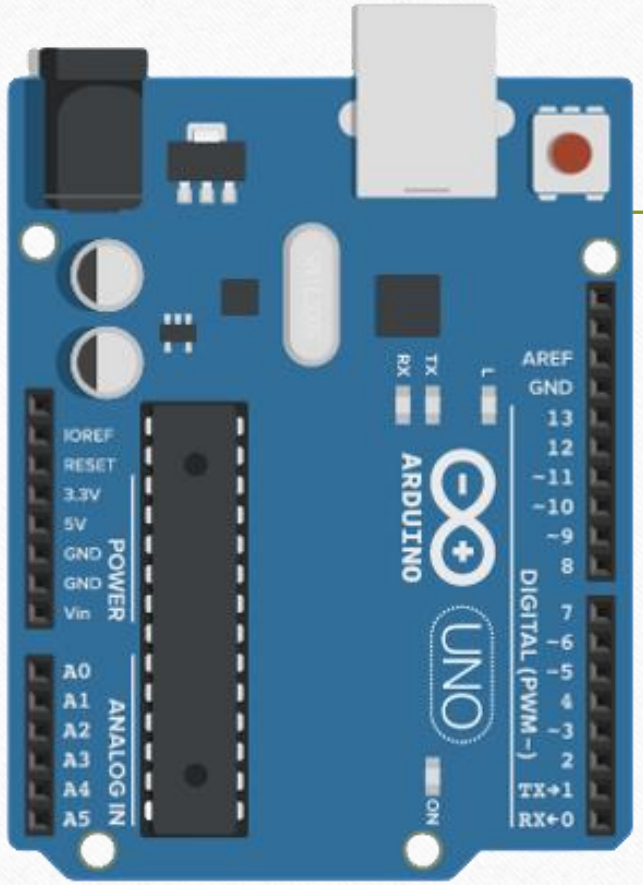
หน่วยที่ 4

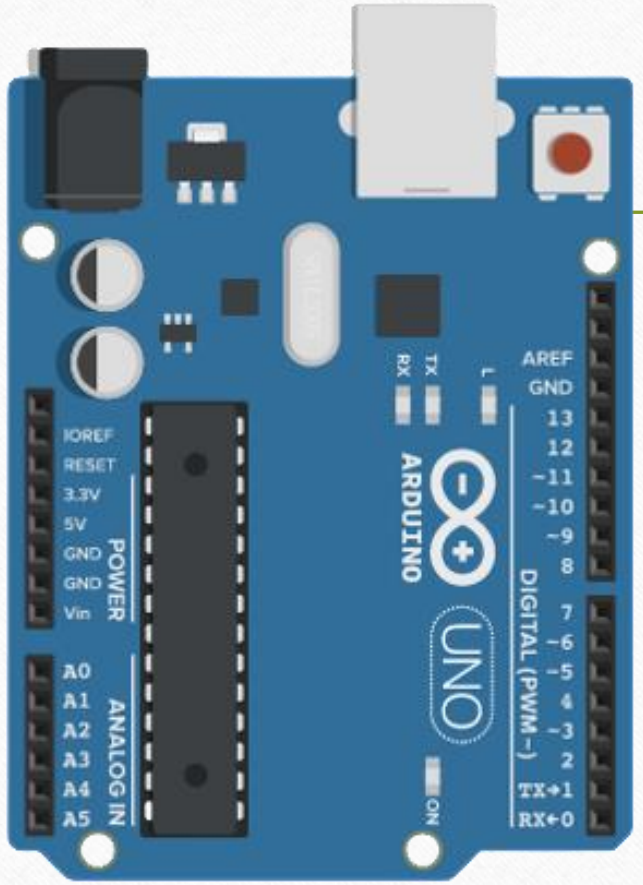
Arduino กับพอร์ตดิจิตอลเอาต์พุต

บอร์ด Arduino



- ซิปไอซีไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega328
- พอร์ตดิจิตอลจำนวน 14 ขา
- ตั้งแต่ขา D0, D1, D2,....., D13
- มี 2 ขาที่ทำหน้าที่อื่น คือ ขา D0 และ D1 ทำหน้าที่เป็นขา Rx และ Tx ใช้สำหรับติอต่อสื่อสารกับคอมพิวเตอร์แบบอนุกรม
- ถ้าเสียบสาย USB กับคอมพิวเตอร์ขา D0 และ D1 จะถูกใช้งานรับส่งข้อมูลไม่สามารถใช้งานเป็นพอร์ตปกติได้

คุณสมบัติของพอร์ตดิจิตอลเอาต์พุต



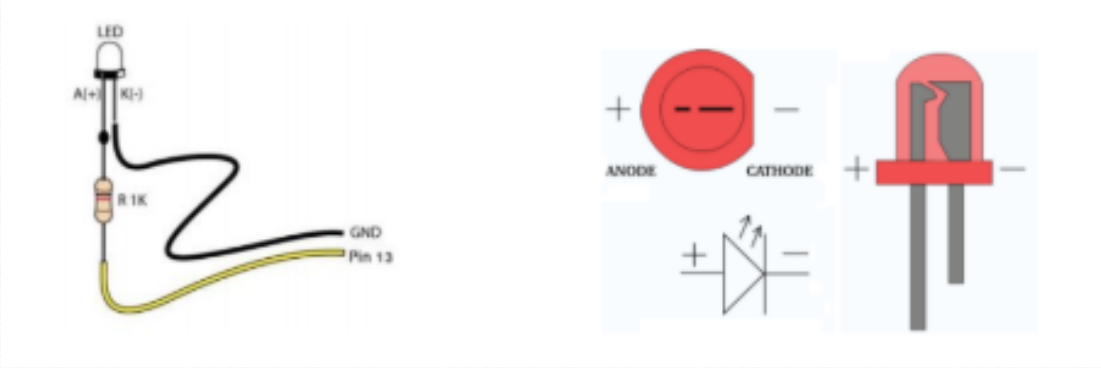
- พอร์ตดิจิตอลทั้ง 14 พอร์ตสามารถจ่ายกระแสให้กับอุปกรณ์อิเล็กทรอนิกส์
- กระแสที่จ่ายได้คือ 40 มิลลิแอมป์
- กระแส 40 มิลลิแอมป์สามารถนำไปขับหลอด LED ให้ติดสว่างได้แต่ต้องต่อตัวต้านทานเพื่อจำกัดกระแสไว้ด้วย
- ส่วนอุปกรณ์อิเล็กทรอนิกส์ที่กินกระแสมากขึ้นต้องต่อวงจรขับกระแส เช่น ใช้ทรานซิสเตอร์หรือเฟตเพื่อขับโหลดที่กินกระแสมากๆ

การใช้งาน Arduino กับอุปกรณ์เอาต์พุตพื้นฐาน

- การใช้งานไมโครคอนโทรลเลอร์ Arduino สามารถใช้ร่วมกับอุปกรณ์อินพุตและเอาต์พุตได้ทั้งแบบดิจิตอลและแอนาล็อก โดยมีฟังก์ชันสำเร็จรูปและไฟล์ไลบรารีสนับสนุนการใช้และการใช้งานไมโครคอนโทรลเลอร์ Arduino กับอุปกรณ์เอาต์พุตพื้นฐาน คือ ไดโอดเปล่งแสง (light emitting diode)

การต่อ Arduino กับหลอดแอลอีดี

- หลอดแอลอีดี (LED) เป็นไดโอดชนิดหนึ่งที่สามารถเปล่งแสงออกมาได้เมื่อได้รับการไบอัสตรง แสงที่เปล่งออกมามีหลายสี ขึ้นอยู่กับชนิดของสารกึ่งตัวนำที่นำมาทำ เช่น แสงอินฟราเรด (Infrared) แสงสีแดง สีเหลือง สีส้ม สีเขียว เป็นต้น



การต่อ Arduino กับหลอดแอลอีดี

- การต่อบอร์ด Arduino เข้ากับหลอดแอลอีดีที่พอร์ตเอาต์พุตของบอร์ด Arduino มีแรงดันขณะส่งลอจิก “1” อยู่ที่ 5 โวลต์ ดังนั้นจึงต้องนำตัวต้านทานมาต่อเพื่อจำกัดกระแสให้น้อยลง ถ้าแรงดันที่พอร์ตเอาต์พุตมีค่า 5 โวลต์ ให้มีแรงดันตกคร่อมหลอดแอลอีดีเท่ากับ 2 โวลต์ ดังนั้นแรงดันตกคร่อมตัวต้านทานเท่ากับ 3 โวลต์ กำหนดให้กระแสที่ไหลผ่านหลอดแอลอีดี 10 มิลลิแอมป์ หาค่าตัวต้านทาน

การต่อ Arduino กับหลอดแอลอีดี

ค่าของตัวต้านทาน (โอห์ม) = (แรงดันจากไมโครฯ - แรงดันของ LED) (V) / กระแส LED (A)

$$R = \frac{E_T - E_{LED}}{I} = \frac{5V. - 2V.}{10mA.} = \frac{3V.}{10mA.} = 300\Omega$$

การต่อหลอดแอลอีดีกับบอร์ด Arduino ตัวต้านทานที่นิยมใช้กันมีค่า 330Ω กับ 220Ω

การต่อ Arduino กับหลอดแอลอีดี

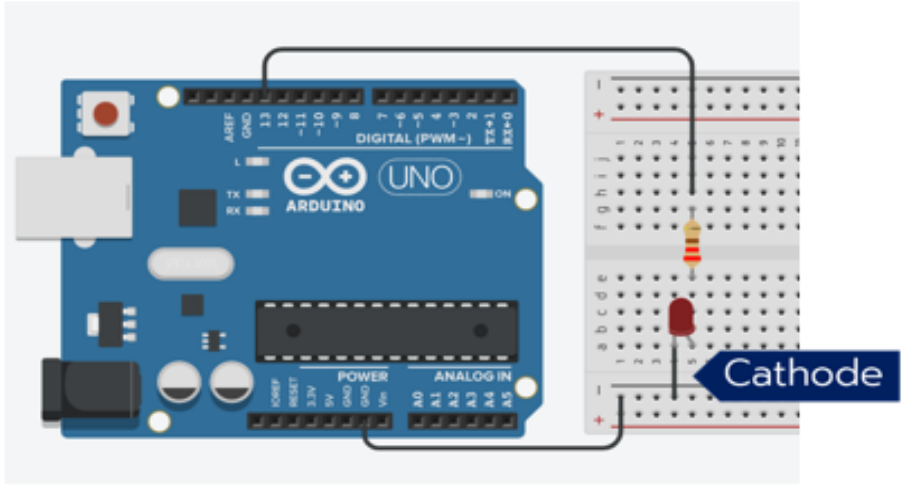
การเชื่อมต่อขาพอร์ตของบอร์ด Arduino กับวงจรแอลอีดี มีรูปแบบการต่อ 2 แบบ

1. แบบควบคุมการทำงานด้วยการจ่ายกระแสออก (Source current) หรือที่เรียกว่าการต่อแบบแคโทด

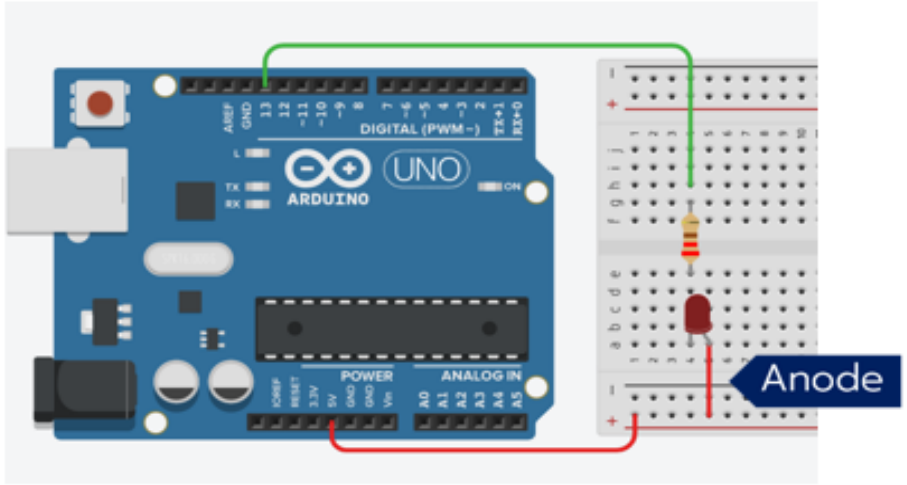
2. แบบควบคุมการทำงานด้วยการรับกระแสเข้า (Sink current) หรือที่เรียกว่าการต่อแบบแอโนด

การต่อ Arduino กับหลอดแอลอีดี

- รูปแบบการต่อขาพอร์ตของบอร์ด Arduino กับวงจรแอลอีดี

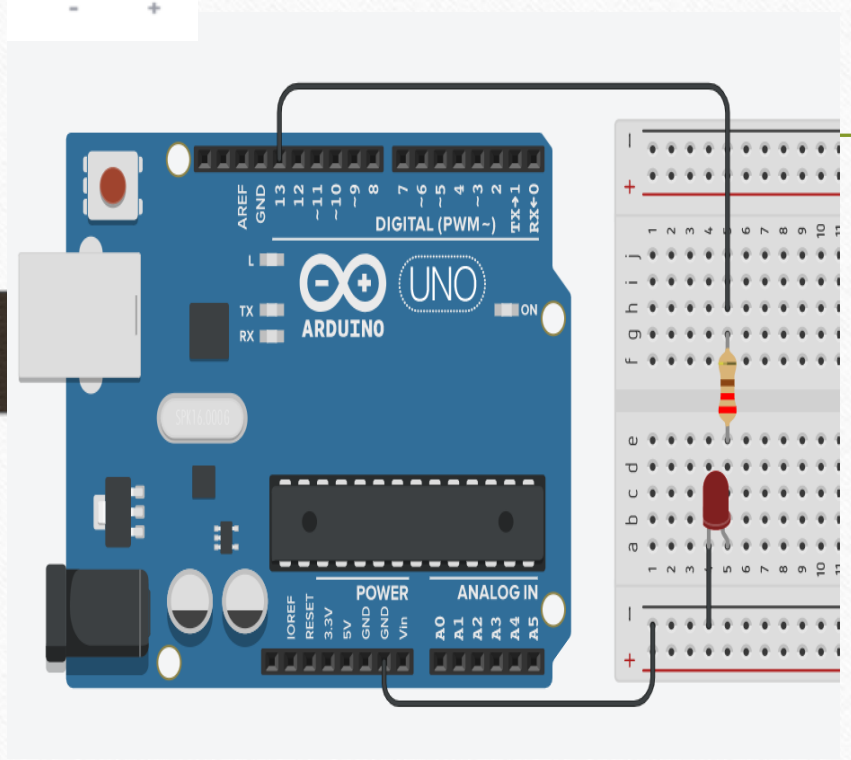


การจ่ายกระแสออก



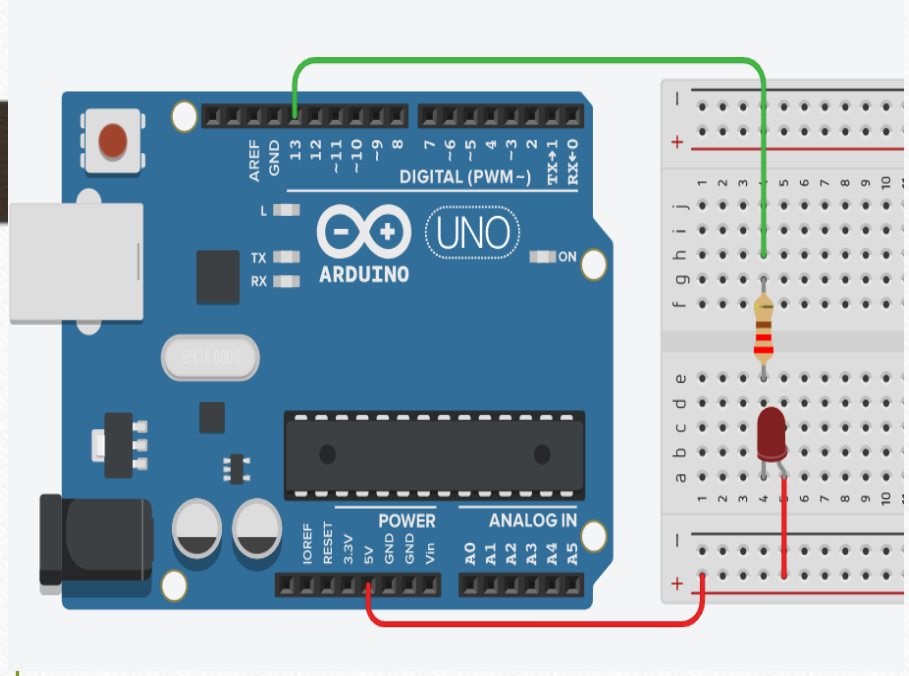
การรับกระแสเข้า

การต่อ Arduino กับหลอดแอลอีดี



จากรูปการควบคุมการติดดับของ LED ที่ต่อแบบการจ่ายกระแสออก (Source Current) หรือการต่อแบบแคโทดคือขาแอโนด (A) ของหลอดแอลอีดีต่อเข้ากับไมโครคอนโทรลเลอร์โดยผ่านตัวต้านทานและขาแคโทด (K) ต่อลงกราวนด์ หากต้องการให้ LED13 ติดสว่างต้องส่งค่าลอจิก HIGH และถ้าต้องการให้ LED13 ดับ ต้องส่งค่าลอจิก LOW ที่ขาพอร์ต Digital Pin13

การต่อ Arduino กับหลอดแอลอีดี



ส่วนการควบคุมการติดดับของ LED ที่ต่อแบบการรับกระแสเข้า (Sink Current) หรือการต่อแบบแอโนดคือขาแอโนด (A) ของหลอดแอลอีดีถูกต่อเข้ากับแรงดันไฟ VCC หรือไฟ +5 โวลต์ และขาแคโทด (K)ต่อเข้าไมโครคอนโทรลเลอร์โดยผ่านตัวต้านทาน ถ้าต้องการให้ LED13 ติดสว่างต้องส่งค่าลอจิก LOW และถ้าต้องการให้ LED13 ดับต้องส่งค่าลอจิก HIGH ที่ขาพอร์ต Digital Pin13

การเขียนโปรแกรมควบคุม LED

- อุปกรณ์เอาต์พุต LED สามารถเชื่อมต่อและควบคุมผ่านขาพอร์ตของบอร์ด Arduino ได้แก่ ขาพอร์ต Digital Pin0 ถึง Digital Pin13 การเขียนโปรแกรมเพื่อใช้งานเป็นพอร์ตดิจิตอลเอาต์พุต เริ่มต้นด้วยการกำหนดโหมดการทำงานให้กับพอร์ตโดยใช้คำสั่ง

การเขียนโปรแกรมควบคุม LED

1. pinMode (Pin, Mode)

- เป็นคำสั่งสำหรับกำหนดการทำงานของพอร์ตที่ต้องการใช้งานให้เป็นพอร์ตเอาต์พุตหรือพอร์ตอินพุต

```
pinMode (ตำแหน่งขาพอร์ต, โหมด INPUT หรือ OUTPUT);
```

```
pinMode (7, OUTPUT);
```

การเขียนโปรแกรมควบคุม LED

2. digitalWrite (Pin, Status)

- ใช้ทำหน้าที่สำหรับกำหนดสถานะทางสัญญาณเอาต์พุต ต้องการให้มีสภาวะลอจิกเป็น 1 หรือ 0 ซึ่งในสภาวะ 1 สามารถใช้แทนได้คือคำสั่ง HIGH และ 0 คือ LOW ขาสัญญาณจะใช้คำสั่ง digitalWrite () ต้องเป็นขาสัญญาณที่เป็นเอาต์พุต

```
digitalWrite (ตำแหน่งขาพอร์ต, ค่า HIGH หรือ LOW);
```

```
digitalWrite (7, HIGH);
```

การเขียนโปรแกรมควบคุม LED

3. delay(millisecond หรือ ms);

- เป็นคำสั่งสำหรับหน่วงเวลามีหน่วยเป็นมิลลิวินาที

1 มิลลิวินาที = 0.001 วินาที หรือ 1/1000 วินาที

แปลงหน่วยมิลลิวินาทีเป็นวินาที = หารด้วย 1000

แปลงหน่วยวินาทีเป็นมิลลิวินาที = คูณด้วย 1000

การเขียนโปรแกรมควบคุม LED

4. for()

- เป็นคำสั่งสำหรับใช้ในกรณีที่ทราบจำนวนรอบของการทำงานซ้ำ

```
for ( การกำหนดค่าเริ่มต้นของตัวนับ, เงื่อนไขการทำซ้ำ;
      การเปลี่ยนแปลงค่าตัวนับ(การเพิ่มค่าหรือลดค่า) )
```

```
for (int i = 0, i < 8; i++)
```

```
for (int i = 7, i >=0; i--)
```

การเขียนโปรแกรมควบคุม LED

5. array

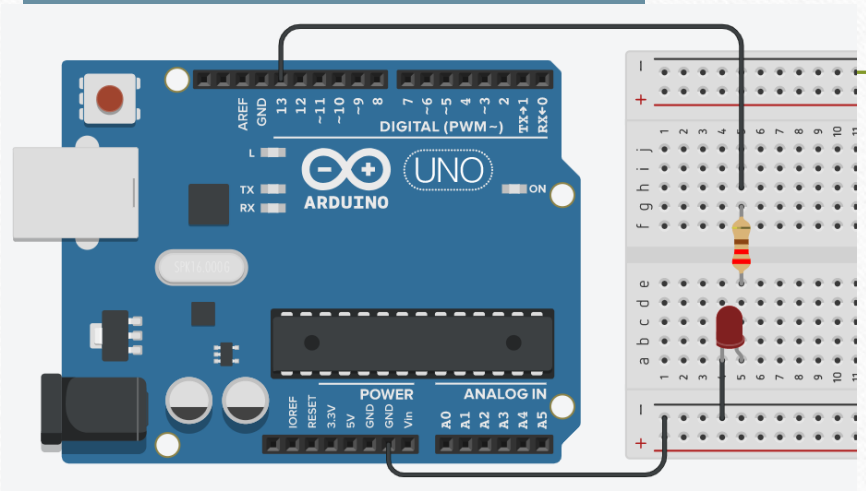
- เป็นตัวแปรสำหรับใช้งานตัวแปรหลายๆตัวที่มีชนิดข้อมูลแบบเดียวกันเพื่อใช้ในการเก็บข้อมูลหรือแสดงผล

ชนิดของข้อมูลที่ให้ตัวแปรนี้เก็บค่าไว้ ชื่อของตัวแปรที่กำหนดขึ้นเอง
[ขนาดของอาร์เรย์] = {รายการค่าของตัวแปรที่ต้องการกำหนดค่าให้กับอาร์เรย์};

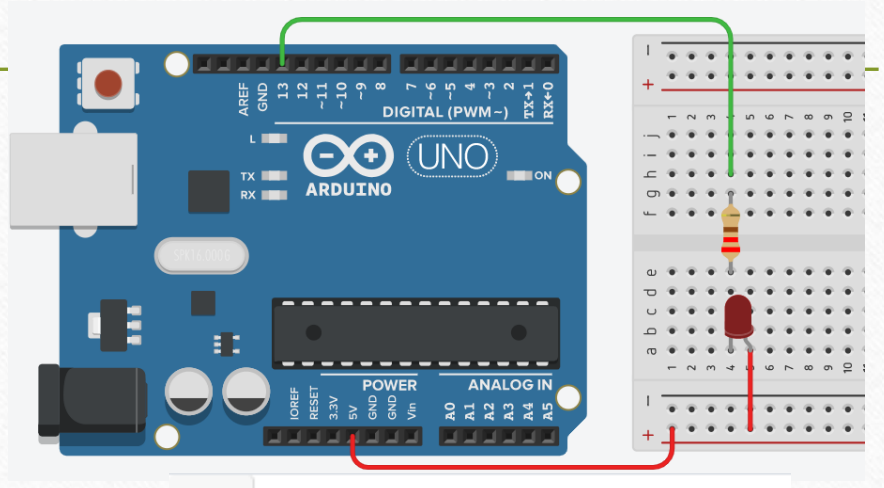
```
const int led [8] = {2,3,4,5,6,7,8,9} ;
```

ตัวอย่างการเขียนโปรแกรมควบคุม LED

LED ติดและดับ 1 ครั้ง



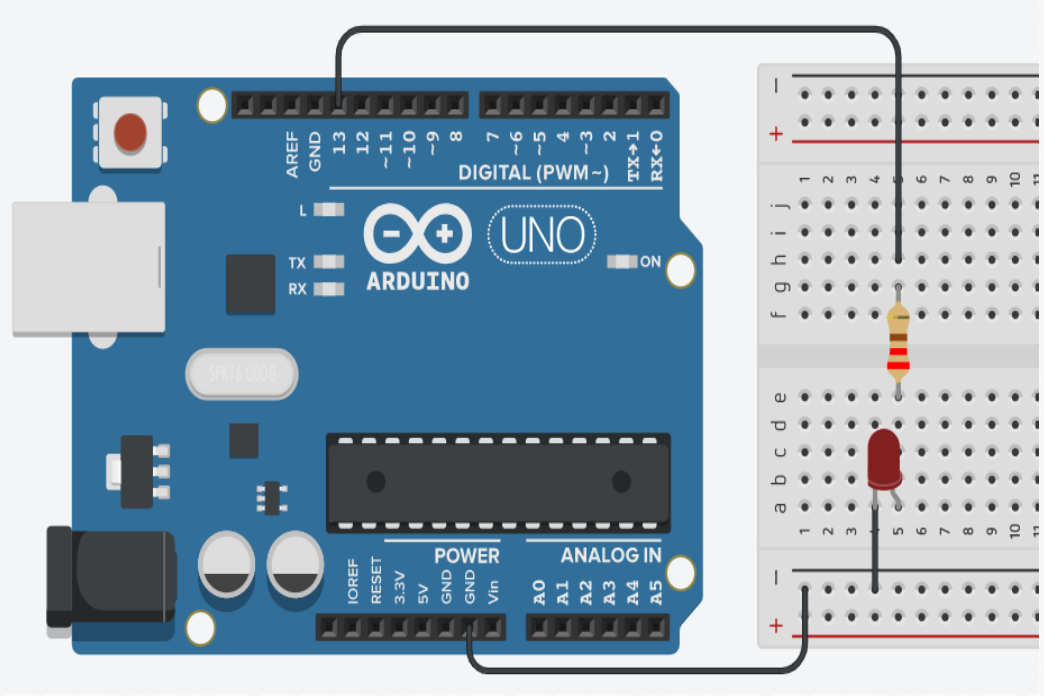

```
1  void setup()
2  {
3    pinMode(13, OUTPUT);
4    digitalWrite(13, HIGH);
5    delay(1000);
6    digitalWrite(13, LOW);
7    delay(1000);
8  }
9
10 void loop()
11 {
12
13 }
```




```
1  void setup()
2  {
3    pinMode(13, OUTPUT);
4    digitalWrite(13, LOW);
5    delay(1000);
6    digitalWrite(13, HIGH);
7    delay(1000);
8  }
9
10 void loop()
11 {
12
13 }
```

ตัวอย่างการเขียนโปรแกรมควบคุม LED

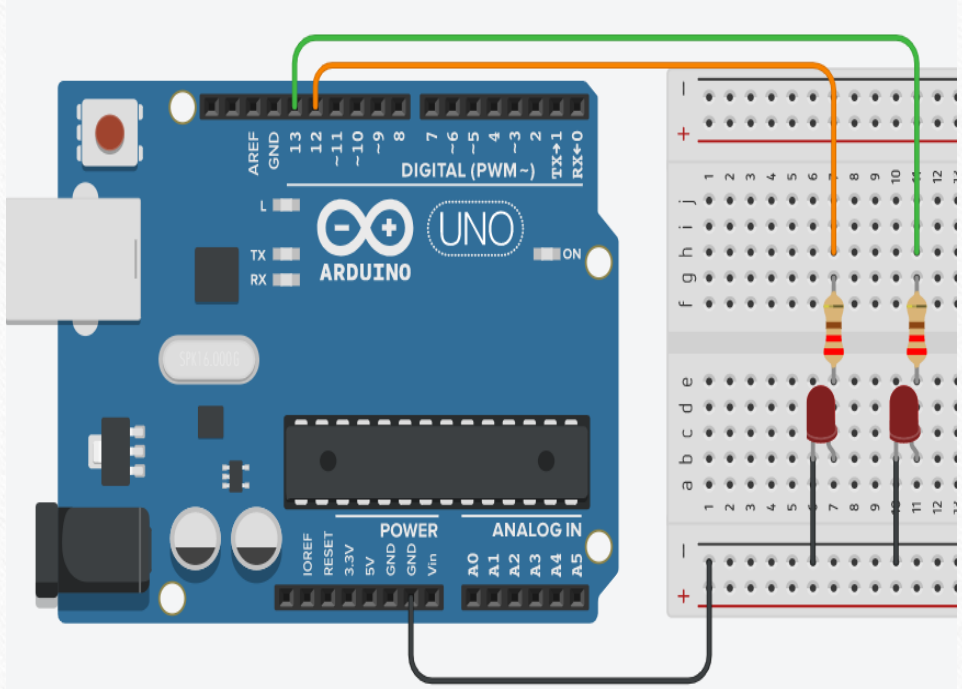
LED ติดและดับ ทำซ้ำตลอดเวลา (กระพริบ)




```
1  void setup()
2  {
3    pinMode(13, OUTPUT);
4  }
5
6  void loop()
7  {
8    digitalWrite(13,HIGH);
9    delay(300);
10   digitalWrite(13, LOW);
11   delay(300);
12 }
```

ตัวอย่างการเขียนโปรแกรมควบคุม LED

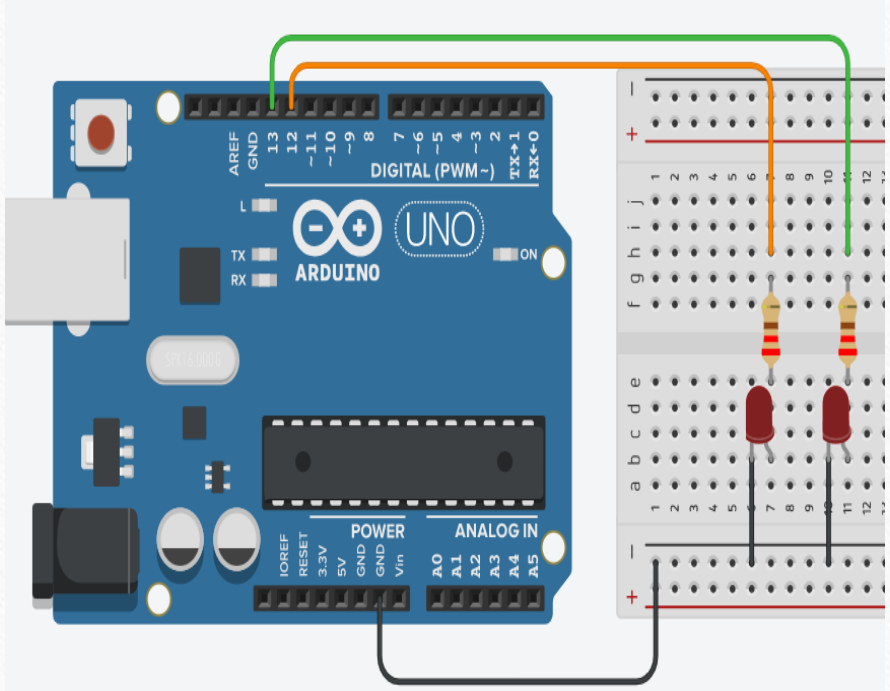
LED 2 ดวง ติดและดับพร้อมกัน ทำซ้ำตลอดเวลา (กระพริบ)




```
1  void setup()
2  {
3    pinMode(13, OUTPUT);
4    pinMode(12, OUTPUT);
5  }
6
7  void loop()
8  {
9    digitalWrite(13,HIGH);
10   digitalWrite(12,HIGH);
11   delay(300);
12   digitalWrite(13, LOW);
13   digitalWrite(12, LOW);
14   delay(300);
15 }
```

ตัวอย่างการเขียนโปรแกรมควบคุม LED

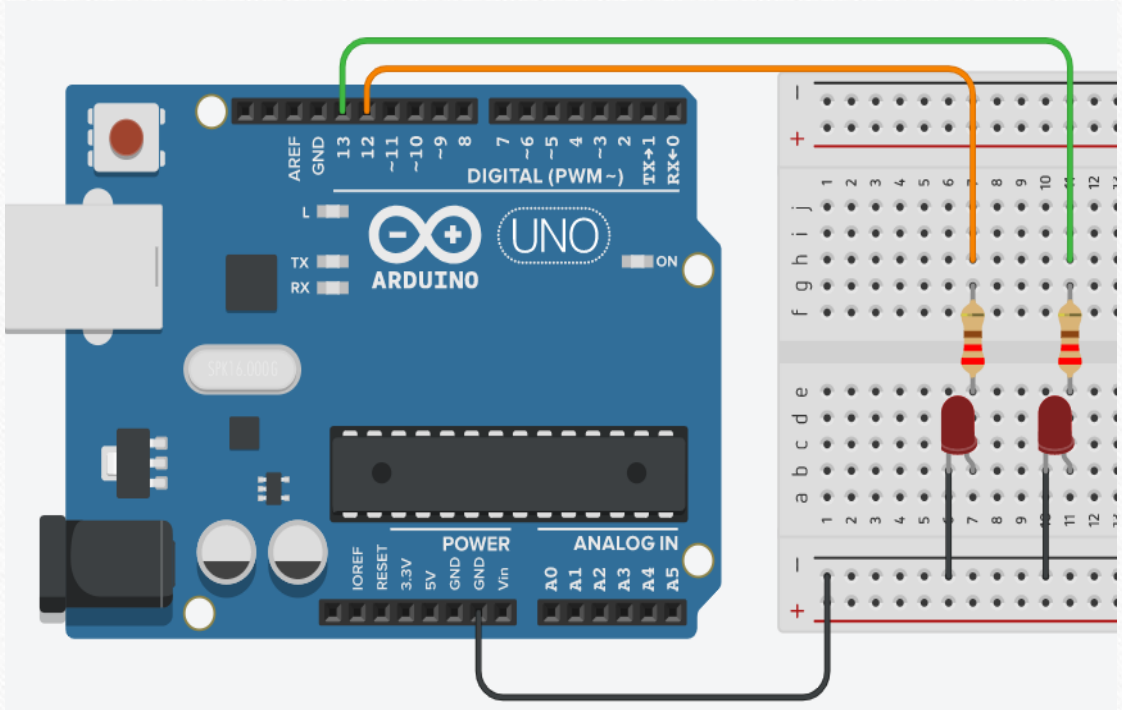
LED สลับกันทำงานโดยกำหนดให้ติดและดับครั้งละ 1 ดวง ทำซ้ำตลอดเวลา




```
1  void setup()
2  {
3    pinMode(13, OUTPUT);
4    pinMode(12, OUTPUT);
5  }
6
7  void loop()
8  {
9    digitalWrite(13,HIGH);
10   delay(500);
11   digitalWrite(13, LOW);
12   delay(500);
13   digitalWrite(12,HIGH);
14   delay(500);
15   digitalWrite(12, LOW);
16   delay(500);
17
18 }
```

ตัวอย่างการเขียนโปรแกรมควบคุม LED

LED ติดเพิ่มขึ้นทีละ 1 ดวง ให้ติดค้างไว้จนครบทุกดวง
และดับพร้อมกัน ทำซ้ำตลอดเวลา




```
1  void setup()
2  {
3    pinMode(13, OUTPUT);
4    pinMode(12, OUTPUT);
5  }
6
7  void loop()
8  {
9    digitalWrite(13,HIGH);
10   delay(1000);
11   digitalWrite(12,HIGH);
12   delay(1000);
13   digitalWrite(13, LOW);
14   digitalWrite(12, LOW);
15   delay(500);
16 }
```

จบการเรียนการสอน