

# Young Digital Maker





## ทีมวิทยากร

- Full-time lecturers at IT, Mahanakorn University of Technology.
- ดร. เอกรัฐ รัชกาญจน์ (อ. ป๋อก)
- ดร. ภากร จูเหล็ง (อ. น้อย)
- อ.มุกกระวี มะตะเรส (อ. มุก)



# Outline

## Day 1 (เช้า)

- เรียนรู้การใช้ Simulation design tool (Tinkercad)
- พื้นฐานอิเล็กทรอนิกส์
- แนะนำ Arduino
- อุปกรณ์อิเล็กทรอนิกส์ที่ใช้ในการเรียน



# Outline

## Day 1 (บ่าย)

- ติดตั้ง Arduino IDE
- เรียนรู้การสร้าง electronic เบื้องต้น จาก simulation สู่ปฏิบัติจริง



# Outline

## Day 2 (เช้า)

- การทดลอง LED: ไฟกระพริบหลายดวง
- การทดลอง LED: ไฟวิ่ง
- การทดลอง LED: 7-Segment



# Outline

## Day 2 (บ่าย)

- การทดลอง LED + Button
- การทดลอง Sensor แสง
- การทดลอง Sensor อุณหภูมิ



# Outline

## Day 3 (เช้า)

- การทดลอง Sensors วัดระยะทาง
- การทดลอง Motor
- แบ่งกลุ่มระดมสมองออกแบบโปรเจค



# Outline

## Day 3 (บ่าย)

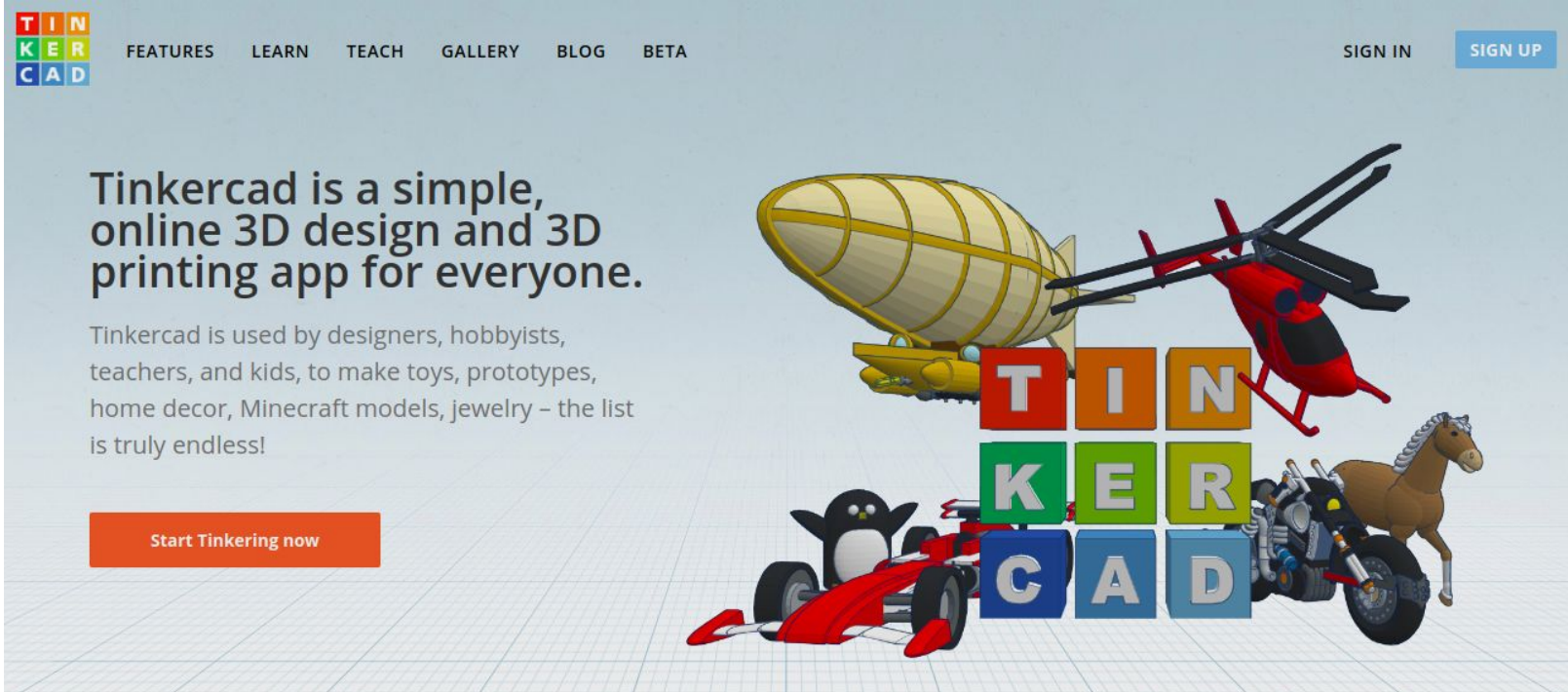
- ลงมือสร้างโปรเจค
- เสนอโปรเจค



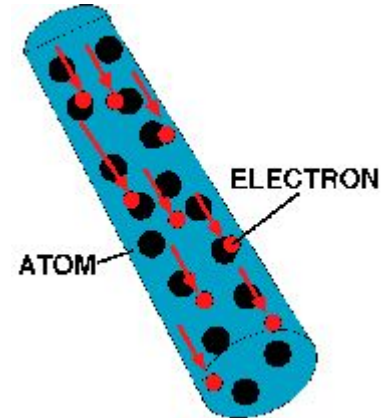
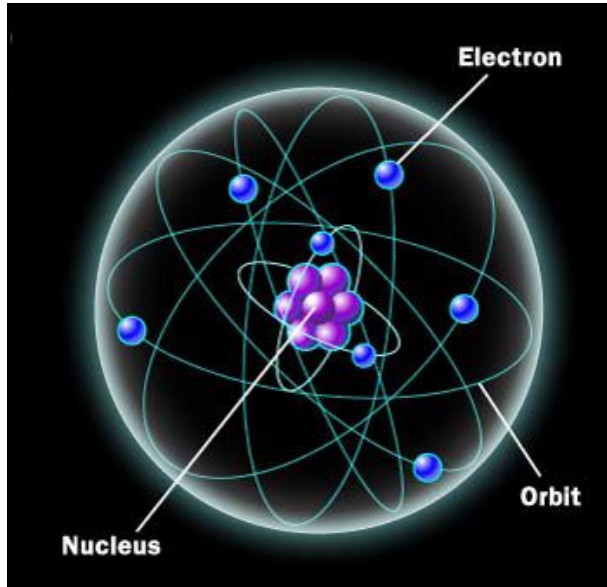


## 1.1 พื้นฐานอิเล็กทรอนิกส์

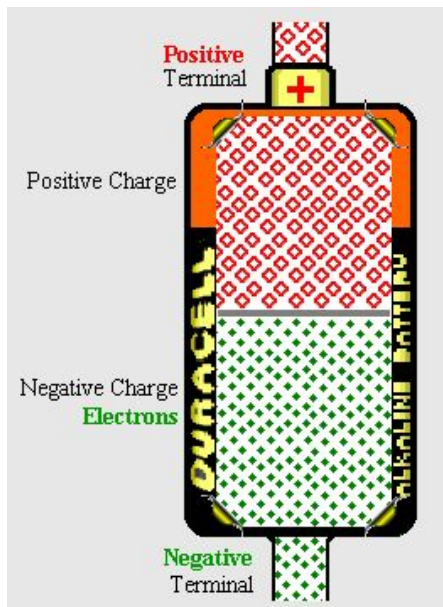
# <https://www.tinkercad.com>



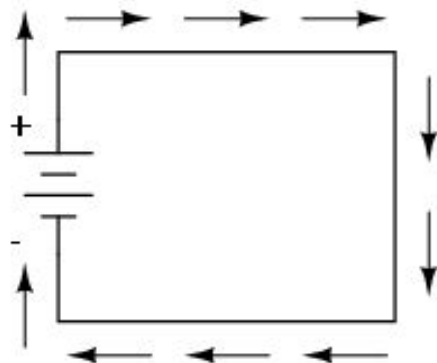
# 1.1 Current



# 1.1 Current

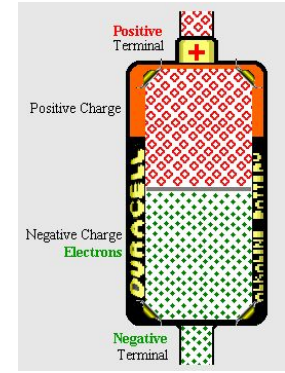


*Conventional flow notation*

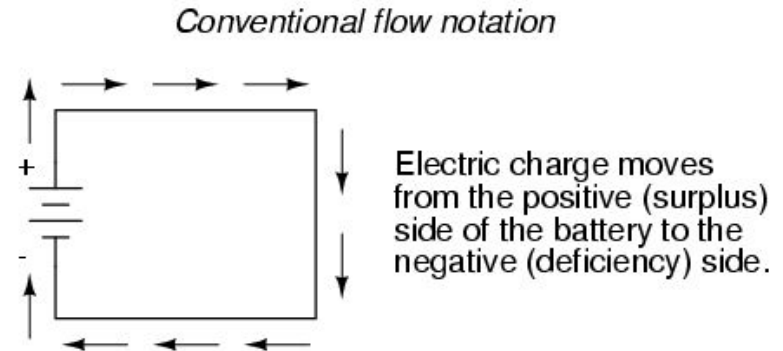
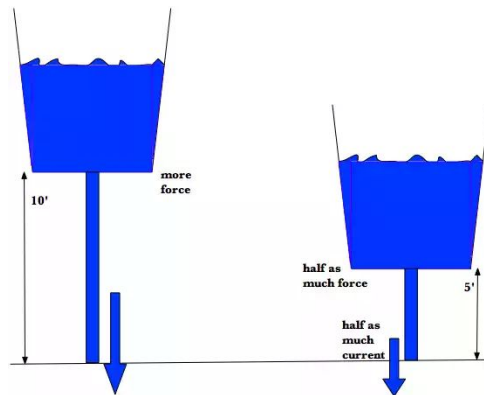


Electric charge moves from the positive (surplus) side of the battery to the negative (deficiency) side.

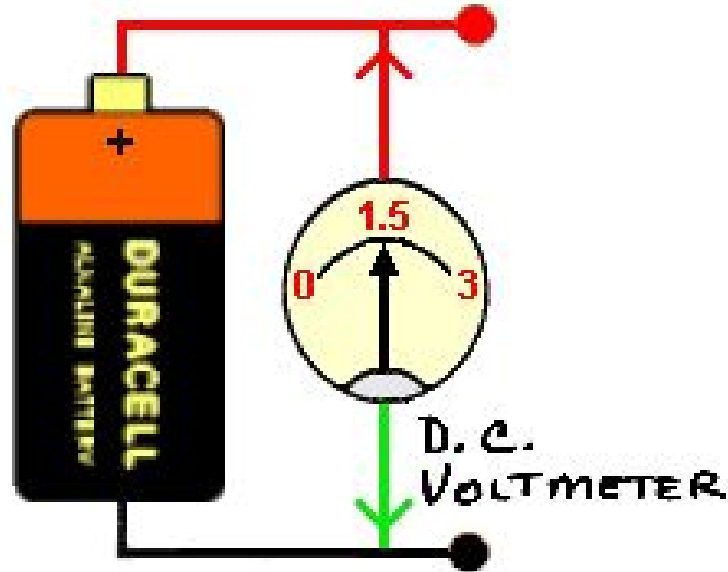
## 1.2 Voltage



- A battery positive terminal (+) and a negative terminal (-). The difference in charge between each terminal is the potential energy the battery can provide. This is labeled in units of volts.



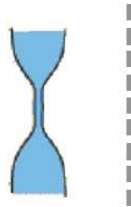
## 1.2 Voltage





## 1.3 Resistance

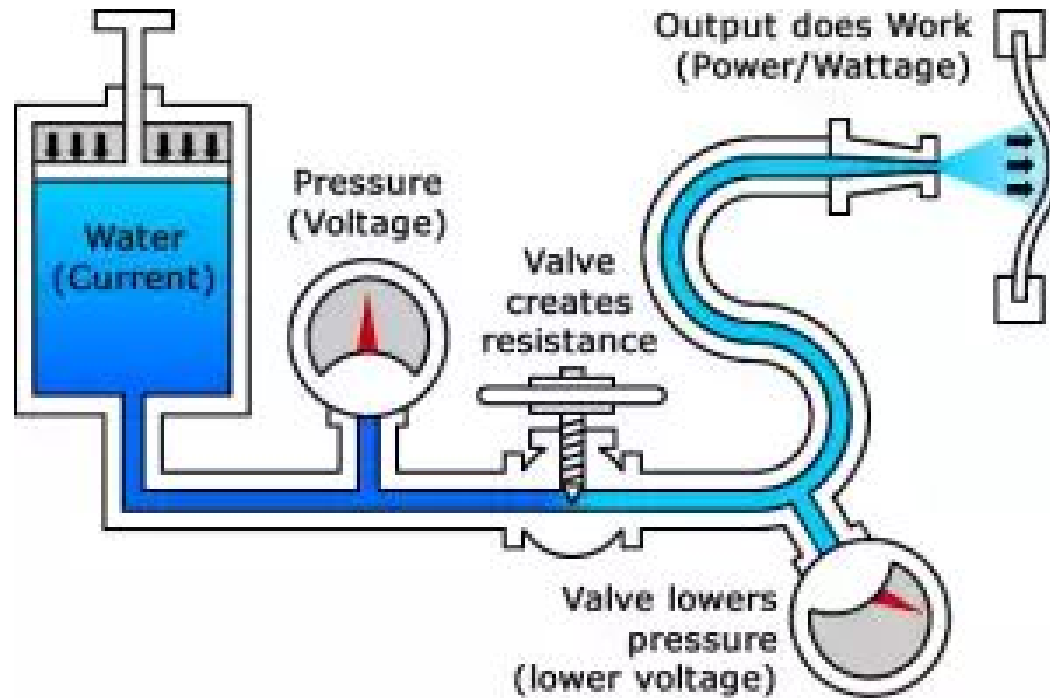
Constriction  
creates  
Resistance to water flow



Resistor creates  
Resistance to current  
flow



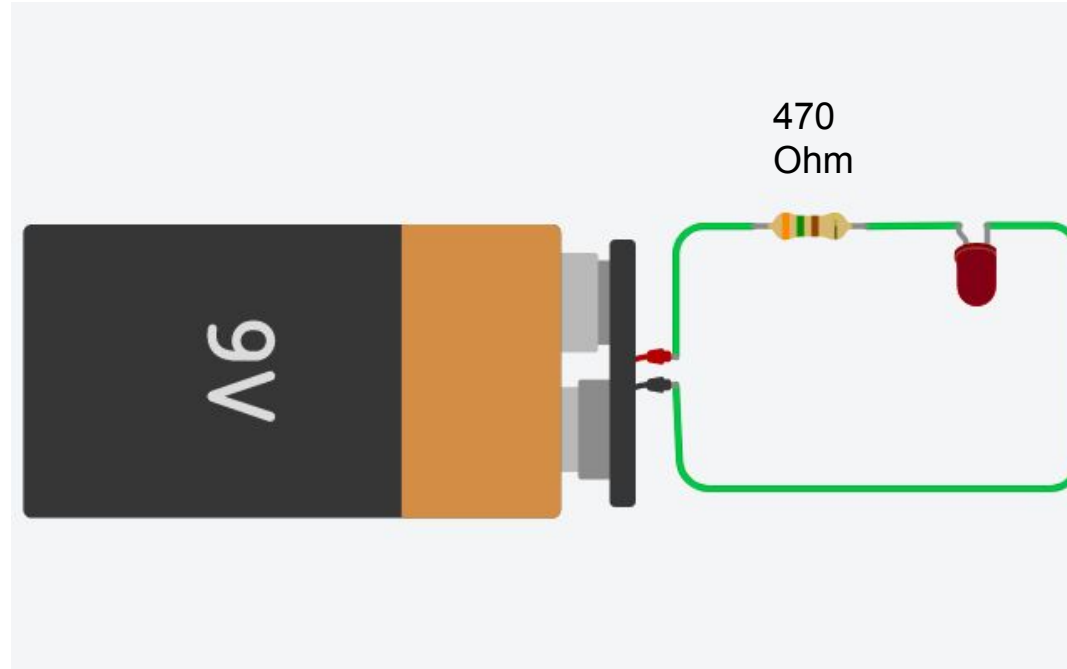
# Summation



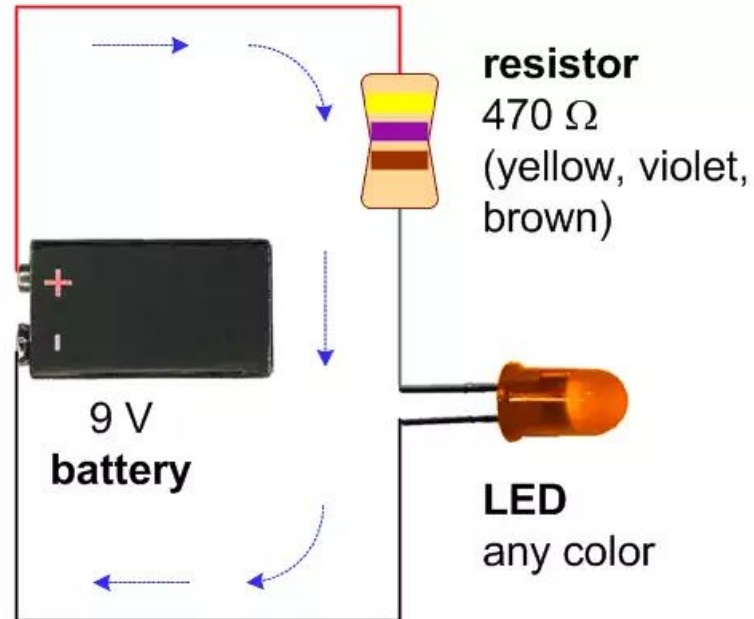




# Experiment 1



# Experiment 1 (Explain)



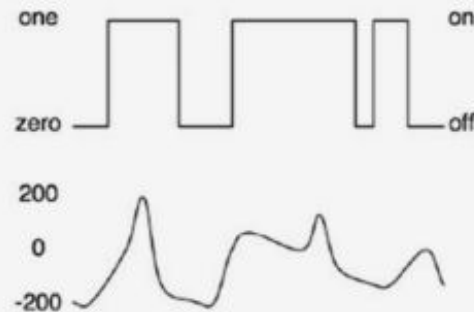


## **1.2 Digital & Analog**

# Digital? Analog?

---

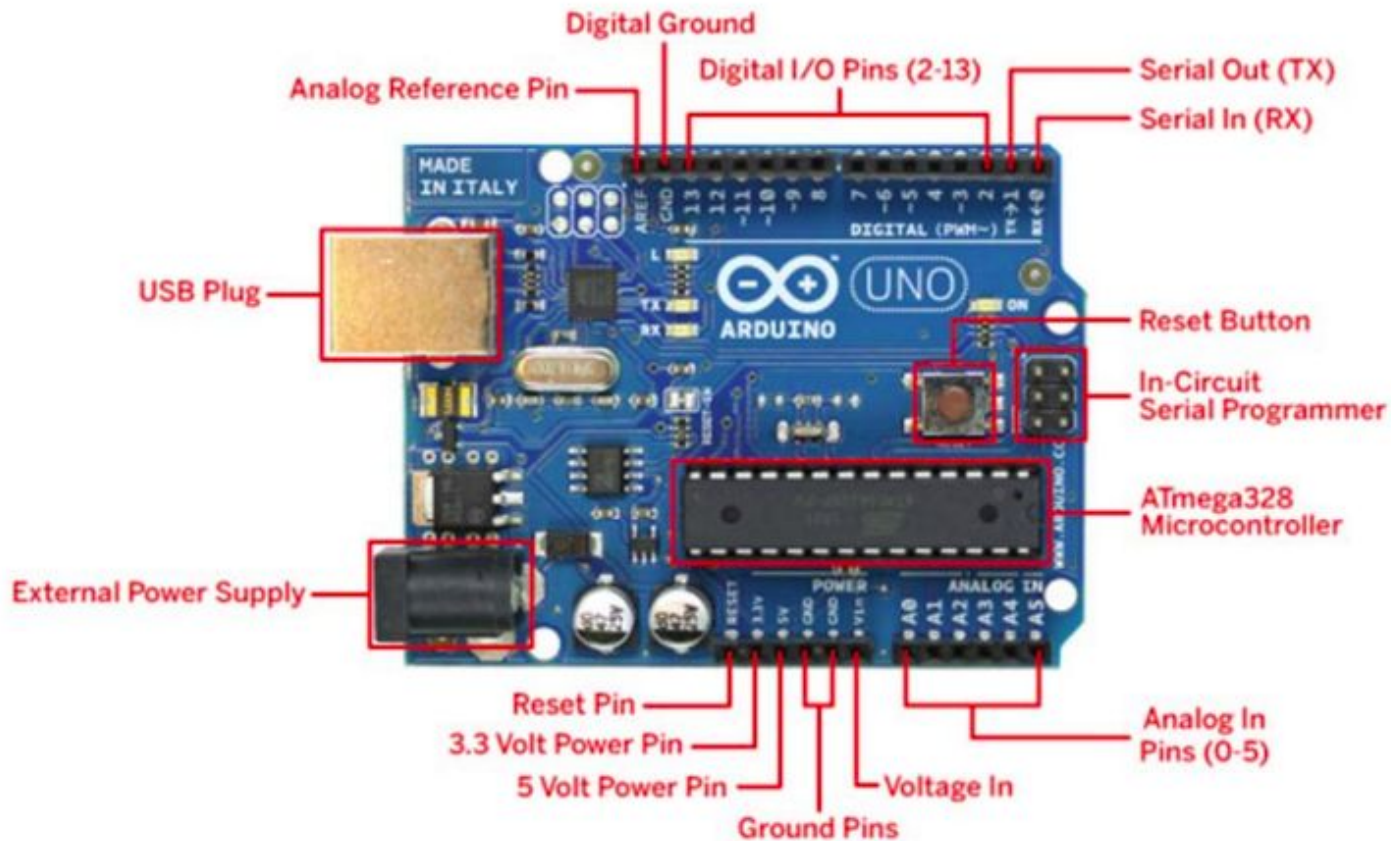
- Digital – only has two values: on/off
- Analog – has many (infinite) values





## 1.3 แนะนำ Arduino

# Arduino Board





## 1.4 อุปกรณ์ที่ใช้ในการเรียน

# อุปกรณ์



1. Arduino



2. Breadboard



3. สายไฟ (Jumper)



# อุปกรณ์



4. resistor, LED, และ switches แบบต่างๆ

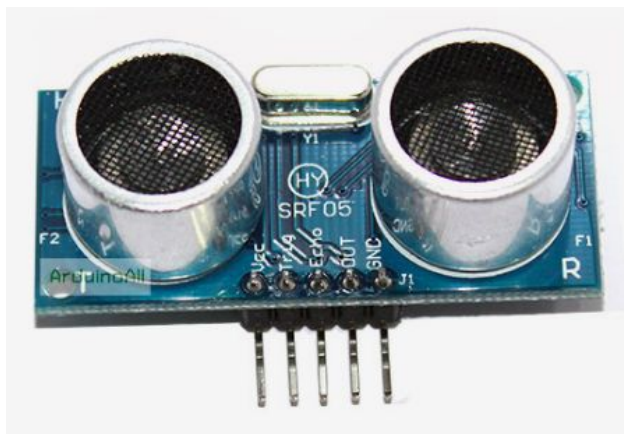


5. 7-Segment LED



6. Sensor แสง

## อุปกรณ์



6. Ultrasonic วัด  
ระยะทาง



7. Sensor อุณหภูมิ



8. Servo motor



## 1.5 ติดตั้ง Ardunio IDE

<https://www.arduino.cc/en/Main/Software>



The screenshot shows the Arduino IDE download page. The header is teal with the Arduino logo and navigation links: HOME, BUY, SOFTWARE (highlighted), PRODUCTS, LEARNING, COMMUNITY, and SUPPORT. On the right of the header are icons for search, a shopping cart, and a link to SIGN IN. The main content area has a light gray background. The heading "Download the Arduino IDE" is centered. Below it, on the left, is a large teal circle containing the Arduino logo. To the right of this circle, the text "ARDUINO 1.8.5" is displayed in bold. Below this, a paragraph describes the IDE as open-source software that runs on Windows, Mac OS X, and Linux. Another paragraph mentions that the software can be used with any Arduino board and refers to the "Getting Started" page for installation instructions. On the right side of the page, there is a teal sidebar with links to download the Windows Installer, a Windows ZIP file for non-admin install, the Windows app (with a "Get" button and Windows logo), Mac OS X 10.7 Lion or newer, Linux 32 bits, Linux 64 bits, and Linux ARM. At the bottom of the sidebar are links for Release Notes, Source Code, and Checksums (sha512).

HOME BUY SOFTWARE PRODUCTS LEARNING COMMUNITY SUPPORT


Download the Arduino IDE

**ARDUINO 1.8.5**

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

**Windows** Installer  
**Windows** ZIP file for non admin install

**Windows app** [Get](#) 

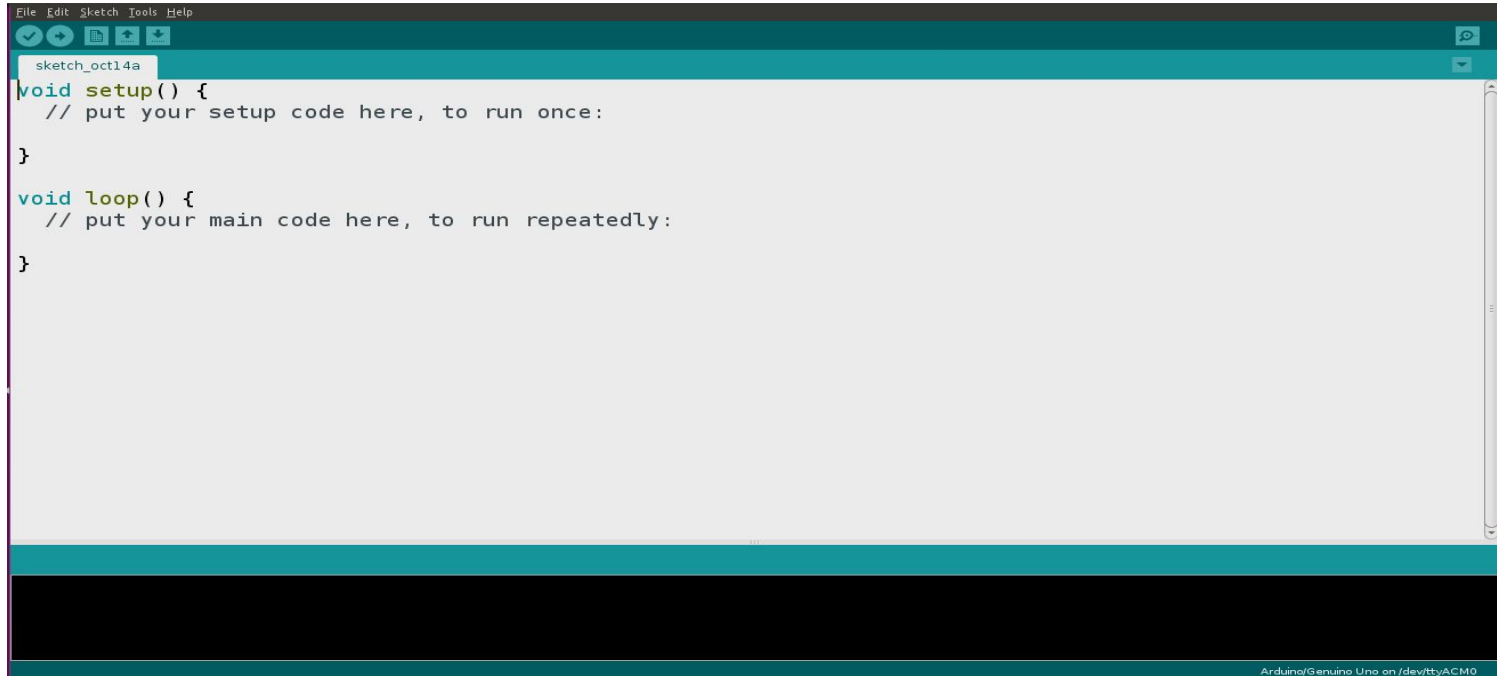
**Mac OS X** 10.7 Lion or newer

**Linux** 32 bits  
**Linux** 64 bits  
**Linux** ARM

[Release Notes](#)  
[Source Code](#)  
[Checksums \(sha512\)](#)



# Run Program



```
File Edit Sketch Tools Help
sketch_oct14a
void setup() {
  // put your setup code here, to run once:
}

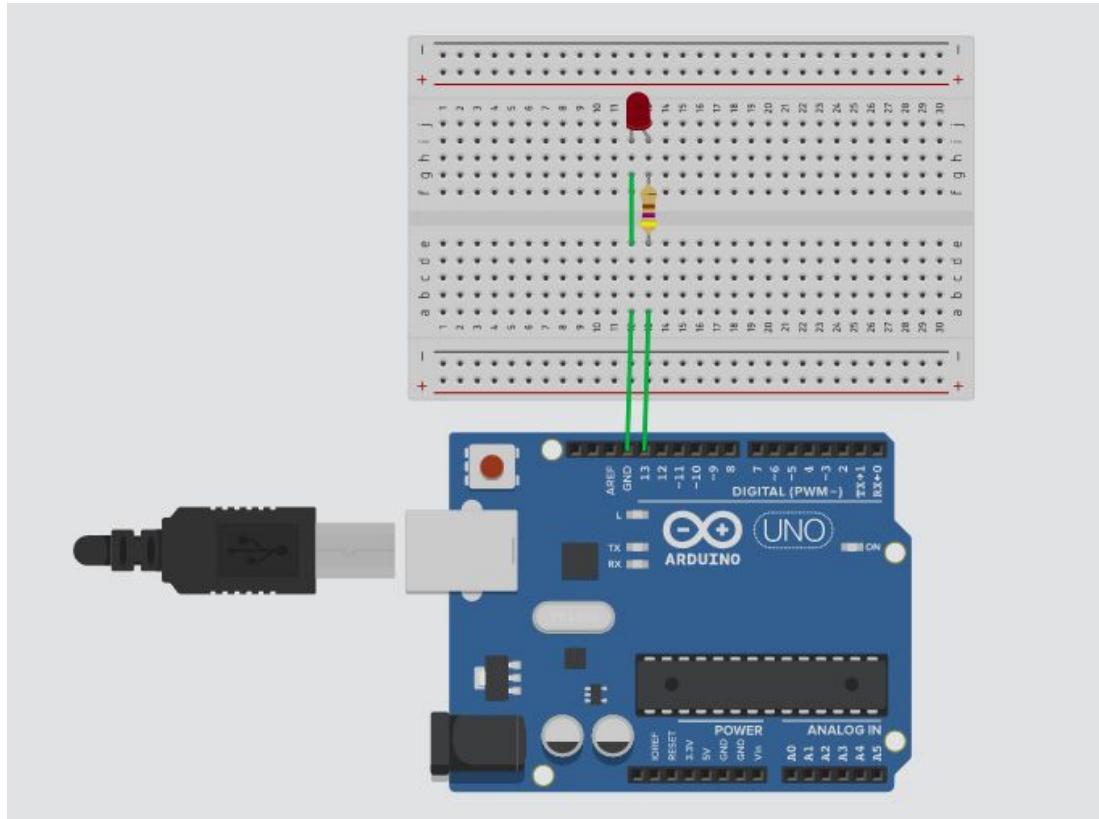
void loop() {
  // put your main code here, to run repeatedly:
}
```

Arduino/Genuino Uno on /dev/ttyACM0



## 1.6 Hello world

## วงจร Hello world: LED กระพริบ





## Code: Hello world: LED กระพริบ

```
void setup()
{
    pinMode(13, OUTPUT);
}

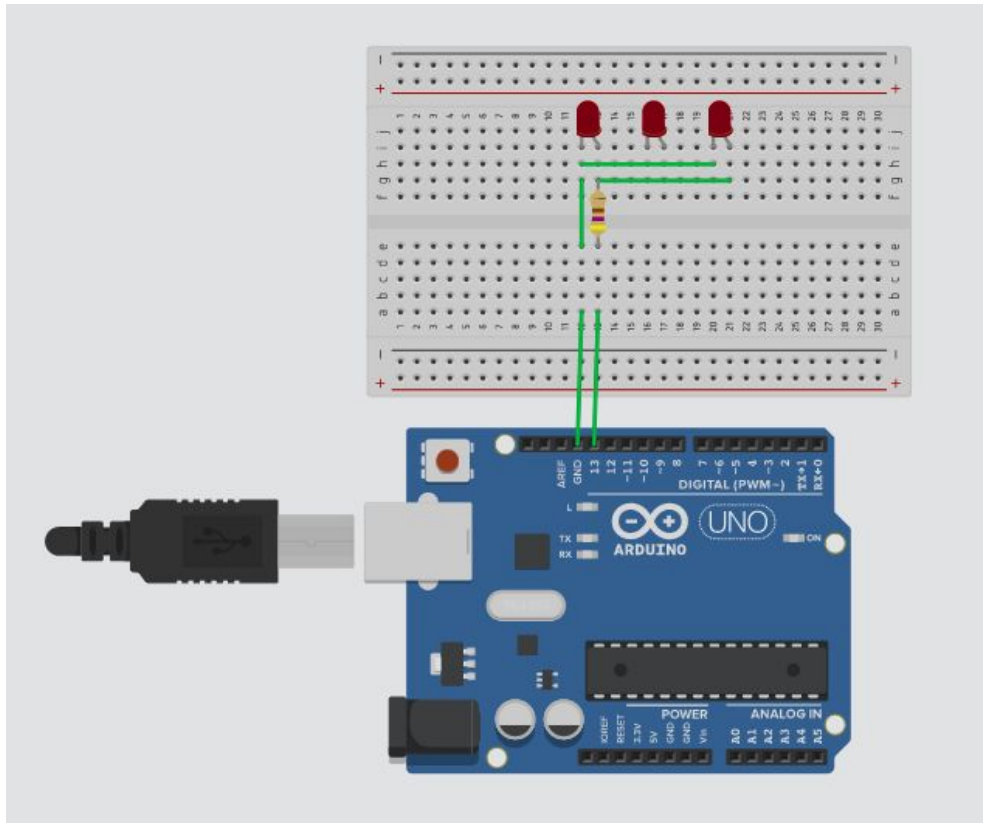
void loop()
{
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);
}
```



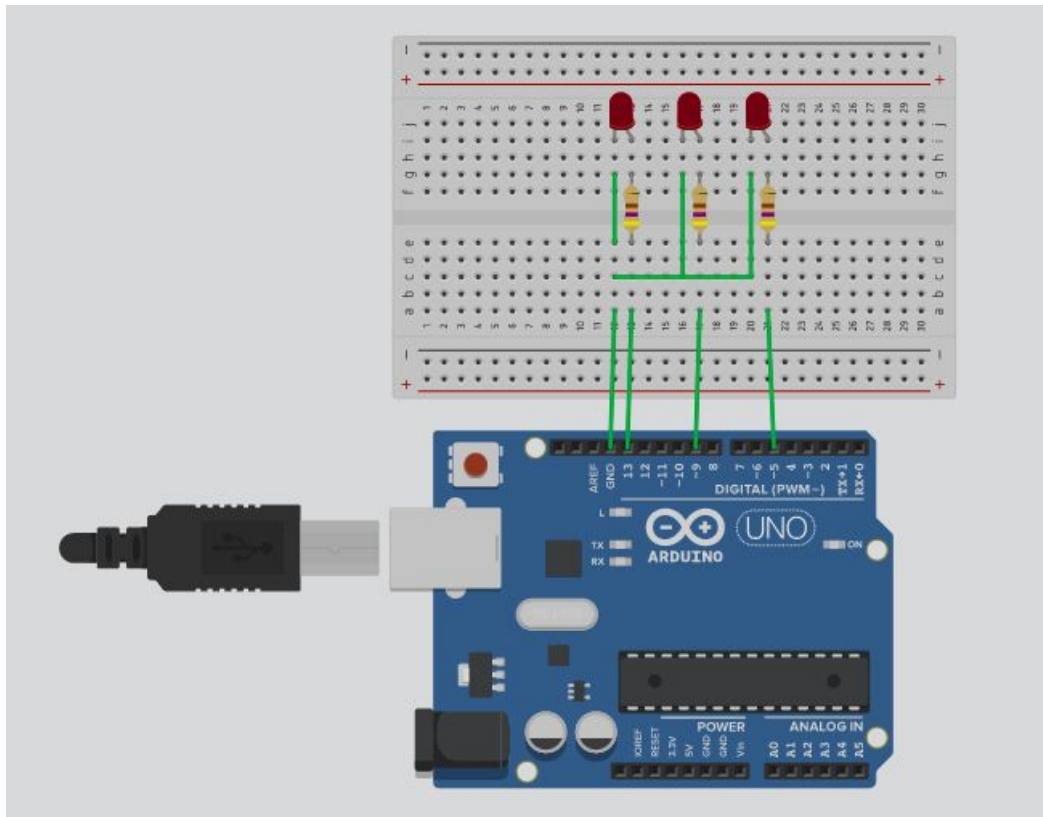


## 2.1 การทดลอง LED: ไฟกระพริบหลายดวง

## วงจร LED กระพริบหลายดวง



# วงจร LED ไฟวิ่ง





# Code: LED ไฟวิ่ง

```
void setup()
{
    pinMode(5, OUTPUT);
    pinMode(9, OUTPUT);
    pinMode(13, OUTPUT);
}
```

```
void loop()
{
    digitalWrite(13, HIGH);
    digitalWrite(5, LOW);
    delay(1000);

    digitalWrite(13, LOW);
    digitalWrite(9, HIGH);
    delay(1000);

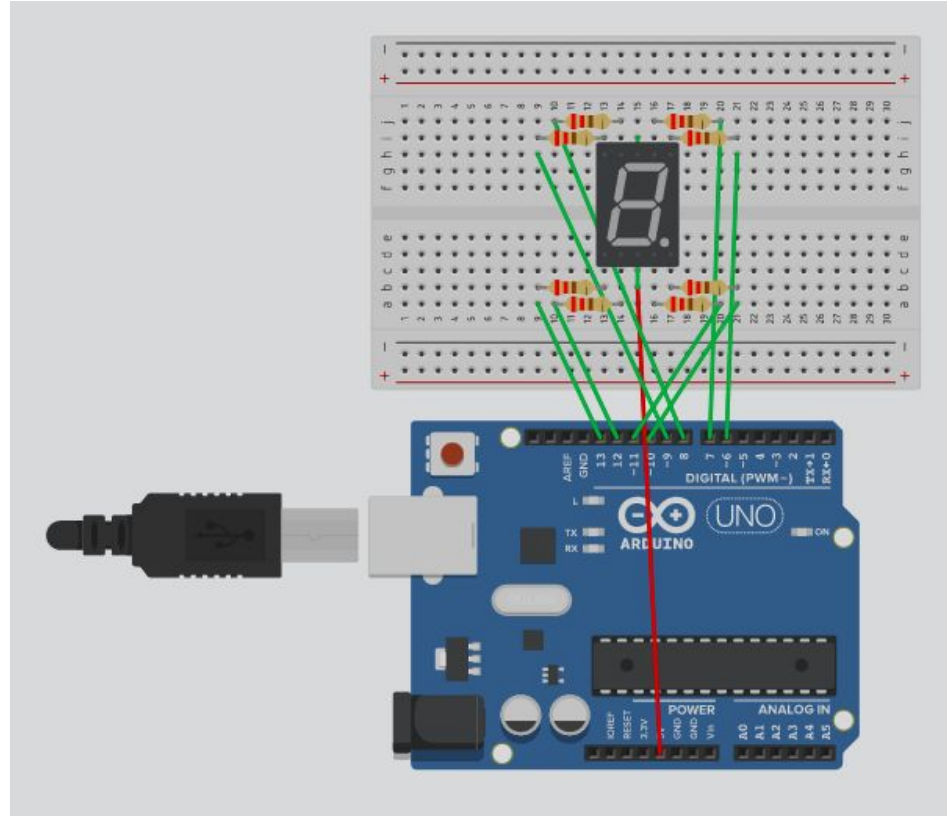
    digitalWrite(9, LOW);
    digitalWrite(5, HIGH);
    delay(1000);
}
```



## Exercise 1

1.1 ให้ทำวงจรไฟ LED ริ่งไปแล้วริ่งกลับ

# 7 Segment





## Code: 7 Segment

```
void setup()
{
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
    pinMode(9, OUTPUT);
    pinMode(10, OUTPUT);
    pinMode(11, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(13, OUTPUT);
}
```



## Code: 7 Segment (Cont.)

```
void loop()
{
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(10, HIGH);
    digitalWrite(11, HIGH);
    digitalWrite(12, HIGH);
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(6, LOW);
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    digitalWrite(10, LOW);
    digitalWrite(11, LOW);
    digitalWrite(12, LOW);
    digitalWrite(13, LOW);
    delay(1000);
}
```



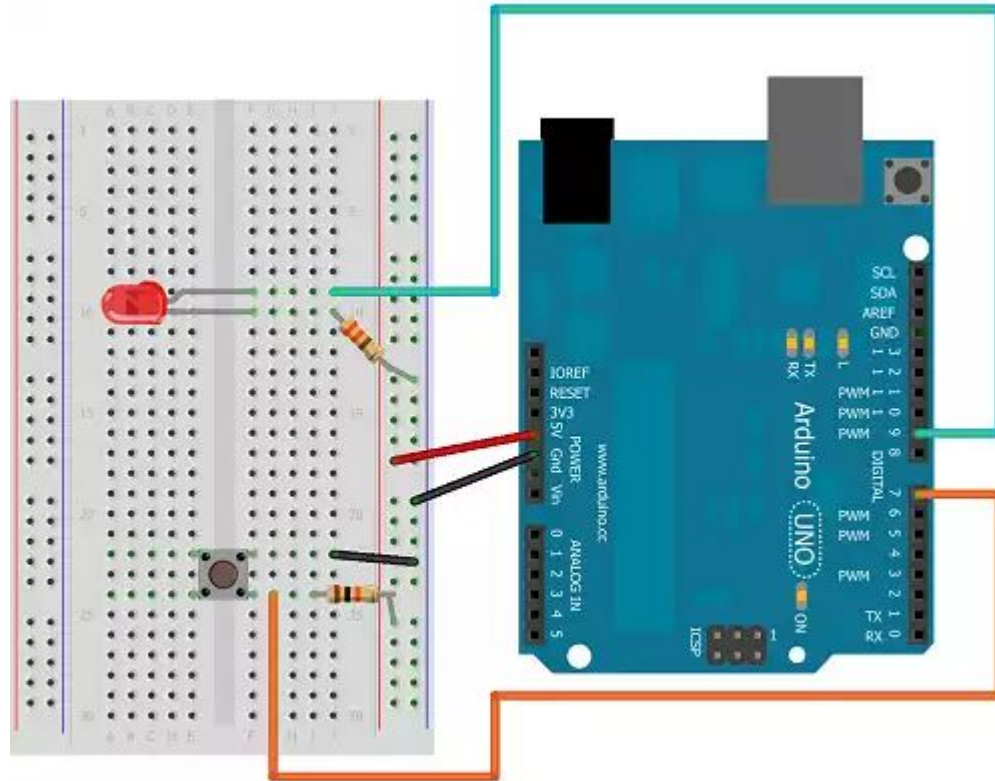


## Exercise 2

2.1 ให้ทำวงจรนับเลข 0-9 ที่ 7 segment

2.2 ให้ทำวงจรนับเลข 2 หลัก โดยที่หลักหน่วยแสดงผลที่ 7 segment หลักสิบแสดงผลเป็นจำนวนไฟ LED

# LED + Button





## Code: LED + Button

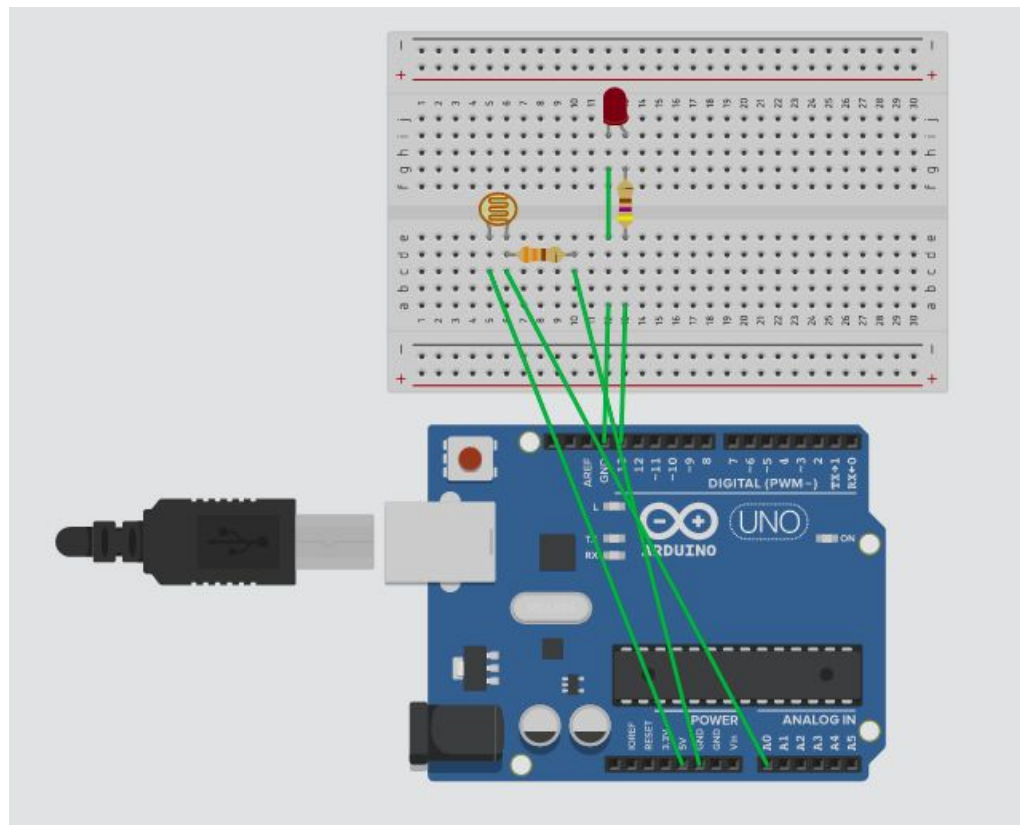
```
void setup()
{
    pinMode(9, OUTPUT);
    pinMode(7, INPUT);
}
void loop()
{
    // read from the button pin
    int button = digitalRead(7);
    if (button == HIGH)
    {
        digitalWrite(9, HIGH);
    }
    else
    {
        digitalWrite(9, LOW);
    }
}
```



## Exercise 3

3.1 ทำ Counter นับเลข 2 หลัก โดยที่หลักหน่วยแสดง  
ผลที่ 7 segment หลักลิบแสดงผลเป็นจำนวนไฟ LED  
ซึ่งจะนับเลขเมื่อมีการกด button

# Sensor แสง



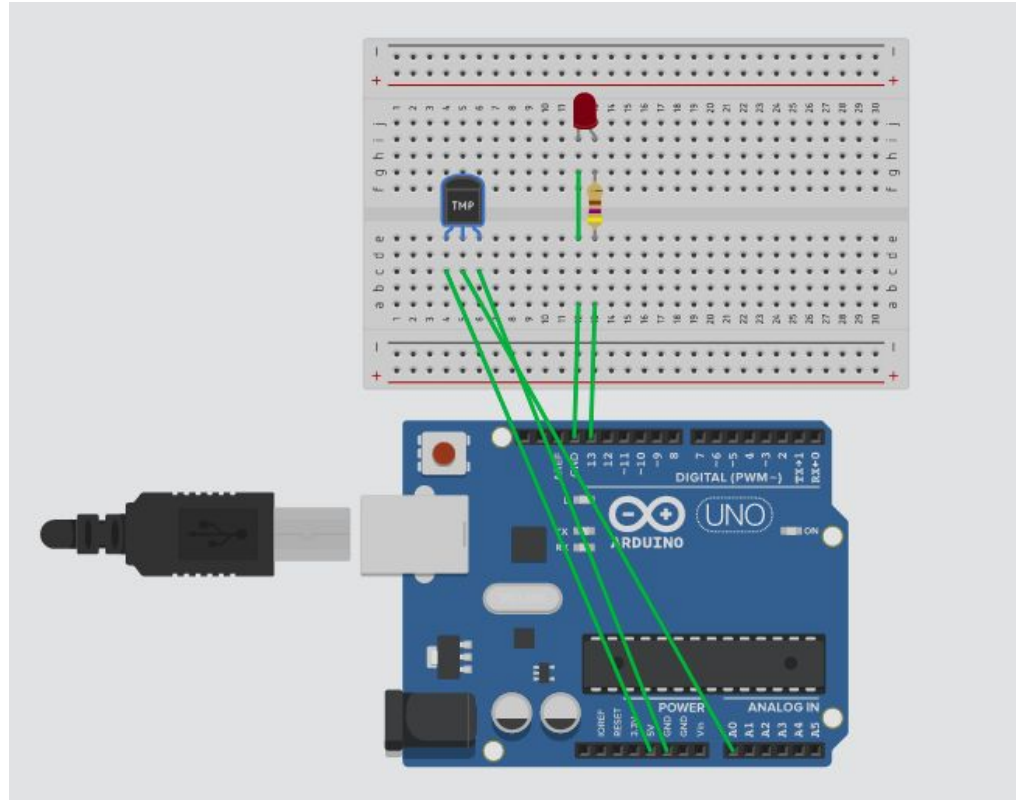


## Code: Sensor แสง

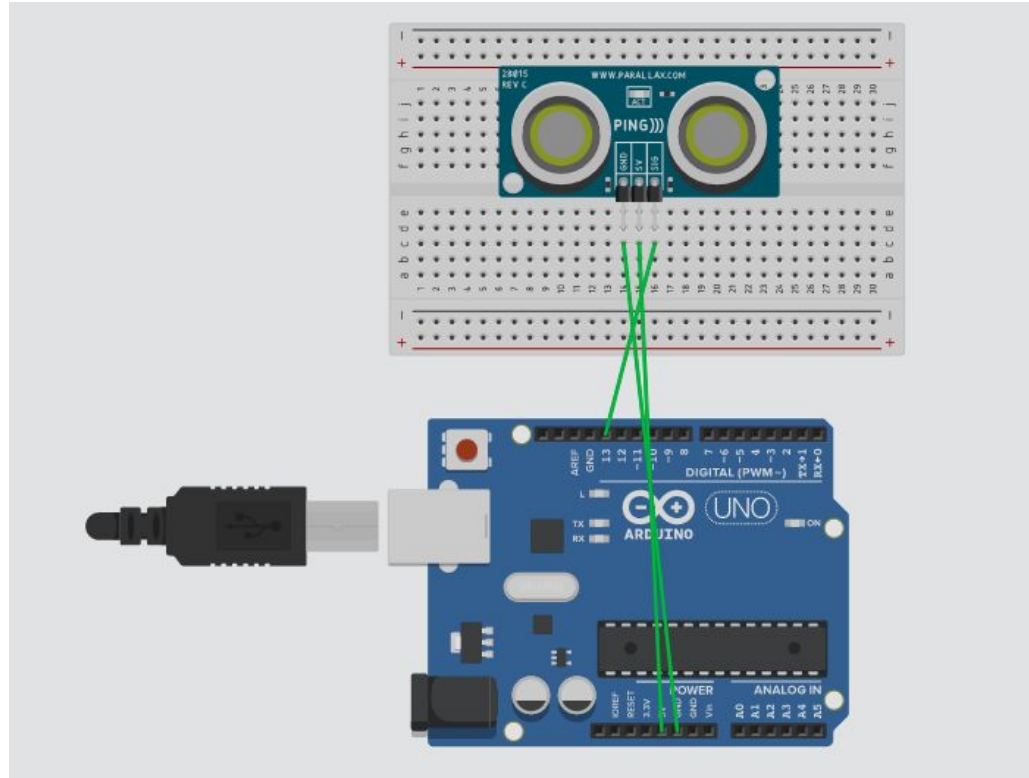
```
void setup()
{
    pinMode(13, OUTPUT);
    Serial.begin(9600);
}

void loop()
{
    int val = analogRead(A0);
    Serial.println(val);
    if(val >= 200){
        digitalWrite(13,HIGH);
    }
    else {
        digitalWrite(13,LOW);
    }
}
```

# Sensor อุณหภูมิ



# Sensor วัดระยะทาง



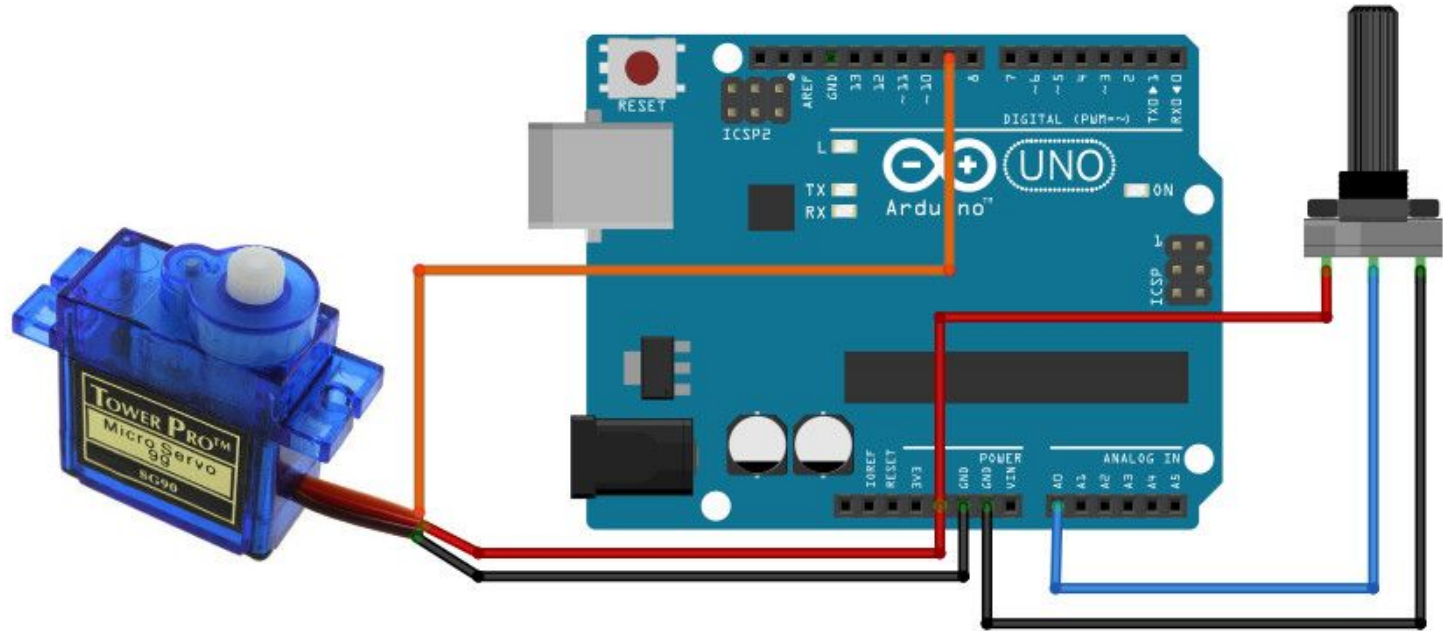




## Code: Sensor วัดระยะทาง

```
#define ping 13
void setup () {
    Serial.begin(9600);
}
void loop() {
    int duration;
    int cm;
    pinMode(ping,OUTPUT);
    digitalWrite(ping,HIGH);
    digitalWrite(ping,LOW);
    pinMode(ping,INPUT);
    duration=pulseIn(ping,HIGH);
    cm=(duration/2)/29;
    Serial.print(cm);
    Serial.println("cm");
    delay(1000);
}
```

# Motor





## Code: Motor

```
#include <Servo.h>
```

```
Servo myservo; // สร้างออปเจกชื่อ myservo จากคลาส Servo เพื่อควบคุม servo micro
```

```
int potpin = 0; // ขาอะนาล็อกใช้ต่อกับตัวต้านทานปรับค่าได้  
int val;       // ใช้เก็บค่าที่อ่านได้จากตัวต้านทานปรับค่าได้
```

```
void setup()  
{  
  myservo.attach(9); // บอกว่าจะต่อ servo ที่ขา 9  
}
```

```
void loop()  
{  
  val = analogRead(potpin); // อ่านค่าอะนาล็อกจากตัวต้านทานปรับค่าได้  
  val = map(val, 0, 1023, 0, 179); // แปลงช่วง 0-1023 ที่อ่านได้ ให้อยู่ในช่วง 0-179  
  myservo.write(val); // ส่งค่าไปควบคุม servo  
  delay(15); // หน่วงเวลาให้ servo ได้ทำงาน  
}
```



## แบ่งกลุ่มระดมสมองสร้างโปรเจค