpRequestData >> " + httpRequestData);  
    if (WiFi.status() == WL_CONNECTED) {  
        HTTPClient http;  
        http.begin(serverName);
```

```
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
int httpResponseCode = http.POST(httpRequestData);
Serial.print("HTTP Response code: ");
Serial.println(httpResponseCode);
http.end();
if (httpResponseCode == 200)
    Serial.println("[Line] --> Successfully sent");
else
    Serial.println("[Line] --> Failed!");
}
else {
    Serial.println("WiFi Disconnected");
}
}

if (digitalRead(sw2) == LOW) {
    String serverName = "http://maker.ifttt.com/trigger/" +
    String(WebHooksEventName_line) + "/with/key/" + String(WebHooksKey);
    String httpRequestData = "value3=" + String("Overheat Alarm");
    Serial.println();
    Serial.println("Server Name >> " + serverName);
    Serial.println("json httpRequestData >> " + httpRequestData);
    if (WiFi.status() == WL_CONNECTED) {
        HTTPClient http;
        http.begin(serverName);
        http.addHeader("Content-Type", "application/x-www-form-urlencoded");
        int httpResponseCode = http.POST(httpRequestData);
        Serial.print("HTTP Response code: ");
        Serial.println(httpResponseCode);
    }
}
```



```
http.end();  
if (httpResponseCode == 200)  
  Serial.println("[Line] --> Successfully sent");  
else  
  Serial.println("[Line] --> Failed!");  
}  
else {  
  Serial.println("WiFi Disconnected");  
}  
}  
  
Blynk.run();  
timer.run(); // running timer every 250ms  
}
```



IFTTT Maker Webhooks Events

ไฟล์ แก้ไข ดู แทรก รูปแบบ ข้อมูล เครื่องมือ ส่วนขยาย ความช่วยเหลือ แก้ไขล่าสุด 2 นาทีที่ผ่านมาโดย ฤกษ์ยา ผ่องทศล

100% B % .0 .00 123 คำเริ่มต้น (A... 10 B I A

	A	B	C	D	E	F	G	H	I	J	K	L
22	April 29, 2022 at test_GSheet	B6226718 Natch	32.7		99.9							
23	April 29, 2022 at test_GSheet	B6226718 Natch	32.7		99.9							
24	April 29, 2022 at test_GSheet		Intruders Alarm									
25	April 29, 2022 at test_GSheet		Intruders Alarm									
26	April 29, 2022 at test_GSheet	B6226718 Natch	32.7		99.9							
27	April 29, 2022 at test_GSheet	B6226718 Natch	32.7		99.9							
28	April 29, 2022 at test_GSheet		Intruders Alarm									
29	April 29, 2022 at test_GSheet	B6226718 Natch	32.6		99.9							
30	April 29, 2022 at test_GSheet		Intruders Alarm									
31	April 29, 2022 at test_GSheet	B6226718 Natch	32.6		99.9							
32	April 29, 2022 at test_GSheet	B6226718 Natch	32.7		99.9							
33	April 29, 2022 at test_GSheet		Intruders Alarm									
34	April 29, 2022 at test_GSheet		Intruders Alarm									
35	April 29, 2022 at test_GSheet	B6226718 Natch	35.2		99.9							
36	April 29, 2022 at test_GSheet		Intruders Alarm									
37	April 29, 2022 at test_GSheet		Intruders Alarm									
38	April 29, 2022 at test_GSheet	B6226718 Natch	32.5		99.9							
39	April 29, 2022 at test_GSheet		Intruders Alarm									
40	April 29, 2022 at test_GSheet	B6226718 Natch	32.5		99.9							

