

練習一：感應燈

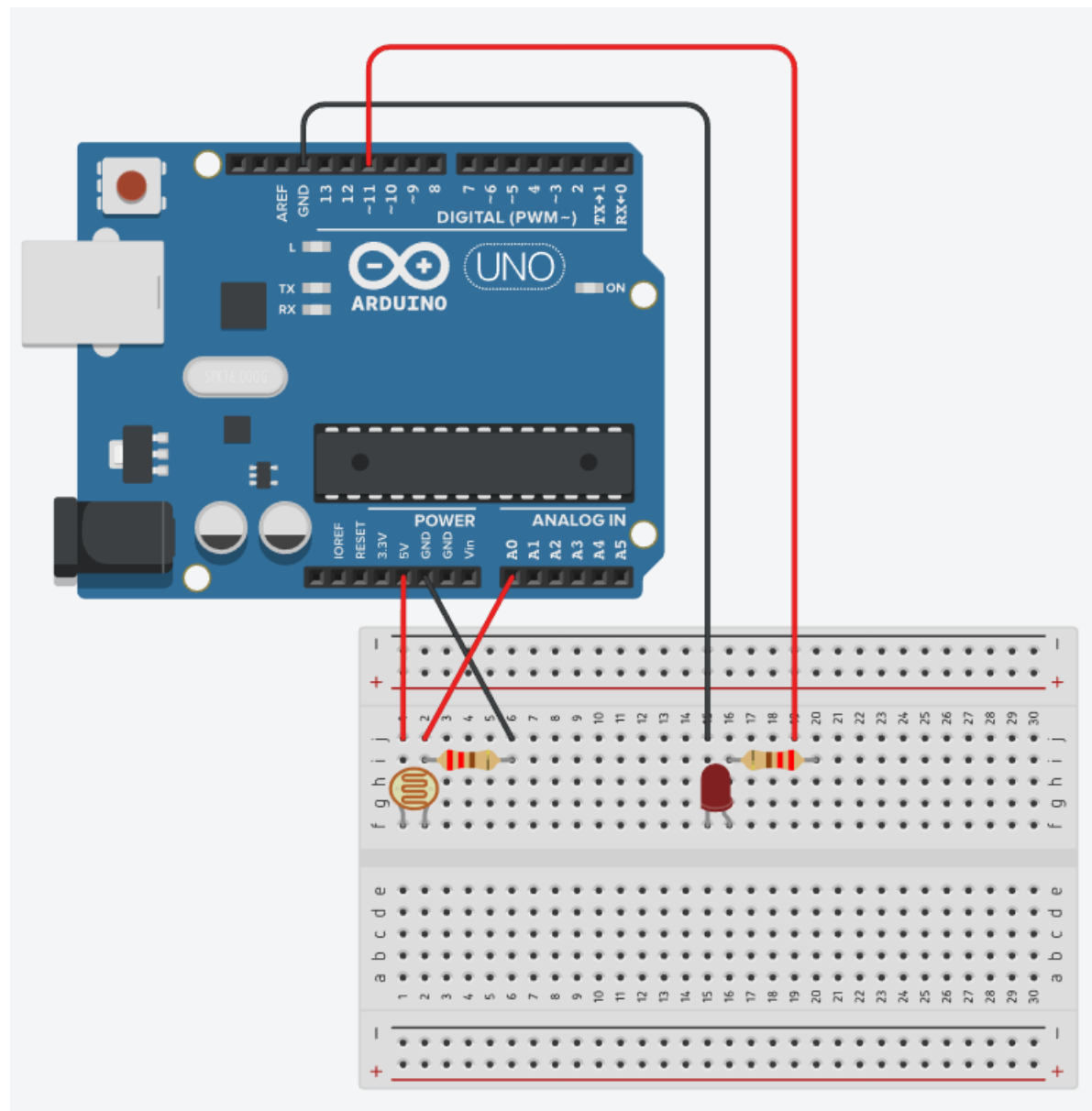
透過光敏電阻感應亮度的變化而開啟LED燈。

所需元件：

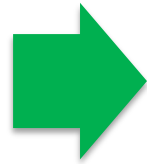
LED燈 x 1

定值電阻 x 2

光敏電阻 x 1



讀取光敏電阻
數值



設定開啟LED燈
門檻值



如果小於門檻值
則開啟LED燈
反之亦然

`analogRead(腳位)`
`Serial.println(輸出內容)`

`If(讀取數值 < 門檻值)`
開啟LED燈

練習二：電子琴

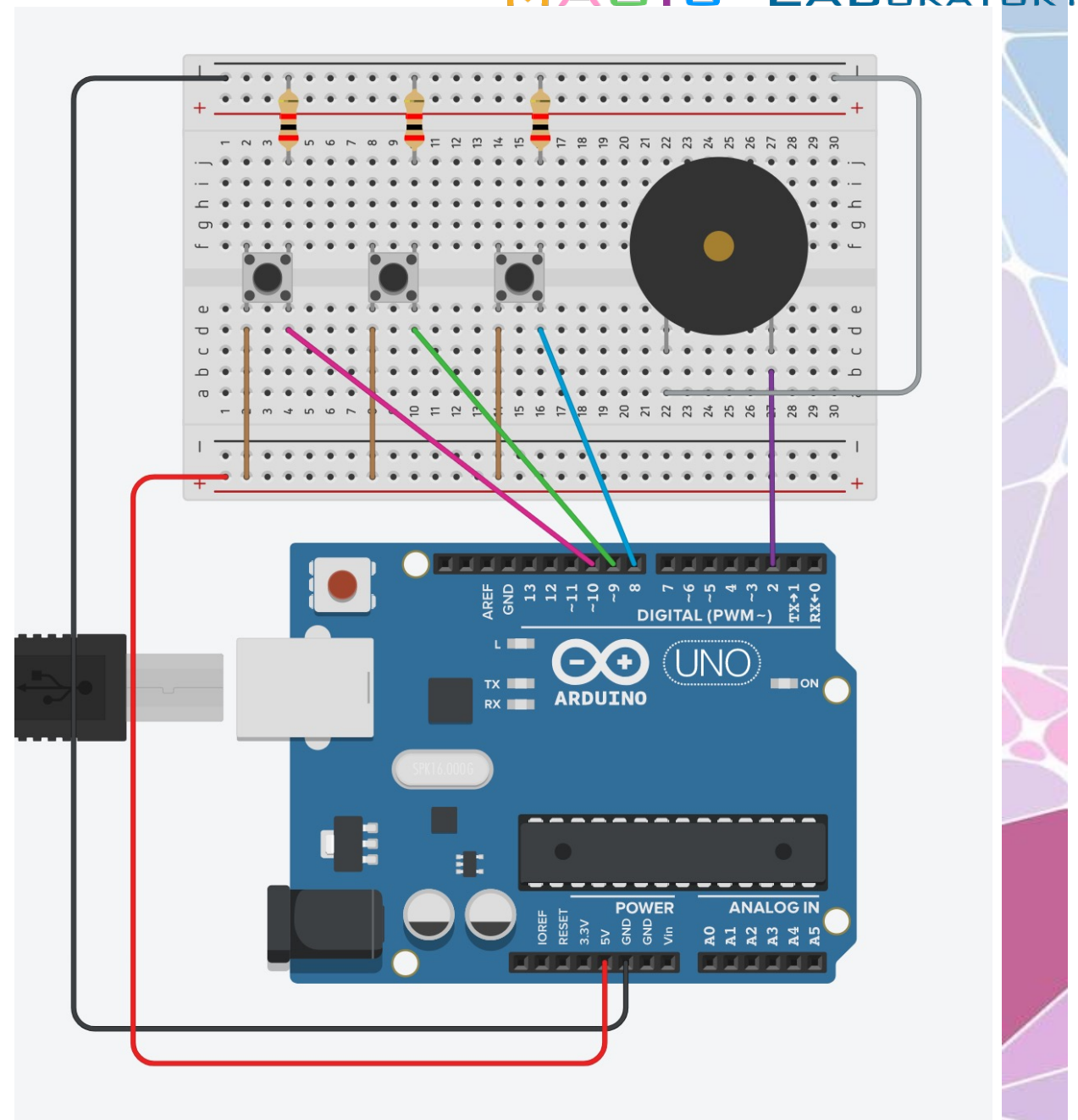
利用無源蜂鳴器搭配按鈕開關，模擬彈奏電子琴。

所需元件：

按鈕開關 x 3

定值電阻 x 3

無源蜂鳴器 x 1



設定按鍵腳位



讀取按鍵開關
狀態



按下去時發出
指定音頻聲音

`pinMode(腳位, INPUT)`
(可以省略)

`digitalRead(腳位)`

`tone(腳位, 頻率, 持續時間)`

音階	C (Do)	D (Re)	E (Mi)	F (Fa)	G (So)	A (La)	B (Si)
音頻	262	294	330	349	392	440	494

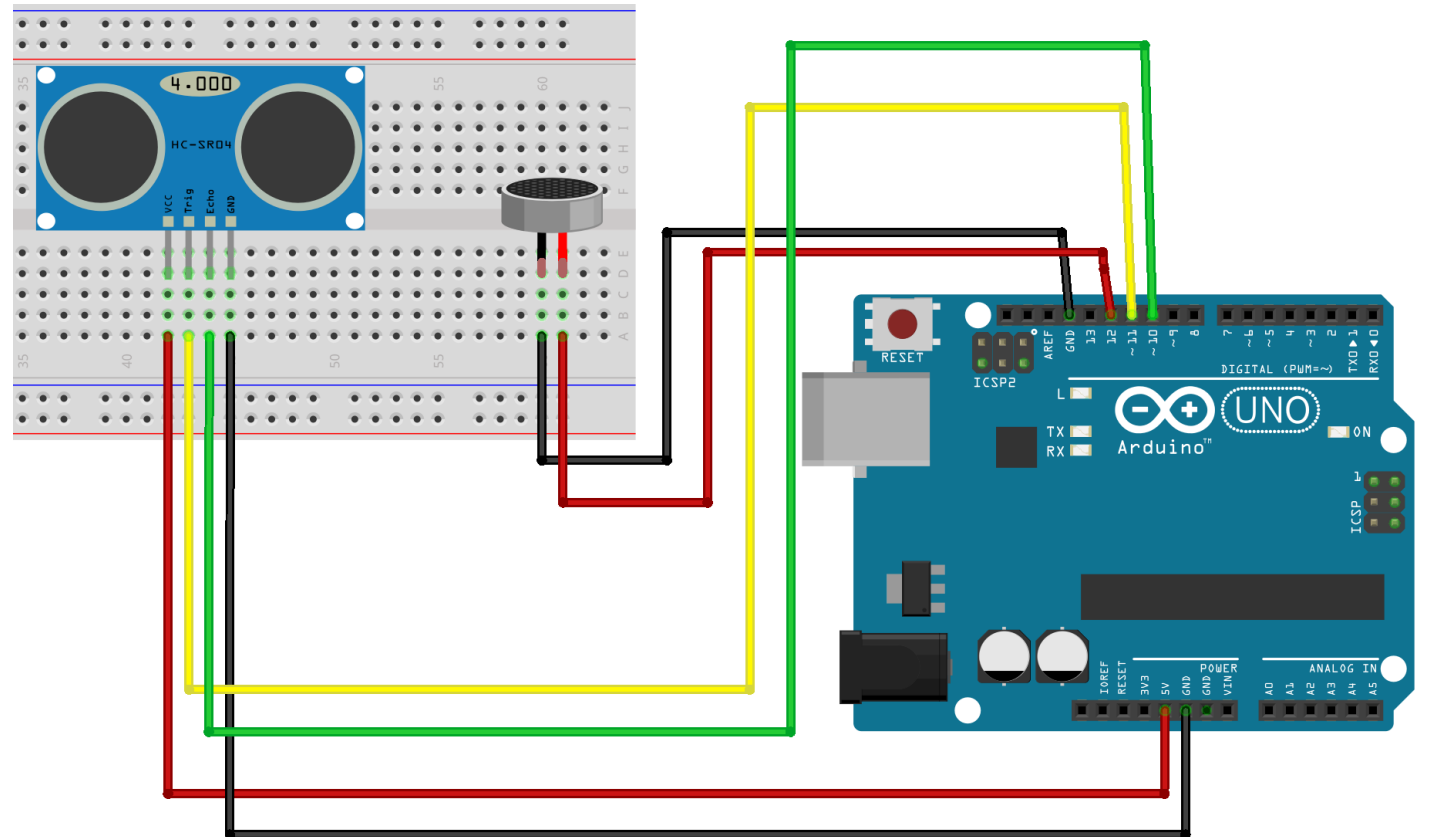
練習三：倒車雷達

利用無源蜂鳴器搭配超音波測距儀，模擬倒車雷達。

所需元件：

無源蜂鳴器 x 1

超音波測距儀 x 1



fritzing

測出距離



讓蜂鳴器隨距離遠近
而有不同鳴叫的頻率



調整成更像
倒車雷達的樣子

使用剛剛練習的程式

`tone(腳位, 頻率, 持續時間)`
`delay(間隔 + 距離)`

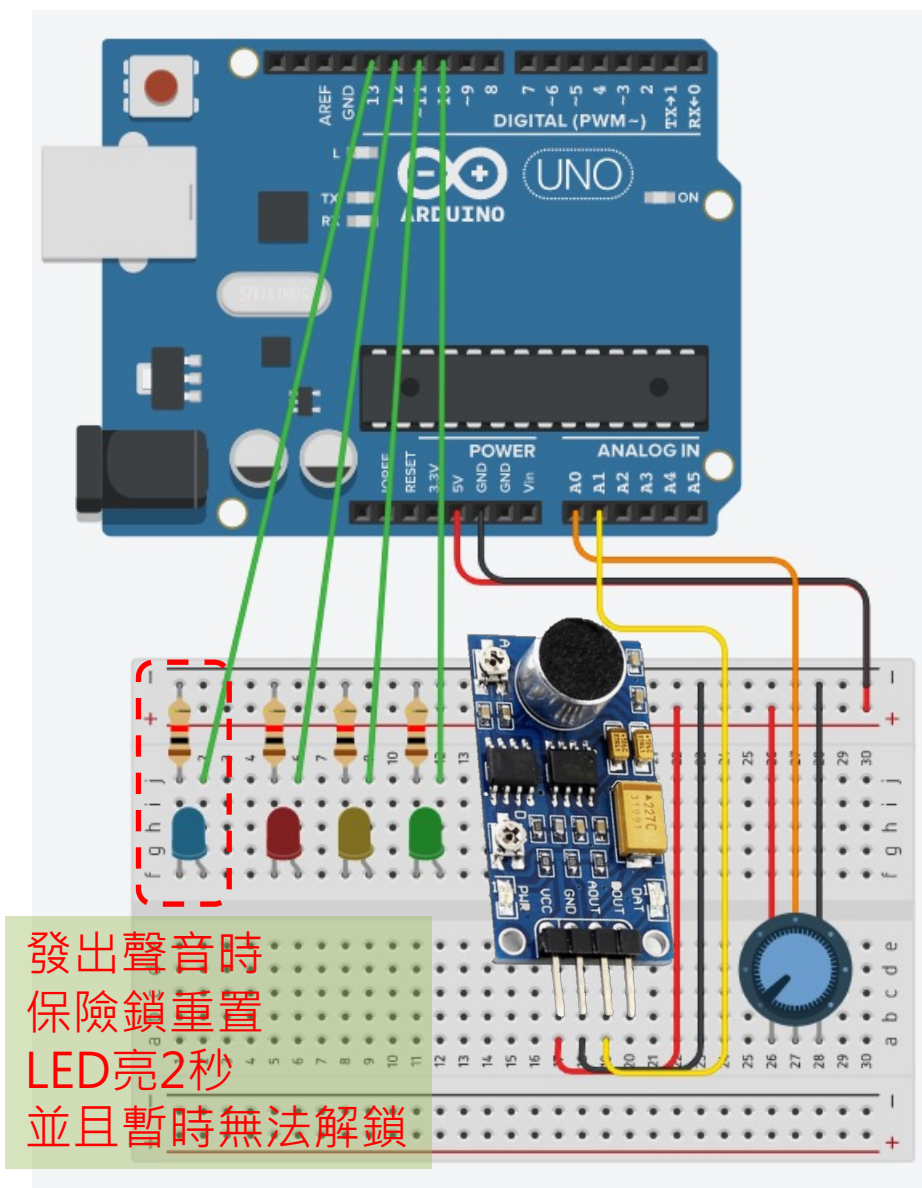
`delay(間隔 + 距離 * 倍數)`

作業四：1. 保險鎖

- 自訂三道鎖的密碼。
- 利用可變電阻當作保險鎖，解鎖時三顆LED會依序發亮
- 當發出聲音時保險鎖會重置且會暫時上鎖。

所需元件：

- 可變電阻 x 1
- LED燈 x 4
- 電阻 x 4
- 聲音感測器 x 1



讀取可變電阻及
聲音感測器數值



判斷聲音是否超過門檻值及
可變電阻是否在轉動



判斷可變電阻數值
是否在解鎖區間內

`value_VR = analogRead(可變電阻)`
`value_sound = analogRead(聲音感測器)`

`if(value_sound < 門檻值)`

`delta_VR = value_VR - last_value_VR`
`if(delta_VR == 0)`

`if(下限 < value_VR && value_VR < 上限)`

作業四：2. DJ盤

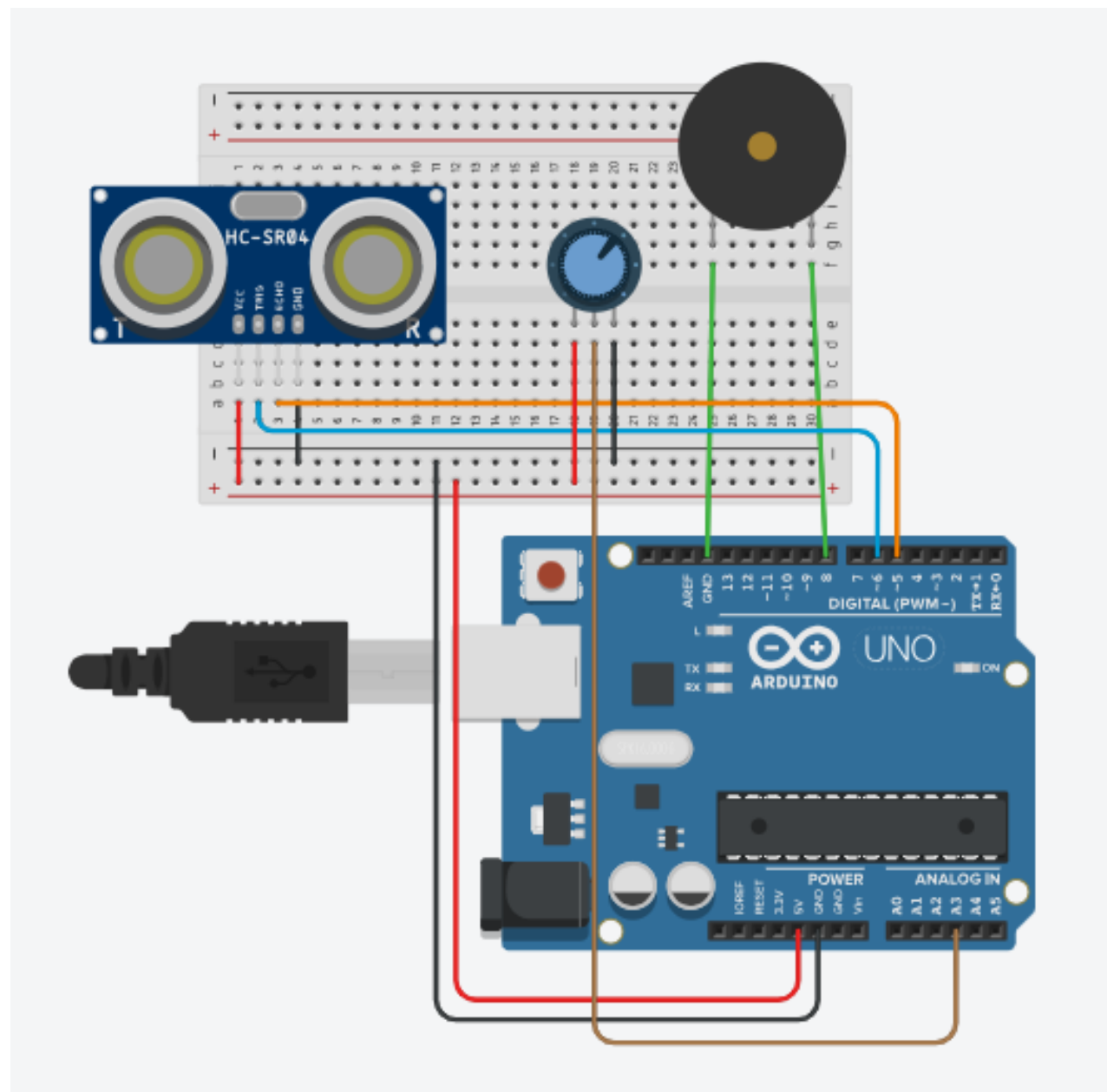
利用無源蜂鳴器搭配可變電阻與超音波測距儀，模擬DJ盤。

所需元件：

無源蜂鳴器 x 1

超音波測距儀 x 1

可變電阻 x 1



測出距離、
讀取可變電阻的值



距離改變蜂鳴器的音調
可變電阻調整音樂速度



細節調整

(速度較慢時，持續
時間也較長)

結合剛剛練習的程式

`tone(腳位, 頻率*音調倍率, 持續時間)`
`delay(100+sensorValue/2)`

持續時間：
`100+ sensorValue/10`

聖誕歌：

```
int frequency[61] =  
{ 330,330,330,0,330,330,330,0,  
  330,392,262,294,330,0,0,0,  
  349,349,349,349,349,330,330,330,  
  330,294,294,330,294,0,392,0,  
  330,330,330,0,330,330,330,0,  
  330,392,262,294,330,0,0,0,  
  349,349,349,349,349,330,330,330,  
  392,392,349,294,262 };
```

`0cm < 距離 ≤ 10cm` → 音調倍率=5
`11cm < 距離 ≤ 20cm` → 音調倍率=4
`21cm < 距離 ≤ 30cm` → 音調倍率=3
`31cm < 距離 ≤ 40cm` → 音調倍率=2
`41cm < 距離` → 音調倍率=1

作業四：3.？

請利用各式元件製作創意專題

- 震動感測器
- 溫溼度感測器
- 土壤濕度感測器
- 水位感測器
- 煙霧氣體感測器
- 心跳感測器