

超音波感測器測距程式

```
int trigPin = 8;           //超音波感測器 Trig 腳接 Arduino pin 8
int echoPin = 9;           //超音波感測器 Echo 腳接 Arduino pin 9
int speakerpin = 7;        //蜂鳴器 + 腳接 Arduino pin 7
long duration, cm;         //宣告計算距離時，需要用到的兩個實數

void setup() {
  Serial.begin(9600);       //設定序列埠監控視窗和 Arduin 資料傳輸速率為 9600 bps
  pinMode(trigPin, OUTPUT); //Arduino 對外啟動距離感測器 Trig 腳，射出超音波
  pinMode(echoPin, INPUT);  //超音波被障礙物反射後，Arduino 讀取感測器 Echo 腳的時間差
  pinMode(speakerpin, OUTPUT); //Arduino 對蜂鳴器送出電壓，使其鳴叫
}

void loop()
{
  //程式計算出距離值 cm

  digitalWrite(trigPin, LOW);
  delayMicroseconds(5);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  pinMode(echoPin, INPUT);
  duration = pulseIn(echoPin, HIGH);
  cm = (duration/2) / 29.1;

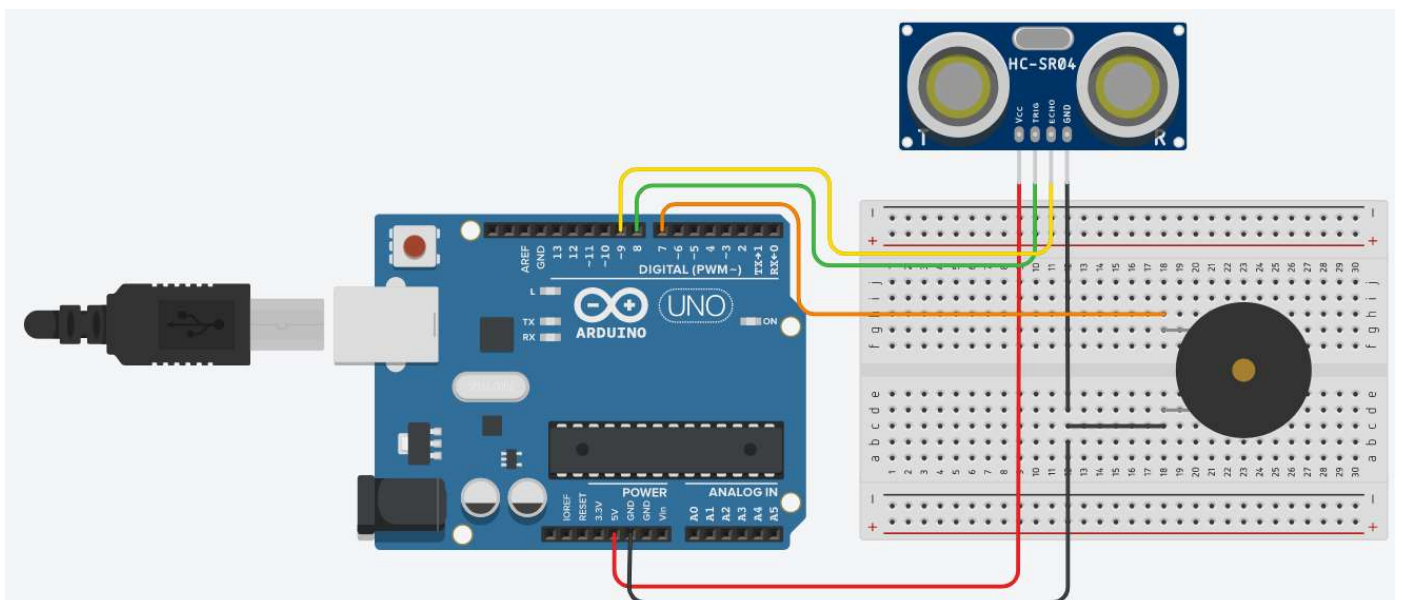
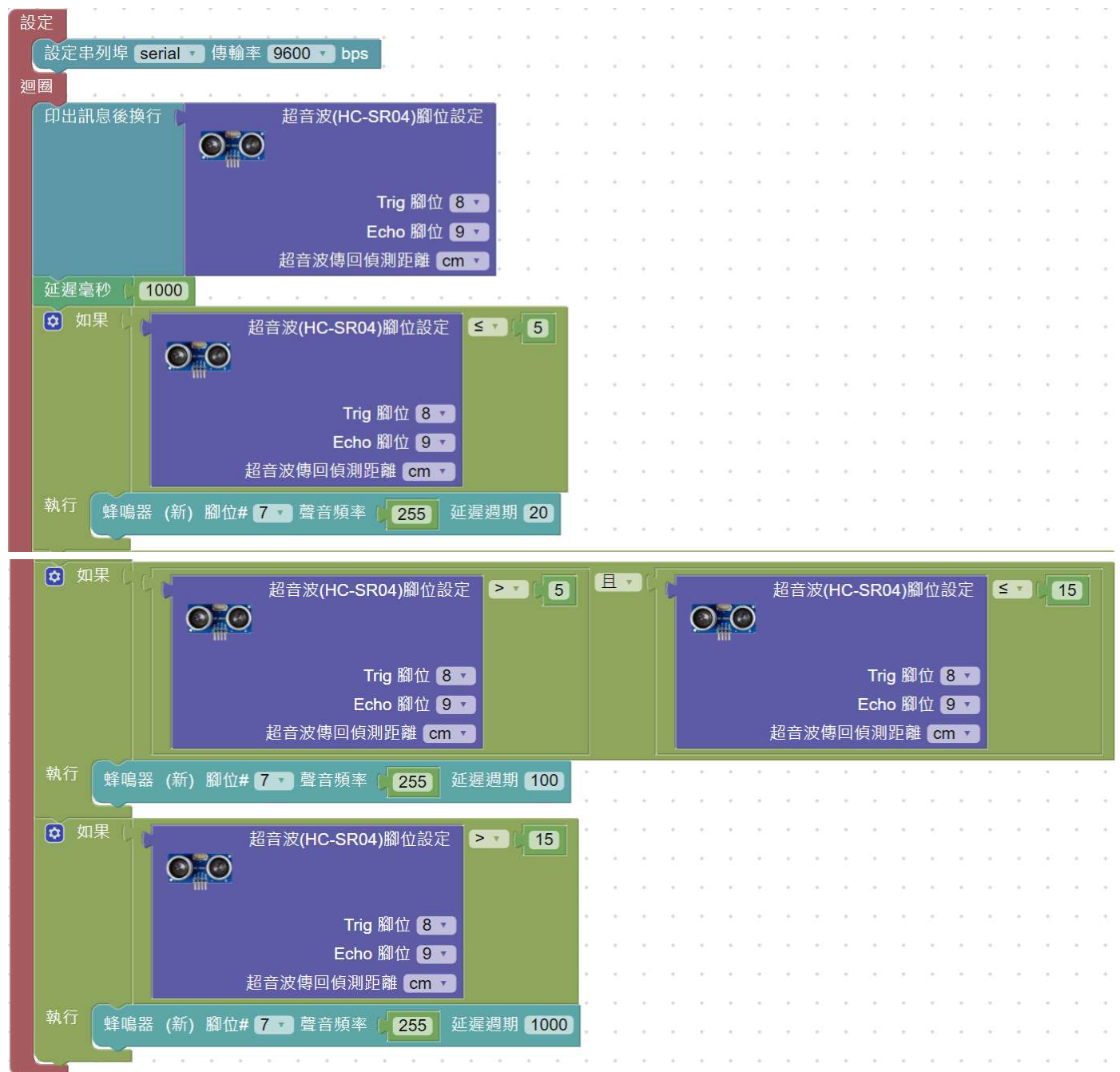
  Serial.print(cm);         //印出距離值 cm 在序列埠監控顯示器"單位公分"
  Serial.println(" cm");

  if (cm <= 5) {             //距離小於 5 公分，蜂鳴器一直叫
    digitalWrite(speakerpin, HIGH);
    delay(20);
  }

  if (cm > 5 && cm <= 15) {   //距離介於 5 到 15 公分，蜂鳴器斷斷續續叫，每次 0.1 秒
    digitalWrite(speakerpin, HIGH);
    delay(100);
    digitalWrite(speakerpin, LOW);
    delay(100);
  }

  if (cm > 15){              // 距離大於 15 公分，蜂鳴器斷斷續續叫，每次 1 秒
    digitalWrite(speakerpin, HIGH);
    delay(1000);
    digitalWrite(speakerpin, LOW);
    delay(1000);
  }
  delay(10);
}
```

motoBlockly 超音波感測器測距程式



SG90 伺服馬達移動角度及旋轉

使用 180 度馬達：

```
#include <Servo.h>    //載入函式庫，這是內建的，不用安裝
Servo myservo;        // 建立 SERVO 物件

void setup()
{
  myservo.attach(9);  // 設定要將伺服馬達接到哪一個 PIN 腳
}

void loop()
{
  myservo.write(0);    //旋轉到 0 度，就是一般所說的歸零
  delay(1000);
  myservo.write(90);   //旋轉到 90 度
  delay(1000);
  myservo.write(180);  //旋轉到 180 度
  delay(1000);
  myservo.write(90);
  delay(1000);
}
```

使用 360 度馬達：

```
void loop()
{
  myservo.write(0);  //不停旋轉(正轉)
  delay(1000);
  myservo.write(90); //馬達不動
  delay(1000);
  myservo.write(180); //不停旋轉(反轉)
  delay(1000);
  myservo.write(90);
  delay(1000);
}
```

motoBlockly SG90 伺服馬達移動角度及旋轉



兩開關控制 SG90360 度伺服馬達正反轉

