

## แนวทางการใช้งานอินเทอร์เน็ทของสรรพสิ่งในระบบการผลิต

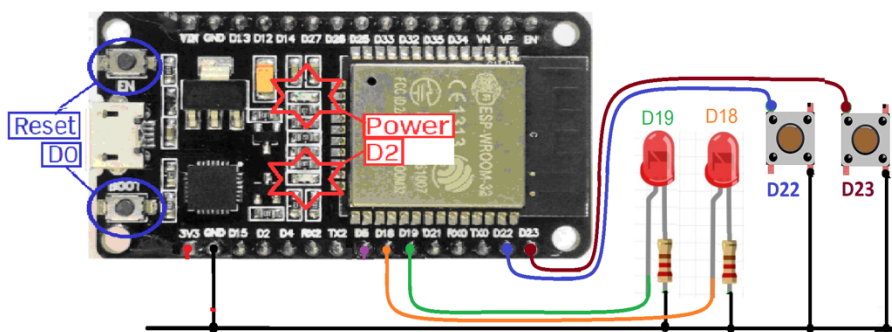
### IoT Approaches to Manufacturing System

ชื่อ-สกุล : B6304577 นายภาณุพงศ์ แคนอินทร์

#### 4/4. 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}



< Test Code >

```
#define BUTTON_PIN1    22 // GPIO22 pin connected to button
#define BUTTON_PIN2    23 // GPIO23 pin connected to button

#define LED_PIN1      18 // ESP32 pin GPIO18, which connected to led
#define LED_PIN2      19 // ESP32 pin GPIO19, which connected to led
#define DEBOUNCE_TIME 50 // the debounce time in millisecond, increase this time if it still chatters

// Variables will change:
int lastSteadyState_led1 = LOW; // the previous steady state from the input pin
int lastSteadyState_led2 = LOW; // the previous steady state from the input pin
int lastFlickerableState_led1 = LOW; // the previous flickerable state from the input pin
int lastFlickerableState_led2 = LOW; // the previous flickerable state from the input pin
int currentState_led1; // the current reading from the input pin
int currentState_led2; // the current reading from the input pin
int led1_state = LOW; // the current state of LED
int led2_state = LOW; // the current state of LED

// the following variables are unsigned longs because the time, measured in
// milliseconds, will quickly become a bigger number than can be stored in an int.
unsigned long lastDebounceTime = 0; // the last time the output pin was toggled

void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);

  pinMode(BUTTON_PIN1, INPUT_PULLUP);
  pinMode(BUTTON_PIN2, INPUT_PULLUP);

  pinMode(LED_PIN1, OUTPUT);
  pinMode(LED_PIN2, OUTPUT);
}
```

```

void led1() {

    // read the state of the switch/button:
    currentState_led1 = digitalRead(BUTTON_PIN1);

    // If the switch/button changed, due to noise or pressing:
    if (currentState_led1 != lastFlickerableState_led1) {
        // reset the debouncing timer
        lastDebounceTime = millis();
        // save the the last flickerable state
        lastFlickerableState_led1 = currentState_led1;
    }

    if ((millis() - lastDebounceTime) > DEBOUNCE_TIME) {
        // whatever the reading is at, it's been there for longer than the debounce
        // delay, so take it as the actual current state:

        // if the button state has changed:
        if (lastSteadyState_led1 == HIGH && currentState_led1 == LOW) {
            led1_state = !led1_state;
            digitalWrite(LED_PIN1, led1_state);
        }

        // save the the last steady state
        lastSteadyState_led1 = currentState_led1;
    }
}

void led2() {

    // read the state of the switch/button:
    currentState_led2 = digitalRead(BUTTON_PIN2);

    if (currentState_led2 != lastFlickerableState_led2) {
        // reset the debouncing timer
        lastDebounceTime = millis();
        // save the the last flickerable state
        lastFlickerableState_led2 = currentState_led2;
    }

    if ((millis() - lastDebounceTime) > DEBOUNCE_TIME) {
        // whatever the reading is at, it's been there for longer than the debounce
        // delay, so take it as the actual current state:
        // if the button state has changed:
        if (lastSteadyState_led2 == HIGH && currentState_led2 == LOW) {
            led2_state = !led2_state;
            digitalWrite(LED_PIN2, led2_state);
        }

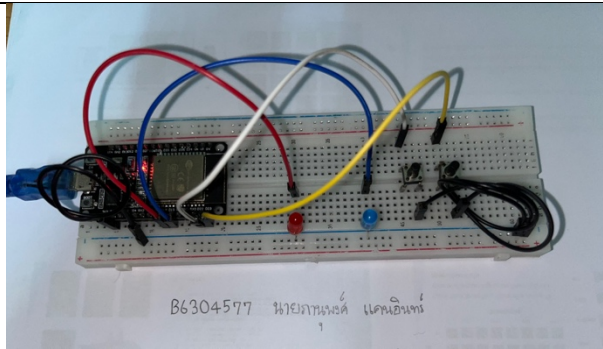
        // save the the last steady state
        lastSteadyState_led2 = currentState_led2;
    }
}

void loop() {

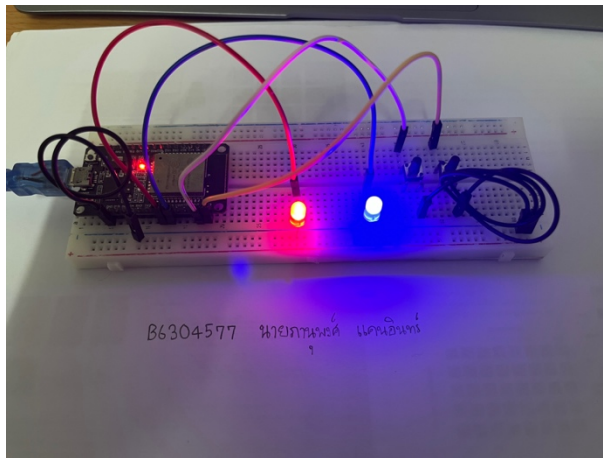
    led1();
    led2();

}

```



## รูปการต่อวงจร - 2



## 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%

< Test Code >

```
#include <DHT.h> //https://www.arduino-libraries.info/libraries/dht-sensor-library
#define DHT_SENSOR_PIN 21 // ESP32 pin GPIO21 connected to DHT22 sensor
#define DHT_SENSOR_TYPE DHT22

DHT dht_sensor(DHT_SENSOR_PIN, DHT_SENSOR_TYPE);

void setup() {
  Serial.begin(9600);
  dht_sensor.begin(); // initialize the DHT sensor
}

void loop() {
  // read humidity
  float humi = dht_sensor.readHumidity();
  // read temperature in Celsius
  float tempC = dht_sensor.readTemperature();
  // read temperature in Fahrenheit
  float tempF = dht_sensor.readTemperature(true);

  // check whether the reading is successful or not
  if ( isnan(tempC) || isnan(tempF) || isnan(humi) ) {
    Serial.println("Failed to read from DHT sensor!");
  } else {

    Serial.print("Temperature: ");
    Serial.print(tempC);
    Serial.print("°C ~ ");
    Serial.print(tempF);
    Serial.print("°F");

    Serial.print(" | ");

    Serial.print("Humidity: ");
    Serial.print(humi);
    Serial.println("%");

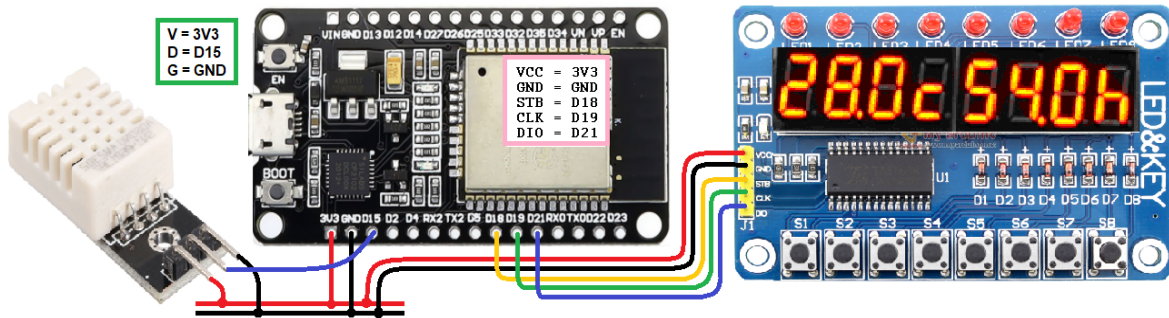
  }

  // wait a 2 seconds between readings
  delay(2000);
}
```



## Quiz\_103 – Read Sensor and Show

- ต่อวงจรเพิ่มเติม ทดสอบการทำงานด้วยโปรแกรมต่อไปนี้ และปรับแก้ให้ถูกต้อง



## &lt; Test Code &gt;

```
#include <TM1638plus.h> [ver 1.9.1]
#include <DHT.h> //https://www.arduinolibraries.info/libraries/dht-sensor-library [ver 1.4.4]
#define DHT_SENSOR_PIN 15 // ESP32 pin GPIO21 connected to DHT22 sensor
#define DHT_SENSOR_TYPE DHT22

#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 , if using high freq CPU > 100 MHz set to true

TM1638plus tm(Brd_STB, Brd_CLK, Brd_DIO, high_freq);

DHT dht_sensor(DHT_SENSOR_PIN, DHT_SENSOR_TYPE);

void setup() {
  Serial.begin(9600);
  dht_sensor.begin(); // initialize the DHT sensor
  tm.displayBegin(); // initialize the TM1638
}

void loop() {
  // read humidity
  float humi = dht_sensor.readHumidity();
  // read temperature in Celsius
  float tempC = dht_sensor.readTemperature();
  // read temperature in Fahrenheit
  float tempF = dht_sensor.readTemperature(true);

  // check whether the reading is successful or not
  if ( isnan(tempC) || isnan(tempF) || isnan(humi) ) {
    Serial.println("Failed to read from DHT sensor!");
  } else {

    Serial.print("Temperature: ");
    Serial.print(tempC);
    Serial.print("°C ~ ");
    Serial.print(tempF);
    Serial.print("°F");
  }
}
```

```

Serial.print(" | ");

Serial.print("Humidity: ");
Serial.print(humi);
Serial.println("%");

tm.displayHex(0,int(tempC/10));
tm.displayASCIIDot(1,int(int(tempC)%10)+'0'); //turn on dot
tm.displayHex(2,int(int(tempC*10)%10);
tm.display7Seg(3,B01011000);

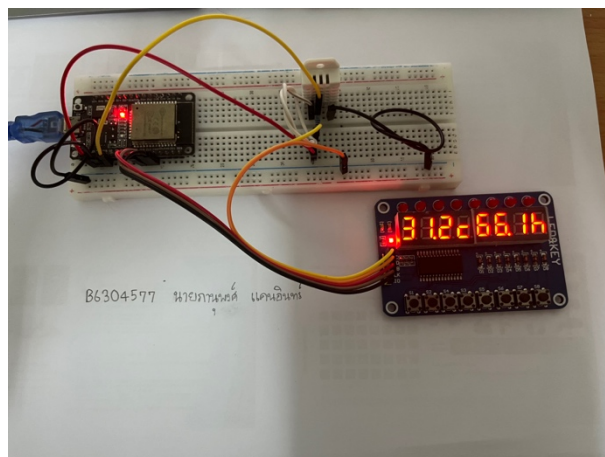
tm.displayHex(4,int(humi/10));
tm.displayASCIIDot(5,int(int(humi)%10)+'0'); //turn on dot
tm.displayHex(6,int(int(humi*10)%10);
tm.display7Seg(7,B01110100);

}

// wait a 2 seconds between readings
delay(2000);
}

```

### รูปการต่อวงจร - 1



### รูปการต่อวงจร - 2

