

ใบงานการทดลอง เรื่อง โปรแกรมแบบเรียงลำดับกับ 7-Segment

จุดประสงค์การเรียนรู้

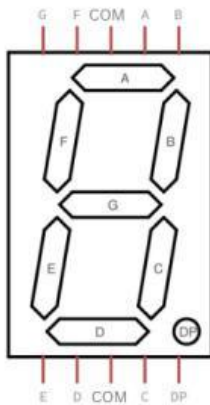
1. นักเรียนสามารถอธิบายหน้าที่และลักษณะของอุปกรณ์ 7-segment display ได้ (K)
2. นักเรียนสามารถต่อวงจรไฟฟ้า 7-segment display และเขียนโปรแกรมเพื่อแสดงผลได้อย่างถูกต้อง (P)
3. นักเรียนปฏิบัติงานด้วยความตั้งใจและจัดเก็บอุปกรณ์เรียบร้อยเมื่อใช้งานเสร็จ (A)

เครื่องมือและอุปกรณ์การทดลอง

1. LED 7 Segment
2. บอร์ด Arduino
3. สายจัมเปอร์
4. สาย USB
5. ตัวต้านทาน
6. โฟโต้บอร์ด
7. โปรแกรม Arduino IDE

ภาคทฤษฎี

7-segment display ประกอบด้วยหลอดไฟ LED ขนาดเล็ก 7 ดวง ซึ่งแต่ละดวงจะถูกตั้งชื่อด้วยตัวอักษรเพื่อใช้ในการอ้างอิงและควบคุมการติด-ดับ ทำให้เกิดเป็นรูปทรงตัวเลขขึ้นมา



- Segment a คือส่วนบนสุดของตัวเลข
- Segment b คือส่วนบนด้านขวา
- Segment c คือส่วนล่างด้านขวา
- Segment d คือส่วนล่างสุด
- Segment e คือส่วนล่างด้านซ้าย
- Segment f คือส่วนบนด้านซ้าย
- Segment g คือส่วนตรงกลาง

นอกจากนี้ยังมี Segment DP (Decimal Point) สำหรับจุดทศนิยมอีกด้วย

รูปแบบ	Common Cathode (CC)	Common Anode (CA)
ขาร่วม (Common)	เชื่อมต่อกับขั้วลบทั้งหมด	เชื่อมต่อกับขั้วบวกทั้งหมด
การต่อวงจร	ต่อขาร่วมเข้ากับ GND	ต่อขาร่วมเข้ากับ 5V
การทำให้ Segment ติด	ส่งสัญญาณ HIGH (5V) ไปยังขา Segment ที่ต้องการ	ส่งสัญญาณ LOW (0V) ไปยังขา Segment ที่ต้องการ
คำสั่ง Arduino	digitalWrite(ขา Segment, HIGH);	digitalWrite(ขา Segment, LOW);

หลักการจำ

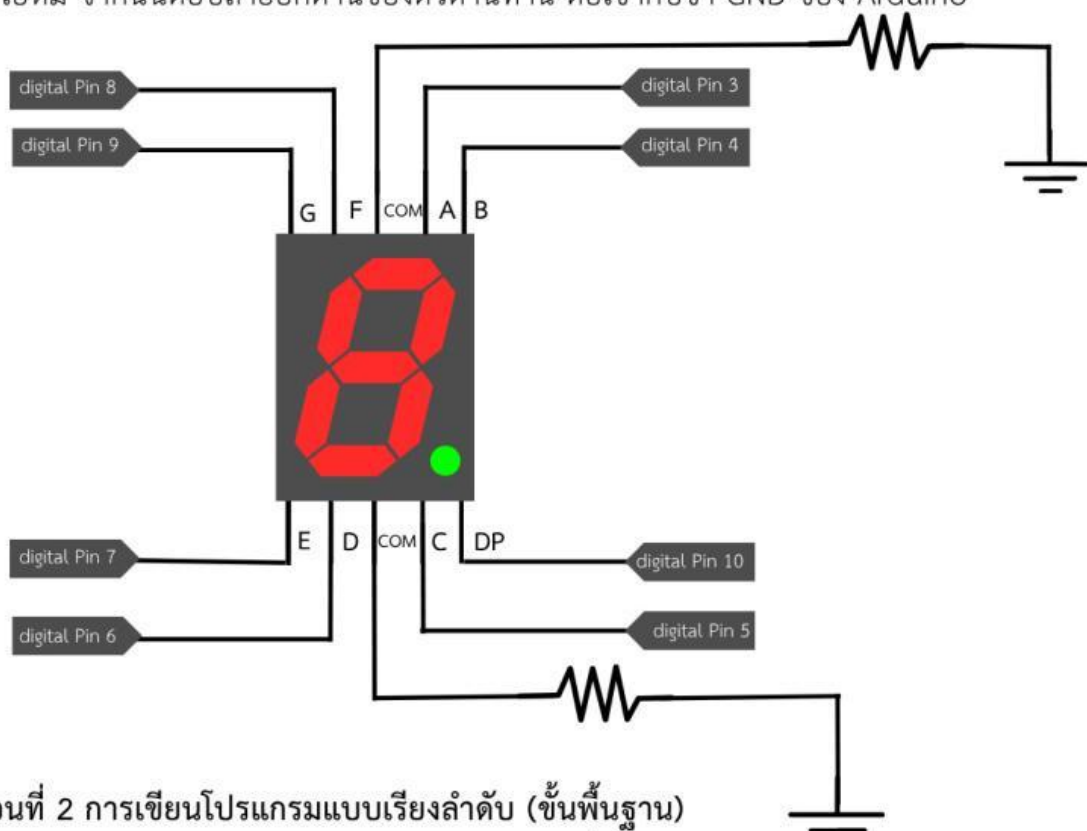
- Cathode (แคโทด) คือขั้วลบ ซึ่งต่อกับกราวด์ (GND) ดังนั้นจึงต้องใช้คำสั่ง HIGH เพื่อให้กระแสไหลและ LED ติด
- Anode (แอโนด) คือขั้วบวก ซึ่งต่อกับไฟเลี้ยง (5V) ดังนั้นจึงต้องใช้คำสั่ง LOW (หรือ 0V) เพื่อให้เกิดความต่างศักย์และกระแสไหล ทำให้ LED ติด

ภาคปฏิบัติ

ขั้นตอนที่ 1 การต่อวงจร 7segment แบบ Cathode

1.1 เสียบ 7-segment display ลงบน Breadboard

1.2 ต่อขา Segment แบบ Common Cathode ขา a ถึง g และ dp โดยแต่ละขาต่อไปยังขา Digital Pin 3 ถึง 10 ตามลำดับดังรูป ต่อขา Common (COM) ของ 7segment เข้ากับตัวต้านทาน 150 โอห์ม จากนั้นต่อปลายอีกด้านของตัวต้านทาน ต่อเข้ากับขา GND ของ Arduino



ขั้นตอนที่ 2 การเขียนโปรแกรมแบบเรียงลำดับ (ขั้นพื้นฐาน)

2.1 เปิดโปรแกรม Arduino IDE และเขียนโค้ดเพื่อแสดงผลตัวเลข 0-9 แบบเรียงลำดับ ดังนี้

```
// กำหนดขาพินสำหรับแต่ละ segment
#define SEG_A 3
#define SEG_B 4
#define SEG_C 5
#define SEG_D 6
#define SEG_E 7
#define SEG_F 8
#define SEG_G 9
#define SEG_DP 10
```

```

void setup() {
    // ตั้งค่าขาพินทั้งหมดเป็น OUTPUT
    pinMode(SEG_A, OUTPUT);
    pinMode(SEG_B, OUTPUT);
    pinMode(SEG_C, OUTPUT);
    pinMode(SEG_D, OUTPUT);
    pinMode(SEG_E, OUTPUT);
    pinMode(SEG_F, OUTPUT);
    pinMode(SEG_G, OUTPUT);
    pinMode(SEG_DP, OUTPUT);
    digitalWrite(SEG_DP, HIGH); // แสดงจุดที่อุปกรณ์ 7segment
    // เริ่มต้น Serial Monitor ด้วยความเร็ว 9600 bps
    Serial.begin(9600);
}

void loop() {
    // --- แสดงตัวเลข 0 ---
    Serial.println("แสดงตัวเลข: 0");
    digitalWrite(SEG_A, HIGH);
    digitalWrite(SEG_B, HIGH);
    digitalWrite(SEG_C, HIGH);
    digitalWrite(SEG_D, HIGH);
    digitalWrite(SEG_E, HIGH);
    digitalWrite(SEG_F, HIGH);
    digitalWrite(SEG_G, LOW);
    delay(1000); // หน่วงเวลา 1 วินาที

    // --- แสดงตัวเลข 1 ---
    Serial.println("แสดงตัวเลข: 1");
    digitalWrite(SEG_A, LOW);
    digitalWrite(SEG_B, HIGH);
    digitalWrite(SEG_C, HIGH);
    digitalWrite(SEG_D, LOW);
    digitalWrite(SEG_E, LOW);
    digitalWrite(SEG_F, LOW);
    digitalWrite(SEG_G, LOW);
    delay(1000); // หน่วงเวลา 1 วินาที
}

```

```
// --- แสดงตัวเลข 2 ---
```

```
Serial.println("แสดงตัวเลข: 2");
```

```
digitalWrite(SEG_A, HIGH);
```

```
digitalWrite(SEG_B, HIGH);
```

```
digitalWrite(SEG_C, LOW);
```

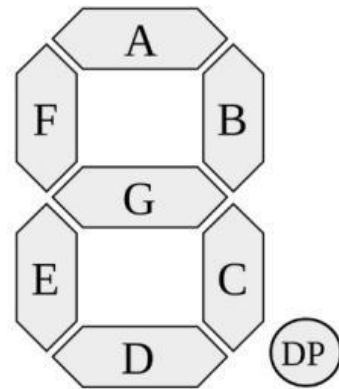
```
digitalWrite(SEG_D, HIGH);
```

```
digitalWrite(SEG_E, HIGH);
```

```
digitalWrite(SEG_F, LOW);
```

```
digitalWrite(SEG_G, HIGH);
```

```
delay(1000);          // หน่วงเวลา 1 วินาที
```



```
// --- แสดงตัวเลข 3 ---
```

```
Serial.println("แสดงตัวเลข: 3");
```

```
digitalWrite(SEG_A, HIGH);
```

```
digitalWrite(SEG_B, HIGH);
```

```
digitalWrite(SEG_C, HIGH);
```

```
digitalWrite(SEG_D, HIGH);
```

```
digitalWrite(SEG_E, LOW);
```

```
digitalWrite(SEG_F, LOW);
```

```
digitalWrite(SEG_G, HIGH);
```

```
delay(1000);          // หน่วงเวลา 1 วินาที
```

```
// เขียนโค้ดเพิ่มเติมด้วยตนเองเพื่อแสดงตัวเลข 4 ถึง 9 ด้วยตนเอง
```

```
}
```

ขั้นตอนที่ 3 การอัปโหลดและทดสอบ

3.1 เขียนโค้ดให้ครบถ้วนสำหรับตัวเลข 0-9

3.2 เชื่อมต่อบอร์ด Arduino กับคอมพิวเตอร์ผ่านสาย USB

3.3 ใน Arduino IDE เลือก Tools > Board > Arduino Uno

3.4 เลือก Tools > Port > COM... (เลือกพอร์ตที่ถูกต้อง)

3.5 คลิกปุ่ม 'Upload' (ลูกศรขวา)

3.6 หลังจากอัปโหลดสำเร็จ ให้สังเกตการแสดงผลบน 7-segment display และเปิด Serial Monitor เพื่อดูว่ามีข้อความแสดงผลตามที่เขียนไว้หรือไม่

บันทึกผลการทดลอง

รอบการทำงาน ของโปรแกรม	ตัวเลขที่ แสดง	สถานะของ Segment ที่ติดสว่าง (ตอบแบบเรียงลำดับตัวอักษร ไล่ , คั่น ดังตัวอย่าง)
1	0	a, b, c, d, e, f
2		
3		
4		
5		
6		
7		
8		
9		
10		

คำถามท้ายการทดลอง

1. คุณสมบัติที่สำคัญที่สุดของ 7-segment display แบบ Common Cathode คืออะไร และเราจะทำให้ Segment ติดสว่างได้อย่างไร?

.....

.....

2. หากต้องการให้ Segment ที่ตำแหน่ง c ของ 7-segment display ติดสว่าง จะต้องส่งสัญญาณใดไปที่ขาพินที่เชื่อมต่อกับ Segment นั้น และใช้คำสั่งอะไร

.....

.....

คำถามท้ายการทดลอง

3. จากใบงานการทดลองนี้ การต่อวงจร 7-segment แบบ Common Cathode ขา Common ต้องต่อเข้ากับขาใดบนบอร์ด Arduino

.....

.....

4. คำสั่งใดที่ใช้สำหรับสั่งให้ Segment ที่ต้องการติดสว่าง? และในกรณีนี้ต้องส่งสัญญาณเป็นค่าใด (HIGH หรือ LOW)

.....

.....

5. เหตุใดจึงต้องต่อตัวต้านทาน 220 โอห์ม อนุกรมเข้ากับแต่ละ Segment ของ 7-segment display

.....

.....

6. หากคุณต้องการให้ 7-segment แสดงผลเป็นตัวเลข '4' คุณจะต้องทำให้ Segment ใดบ้างติดสว่าง

.....

.....

7. ถ้าต้องการให้แต่ละตัวเลขแสดงผลนานขึ้นจาก 1 วินาที เป็น 3 วินาที จะต้องแก้ไขโค้ดอย่างไร

.....

.....

8. คำสั่ง `Serial.begin(9600);` และ `Serial.println();` มีหน้าที่แตกต่างกันอย่างไรในการทดลองนี้?

.....

.....

9. เหตุใดจึงต้องใช้ `#define` เพื่อกำหนดค่าพินแทนการใช้ตัวเลขโดยตรงในโค้ด และมีข้อดีอย่างไร?

.....

.....

สรุปการทดลอง

.....

.....

.....

.....

.....

.....

.....

.....

คำถามท้ายบท

คำชี้แจง ให้นักเรียนทำเครื่องหมายกากบาท (X) ลงบนคำตอบที่ถูกต้องที่สุดเพียงคำตอบเดียว

1. โค้ดส่วนใดในโปรแกรม Arduino ที่ทำหน้าที่ในการตั้งค่าเริ่มต้นและจะทำงานเพียง ครั้งเดียว เมื่อบอร์ดเริ่มต้นทำงาน?

- ก. void loop() ข. void setup() ค. digitalWrite() ง. delay()

2. คำสั่ง digitalWrite(LED_PIN, HIGH); มีความหมายว่าอย่างไร?

- ก. เป็นการสั่งให้ LED ดับลง
ข. เป็นการสั่งให้ LED ติดสว่าง
ค. เป็นการหน่วงเวลาการทำงานของ LED
ง. เป็นการกำหนดโหมดการทำงานของขา LED_PIN

3. ข้อใดคือความหมายที่ถูกต้องของ 7-segment display แบบ Common Cathode และวิธีการ ทำให้ Segment ติดสว่าง?

- ก. ขา Common ต้องต่อกับไฟเลี้ยง 5V และใช้สัญญาณ HIGH เพื่อทำให้ Segment ติดสว่าง
ข. ขา Common ต้องต่อกับกราวด์ (GND) และใช้สัญญาณ HIGH เพื่อทำให้ Segment ติดสว่าง
ค. ขา Common ต้องต่อกับไฟเลี้ยง 5V และใช้สัญญาณ LOW เพื่อทำให้ Segment ติดสว่าง
ง. ขา Common ต้องต่อกับกราวด์ (GND) และใช้คำสั่ง delay() เพื่อทำให้ Segment ติดสว่าง

4. คำสั่ง delay(1000); ในโปรแกรม Arduino มีความหมายว่าอย่างไร?

- ก. หน่วงเวลาการทำงานของโปรแกรมเป็นเวลา 1000 วินาที
ข. ทำให้โปรแกรมทำงานเร็วขึ้น 1000 เท่า
ค. หน่วงเวลาการทำงานของโปรแกรมเป็นเวลา 1 วินาที
ง. ตรวจสอบสถานะของขา Digital Pin 1000

5. หากต้องการให้ 7-segment display แสดงผลตัวเลข '1' บนหน้าจอ สิ่งที่คุณต้องควบคุม คืออะไร?

- ก. ควบคุมเฉพาะขาที่ต่อกับ Segment 'a' เท่านั้น
ข. ควบคุมขา Digital ทุกขาบน Arduino พร้อมกัน
ค. ควบคุมสถานะ (เปิด/ปิด) ของ Segment ที่จำเป็นสำหรับตัวเลข '1' เท่านั้น
ง. ควบคุมขา Digital ที่ต่อกับ Serial Monitor

6. ข้อใดอธิบายความแตกต่างหลักระหว่าง Serial.print() และ Serial.println() เมื่อแสดงผลบน Serial Monitor ได้ถูกต้องที่สุด?

- ก. ทั้งคู่แสดงผลเหมือนกัน แต่ Serial.println() จะขึ้นบรรทัดใหม่หลังแสดงผล
- ข. Serial.print() แสดงผลเป็นตัวเลขเท่านั้น ส่วน Serial.println() แสดงผลเป็นข้อความเท่านั้น
- ค. Serial.println() ใช้สำหรับส่งข้อมูลไปยัง 7-segment display ส่วน Serial.print() ใช้กับ Serial Monitor
- ง. Serial.print() ทำงานเร็วกว่า Serial.println() มาก

7. การเขียนโปรแกรมสำหรับ 7-segment แบบ Common Anode มีหลักการทำงานอย่างไร?

- ก. ต่อขาร่วมกับกราวด์ (GND) และใช้สัญญาณ HIGH เพื่อเปิด Segment
- ข. ต่อขาร่วมกับไฟเลี้ยง (5V) และใช้สัญญาณ HIGH เพื่อเปิด Segment
- ค. ต่อขาร่วมกับไฟเลี้ยง (5V) และใช้สัญญาณ LOW เพื่อเปิด Segment
- ง. ต่อขาร่วมกับกราวด์ (GND) และใช้สัญญาณ LOW เพื่อเปิด Segment

8. ข้อใดอธิบายหน้าที่และลักษณะของ 7-segment display ได้ถูกต้องที่สุด?

- ก. เป็นอุปกรณ์ที่ใช้แสดงผลข้อความขนาดใหญ่ มีหลอดไฟ LED 7 ดวง
- ข. เป็นอุปกรณ์ที่ใช้แสดงผลตัวเลข มีหลอดไฟ LED 7 ดวงที่สามารถเปิด-ปิด เพื่อสร้างรูปทรงตัวเลขได้
- ค. เป็นอุปกรณ์ที่รับค่าตัวเลขจากผู้ใช้ มีหลอดไฟ LED 7 ดวง
- ง. เป็นอุปกรณ์ที่ใช้สำหรับวัดค่าอุณหภูมิ มีหลอดไฟ LED 7 ดวง

9. หากคุณต้องการให้ 7-segment display แสดงผลตัวเลข '7' บนหน้าจอ โดยใช้การเขียนโปรแกรมแบบเรียงลำดับ คุณจะต้องส่งสัญญาณอย่างไรไปยัง Segment ที่เกี่ยวข้อง?

- ก. ส่งสัญญาณ HIGH ไปที่ Segment a, b, c, g และ LOW ไปที่ Segment อื่นๆ
- ข. ส่งสัญญาณ HIGH ไปที่ Segment b, c, g และ LOW ไปที่ Segment อื่นๆ
- ค. ส่งสัญญาณ HIGH ไปที่ Segment a, b, c และ LOW ไปที่ Segment อื่นๆ
- ง. ส่งสัญญาณ LOW ไปที่ Segment a, b, c และ HIGH ไปที่ Segment อื่นๆ

10. ในการเขียนโปรแกรมควบคุม 7-segment display ข้อใดคือเหตุผลหลักที่ควรใช้ #define เพื่อกำหนดค่าพินแทนการใช้ตัวเลขโดยตรง?

- ก. เพื่อให้โปรแกรมทำงานเร็วขึ้น
- ข. เพื่อป้องกันการเขียนโค้ดผิดพลาดและทำให้แก้ไขได้ง่ายขึ้นในภายหลัง
- ค. เพื่อลดขนาดของไฟล์โปรแกรม
- ง. เพื่อให้สามารถต่อ 7-segment display ได้หลายตัวพร้อมกัน