

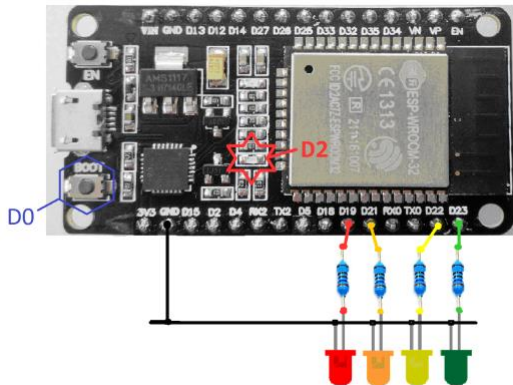
แนวทางการใช้งานอินเทอร์เน็ตของสรรพสิ่งในระบบการผลิต

IoT Approaches to Manufacturing System

ชื่อ-สกุล : นายธนพล กาศักดิ์

3/3. คำถามท้ายบทเพื่อทดสอบความเข้าใจ

Quiz_301 – 4 External LED Control

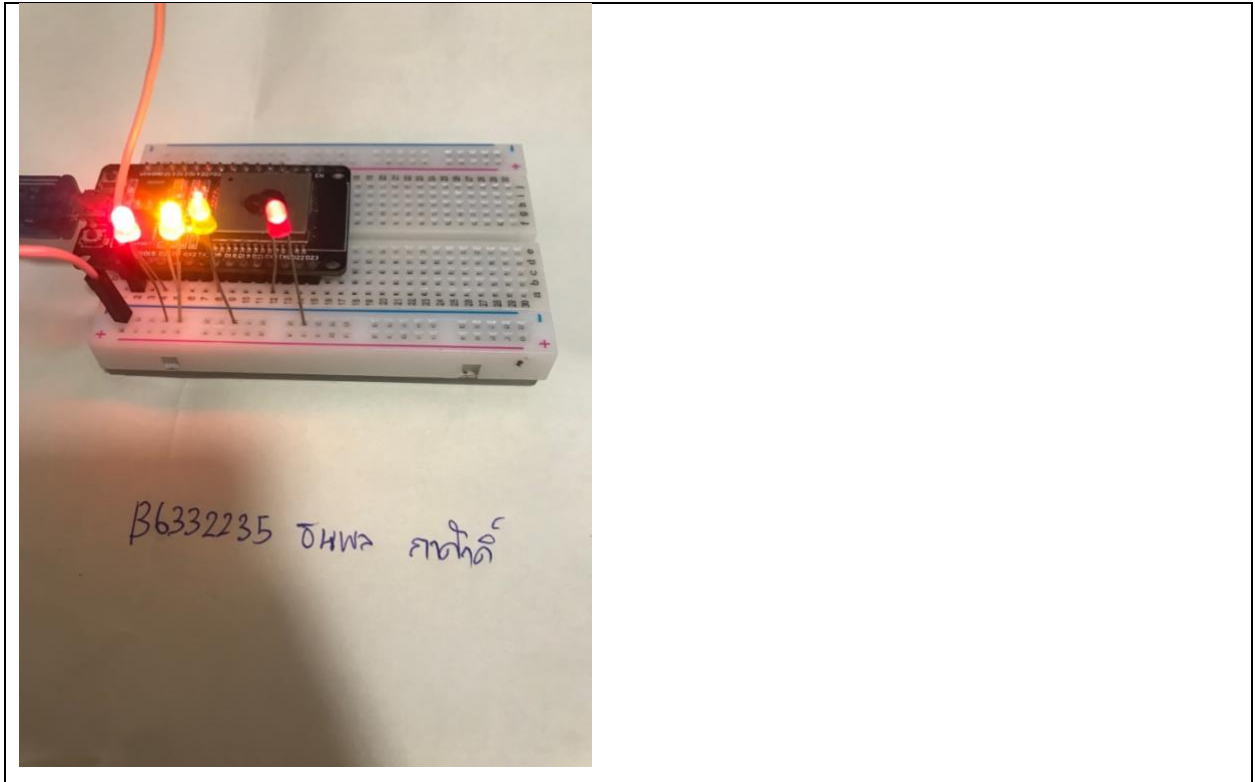


```
#define BLYNK_PRINT Serial
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
char auth[] = "gOVELUAYorH-U-5YCExxEemXYjjhcl1S";
const char* ssid = "iPhoneOhm";
const char* password = "2444666668888888";
void setup()
{
  // Debug console
  Serial.begin(115200);
  Blynk.begin(auth, ssid, pass);
}

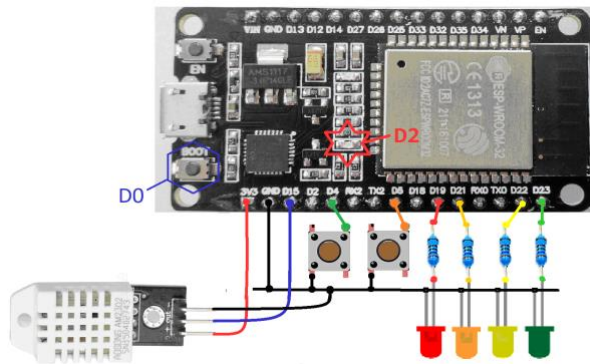
void loop()
{
  Blynk.run();
}
```

รูปภาพจอ Blynk





Quiz_302 – DHT22 + 4 LED + 2 Switch



< Test Code >

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

#define DHT22_Pin 15
#define sw1 4
#define sw2 21

char auth[] = "gOVELUAYorH-U-5YCExxEemXYjjhcl1S";
const char* ssid = "iPhoneOhm";
const char* password = "2444666668888888";
DHTesp dht;

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

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);
}

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 loop()
{ Blynk.run();
  timer.run(); // running timer every 250ms
}

```

รูปหน้าจอ Blynk

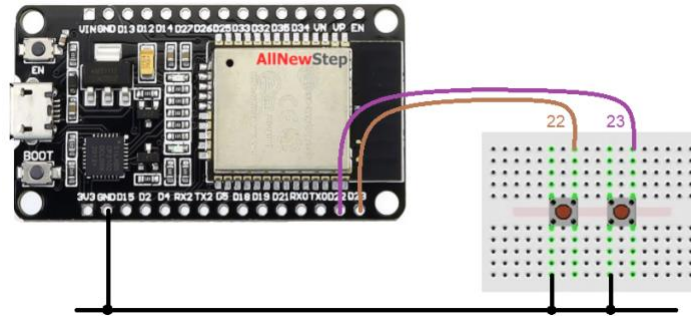
รูปการต่อวงจร – 1

รูปการต่อวงจร – 2

Quiz_303 – Social Alert

ทดสอบการส่งข้อมูลไป ☐ LINE สำหรับสวิตช์กด 3 ตัว

- กดปุ่ม B ที่ต่อกับ ESP32- ให้ส่งข้อความ “Door Open Alarm”
- กดปุ่ม C ที่ต่อกับ ESP32- ให้ส่งข้อความ “Intruders Alarm”



```
#include <WiFi.h>
#include <HttpClient.h>

#define WIFI_SSID "iPhoneOhm"
#define WIFI_PASS "2444666668888888"
#define WebHooksKey "c6Dw2mHZ9znYfzd0i5Im4U_wZEYKTCtNmV19i1dyUvI"
#define WebHooksEventName "ohm"

#define testSwitch1 22
#define testSwitch2 23

void setup() {
  Serial.begin(115200);
  WiFi.begin(WIFI_SSID, WIFI_PASS);
  Serial.println("Connecting");
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());
  pinMode(testSwitch1, INPUT_PULLUP);
  pinMode(testSwitch2, INPUT_PULLUP);
  randomSeed(analogRead(33));
}
```

```

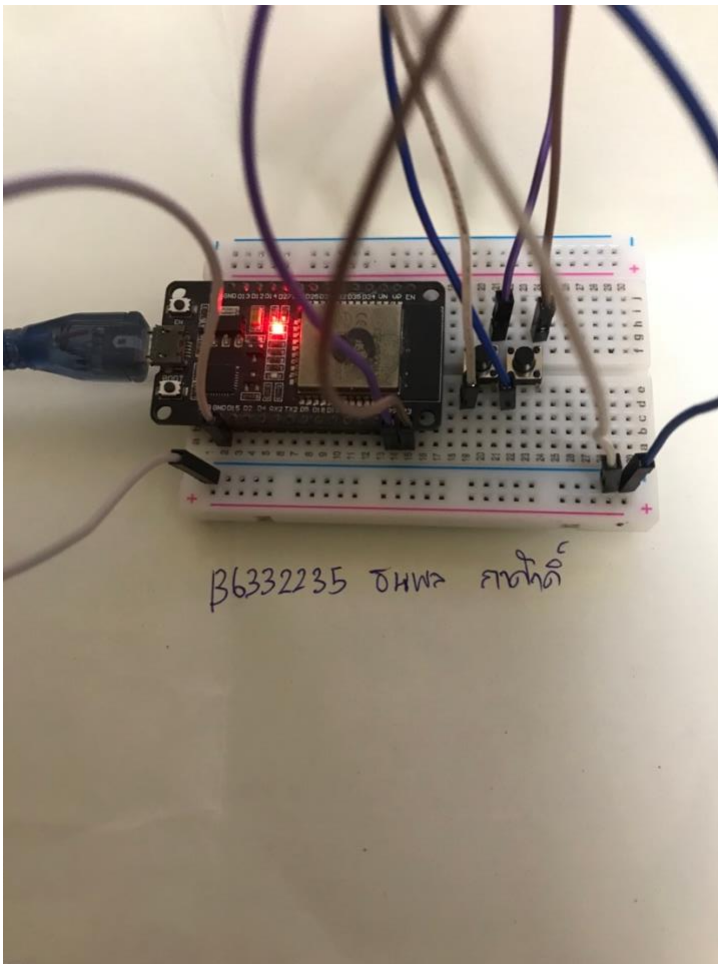
void loop() {
  if (digitalRead(testSwitch1) == LOW) {
    String serverName = "http://maker.ifttt.com/trigger/" +
      String(WebHooksEventName) + "/with/key/" + String(WebHooksKey);
    String httpRequestData = "value1=" + String("Door Open Alarm");
    Serial.println("Server Name : " + serverName);
    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("Successfully sent");
      else
        Serial.println("Failed!");
    }
    else {
      Serial.println("WiFi Disconnected");
    }
  }
  if (digitalRead(testSwitch2) == LOW) {
    String serverName = "http://maker.ifttt.com/trigger/" +
      String(WebHooksEventName) + "/with/key/" + String(WebHooksKey);
    String httpRequestData = "value1=" + String("Intruders Alarm");
    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);
    }
  }
}

```

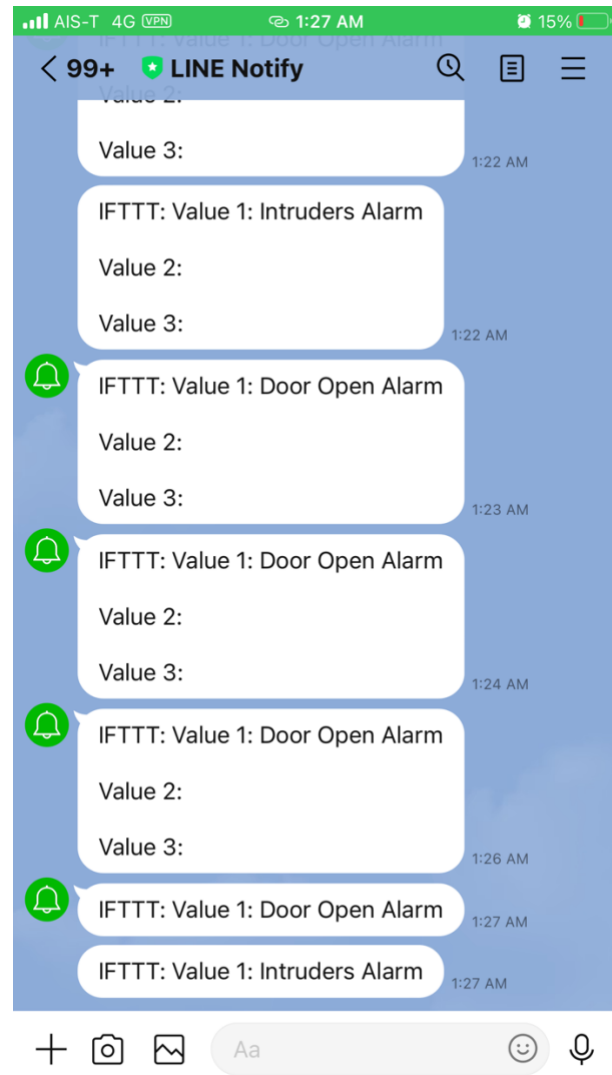
```
http.end();  
if (httpResponseCode == 200)  
    Serial.println("Successfully sent");  
else  
    Serial.println("Failed!");  
}  
else {  
    Serial.println("WiFi Disconnected");  
}  
}  
}
```

รูปการต่อวงจร - 1

รูปการต่อวงจร - 2

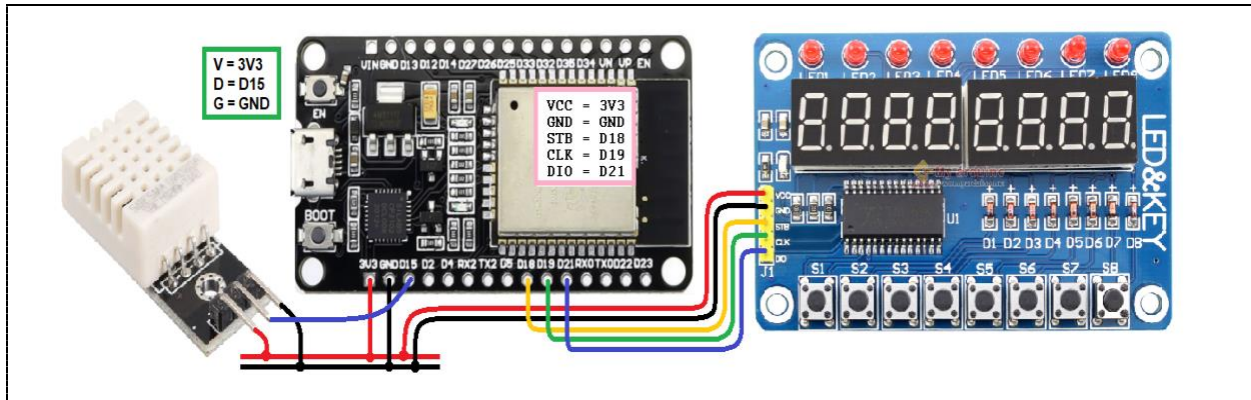


รูปหน้าจอ LINE ผลการทดสอบ



Quiz_304 – Data Logger and Social Alarm

- ส่งข้อมูลอุณหภูมิไปยัง Google Spreadsheet (ทำแล้วในข้อ QB4)
- หากอุณหภูมิที่อ่านได้เกิน 28°C ให้แจ้งเตือนผ่าน ____ และบอกด้วยว่าอุณหภูมิเท่าใด
☐ SMS, ☐ FB Page, ☐ FB Massager, ☐ Twitter, ☒ LINE
- แสดงอุณหภูมิที่ 7_Segment Display TM1638 Board



< Test Code >

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <TM1638plus.h>
#define DHT22_Pin 15
#include "DHTesp.h"
DHTesp dht;

#define WIFI_SSID "iPhoneOhm"
#define WIFI_PASS "2444666668888888"
#define WebHooksKey "c6Dw2mHZ9znYfzd0i5Im4U_wZEYKTCtNmV19i1dyUvI"
#define WebHooksEventName "ohm"
#define WebHooksEventName_line " ohm_line"

#define My_NAME "B6214005 Varasiri Limprasert"
#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
WiFi.begin(WIFI_SSID, WIFI_PASS);
Serial.println("Connecting");
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.print("Connected to WiFi network with IP Address: ");
Serial.println(WiFi.localIP());
}

void loop() {
    float humidity = dht.getHumidity();
    float temperature = dht.getTemperature();
    Serial.println();
    Serial.print("\nTemperature('C) = ");
    Serial.print(temperature, 1);
    Serial.print("\tHumidity(%) = ");
    Serial.print(humidity, 1);
    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);
    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("[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");
    }
}
/*Display */
int t = int(temperature * 100);

```

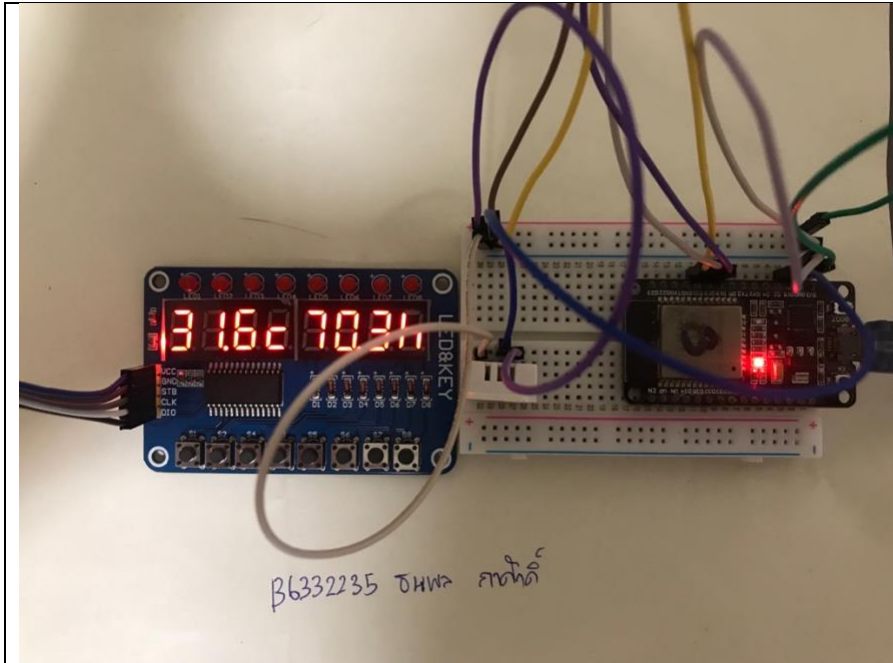
```

int Tempp2 = (int)temperature / 10; int Tempp1 = (int)temperature % 10; int Tempp0 =(int)(temperature * 10) %
10;
int Humi2 = (int)humidity / 10; int Humi1 = (int)humidity % 10; int Humi0 =(int)(humidity * 10) % 10;
tm.displayHex(0, Tempp2);
tm.displayASCIllwDot(1, Tempp1 + '0'); // turn on dot
tm.displayHex(2, Tempp0);
tm.display7Seg(3, B01011000); // Code=tgfedcba
tm.displayHex(4, Humi2);
tm.displayASCIllwDot(5, Humi1 + '0'); // turn on dot
tm.displayHex(6, Humi0);
tm.display7Seg(7, B01110100); // Code=tgfedcba
delay(2000);
int WaitTime = 60;
Serial.print(" >> Wait for next time --> ");
for (int i = WaitTime; i >= 0; i -= 5) {
    Serial.print(",");
    Serial.print(i);
    delay(5000);
}
}

```

รูปการต่อวงจร – 1

รูปการต่อวงจร – 2



รูปภาพจอ LINE ผลการทดสอบ

| | | | |
|----------------------|------------------------|------|------|
| June 13, 2023 at ohm | B6332235 tanapon kasak | 31.6 | 70.3 |
| June 13, 2023 at ohm | B6332235 tanapon kasak | 30.9 | 69.7 |
| June 13, 2023 at ohm | B6332235 tanapon kasak | 31.6 | 70.3 |
| June 13, 2023 at ohm | B6332235 tanapon kasak | 31.5 | 70.3 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

