

ใบงานการทดลองที่ 2

เรื่อง ความรู้เกี่ยวกับไมโครคอนโทรลเลอร์เบื้องต้น

1. จุดประสงค์

2.7.1 ทดลองวงจร LED พื้นฐาน

2. ทฤษฎี

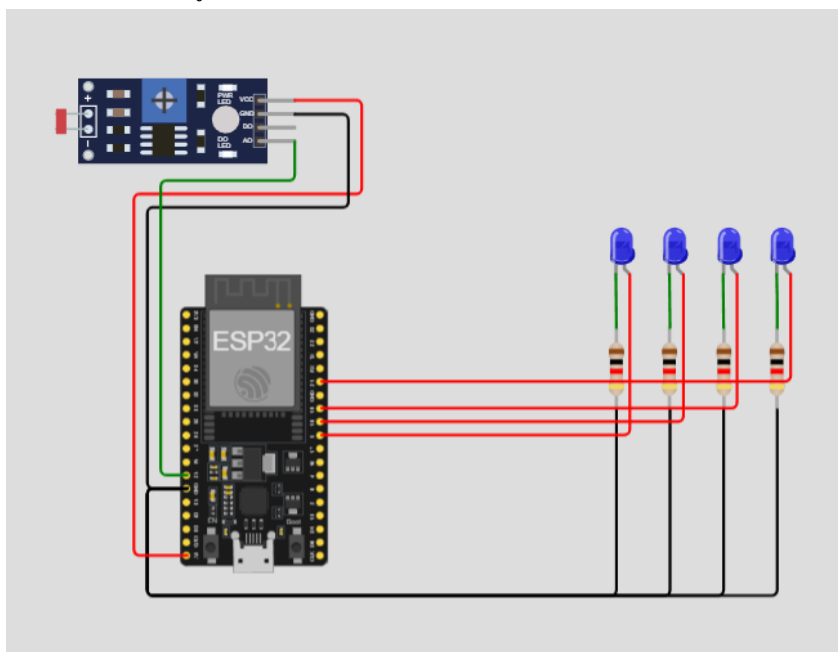
3. เครื่องมือและอุปกรณ์การทดลอง

3.1 โปรแกรม Arduino IDE 1.8.16 หรือสูงกว่า	1	โปรแกรม
3.2 สาย USB สำหรับ ESP8266 หรือ ESP32	1	เส้น
3.3 ชุดทดลอง ESP8266 หรือ ESP32	1	ชุด
3.4 สายต่อวงจร	1	ชุด
3.5 เครื่องคอมพิวเตอร์แบบพกพา	1	เครื่อง
3.6 แผงต่อวงจร	1	ตัว
3.7 ตัวต้านทาน 1 k Ω	8	ตัว
3.8 ไดโอดเปล่งแสง	8	ตัว

4. ลำดับขั้นตอนการทดลอง

การทดลองที่ 1

1.1. ทำการต่อวงจรตามรูป



1.2. เขียนโปรแกรมดังนี้

```
#include <stdexcept>

const int LED_PIN_1 = 5; // หมายเลขขาที่ต่อกับ LED 1
const int LED_PIN_2 = 18; // หมายเลขขาที่ต่อกับ LED 2
const int LED_PIN_3 = 19; // หมายเลขขาที่ต่อกับ LED 3
const int LED_PIN_4 = 21; // หมายเลขขาที่ต่อกับ LED 4
const int LDR_PIN = 12; // หมายเลขขาที่ต่อกับ LDR Sensor

void setup() {
    pinMode(LED_PIN_1, OUTPUT);
    pinMode(LED_PIN_2, OUTPUT);
    pinMode(LED_PIN_3, OUTPUT);
    pinMode(LED_PIN_4, OUTPUT);
    pinMode(LDR_PIN, INPUT);
    Serial.begin(9600);
}

void loop() {
    try {
        int sensorValue = readLDR();
        controlLEDs(sensorValue);
    } catch (const std::exception& e) {
        Serial.print("Error: ");
        Serial.println(e.what());
        cleanup();
    }
    delay(1000);
}

int readLDR() {
    int sensorValue = analogRead(LDR_PIN);

    if (sensorValue < 0 || sensorValue > 10230) {
        throw std::out_of_range("LDR Sensor value out of range");
    }

    return sensorValue;
}
```

```
void controlledLEDs(int sensorValue) {
  if (sensorValue >= 1 && sensorValue <= 200) {
    Serial.print("%d,sensorValue");
    turnOnLED(LED_PIN_1);
    turnOffLED(LED_PIN_2);
    turnOffLED(LED_PIN_3);
    turnOffLED(LED_PIN_4);
  } else if (sensorValue >= 201 && sensorValue <= 400) {
    turnOnLED(LED_PIN_1);
    turnOnLED(LED_PIN_2);
    turnOffLED(LED_PIN_3);
    turnOffLED(LED_PIN_4);
  } else if (sensorValue >= 401 && sensorValue <= 600) {
    turnOnLED(LED_PIN_1);
    turnOnLED(LED_PIN_2);
    turnOnLED(LED_PIN_3);
    turnOffLED(LED_PIN_4);
  } else {
    turnOnLED(LED_PIN_1);
    turnOnLED(LED_PIN_2);
    turnOnLED(LED_PIN_3);
    turnOnLED(LED_PIN_4);
  }
}

void turnOnLED(int pin) {
  digitalWrite(pin, HIGH);
}

void turnOffLED(int pin) {
  digitalWrite(pin, LOW);
}

void cleanup() {
  turnOffLED(LED_PIN_1);
  turnOffLED(LED_PIN_2);
  turnOffLED(LED_PIN_3);
  turnOffLED(LED_PIN_4);
  Serial.end();
}
```

1.3. สังเกตผลการทดลองและบันทึกผลการทดลอง

.....
 เมื่อเริ่มการทดลองจะพบว่าเมื่อมีค่าแสงมากขึ้น LED จะดับทีละดวง

1.4. สรุปผลการทดลอง

.....
 เมื่อทำการปรับค่าแสงของ LDR เพิ่มขึ้น โปรแกรมก็จะทำงานตามค่าที่ได้รับคือการดับ LED ทีละดวงตามค่า
 แสงที่ได้รับ

การทดลองที่ 2

2.1 เพิ่มฟังก์ชันดังต่อไปนี้

```
void checkOpenInterval() {
    // ตรวจสอบว่าเวลาผ่านไปพอสำหรับเปิด LED หรือไม่
    unsigned long currentTime = millis();
    if (currentTime - lastOpenTime >= OPEN_INTERVAL) {
        // เวลาผ่านไปพอสำหรับเปิด LED
        openLEDs();
        lastOpenTime = currentTime;
    }
}
```

2.2 ที่ ฟังก์ชัน void loop() ให้เพิ่ม ฟังก์ชัน checkOpenInterval(); ลงไป

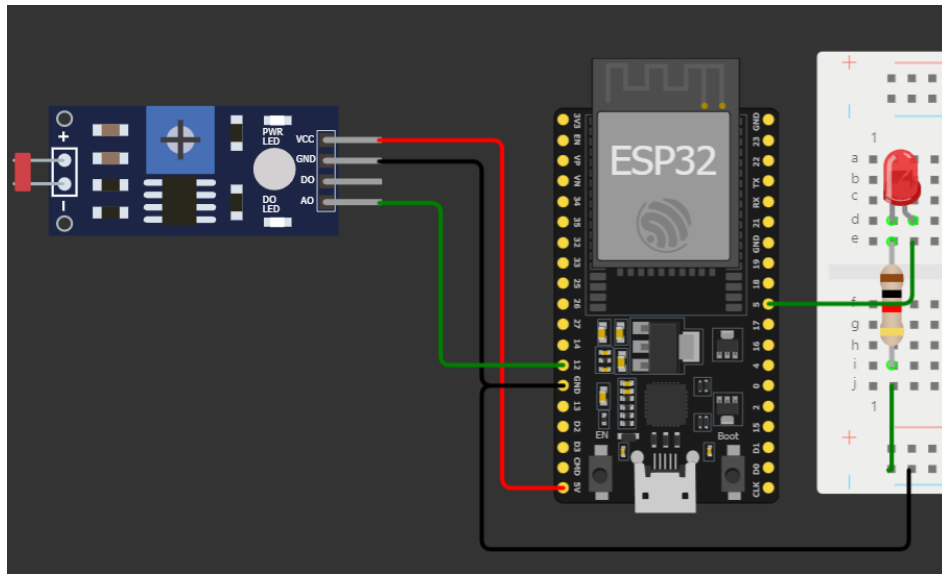
```
void loop() {
    try {
        int sensorValue = readLDR();
        controlLEDs(sensorValue);
        checkOpenInterval();
    } catch (const std::exception& e) {
        Serial.print("Error: ");
        Serial.println(e.what());
        cleanup();
    }
    delay(1000);
}
```

2.3 สังเกตผลการทดลองและบันทึกผลการทดลอง

.....
เมื่อเริ่มการทดลอง LED จะดับอยู่และเมื่อปรับค่า LDR ก็ทำให้ LED ติด
.....
.....

2.4 สรุปผลการทดลอง

.....
เมื่อยังไม่มีส่วนของ LDR จะสั่งให้ LED ดับทุกดวงแต่เมื่อมีส่วน LED จะติดเพิ่มขึ้นเรื่อยๆ และจะค่อยๆดับตาม
ค่าของเวลาที่ใส่เพิ่มไป
.....



```

1  #include <stdexcept>
2  const int LED_PIN_1 = 5; // หมายเลขขาที่ต่อกับ LED 1
3  const int LDR_PIN = 12; // หมายเลขขาที่ต่อกับ LDR Sensor
4  void setup() {
5      pinMode(LED_PIN_1, OUTPUT);
6      pinMode(LDR_PIN, INPUT);
7      Serial.begin(9600);
8  }
9  void loop() {
10     try {
11         int sensorValue = readLDR();
12         controlledLEDs(sensorValue);
13     } catch (const std::exception& e) {
14         Serial.print("Error: ");
15         Serial.println(e.what());
16         cleanup();
17     }
18     delay(1000);
19 }
20
21 int readLDR() {
22     int sensorValue = analogRead(LDR_PIN);
23     if (sensorValue < 0 || sensorValue > 10230) {
24         throw std::out_of_range("LDR Sensor value out of range");
25     }
26     return sensorValue;
27 }
28 void controlledLEDs(int sensorValue) {
29     if(sensorValue <= 40){
30         sensorValue = 0;
31     }
32     Serial.println(sensorValue);
33     turnOnLED(LED_PIN_1,sensorValue);
34 }
35 void turnOnLED(int pin,int light) {
36     light = light/16;
37     analogWrite(pin, light);
38 }
39 void turnOffLED(int pin) {
40     analogWrite(pin, 0);
41 }
42 void cleanup() {
43     turnOffLED(LED_PIN_1);
44     Serial.end();
45 }

```

สังเกตผลการทดลองและบันทึกผลการทดลอง

เมื่อเริ่มจะพบว่า LED จะสว่างขึ้นและเมื่อเริ่มปรับค่าของ LDR ความสว่างของ LED จะเพิ่มขึ้นลงตามความสว่าง

สรุปผลการทดลอง

เมื่อเริ่ม LDR จะรับค่าแสงมาจากนั้นจะทำการแปลงไดนามิกค่าที่อ่านมาหาร 16 เพื่อให้สามารถปรับค่าความสว่างของ LED ได้โดยเมื่อค่าของ LDR

อยู่ในจุดสว่างสุด จะบังคับให้ค่าที่ตั้งให้ LED ทำงานเป็น 0 เพื่อให้ LED ดับ