

創意編程

設計大賽

2020/21

Creative Coder Competition

Arduino  
進階工作坊  
Advanced Training Workshop

合辦機構  
Co-organizer



香港青年協會  
the hongkong federation of youth groups



Department of Electrical and  
Electronic Engineering  
電機電子工程系

贊助  
Sponsor



創新科技署  
Innovation and  
Technology Commission

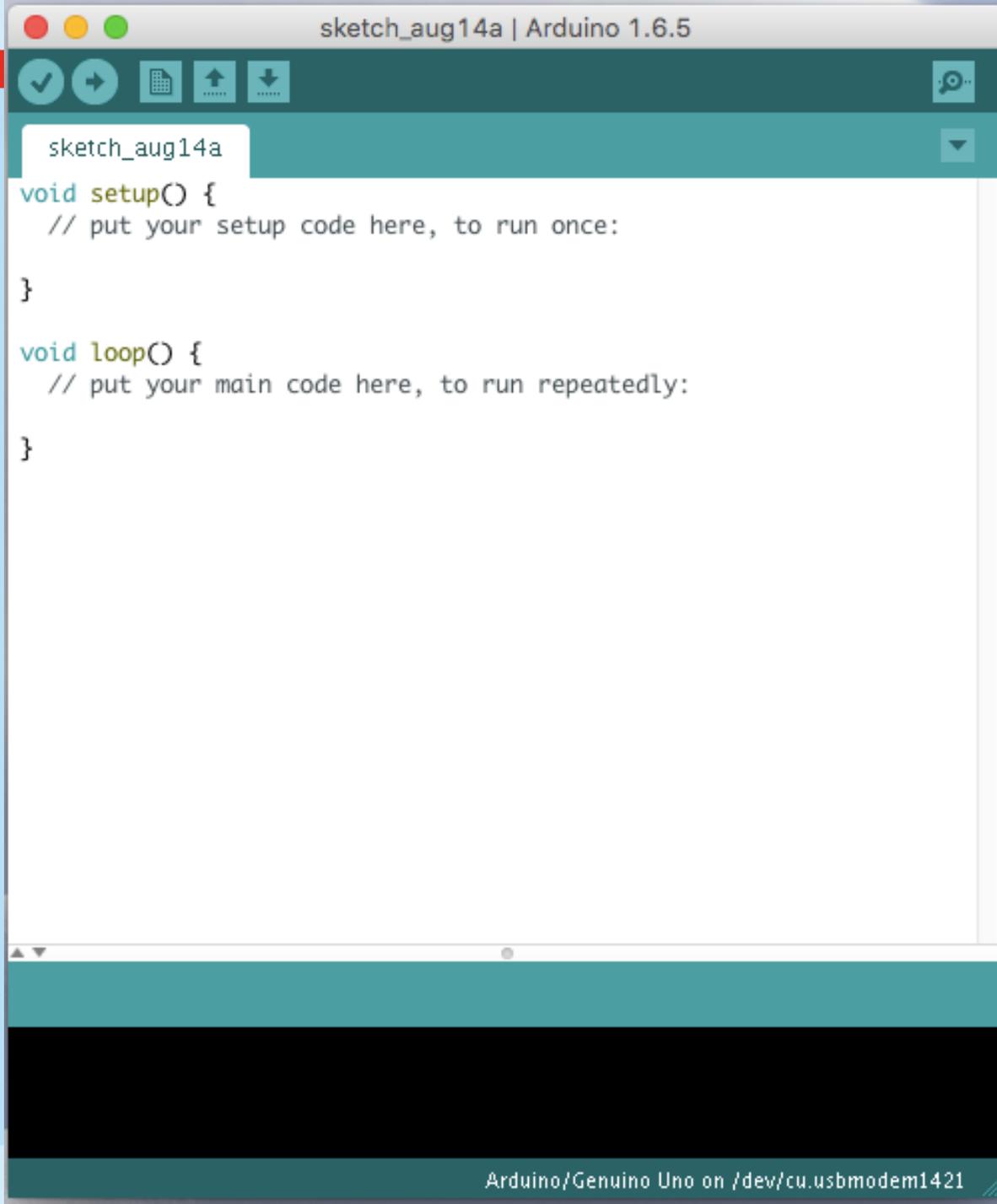
# 使用mBlock5步驟

1. 電腦連接Arduino UNO
  2. 開啟mBlock5程式
  3. [設備] → [添加] → 選擇[Arduino UNO](首次使用需要連網獲取)
  4. [連接] → 選擇Arduino UNO的COM → [連接]
- 模式：
    - 上傳：所有程式需要上傳才會作用
    - 在線：雙擊程式可以即時作用

# 程式編寫介面

arduino IDE (Integrated Development Environment)



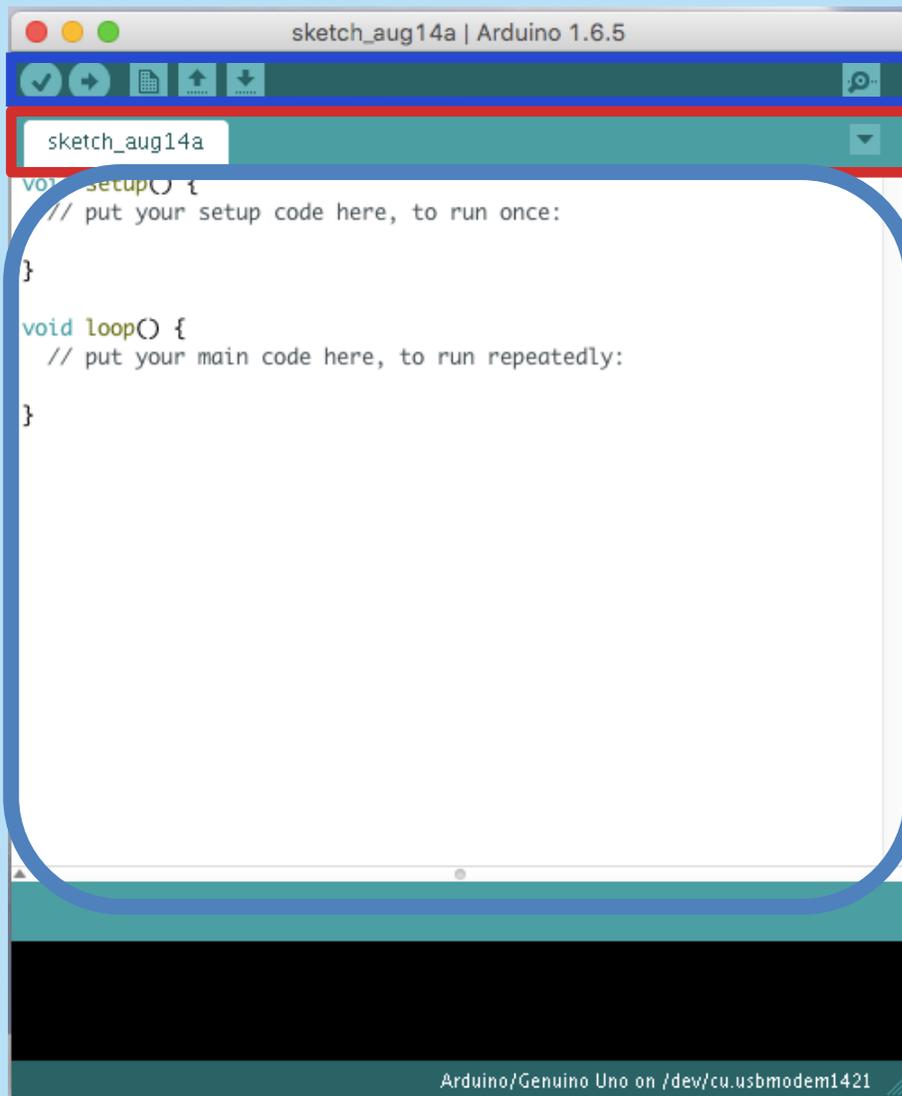


The image shows a screenshot of the Arduino IDE interface. The title bar at the top reads "sketch\_aug14a | Arduino 1.6.5". Below the title bar is a toolbar with icons for checkmark, run, copy, paste, and save. The main editor area contains the following code:

```
sketch_aug14a
void setup() {
  // put your setup code here, to run once:
}

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

At the bottom of the IDE, there is a status bar that reads "Arduino/Genuino Uno on /dev/cu.usbmodem1421".



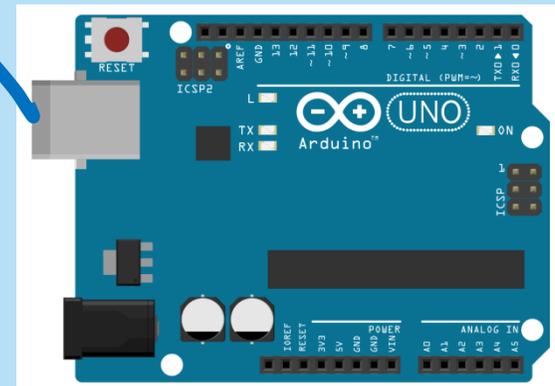
工具列  
程式碼分頁

程式內容

# 使用Arduino IDE步驟

1. 連接Arduino UNO
2. 開啟Arduino IDE
3. "Tools" -> "Board"  
選擇Arduino型號 (Arduino UNO)
  1. "Tools" -> "Port" ->  
選擇最大COM Port (Windows) /  
最大modem (MacOS)

**如果連接出現任何問題，  
請重覆以上步驟**



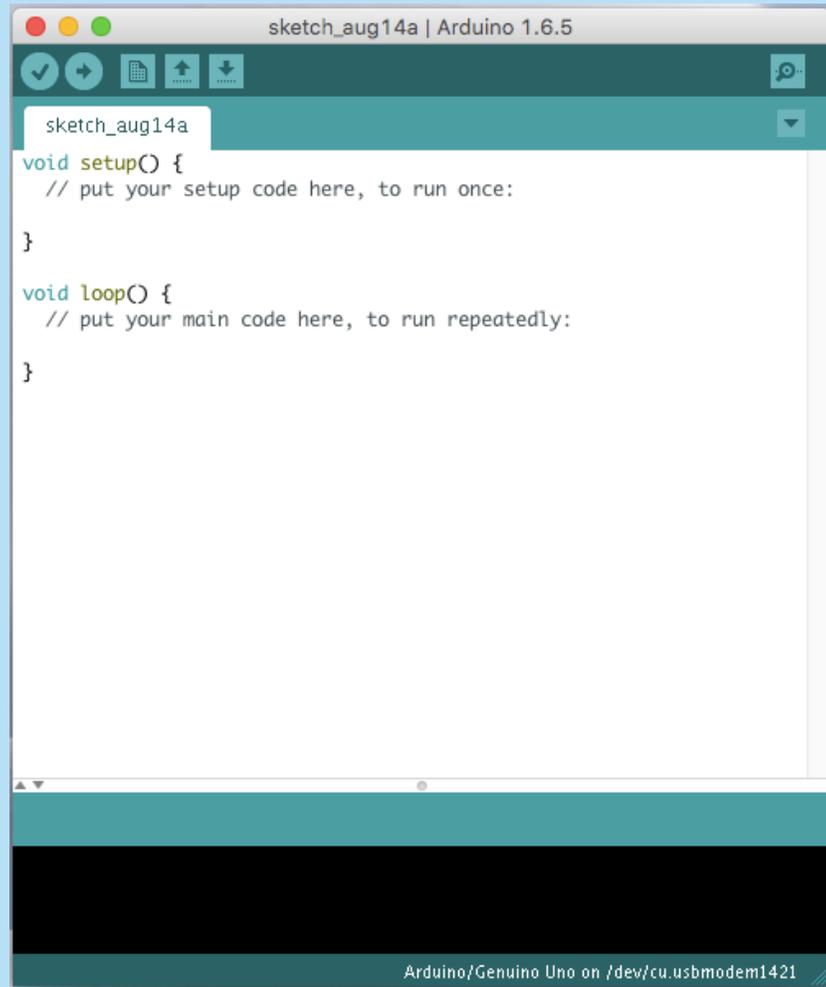
```
sketch_aug14a | Arduino 1.6.5
sketch_aug14a
void setup() {
  // put your setup code here, to run once:
}
void loop() {
  // put your main code here, to run repeatedly:
}
```



初始設定區  
(只執行一次)

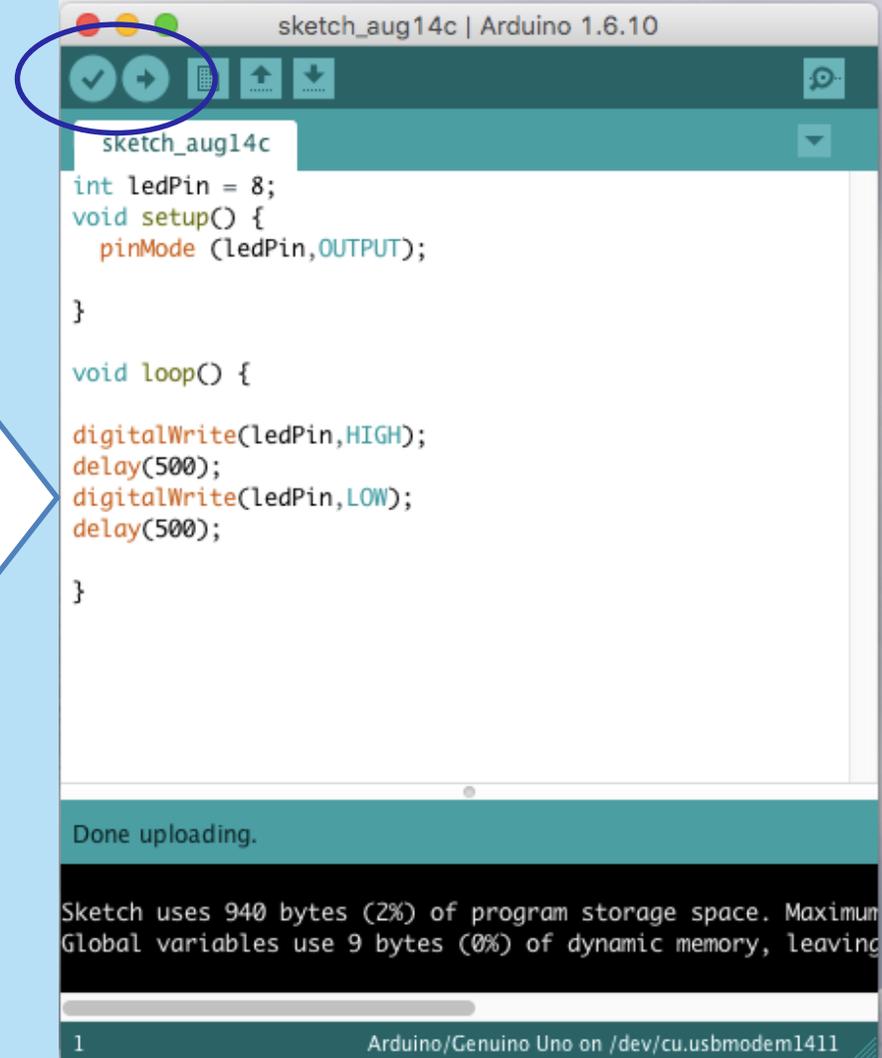


重複執行區  
(不斷重複)



```
sketch_aug14a | Arduino 1.6.5  
sketch_aug14a  
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/cu.usbmodem1421



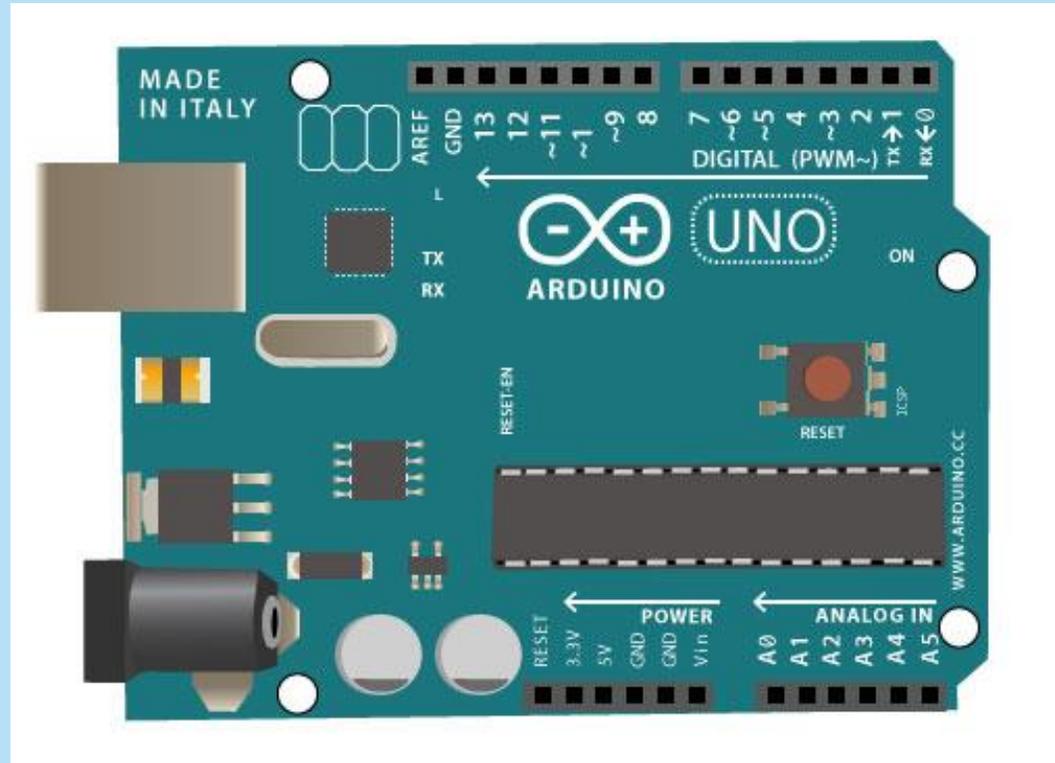
```
sketch_aug14c | Arduino 1.6.10  
sketch_aug14c  
int ledPin = 8;  
void setup() {  
  pinMode (ledPin,OUTPUT);  
}  
  
void loop() {  
  
  digitalWrite(ledPin,HIGH);  
  delay(500);  
  digitalWrite(ledPin,LOW);  
  delay(500);  
}
```

Done uploading.

Sketch uses 940 bytes (2%) of program storage space. Maximum is 32256 bytes.  
Global variables use 9 bytes (0%) of dynamic memory, leaving 2047 bytes free. Maximum is 2048 bytes.

1 Arduino/Genuino Uno on /dev/cu.usbmodem1411

# 活動1：閃爍發光二極管 電子元件接駁



# 活動1：閃爍發光二極管

```
int ledPin = 8; // 設定腳位8為LED的腳位

void setup() {
  pinMode (ledPin,OUTPUT); // 設定LED腳位為輸出
}

void loop() {
  digitalWrite(ledPin,HIGH); // LED腳位高電壓(亮燈)
  delay(1000); // 延遲1000毫秒(1秒)
  digitalWrite(ledPin,LOW); // LED腳位低電壓(關燈)
  delay(1000); // 延遲1000毫秒(1秒)
}
```

小心！  
(1)大小階(2)分號(3)形式  
均要正確 (Case Sensitive)

# 活動1.2：閃爍發光二極管

如要加快閃爍速度...

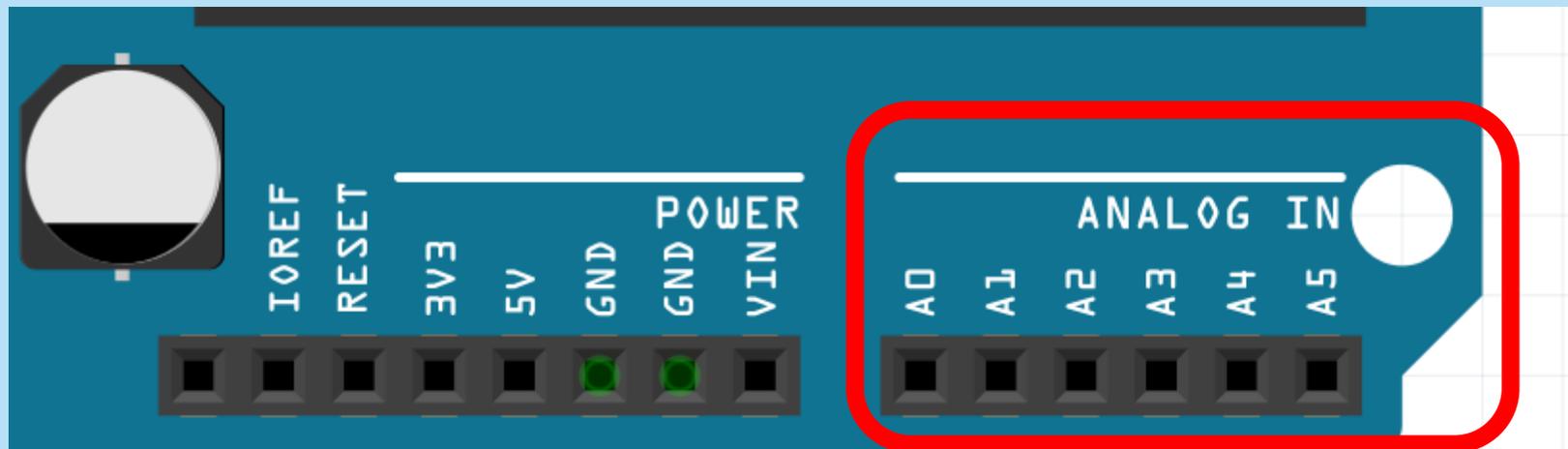
```
int ledPin = 8;

void setup() {
  pinMode (ledPin,OUTPUT);
}

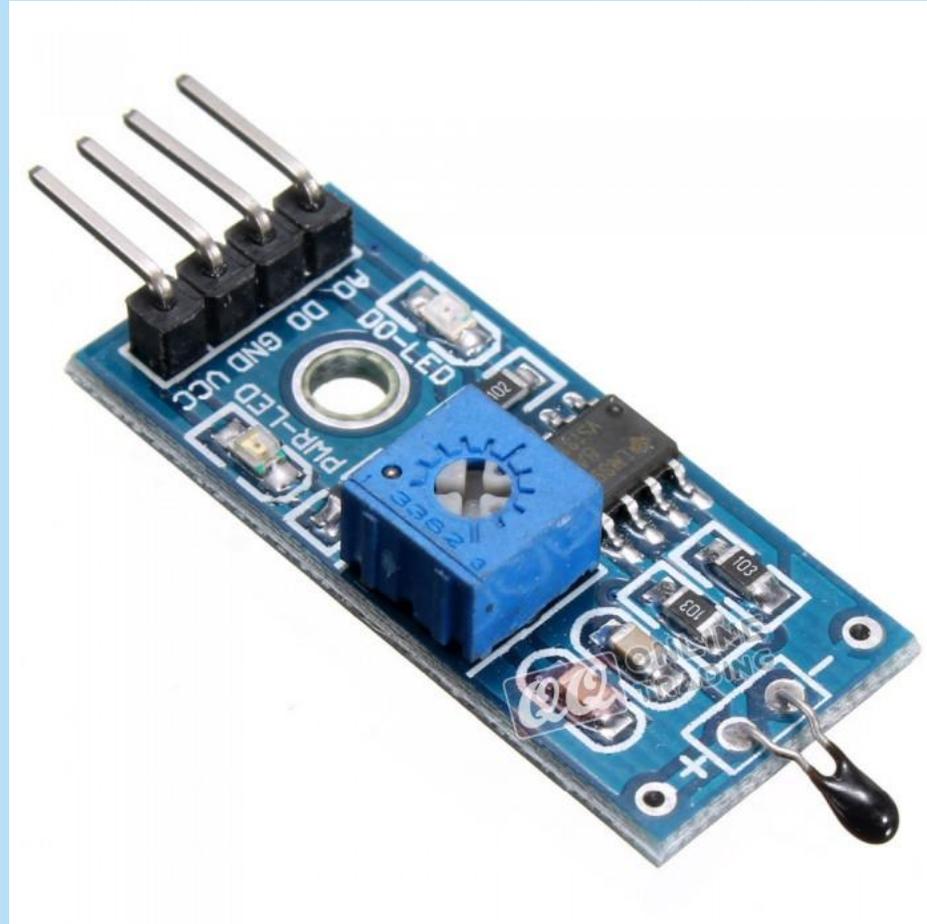
void loop() {
  digitalWrite(ledPin,HIGH);
  delay(1000);
  digitalWrite(ledPin,LOW);
  delay(1000);
}
```

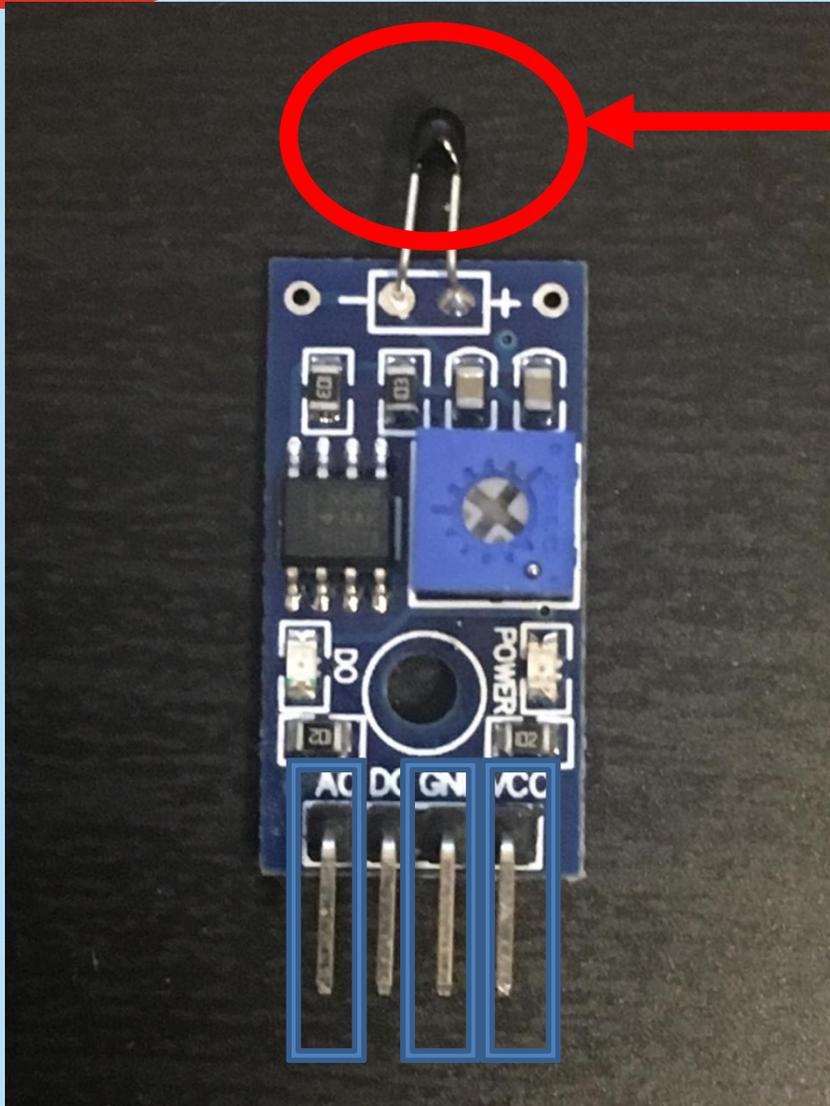
# 重溫：類比輸入 Analog-in

1. Arduino部分接腳位設定為類比數碼轉換器 (Analog to Digital Converter, ADC)
2. 數值由0至1023 (總數：1024或 $2^{10}$ )
3. 例子：可變電阻 (繼電器)、光敏電阻 (LDR)



# 熱敏電阻模組 Thermistor Module





## 熱敏電阻

VCC：連接5V電源接孔

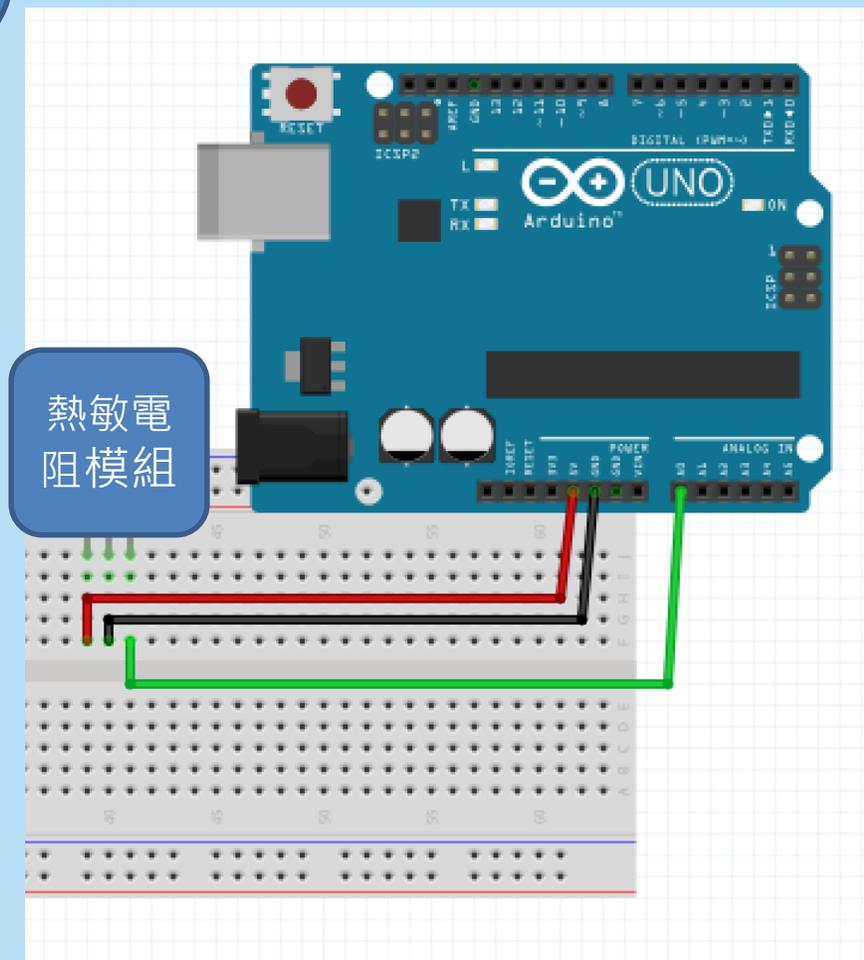
GND：連接GND接地接孔

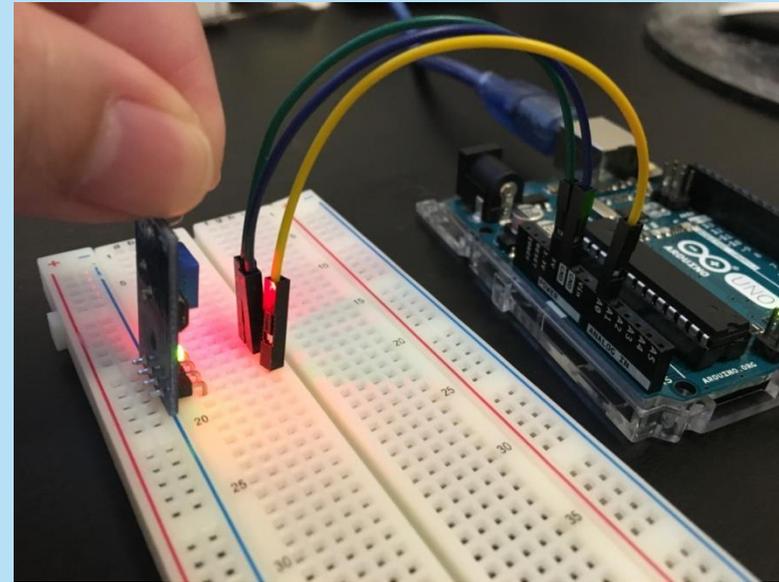
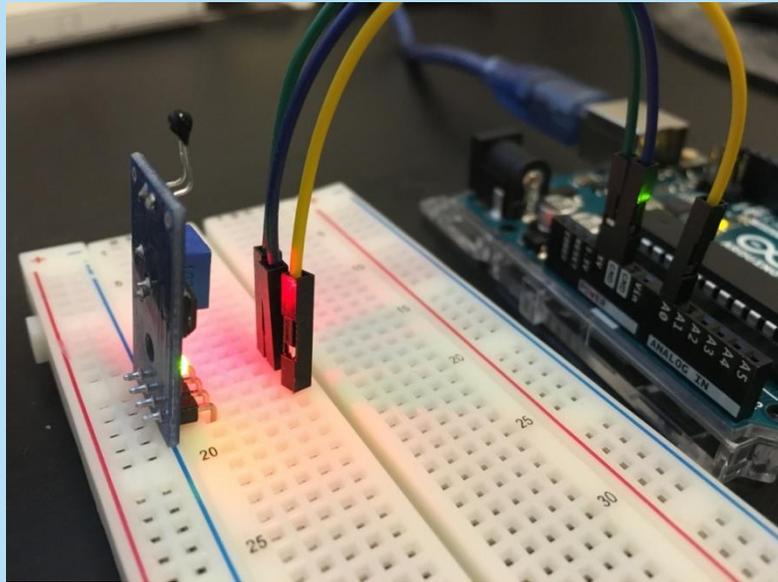
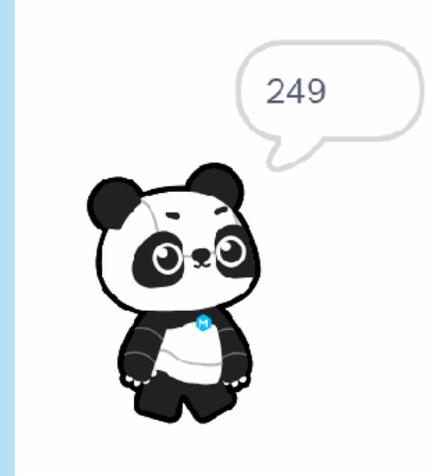
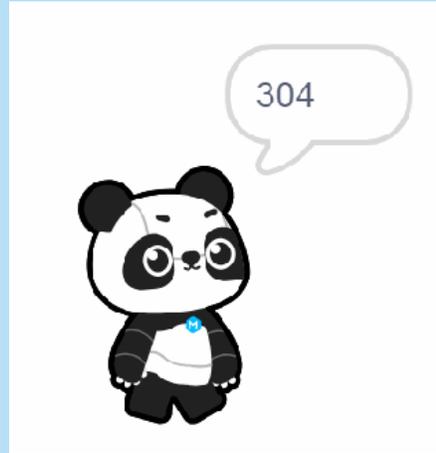
A0：連接類比輸入接孔

# 測試溫度

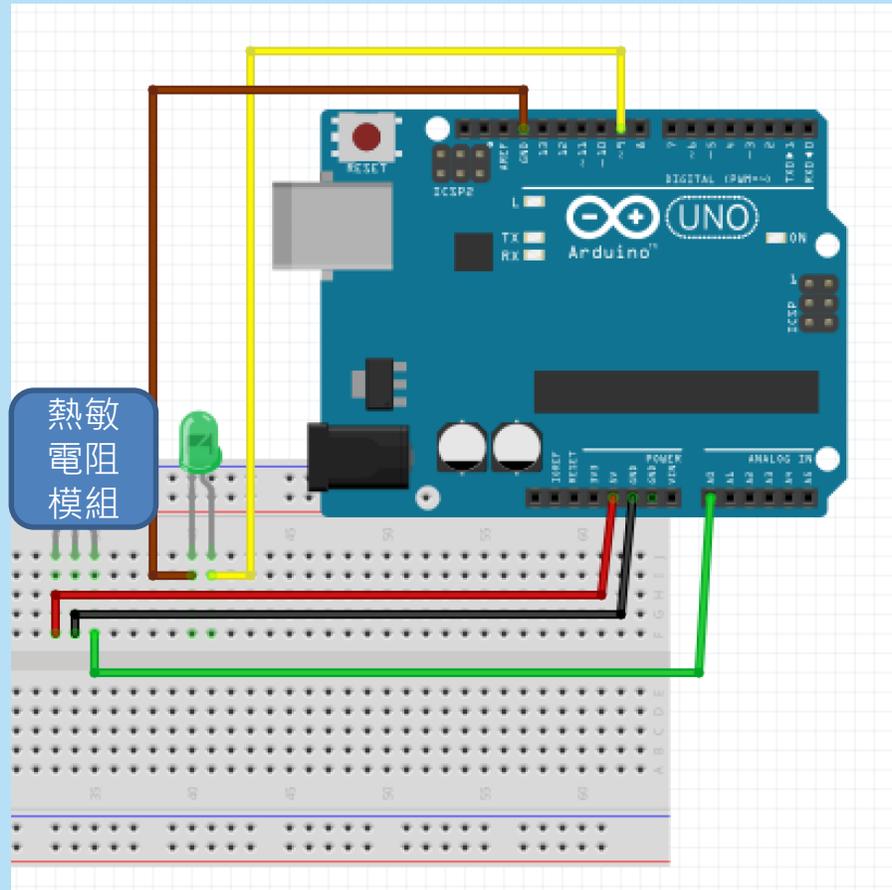


A0 – 接孔A0  
VCC – 接孔5V  
GND – 接孔GND





# 活動：溫度計



當 Arduino Uno 啟動時

不停重複

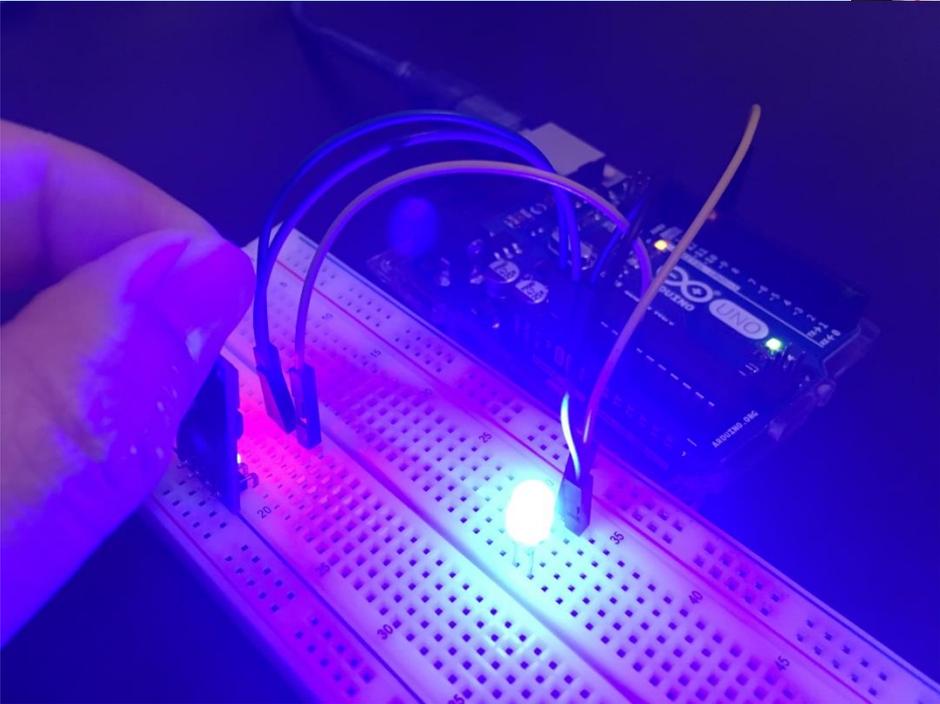
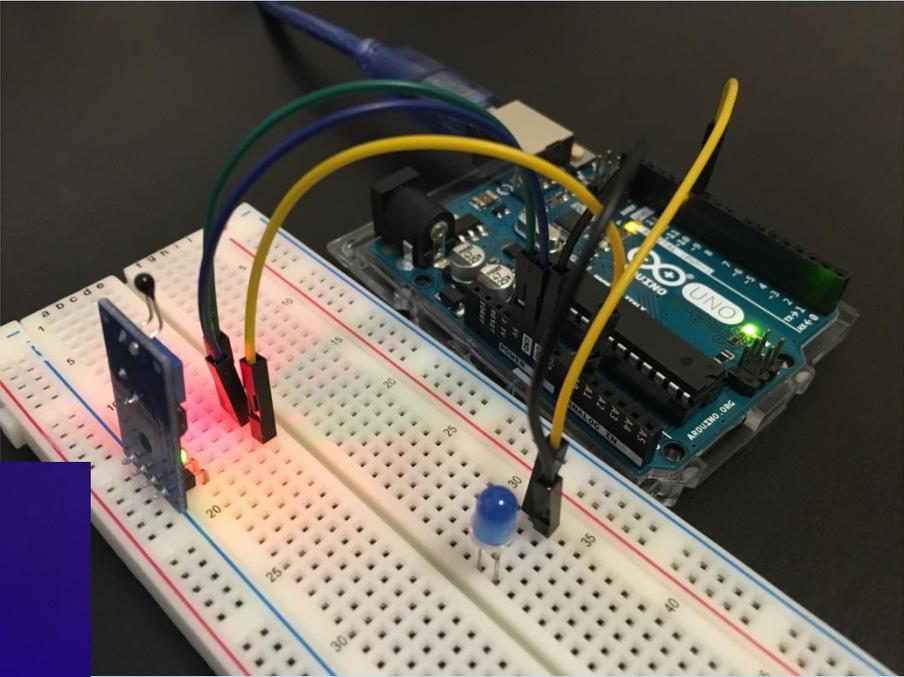
如果  $\infty$  類比埠 (A) 0 小於 290 那麼

$\infty$  設置數位 pin 11 輸出為 高電位 ▾

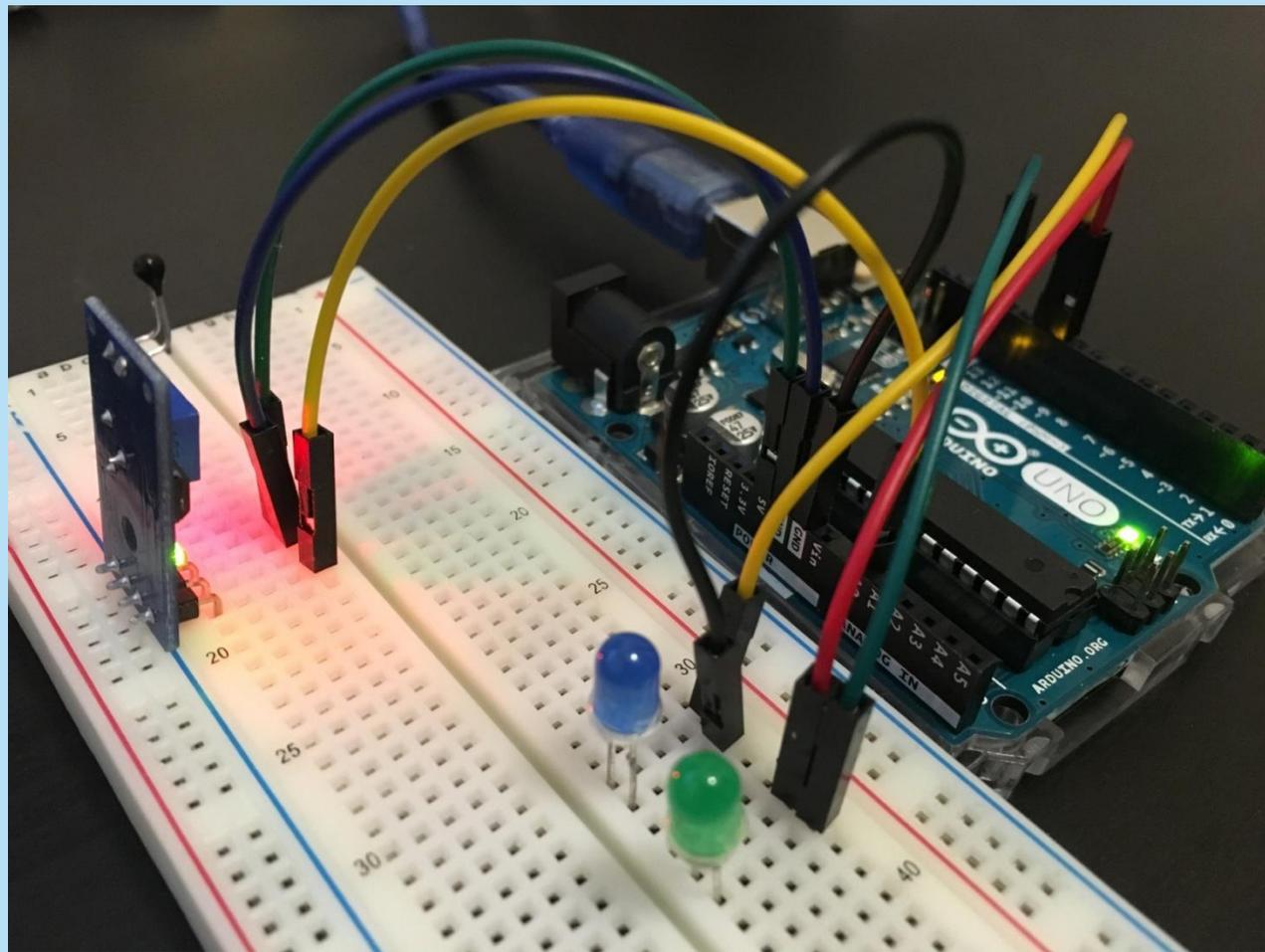
否則

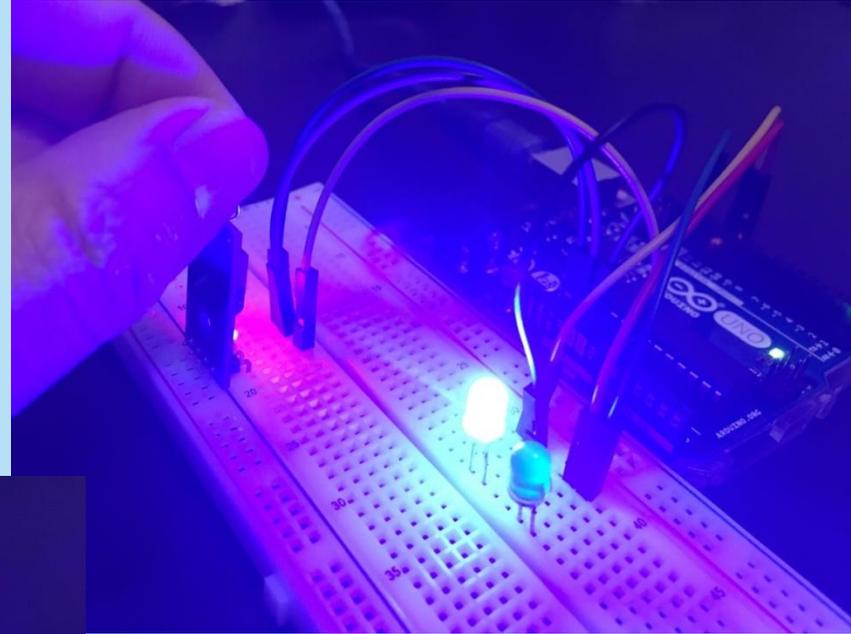
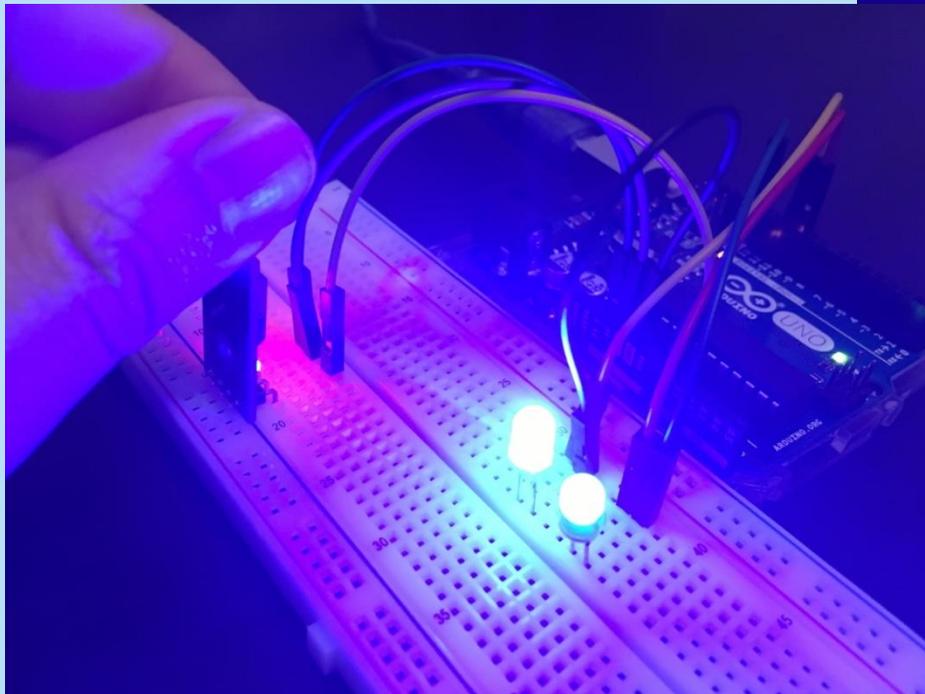
$\infty$  設置數位 pin 11 輸出為 低電位 ▾





如果連接2顆LED...





當 Arduino Uno 啟動時

不停重複

如果 類比埠 (A) 0 小於 250 那麼

設置數位 pin 11 輸出為 高電位

設置數位 pin 10 輸出為 高電位

否則

如果 類比埠 (A) 0 小於 290 那麼

設置數位 pin 11 輸出為 高電位

設置數位 pin 10 輸出為 低電位

否則

設置數位 pin 11 輸出為 低電位

設置數位 pin 10 輸出為 低電位



# 愈熱，愈亮！

```
當 Arduino Uno 啟動時
  不停重複
    如果 類比埠 (A) 0 小於 250 那麼
      設置PWM 11 輸出為 255
    否則
      如果 類比埠 (A) 0 小於 270 那麼
        設置PWM 11 輸出為 150
      否則
        如果 類比埠 (A) 0 小於 285 那麼
          設置PWM 11 輸出為 100
        否則
          如果 類比埠 (A) 0 小於 295 那麼
            設置PWM 11 輸出為 50
          否則
            設置PWM 11 輸出為 0
```

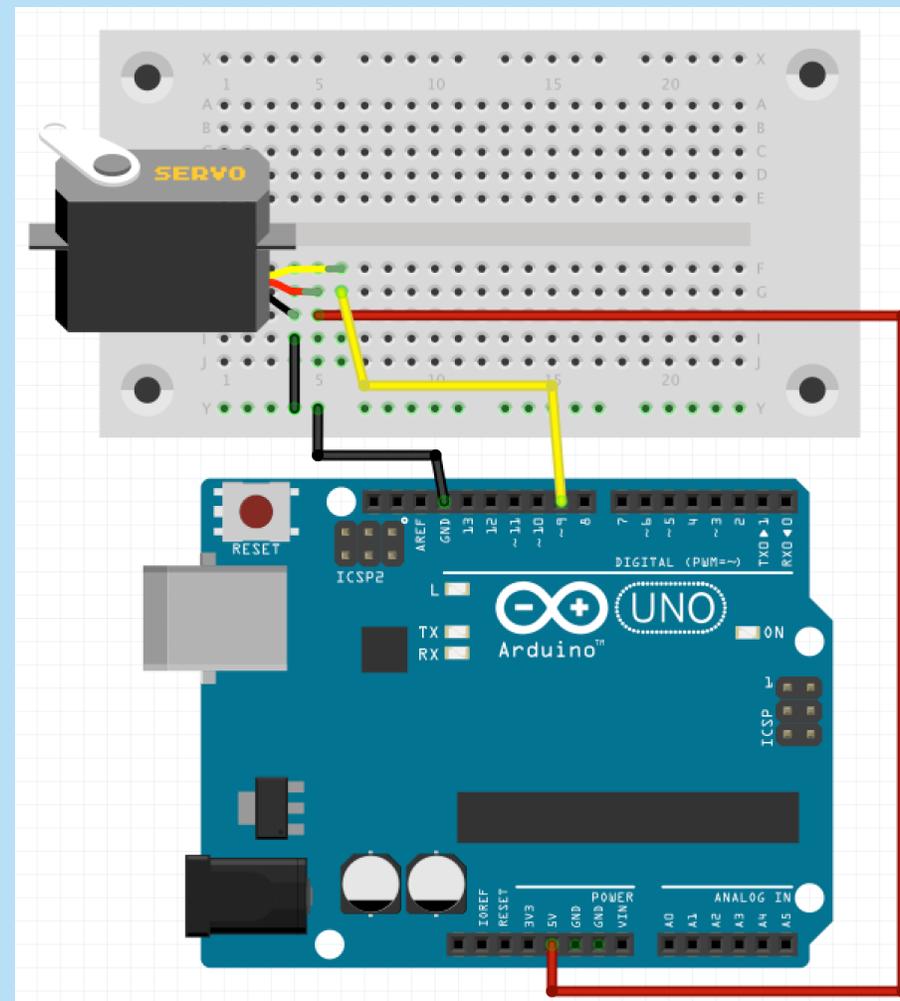
# 伺服摩打(舵機)

## Servo Motor

橙色線：訊號線

紅色線：電源

啡色線：接地



# 伺服摩打(舵機) Servo Motor

當 Arduino Uno 啟動時

不停重複

∞ 設置 9 腳位伺服馬達角度為 0

等待 1 秒

∞ 設置 9 腳位伺服馬達角度為 45

等待 1 秒

∞ 設置 9 腳位伺服馬達角度為 90

等待 1 秒

∞ 設置 9 腳位伺服馬達角度為 135

等待 1 秒

∞ 設置 9 腳位伺服馬達角度為 180

等待 1 秒

∞ 設置 9 腳位伺服馬達角度為 135

等待 1 秒

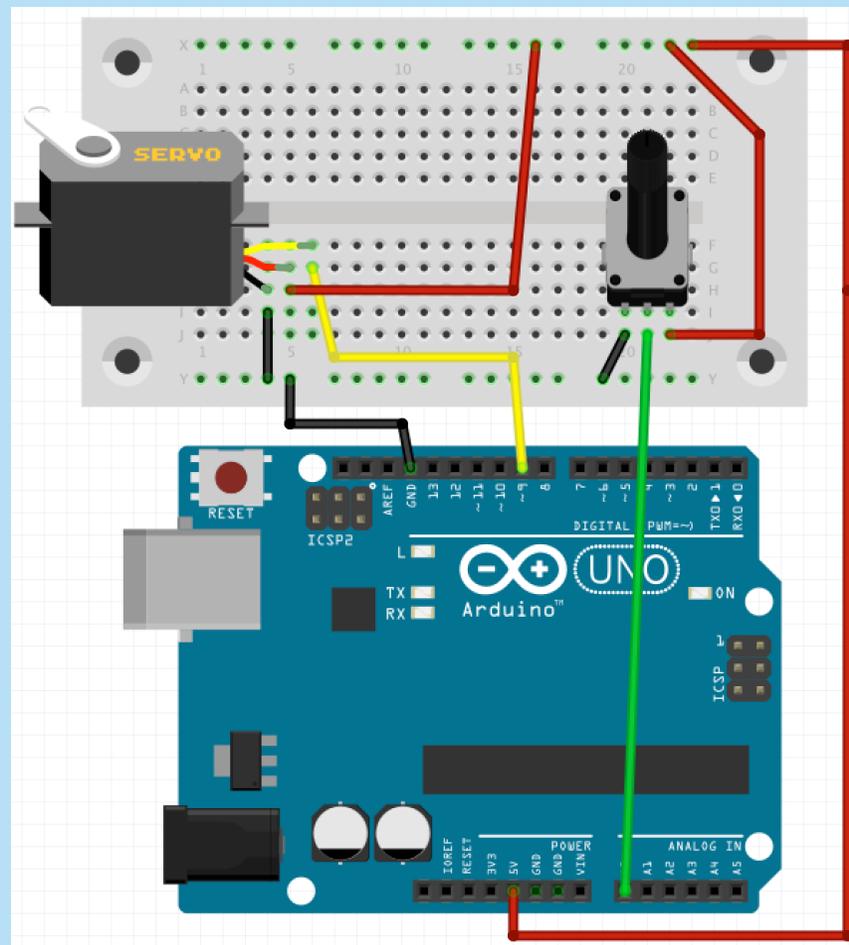
∞ 設置 9 腳位伺服馬達角度為 90

等待 1 秒

∞ 設置 9 腳位伺服馬達角度為 45

等待 1 秒

# 活動： 可變電阻控制 伺服摩打



# 活動： 可變電阻控制伺服摩打

$$180 / 1023 = 0.175$$

The image shows a Scratch script for an Arduino Uno. It starts with a 'When Arduino Uno starts' block, followed by a 'Repeat forever' loop. Inside the loop, there is a 'Set servo motor angle to' block. The 'servo motor angle' is set to '9' and the 'analog pin (A)' is set to '0'. The angle is multiplied by a value of '0.175'. Below this is a 'Wait 0.1 seconds' block. A yellow box with the equation  $180 / 1023 = 0.175$  has an arrow pointing to the '0.175' value in the code block.

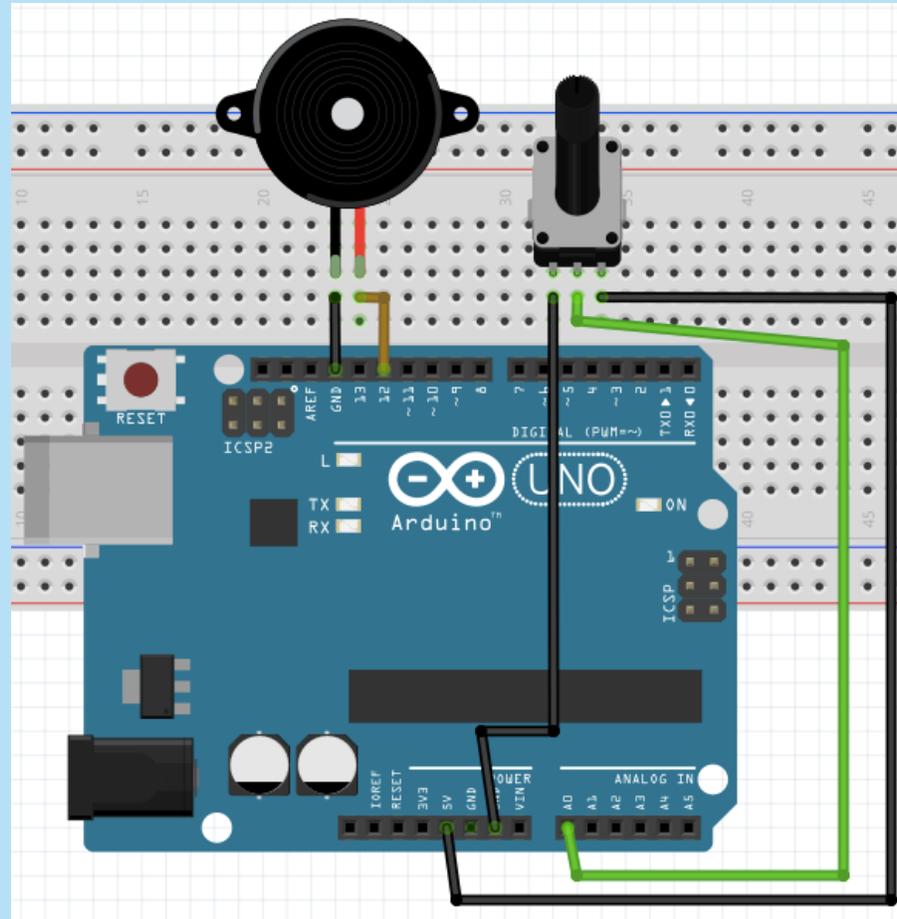
當 Arduino Uno 啟動時

不停重複

設置 9 腳位伺服馬達角度為 類比埠 (A) 0 \* 0.175

等待 0.1 秒

# 活動7：製作節拍器



# LED呼吸燈

# 超聲波感應器 Ultrasonic Sensor



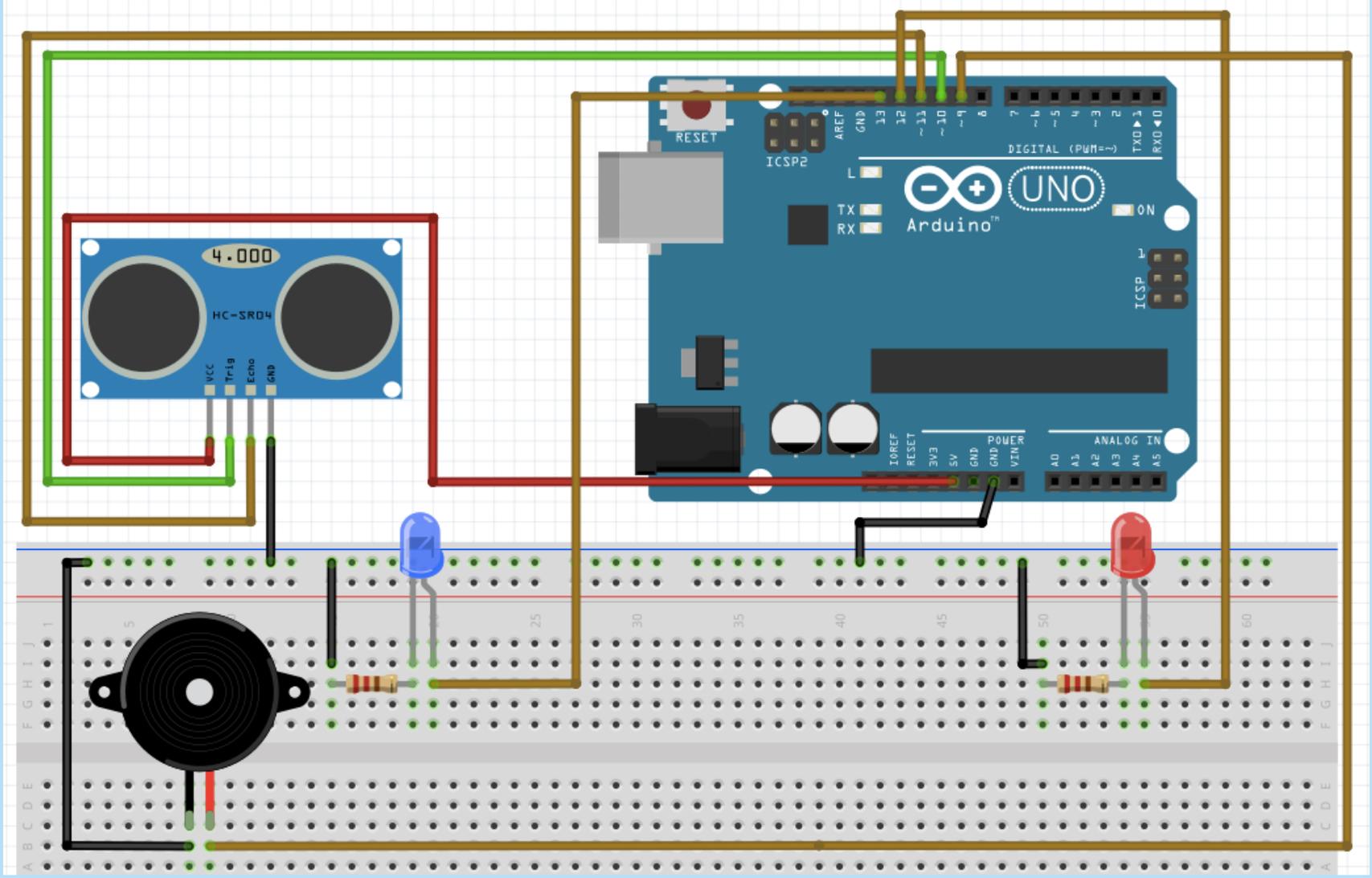
HC-SR04型號：

VCC：供電

Trig：發射超音波

Echo：接收超音波

GND：接地



當 Arduino Uno 啟動時

不停重複

如果  $\infty$  超音波距離感測器 trig腳位 10 echo腳位 11 大於 30 那麼

$\infty$  設置數位 pin 12 輸出為 低電位 ▾

$\infty$  設置數位 pin 13 輸出為 高電位 ▾

否則

如果  $\infty$  超音波距離感測器 trig腳位 10 echo腳位 11 小於 30 且  $\infty$  超音波距離感測器 trig腳位 10 echo腳位 11 大於 10 那麼

$\infty$  設置數位 pin 12 輸出為 高電位 ▾

$\infty$  設置數位 pin 13 輸出為 低電位 ▾

否則

$\infty$  設置數位 pin 12 輸出為 高電位 ▾

$\infty$  設置數位 pin 13 輸出為 低電位 ▾

$\infty$  第 9 腳位彈奏音符 C4 ▾ 1 拍



# 裝置挑戰

- 接駁超聲波傳感器和蜂鳴片，組成一個感測距離的裝置。
- 當距離少於10cm，裝置就會發出連續警號，當距離越近，警號越頻密。