

新竹市第 39 屆中小學科學展覽會

作品說明書

科別：生活與應用科學(一)機電與資訊

組別：國小組

作品名稱：如何利用超音波陣列模擬雷達偵測隱形外型飛機？

關鍵字：超音波、雷達、隱形戰機、陣列、Arduino、CD74HC4067、多工/解多工

編號：



目錄

摘要.....	3
壹、研究動機.....	3
貳、研究目的.....	5
參、研究設備及器材.....	5
肆、研究過程與方法.....	7
伍、研究成果.....	27
陸、討論與結論.....	29
柒、參考資料及其他.....	30

摘要

我們希望透過自身製作的超音波陣列雷達，來實現能夠探測現今的一大軍事利器—隱形戰機，於是我們開始利用每週二的早自修進行研究。

我們首先調查的隱形戰機的設計原理，並了解為何它可以稱為隱形飛機。接著老師開始帶我們認識 Arduino，並利用 IDE 來開發 Arduino 的控制程式。在一切 Arduino 所需要的基本控制能力都具備以後，我們開始進入本實驗的主題—如何利用超音波陣列模擬雷達偵測隱形外型飛機？

初步我們研究了超音波的最大偵測範圍與偵測物體的材質影響，有了實驗數據後，第二步即增加超音波測距模組的數量，研究排列方式與陣列開合方向的實驗，透過這些實驗，我們有了製作方向。最終，我們用三十顆超音波測距模組排列出超音波陣列雷達。

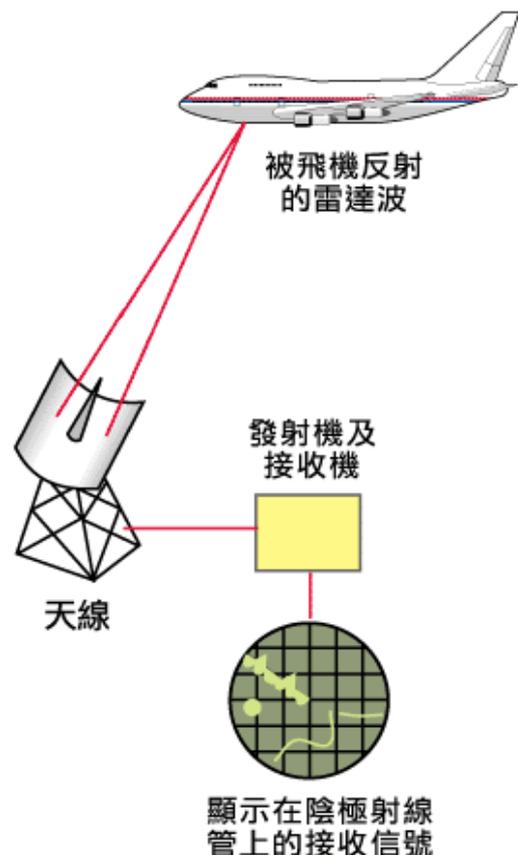
超音波陣列雷達的造型往內彎曲，採圓弧式排列。利用多顆不同角度的超音波測距模組做事當排列，讓我們的雷達的單邊有效探測範圍可達 60 多度，比單一顆超音波測距模組的有效探測範圍整整多了約一倍。

壹、研究動機

隨著科技的發展，現在已經有了連軍用雷達都偵測不到的隱形戰機。隱形戰機，這個名詞讓我們感到興趣的是，飛機是如何做到隱形？經過我們大量查閱資料與科普知識，我們終於了解到，原來隱形戰機指的是雷達隱身，並非視覺隱身。

要達成雷達隱身，常見的方式有三種：

第一種是飛機的外型需要盡量滿足可以減少雷達反射波的設計，讓雷達波被折射到雷達接收不到的方向。第二種手段是在飛機的表面塗上吸波材質，讓雷達波被吸收，進而達到減少反射雷達波的目的。第三種就是使用利用電子戰吊艙，強行發射大功率電子干擾手段，讓敵人的雷達接收不到飛機反射回去的雷達反射波。

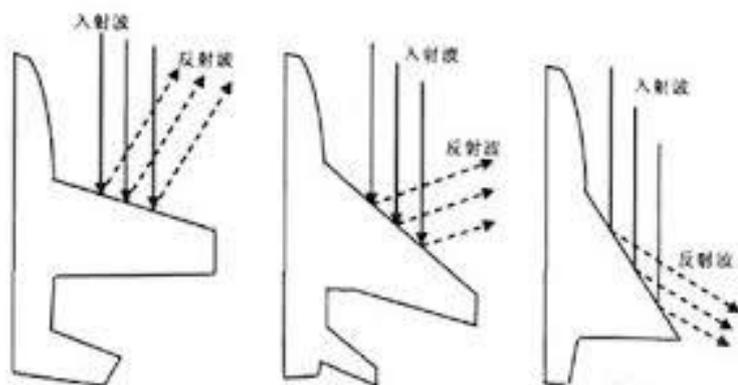


在我們了解以上的基本知識後，我們想到學校電腦課中，老師曾經教過我們使用一個很簡單的小型元件 **HC-SR04**，這是一種超音波測距感測器。我們知道，雷達發出的雷達波是一種電磁波，然而超音波是一種聲波，基本上，電磁波和聲波是兩種完全不同的東西，但是所有的波在一些標準情況下，都會出現共同的性質與行為表現：

- **反射**—波的傳遞方向經碰撞反射面而改變。
- **折射**—波的傳遞方向因行經不同介質而改變。
- **繞射**—波與路徑上的障礙物交互作用而改變方向，當波長與障礙物的大小數量級差不多時，此現象會變得更加顯著。
- **干涉**—兩個波互相疊加。
- **分散**—合成波因頻率不同而分裂。
- **直線傳播**—光波在一條直線上的運動。

基於這些共同性質，我們團隊認為我們可以使用我們熟悉的 **Arduino** 等工具，對隱形戰機的外型展開研究，於是我們開始了破解隱形戰機外型奧秘的實驗。

我們首先調查了戰機外型的設計原理：一般的戰機為了減低在飛行時受到的空氣阻力，所以外殼大部份都是由彎曲的表面所組成的。因此，不論雷達電磁波從那一個角度射到機身，總是有部分會被反射到原來的方向(如下圖)，使雷達探測器收到很強的訊號。現代有些隱形戰機的機殼是由眾多平直的表面所構成，再加上特別設計過的機身角度，所以電磁波不易被反射到原來的方向。除此之外，製造外殼的材料也是經過特別選擇，可以有效地吸收雷達信號的能量。材料多是使用碳、碳纖維聚合物和磁性鐵酸鹽類的物質。結果，隱形戰機在雷達中幾乎消失，就只像一隻小鳥！



接著我們開始設計地表最強雷達，在不考慮吸波材質與電子干擾手段的因素下，**我們選擇針對隱形戰機的外型來做研究**。因為要偵測隱形戰機，就必須要讓我們的雷達克服對平直面的探測能力(如上圖)，所以需要克服折射角度的問題，我們想到的是：把很多個超音波測距模組排列成不同角度，做成圓弧形的矩陣，這樣不論隱身戰機的外型怎麼設計，一定會被其中一些超音波測距模組探測到從機身反彈回來的訊號。實現對隱形戰機的成功探測。

在軍事設計上，戰機的設計技術越來越精良，為了國防安全，偵測敵軍的效果也要相對的提升，如果真的能掌握到偵測隱形戰機的技术，那麼我們就可以在最短時間內發現敵軍，讓我們的國軍在有限的時間內做好應對及準備，保護中華民國人民的安全，以上為我們主要的研究動機。

貳、研究目的

我們想要設計能夠偵測隱形戰機外型的雷達，希望我們的雷達能不論用哪一個角度來掃描隱形外型的飛機，我們都能夠收集到最多的反射訊號。依據本研究主題設計實驗流程，以設計出，實驗目的如下。

1. 研究超音波測距模組的功能。
2. 研究超音波測距模組對於不同材質之平面物在不同角度下的反射效果。
3. 研究兩個超音波測距模組在不同開合角度時對於偵測效果的影響。
4. 實作 6x5 超音波矩陣雷達並測試雷達之效能。

參、研究設備及器材

本研究所使用的主要元件彙整如下

<p>Arduino UNO</p> <p>x1</p>		<p>一種基於開源硬體和開源軟體的單板微控制器和微控制器套件，用於構建數位裝置和互動式物件，以便在物理和數位世界中感知和控制物件。</p> <p>我們在實驗的初期，為了降低成本與方便學習，老師主要使用 Arduino UNO 來指導我們學習有關 Arduino 的相關知識。</p>
<p>超音波測距模組 (HC-SR04)</p> <p>x30</p>		<p>超音波(Ultra Sonic)模組，用來測量前方障礙物的距離。</p> <p>它的運作原理很簡單，模組會送出 8 個 $\times 40\text{kHz}$ 的方波，如果前方有障礙物，信號就會返回，模組收到信號後，再利用返回的時間，去計算該障礙的距離。</p> <p>HC-SR04 的偵測功能不是全方位的，依據網路上的文件說，只要超過 30 度的偏角，便會讓 HC-SR04 無法偵測。</p>

<p>Arduino Pro mini</p> <p>×1</p>		<p>Arduino Mini Pro 是一個微控制器電路板，基本上具有與 Arduino UNO 一樣的功能，擁有 14 個數位輸入/輸出引腳，其中 6 個可作為 PWM 輸出，8 個類比輸入，一個 16MHz 的石英振盪器，一個重置按鈕。</p> <p>Arduino Mini Pro 為了減輕體積，把 USB 插頭刪掉，使其重量減至 2g，厚度只有 6mm，是開發超微型電子及穿戴式科技的佳品。</p>
<p>CD74HC4067</p> <p>類比多工器</p> <p>×2</p>		<p>CD74HC4067 是非常方便的 16 通道的(數位/類比)多工/解多工器。</p> <p>該晶片就像一個旋轉開關，利用四個位址引腳(S0-S3)分別輸出 0 或 1，用以決定公共引腳(電路板上的 SIG)到底是與另一側 16 個通道的哪一個引腳相連接。</p> <p>這將允許我們僅使用 5 個引腳就可以將最多 16 個傳感器連接到我們的系統。</p> <p>由於我們的超音波陣列會連接 30 個超音波偵測器，但不論是 Arduino UNO，或 Arduino Pro Mini 卻都只有 13~14 個引腳，因此引腳明顯不足。所以 30 個裝置需要使用 2 個 CD74HC4067 來連接，這樣我們就可以用 Arduino 上的 10 個引腳來控制最多 32 個 HC-SR04 超音波測距器。</p>
<p>USB to TTL</p> <p>×1</p>		<p>因為小型化，所以我們的成品將使用不具備 USB 通訊的 Arduino Pro Mini 來作為控制核心。</p> <p>為了能夠寫入 Arduino 程式，我們需要將 Arduino Pro Mini 微控制器上的序列通訊，通過微控制器的 RXD、TXD、VCC、GND 四個引腳遵照 TTL 原則在電腦與微控制器之間建立起通訊。</p> <p>USB to TTL 就是一個具有這樣功能的轉接器。</p>

<p>SSD1306 OLED</p> <p>×1</p>		<p>OLED 有機發光二極體(又稱為有機電雷射)顯示器(Organic Light-Emitting Diode, OLED)。</p> <p>OLED 顯示技術具有自發光的特性，採用非常薄的有機材料塗層和玻璃基板，當有電流通過時，這些有機材料就會發光，而且 OLED 顯示螢幕可視角度大，並且能夠節省電能。</p> <p>這將使用在我們的最後的成品上，主要用來顯示掃描到的數量及距離</p>
<p>5V2A</p> <p>電源供應器</p> <p>×1</p>		<p>根據文件，Arduino UNO 的耗電量約為 50mA，因此 Arduino Pro Mini 應該也相差不多。一個 HC-SR04 超音波偵測器的耗電量約為 2mA，30 個超音波串起來全部耗電約為 60mA，SSD1306 OLED 的耗電量約為 25mA，經過計算，整個超音波陣列雷達的耗電量約為：</p> <p>50mA+60mA+25mA=135mA</p> <p>所以我們使用一個最大 2A(約為 2000mA)的電源供應器來為我們的雷達系統提供一個穩定的外部電源。</p>
<p>角度控制器</p> <p>(自製)</p>		<p>為了能夠控制開合方向不會偏向某一個方，因此設計了角度控制器。只要拉後方黏合的部分，即可同時控直左右兩邊的紙板，調整超音波的偵測角度。</p>

肆、研究過程與方法

一、研究流程

我們利用每星期二的早修時間，在一起討論研究方法，並請老師協助指導。期望使用 Arduino 製作出可偵測隱形戰機的超音波陣列雷達。

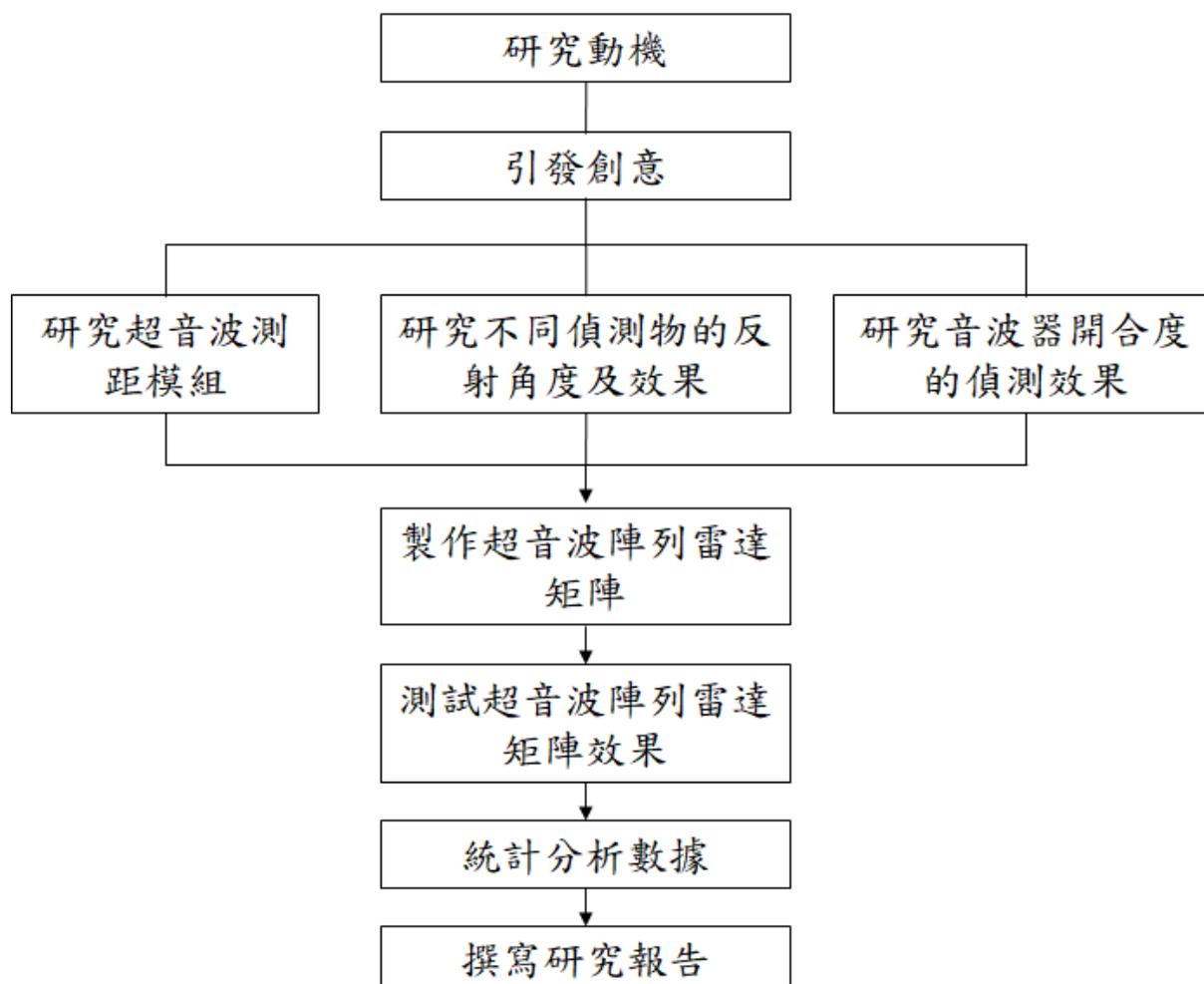
一開始，老師就希望我們屏除積木程式語言，直接指導我們從 Arduino 的 IDE 來學習基礎的程式設計。例如使用 `pinMode(OUTPUT/INPUT)`來設定引腳的模式、`digitalRead(pin)`讀取

引腳是否有電流通過、`digitalWrite(pin,value(HIGH/LOW))`輸出電流給引腳、`if...else` 來完成一個基本的邏輯判斷式.....等。

二、我們的研究步驟與方法為

1. 研究超音波測距模組的功能。
(如何控制單個超音波測距模組與同時連接兩個超音波測距模組之研究)
2. 研究超音波測距模組對於不同材質之平面物在不同角度下的反射效果。
3. 研究兩個超音波測距模組在不同開合角度時對於偵測效果的影響。
4. 實作一個 6×5 的超音波陣列雷達，並使用該達對平直面物體進行測試，看看是否能達到預期的效果。

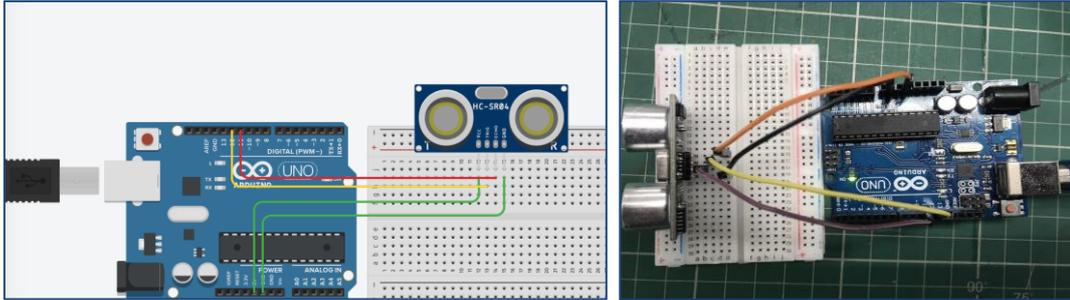
三、研究流程圖



四、研究過程內容與說明

研究超音波測距模組的功能

1. 利用 ArduinoUNO 控制單一 HC-SR04 超音波測距模組

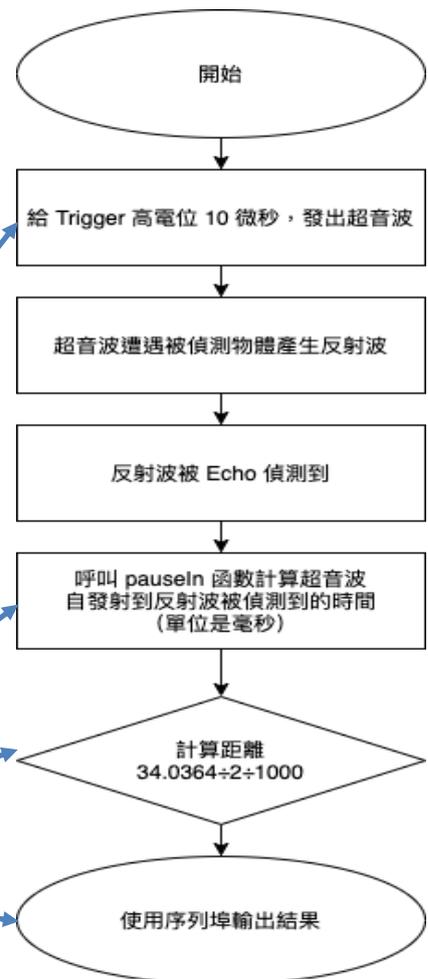


程式碼：

```
int trig=11;
int echo=12;
long duration; //----- 偵測到反射波的時間(微秒)
float cm;

void setup()
{
  pinMode(trig,OUTPUT);
  pinMode(echo,OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  digitalWrite(trig,LOW);
  delayMicroseconds(5);
  digitalWrite(trig,HIGH);
  delayMicroseconds(10);
  digitalWrite(trig,LOW);
  duration=pulseIn(echo,HIGH);
  cm=duration*34.0364/2/1000;
  Serial.println(cm);
  delay(500);
}
```



實驗器材：Ariduno UNO×1、HC-SR04×1、杜邦線×4。

程式設計的流程是先由超音波測距模組的 **trigger** 引腳輸出低電位(或 0)，讓 **trigger** 狀態歸零。然後再由 **trigger** 引腳輸出高電位(或 1)，且至少持續維持 10 微秒(10×10^{-6} 秒)，誘使超音波測距模組向前發出一段聲波。第二，關閉 **trigger** 引腳的高電位，停

止發出聲波。第三，由超音波測距模組的 `echo` 引腳讀取是否有被反射的聲波被接收到，並由 `pulseIn(echo,HIGH)`指令計算接收時的經過時間。

最後，由六下數學第二單元（速度）裡學到的知識，我們知道：

因為，速度 = 距離 ÷ 時間。

所以，距離 = 速度 × 時間

在這裡我們必須注意，由 `pulseIn(echo,HIGH)`讀取到的時間是聲波來回的時間，所以單趟的時間需要先除以 2。我們由網路上查到的資料顯示，聲音的速度(又稱為音速)約為：

$$\begin{aligned} & 340.364(\text{公尺/秒}) \\ & = 340.364 \times 100 \div 1000(\text{公分/毫秒}) \\ & = 34.0364(\text{公分/毫秒}) \end{aligned}$$

利用上面的公式，我們就可以計算出由超音波測距模組所計算出來的距離。

```
digitalWrite(trig,LOW);
```

`trigger` 引腳輸出低電位(或 0)，讓 `trigger` 狀態歸零。

```
delayMicroseconds(5);
```

狀態歸零需持續維持 5 微秒(5×10^{-6} 秒)。

```
digitalWrite(trig,HIGH);
```

`trigger` 引腳輸出高電位(或 1)，誘使 `trigger` 向前發出一段聲波。

```
delayMicroseconds(10);
```

發出聲波需持續維持 10 微秒(10×10^{-6} 秒)。。

```
digitalWrite(trig,LOW);
```

`trigger` 引腳輸出低電位(或 0)，讓 `trigger` 狀態歸零。

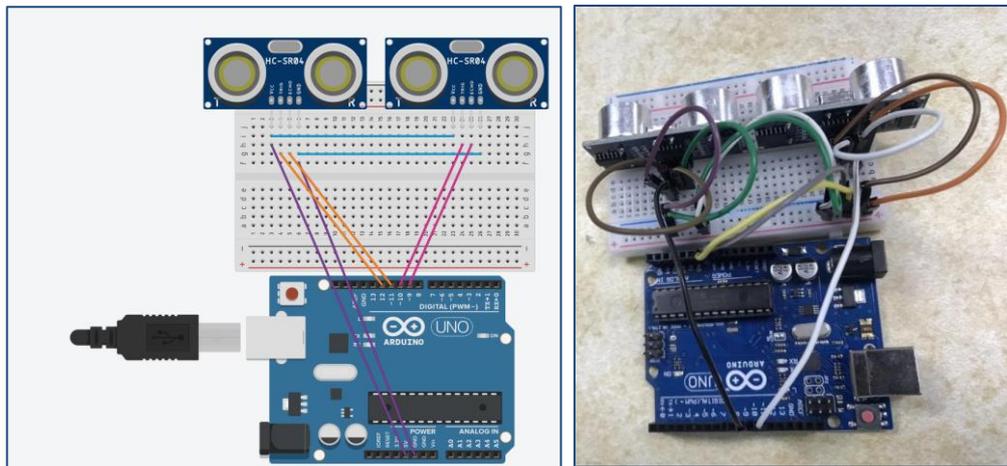
```
duration=pulseIn(echo,HIGH);
```

`echo` 引腳偵測並計算接收到反射波的時間。

```
cm=duration*34.0364/2/1000;
```

利用公式計算超音波測距模組與前方障礙物的距離。

2. 利用 ArduinoUNO 同時控制兩個 HC-SR04 超音波測距模組



程式碼

```

...
void loop()
{
    //---- 這裡先讓第一個超音波偵測與前方障礙物的距離
    digitalWrite(trig0,LOW);
    delayMicroseconds(5);
    digitalWrite(trig0,HIGH);
    delayMicroseconds(10);
    digitalWrite(trig0,LOW);
    d0=pulseIn(echo0,HIGH);
    cm0=d0*34.0364/2/1000;

    //---- 然後讓第二個超音波偵測與前方障礙物的距離
    digitalWrite(trig1,LOW);
    delayMicroseconds(5);
    digitalWrite(trig1,HIGH);
    delayMicroseconds(10);
    digitalWrite(trig1,LOW);
    d1=pulseIn(echo1,HIGH);
    cm1=d1*34.364/2/1000;

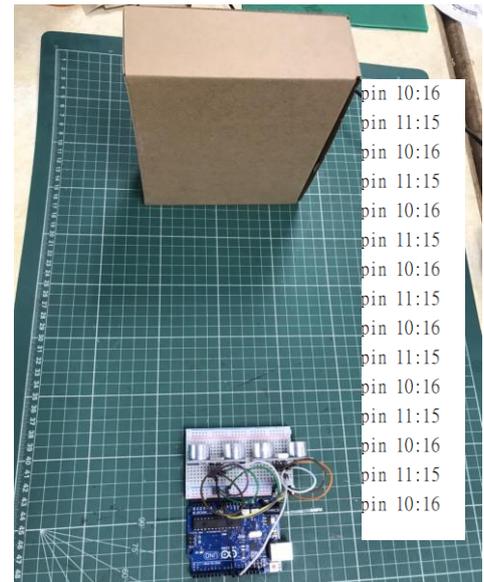
    //---- 利用 Serial Monitor 將兩個超音波偵測的結果列印出來
    Serial.print(cm0);
    Serial.print(" , ");
    Serial.println(cm1);
}

```

實驗器材：Ariduno UNO×1、HC-SR04×2、杜邦線×10。

將超音波偵測器模組、電源連結於開發板上，觀察連接後的超音波都可以經過我們的程式正常工作，只是不同超音波之間所偵測出來的距離，似乎會有誤差，如下面表格所示。

Time (s)	pin 11 (HC-SR04-1)	pin 12 (HC-SR04-2)
17	15cm	16cm
18	15cm	16cm
19	15cm	16cm
20	15cm	16cm

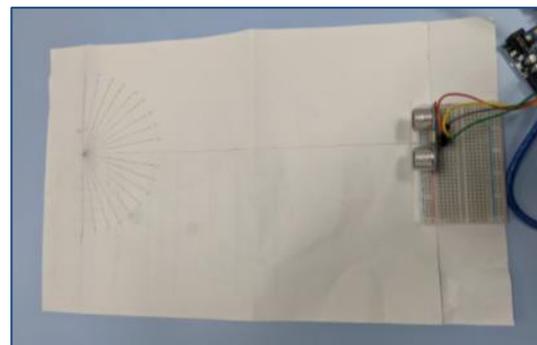


這樣的結果，應該是在超音波測距模組的容許誤差範圍內。這些許的誤差範圍，我們可以利用簡易的程式來予以修正。

研究超音波測距模組對於不同材質之平面物在不同角度下的反射效果

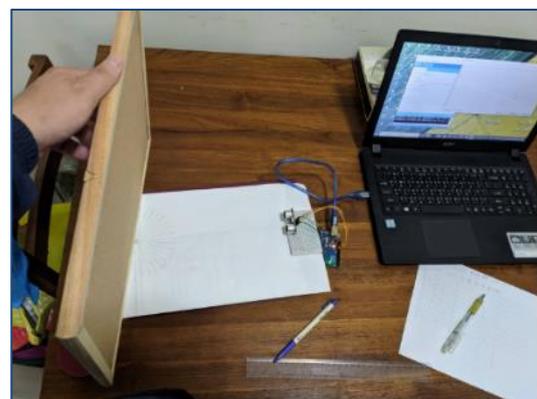
實驗一：

從相關資料得知，超音波在平滑面時反射效果較好，在粗糙面時會有漫射的情形，為了確認超音波對於粗糙面、平滑面的反射效果，我們進行了超音波測距模組對於不同材質的平面障礙物的不同角度的探測實驗。

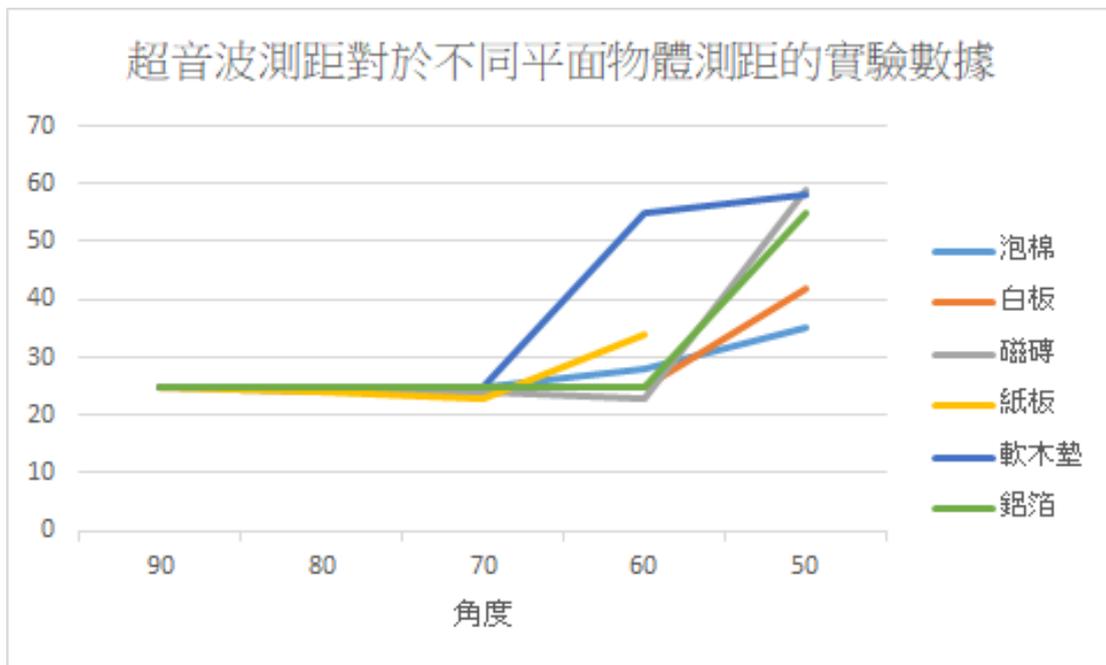


我們在紙上黏上單個超音波偵測器，並且在另一端畫上角度，即可開始進行不同平面物的角度反射效果。

1. 首先先測量超音波和待測物 90 度的距離為標準。
2. 進行不同角度的測試。
3. 最後進行測試，以 90 度為標準，比較不同角度及待測物的距離。



超音波測距對於不同平面物體測距的實驗數據如下表所示。



	泡綿	白板	磁磚	紙板	軟木墊	鋁箔
90 度	25cm	25cm	25cm	25cm	25cm	25cm
80 度	25cm	25cm	24cm	24cm	25cm	25cm
70 度	25cm	25cm	24cm	23cm	25cm	25cm
60 度	28cm	47cm	55cm	34cm	27cm	25cm
50 度	35cm	42cm	59cm	38cm	58cm	55cm

從上面的表格我們知道，超音波在偵測障礙物時，偵測的效果與被偵測物的表面材質似乎有很大的相關，例如：表中顯示，白板、磁磚、紙板等都在 60 度左右就讓超音波喪失訊號，而泡棉、軟木墊、鋁箔則需要偏移角度到達 50 度才會喪失訊號。我們從這六種材質的外表特徵，我們大約可以歸納出下面的結論。

實驗一結論：

如果被偵測的物體，如果表面越光滑，那超音波對其偵測的有效範圍就越小。

反之，

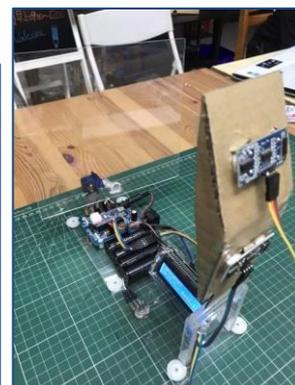
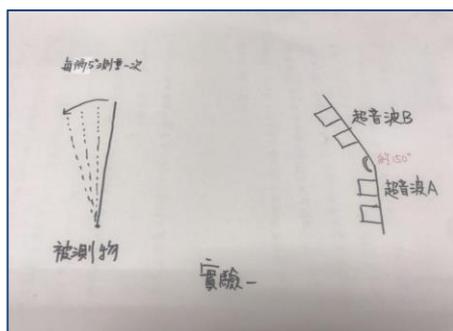
如果被偵測的物體，如果表面越粗糙，那超音波對其偵測的有效範圍就越大。

研究兩個超音波測距模組在不同開合角度時對於偵測效果的影響

因為我們要使用超音波建立陣列模擬雷達，所以由實驗一的結論，如果所有的超音波都在同一個平面使用相同的角度來偵測障礙物，那麼就失去建立陣列的意義。因此，我們需要知道兩個超音波以不同角度來偵測障礙物時效果的差異。

實驗二：兩個超音波方向夾角約為 150 度，呈現向中間夾擠、集中的方向

角度 (度)	A 公分(cm)	B 公分(cm)
90	19.7	20.3
85	20.4	21.4
80	21.2	22.7
75	21.7	22.7
70	22.4	23.1
65	22.9	23.7
60	23.5	24.1
55	239.2(訊號遺失)	24.3
50	236.0(訊號遺失)	25.0
45		25.9
40		95.4(訊號遺失)



實驗二結論：

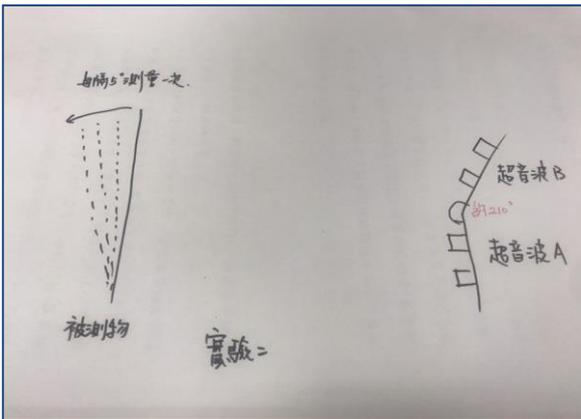
以上的實驗我們可以得知，如果兩個超音波以向中間夾擠、集中的方向來偵測障礙物時，當被偵測障礙物的偏移角度大到其中一個超音波已經喪失訊號，另一個超音波依然可以有效偵測障礙物。這代表了：

我們可以用兩個方向向中間集中的超音波測距模組來擴大超音波的有效偵測範圍。

實驗三：兩個超音波方向夾角約為 210 度，呈現向兩側擴散的方向

角度(度)	A 公分(cm)	B 公分(cm)
-------	----------	----------

90	20.8	207.5(訊號遺失)
85	22.1	209.2(訊號遺失)
80	22.7	208.4(訊號遺失)
75	23.6	217.1(訊號遺失)
70	24.7	24.7(訊號遺失)
65	26.9	214.4(訊號遺失)
60	242.0(訊號遺失)	208.0(訊號遺失)



實驗三結論：

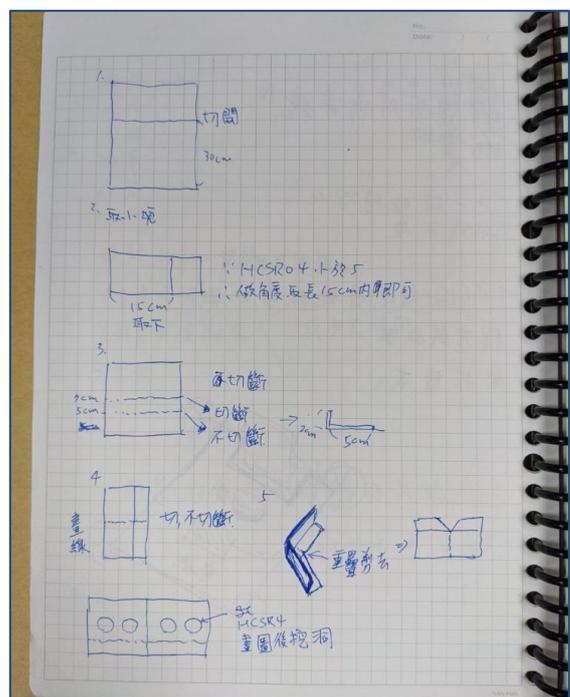
從實驗三的數據我們可以得知，如果兩個超音波以向兩側擴散的方向來偵測障礙物時，方向向外擴散的超音波一開始就遺失訊號，一直到另外一個超音波遺失訊號，方向向外擴散的超音波從頭至尾都沒有有效數值。這代表了：

我們無法用兩個方向向外擴散的超音波向來擴大超音波的有效偵測範圍。

實驗四：

我們為了讓超音波陣列雷達對平直面障礙物的偵測能力能夠到達偏轉 90 度也能做有效探測，為此我們設計一個超音波器開合角度的測試實驗。

1. 先依照左側的設計圖，將開合腳實驗支架用瓦楞紙製作出來。成品如下圖。



- 我們將兩個超音波從 150 度開始，距離約 25 公分，從正面開始實際偵測紙板，並以每隔 10 度開始偏轉，看看在 150 度的開合角，對平面紙板的最大有效偵測範圍可以達幾度的偏角？
- 然後調整超音波的角度，讓其夾角為 140 度，重複以上的實驗，並記錄數值。
- 持續上述步驟，每次都逐漸減少超音波的開合角度，直到紙板可以偏轉 90 度為止。



超音波不同開合角度有效偵測範圍之實驗數據如下表所示。

紙板偏轉	超音波開合角 150 度		超音波開合角 140 度		超音波開合角 130 度	
	超音波 A	超音波 B	超音波 A	超音波 B	超音波 A	超音波 B
0 度	24	22	24	24	22	21
10 度	21	20	19	20	19	20
20 度	20	20	19	21	28	24
30 度	18	20	18	20	17	154
40 度	17	34	15	40	16	153
50 度	21	40	17	44	14	26
60 度	1208	38	48	60	14	50
70 度					151	1197

紙板偏轉	超音波開合角 120 度		超音波開合角 110 度	
	超音波 A	超音波 B	超音波 A	超音波 B
0 度	24	23	22	55
10 度	20	23	25	24
20 度	22	42	26	17
30 度	16	23	26	16
40 度	14	19	24	16
50 度	13	25	48	10
60 度	15	30	30	85
70 度	55	54	30	1209



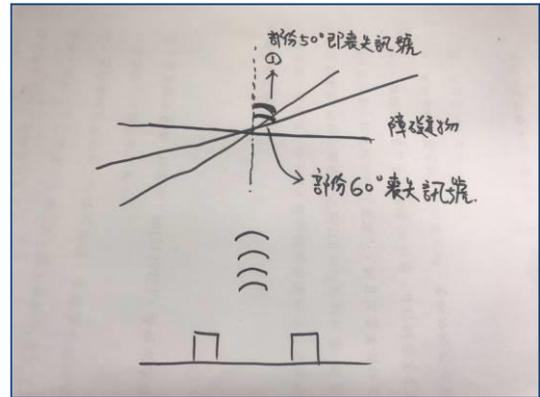
實驗四結論：

- 從以上的實驗結果顯示，兩個超音波開合角在 150 度時，對紙板的最大有效偵測範圍可以達到偏轉 50 度。
- 超音波開合角逐漸減小到 130 度時，對紙板的有效偵測範圍也逐漸增大到可以偏轉 60 度。
- 所以我們可以確定：超音波開合角在大於 90 度的狀態下，開合角越小，對紙板偏轉的有效偵測範圍會越大。

實驗一、二、三、四，階段結論：

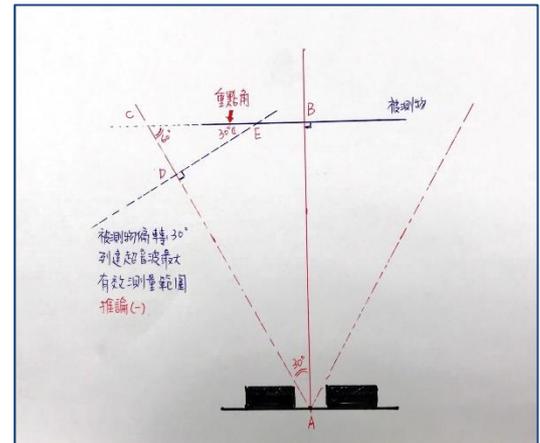
1. 單一個超音波對各種不同材質障礙物的有效偵測範圍從 60 度到 50 度不等，為了讓我們的雷達能適應各種不同的障礙物，我們以 60 度為標準。

2. 從兩個超音波不同方向的實驗數據可以知道，我們無法用兩個方向向外擴散的超音波測距模組來擴大超音波的有效偵測範圍，所以我們可以確定超音波方向向中間集中的方式優於超音波方向向外擴散的方式。

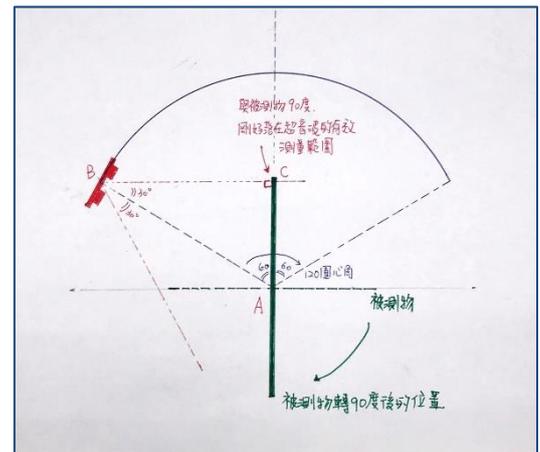


3. 超音波陣列最左側到最外側需要多大的圓心角，才能滿足一個平面障礙物即使轉了 90 度依然可以被有效偵測？

由第一點我們知道對單一個超音波感測器而言，其最大有效偵測範圍是 60 度，所以單邊有效範圍是 30 度。從△ABC 來看，∠A 是 30 度、∠B 是 90 度，∠C 可以算出是 60 度。



如果我們把被測物旋轉，一直到超音波的最大偵測範圍時，我們可以知道，∠D 是 90 度。從△CDE 來看，∠E 為 30 度。所以，被測物從原來的位置旋轉 30 度，是超音波感測器的最大有效偵測位置。



由右圖所示，一個圓弧的雷達陣列前有平面的被測物(綠色虛線)，我們希望當它旋轉 90 度(綠色粗線)時，依然在我們超音波的有效偵測範圍，所以∠C 為 90 度，因為超音波的單邊有效範圍是 30 度，所以∠A 為 60 度，因此我們的陣列雷達若要滿足需求(平面障礙物即使轉了 90 度依然可以被有效偵測)，需要把雷達建置成圓心角為 120 度圓弧狀。

假設我們的雷達版面長度為 40 公分

$$2 \times 3.14 \times \text{半徑} \times \frac{120}{360} = 40$$

$$\text{半徑} = 40 \times 3 \div 2 \div 3.14 = 60 \div 3.14 = 19.11$$

基於以上的結論，我們將利用珍珠板來完成一個半徑約為 20 公分，圓心角為 120 度內嵌 6x5 個超音波測距模組陣列的雷達系統。

當我們依據實驗結果，決定要完成一個 6x5 的超音波陣列雷達時，我們忽然想到一個問題：這麼多超音波擠在一起發射訊號，會不會互相干擾呢？會不會有 A 發 B 收的情形呢？

為了驗證我們的疑問，我們又設計了一個實驗來驗證我們的疑問。

實驗五：驗證 HC-SR04 能否 A 發 B 收

- 使用一個 Arduino UNO 控制板透過麵包板連接兩個 HC-SR04，其中一個只接 Trigger，另外一個只接 Echo。
- 使用一個 Arduino UNO 控制板透過麵包板連接兩個 HC-SR04，這兩個都正常接線，但程式只由其中一個 HC-SR04 發出 Trigger，再由另外一個 HC-SR04 接收 Echo。
- 使用一個 Arduino UNO 控制板透過麵包板連接兩個 HC-SR04，這兩個都正常接線，讓這兩個超音波測距模組背對背排列，分別針對相反方向來偵測，其中一個 HC-SR04 的正前方 20 公分放置障礙物，另一個 HC-SR04 正前方 50 公分放置障礙物，兩個超音波感測器都使用正常程式來執行。

程式碼

```
int trigger=7; //連接到超音波 1
int echo=8; //連接到超音波 2
digitalWrite(trigger,LOW);
delayMicroseconds(5);
digitalWrite(trigger,HIGH);
delayMicroseconds(10);
digitalWrite(trigger,LOW);
Serial.println(pulseIn(echo,HIGH)*34.0364/2/1000);
```

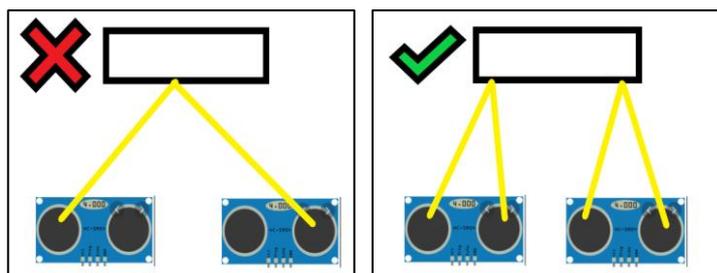
這段程式就是
A 發 B 收的程式碼

實驗結果：

A、B 完全讀數不到數值，C 則為各有各的數據，完全互不干擾。

當然除了實驗證明以外，老師在我們實驗後告訴我們，其實我們完全可以利用程式邏輯，就可以避免可能會互相干擾的情形。例如：只要我們循序執行，確保每一個

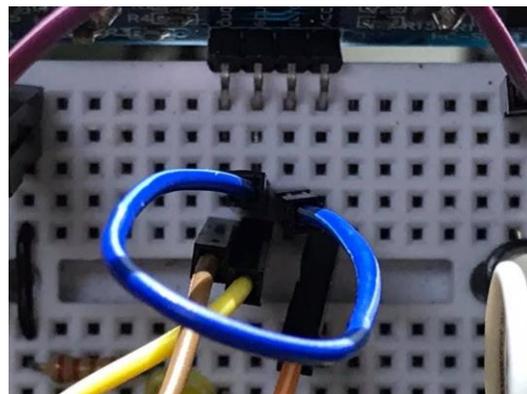
超音波都在 `pulseIn()` 結束以後才執行下一個超音波的掃描，這樣就可以確保每一個超音波的 Trigger 訊號不會互相干擾。



實作 6x5 超音波矩陣雷達並測試雷達之效能

實驗六：只用一個 IO 來控制 HC-SR04

由於 Arduino 控制板上只有 13 個 IO，因此我們希望能夠用更少的 IO 來控制 HC-SR04 超音波偵測器，我們查閱網路資料，發現有人實作出只用一個 IO 就可控制超音波，於是我們開始跟著實驗。



程式碼

```
pinMode(SIG,OUTPUT);  
digitalWrite(SIG,LOW);  
delayMicroseconds(5);  
digitalWrite(SIG,HIGH);  
delayMicroseconds(10);  
digitalWrite(SIG,LOW);  
pinMode(SIG,INPUT);  
duration=pulseIn(SIG,HIGH);  
distance=duration*34.0364/2/1000;
```

共用 IO 時，不論在輸入或是輸出前都需要先呼叫 `pinMode()` 來設定引腳的模式，才可以正常工作。

要使用一個 IO 來控制超音波測距模組其實只有一個技巧，那就是在 `trigger` 要送出訊號時，必須先呼叫 `pinMode(SIG,OUTPUT)`;將控制引腳設定為輸出狀態，然後在 `echo` 要偵測反射波時，也要再次呼叫 `pinMode(SIG,INPUT)`;將控制引腳設定為輸入狀態。

由於 `trigger` 與 `echo` 共用同一個引腳，所以我們無法使用以前寫程式的習慣，先在 `setup()` 中設定好，必須只能跟著程式需求，隨時改變引腳的狀態。

實驗七：控制 CD74HC4067 晶片

由於我們的超音波陣列會連接 30 個超音波測距模組，但不論是 Arduino UNO，或 Arduino Pro Mini 卻只有 13~14 個引腳，因此引腳明顯不足。所以我們會連接 2 個 CD74HC4067 的 IO 擴展晶片，這樣我們就可以用 10 個引腳來控制最多 32 個 HC-SR04 超音波測距器。

然而，為了要學習控制 CD74HC4067 晶片，我們翻找了網路上的說明文件(Google 搜尋 CD74HC4067 教學)，閱讀後我們對如何操作位址引腳的部分還是不明白，請教老師才知道這是二進位的概念。在我們學習了二進位的基礎概念後，我們才明白四個位址引腳可以相當於一個四位數的二進位數字，所以 `s0=0,s1=0,s2=0,s3=0` 時，可以對應到編號是 0 的通道，而 `s3=0,s2=1,s1=0,s0=1` 時，對應到的通道是編號是 5。這好像就是用另外一種方式來記錄 0~16，然而用這種讀寫方式是因為 Arduino 的引腳只能讀寫高電位(1)或低電位(0)。因此，二進位能算是個貼近電腦的真實記錄方式。

通道編號	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
S2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
S1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
S0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

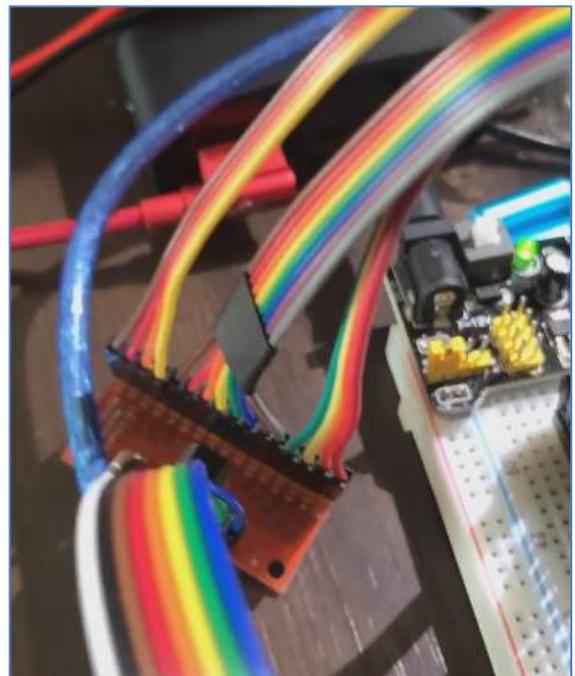
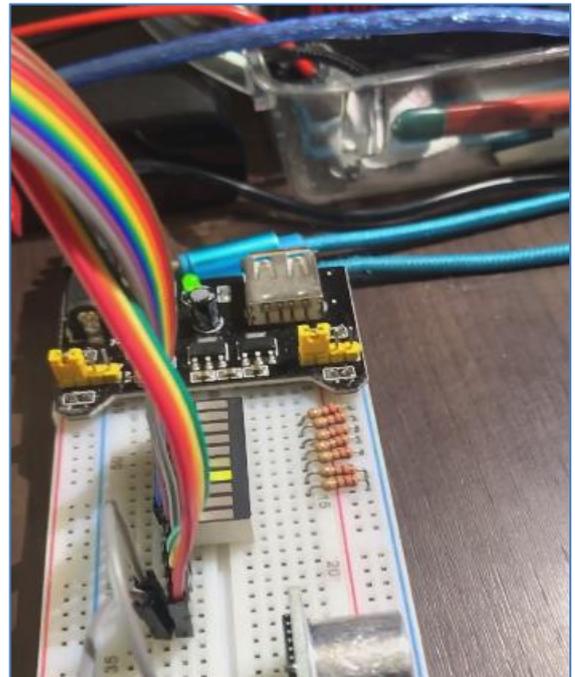
從一個 CD74HC4067 模組能連接 16 個設備我們知道，我們只要熟悉 0~15 的二進位記錄方式就可以寫程式了。這裡，我們使用一個 Arduino 控制板、一個 LED 燈排、數個 220Ω 的電阻、杜邦線等器材來進行實驗。

程式碼

```
int s0 = 7; //---- 位址引腳 0
int s1 = 6; //---- 位址引腳 1
int s2 = 5; //---- 位址引腳 2
int s3 = 4; //---- 位址引腳 3
int SIG_pin = 0; //---- Arduino 與
                  CD74HC4067 連接的通道引腳

void writeToMux(int channel)
{
    int controlPin[] = {s0, s1, s2, s3};

    int muxChannel[16][4] =
    {
        {0, 0, 0, 0}, //channel 0
        {1, 0, 0, 0}, //channel 1
        {0, 1, 0, 0}, //channel 2
        {1, 1, 0, 0}, //channel 3
        {0, 0, 1, 0}, //channel 4
        {1, 0, 1, 0}, //channel 5
        {0, 1, 1, 0}, //channel 6
        {1, 1, 1, 0}, //channel 7
        {0, 0, 0, 1}, //channel 8
        {1, 0, 0, 1}, //channel 9
        {0, 1, 0, 1}, //channel 10
        {1, 1, 0, 1}, //channel 11
        {0, 0, 1, 1}, //channel 12
        {1, 0, 1, 1}, //channel 13
        {0, 1, 1, 1}, //channel 14
        {1, 1, 1, 1} //channel 15
    };
};
```



```

//---- 利用寫入上面的值來設定我們將使用哪一個通道
for(int i = 0; i < 4; i ++)
{
    digitalWrite(controlPin, muxChannel[channel]);
}

//---- 給這個通道輸出高電位
digitalWrite(SIG_pin,HIGH);
}

```

在使用上面這段程式成功用 CD74HC4067 控制 LED 燈排之後，我們越看越覺得這段程式好像很醜，程式冗長所以編寫無效率。於是我們請教老師，老師提示我們可以去查閱 Arduino reference 中的 bitRead() 頁面，打開網頁，我們嚇了一跳，這好像是我們第一次閱讀英文文獻。閱讀後帶著不清不楚的心情再次請老師協助，老師用一個簡單的例子就直接讓我們馬上對 bitRead() 的用法馬上一請二處，於是，在老師的協助下，我們成功更改了程式碼。

程式碼

```

void writeToMux(int channel)
{
    //---- here is select what channel that we want to
    digitalWrite(s0,bitRead(channel,0)); //---- 設定通道 s0
    digitalWrite(s1,bitRead(channel,1)); //---- 設定通道 s1
    digitalWrite(s2,bitRead(channel,2)); //---- 設定通道 s2
    digitalWrite(s3,bitRead(channel,3)); //---- 設定通道 s3
    //---- 給這個通道輸出高電位
    digitalWrite(SIG_pin,HIGH);
}

```

果然程式的執行效果一模一樣，但，我們的程式精簡了很多。接下來，我們就需要實際寫出使用 CD74HC4067 晶片來控制 HC-SR04 超音波偵測模組。

實驗八：使用 CD74HC4067 晶片來控制 HC-SR04 超音波偵測模組

這裡，我們使用一個 Arduino 控制板、一個 CD74HC4067、兩個 HC-SR04 超音波偵測模組。杜邦線等器材來進行實驗。

程式碼

```

void sonarSelect(int channel)
{
    digitalWrite(s0,bitRead(channel,0)); //---- 設定通道 s0

```

```

        digitalWrite(s1,bitRead(channel,1)); //----- 設定通道 s1
        digitalWrite(s2,bitRead(channel,2)); //----- 設定通道 s2
        digitalWrite(s3,bitRead(channel,3)); //----- 設定通道 s3
    }

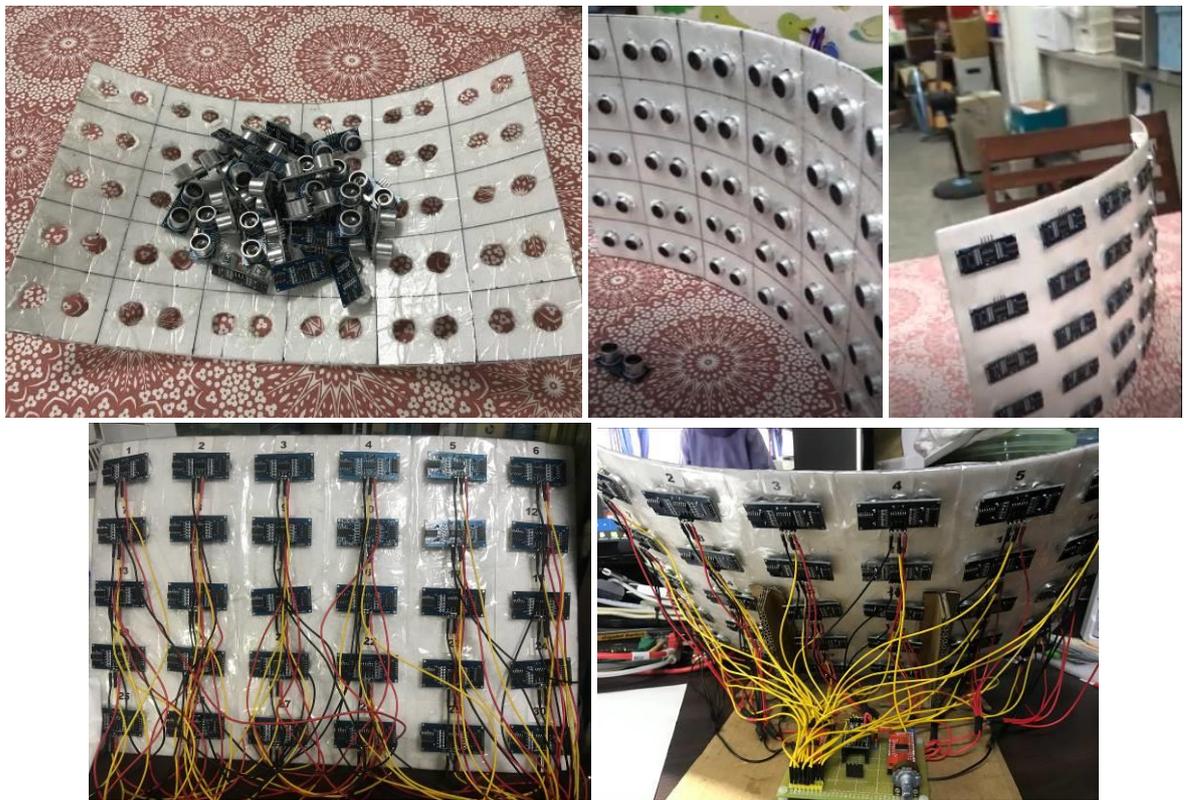
float getDistance()
{
    long duration;
    float distance;
    pinMode(SIG,OUTPUT); //----- 令 SIG 為 OUTPUT 接腳以發出超音波
    digitalWrite(SIG,LOW); //----- 歸零 Trigger 接腳
    delayMicroseconds(5); //----- 持續 5 微秒
    digitalWrite(SIG,HIGH); //----- 發出 Trigger 高電位信號
    delayMicroseconds(10); //----- 持續 10 微秒
    digitalWrite(SIG,LOW); //----- 歸零 Trigger 接腳
    pinMode(SIG,INPUT); //----- 令 SIG 為 INPUT 接腳，以偵測反射波
    duration=pulseIn(SIG,HIGH); //----- 使用 pulseIn 計算偵測到反射波的時間
    distance=duration*34.0364/2/1000; //----- 利用公式計算：距離＝速度×時間
    return distance;
}

```

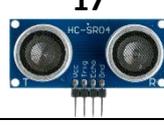
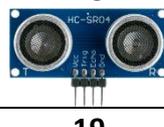
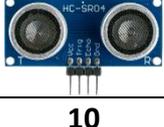
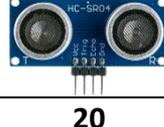
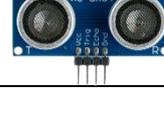
透過 Serial Port 輸出實驗結果，我們觀察到即便我們只使用單一個 IO 且經由控制 CD74HC4067 再二階段控制 HC-SR04 超音波測距模組，我們依然可以有正確的讀數，這代表我們超音波陣列雷達的基礎理論，都已經獲得成功驗證，所以可以進入最後的作品組裝與實驗的階段了。

製作、組裝過程：

1. 利用垂手可得的工具，如小刀、簽字筆、直尺等工具把我們所需要的雷達版面先切割出來。
2. 然後將小刀與剪刀搭配使用，在雷達版面上依序挖出 30 個超音波測距模組的安裝位置。
3. 利用老師在教室的教具-大型粉筆圓規，在黑板上畫出一個圓心角為 120 度的圓弧，然後將我們的雷達版面照著黑板上的圓弧，徒手慢慢朔形。
4. 朔形的過程需要許多的耐心與毅力，因為珍珠板容易龜裂，且朔形期間不易定型，往往沒有多少時間又回到原來的樣子。
5. 最後使用透明膠帶把整個朔形後的珍珠板密封起來，除了可以增加雷達版面的強度以外，也希望能透過透明膠帶的拉力，把整個雷達版面定型。
6. 完成雷達版面的朔型後，就可以開始將電纜線與訊號線連接到我們準備好的電路板。



超音波陣列雷達的位置編號：

 01	 06	 11	 16	 21	 26
 02	 07	 12	 17	 22	 27
 03	 08	 13	 18	 23	 28
 04	 09	 14	 19	 24	 29
 05	 10	 15	 20	 25	 30

上面的位置編號，是我們程式上的 sonarID，我們會利用 sonarID 來計算 CD74HC4067 的通道編號，這樣才能正確讓指定的超音波測距模組工作。

Arduino 主程式

```
void loop()
{
```

```

int sonarID=1;
int cid;

do
{
    cid=sonarSelect(sonarID); //----- 由 sonarID 設定 CD74HC4067 的通道

    display.clearDisplay();
    display.setTextColor(1);
    display.setTextSize(2); //----- 設定文字大小
    //----- 顯示超音波 ID
    display.setCursor(0,0);
    display.print("SonarID:");
    display.print(String(sonarID));
    //----- 顯示通道 ID
    display.setCursor(0,16);
    display.print(" CID:");
    display.print(String(cid));
    //----- 顯示通道 ID 的二進為表示法
    display.setCursor(0,32);
    display.print("BIN:");
    if(sonarID>15)
    {
        display.print("R-"); //----- 若 sonarID 大於 15 表示為右側 CD74HC4067
    }
    else
    {
        display.print("L-"); //----- 否則為左側 CD74HC4067
    }
    display.print(bitRead(cid,3)); //----- 顯示通道 ID 的 S3 (二進位值)
    display.print(bitRead(cid,2)); //----- 顯示通道 ID 的 S2 (二進位值)
    display.print(bitRead(cid,1)); //----- 顯示通道 ID 的 S1 (二進位值)
    display.print(bitRead(cid,0)); //----- 顯示通道 ID 的 S0 (二進位值)
    //----- 顯示超音波測量出來的距離
    display.setCursor(0,48);
    display.print("Dist=");
    display.print(getDistance()); //----- 取得偵測的距離數值
    //-----
    display.display();
    delay(500);
    //-----
    sonarID++;
}

```

```

    }
    while(sonarID<=30);
}

```

通道設定程式

```

int sonarSelect(int sid)
{
    if(sid<16) //---- 超音波編號 0~15 為左側多工控制器
    {
        SIG=LSIG;
        s0=Ls0;
        s1=Ls1;
        s2=Ls2;
        s3=Ls3;
    }
    else //---- 超音波編號 16~29 為右側多工控制器
    {
        sid=-15; //---- 對右側多工器而言超音波編號依然需要由 1 開始
        SIG=RSIG;
        s0=Rs0;
        s1=Rs1;
        s2=Rs2;
        s3=Rs3;
    }

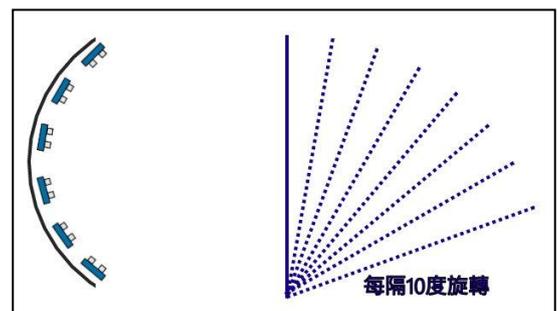
    sid=-1; //---- 將超音波編號轉為通道編號

    digitalWrite(s0,bitRead(sid,0)); //---- 設定通道 s0
    digitalWrite(s1,bitRead(sid,1)); //---- 設定通道 s1
    digitalWrite(s2,bitRead(sid,2)); //---- 設定通道 s2
    digitalWrite(s3,bitRead(sid,3)); //---- 設定通道 s3

    return sid;
}

```

實驗九：在超音波雷達前約 20 公分處，正面偵測紙板，然後依次將紙板分別以差距 10 度的間隔偏轉，然後再次利用超音波陣列雷達來偵測。



(一)正面實驗數據(單位：公分)

58.3	20.6	20.3	20.4	20.5	59.1
------	------	------	------	------	------

57.6	20.4	20.5	20.7	20.6	58.5
58.5	20.3	20.4	20.8	20.9	58.3
58.4	20.5	20.3	20.3	20.8	58.2
58.3	20.5	20.7	20.5	20.7	58.7

兩側最外的超音波測距模組都遺失訊號(紅色數字)

(二)正面偏轉 10 度實驗數據(單位：公分)

28.7	28.1	26.7	25.3	80.8	93.3
28.5	27.9	26.6	25.5	80.9	93.6
28.9	28.2	26.5	25.4	79.7	93.4
29.1	27.8	26.4	25.8	80.5	93.0
28.4	27.7	26.7	25.6	80.1	93.4

靠右側的兩排超音波測距模組都遺失訊號(紅色數字)

(三)正面偏轉 20 度實驗數據(單位：公分)

35.4	34.6	33.1	31.7	84.9	95.4
35.1	34.3	33.8	31.5	84.9	95.2
35.9	34.7	33.2	31.2	84.4	96.1
34.8	34.8	33.5	30.8	84.0	95.8
35.2	34.3	32.7	30.4	84.5	95.7

靠右側的兩排超音波測距模組都遺失訊號(紅色數字)

(四)正面偏轉 30 度實驗數據(單位：公分)

41.3	39.2	75.4	77.9	89.4	97.4
41.2	39.7	74.8	77.3	88.6	97.3
41.9	39.2	74.2	77.8	88.3	97.9
40.7	39.4	75.3	78.4	89.1	98.3
40.8	38.8	75.6	78.6	89.6	97.1

靠右側的四排超音波測距模組都遺失訊號(紅色數字)

(五)正面偏轉 40 度實驗數據(單位：公分)

47.5	45.6	79.8	83.6	93.2	100.7
47.3	45.2	78.2	83.7	93.8	101.2
48.1	45.9	78.6	83.2	93.4	101.5
48.3	45.3	78.4	83.3	93.5	101.2
48.0	45.5	78.7	83.3	93.6	101.3

靠右側的四排超音波測距模組都遺失訊號(紅色數字)

(六)正面偏 50 度實驗數據(單位：公分)

51.6	75.4	78.8	88.6	97.5	105.8
51.3	76.1	78.7	88.5	97.3	105.8
52.4	76.0	78.7	88.9	97.1	106.4
52.1	75.8	78.4	89.3	98.2	106.5
51.9	75.5	78.1	89.5	97.8	106.2

靠右側的五排超音波測距模組都遺失訊號(紅色數字)

(七)正面偏轉 60 度實驗數據(單位：公分)

55.7	82.5	84.4	96.7	103.4	112.7
55.6	82.8	84.5	96.1	104.1	111.9
55.4	82.4	84.3	95.6	103.7	112.8
54.8	82.6	84.7	95.9	103.6	112.3
55.2	83.0	84.9	95.3	103.8	112.5

靠右側的五排超音波測距模組都遺失訊號(紅色數字)

(八)正面偏轉 70 度實驗數據(單位：公分)

84.9	89.9	96.5	107.4	111.1	112.5
84.3	89.9	96.3	107.1	111.9	111.8
83.8	90.7	95.8	107.3	111.3	112.3
83.9	90.3	96.1	106.7	110.7	112.2
84.0	89.5	96.4	107.5	110.9	112.6

全部的超音波測距模組都遺失訊號(紅色數字)

實驗九結論：

從實驗九的數據我們可以得知，利用超音波測距模組陣列來偵測平面，的確可以把超音波的有效偵測範圍擴大。相對於單一個超音波測距模組而言，實驗的結果表明了我們將平面的有效偵測範圍直接擴大了 30~40 度，大約一倍的範圍。

伍、研究成果

一、研究項目及內容

由於我們希望將 30 個超音波測距模組組合並製作成一個 6×5 超音波陣列雷達，以實現對類似隱形戰機外形之平滑物進行偵測，所以針對我們的目標，我們訂定了以下的研究內容與過程，並希望經由這些研究內容與結果，可以建構出我們的理論基礎。

1. 研究超音波測距模組的功能。
(如何控制單個超音波測距模組與同時連接兩個超音波測距模組之研究)
2. 研究超音波測距模組對於不同材質之平面物在不同角度下的反射效果。
3. 研究兩個超音波測距模組在不同開合角度時對於偵測效果的影響。
4. 實作一個 6×5 的超音波矩陣雷達，並使用該達對各種物體進行效能測試及實際應用。

二、研究成果說明

研究超音波測距模組的功能

1. 這個部分，我們學習到如何以 Arduino IDE 撰寫程式來控制 HC-SR04 超音波測距模組。學習完控制單一個超音波模組後，我們開始嘗試以程式來同時控制兩個超音波模組，來成為整個作品的基礎能力。
2. 我們透過實驗輸出的數據發現，每一個超音波測距模組所測量出來的結果，都有些許的誤差。我們猜測，這可能是由超音波測距模組的製造過程中，沒有非常精細所造成的。

研究超音波測距模組對於不同材質之平面物在不同角度下的反射效果

單一個超音波對各種不同材質障礙物的有效偵測範圍從 60 度到 50 度不等，為了讓我們的雷達能適應各種不同的障礙物，我們以 60 度為標準。

研究兩個超音波測距模組在不同開合角度時對於偵測效果的影響

1. 從兩個超音波不同方向的實驗數據可以知道，我們無法用兩個方向向外擴散的超音波測距模組來擴大超音波的有效偵測範圍，所以我們可以確定超音波方向向中間集中的方式優於超音波方向向外擴散的方式。
2. 老師指導我們從實驗的數據分析，顯示超音波陣列從最左側到最右側需要約 120 度的圓心角，才能滿足一個平面障礙物即使轉了 90 度依然可以被有效偵測。因此我們最決定利用珍珠板，來完成一個 120 度圓心角內嵌 6×5 個超音波測距模組陣列的雷達系統。
3. 在決定要製作一個 120 度圓心角內嵌 6×5 個超音波測距模組陣列的雷達系統後，我們忽然想起一個問題：這麼多超音波擠在一起發射訊號，會不會互相干擾呢？會不會有 A 發 B 收的情形呢？於是我們又透過實驗得到超音波測距模組即使在密集模式下工作，也完全不會互相干擾。
4. 除了實驗證明多個超音波在一個範圍以內密集的同時工作也不會互相干擾，老師也提示我們，其實也可以利用程式邏輯來避免可能會發生互相干擾的情形。例如：只要我們循序執行，確保每一個超音波都在 `pauseIn()` 結束以後才執行下一個超音波的掃描，這樣就可以確保每一個超音波的 Trigger 訊號不會互相干擾。

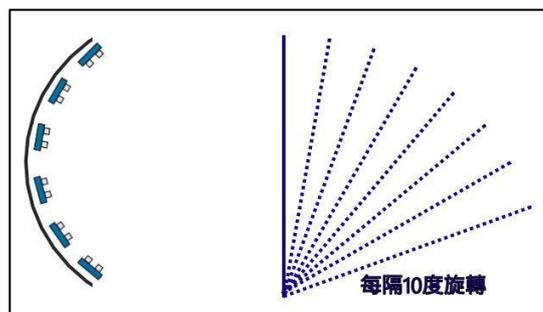
實作一個 6×5 的超音波矩陣雷達，並使用雷達對平直面物體進行效能測試

1. 首先，我們因為 Arduino 先天的物理限制，因此我們必須想辦法節省 IO，將原來需要兩個 IO 才能控制一個 HC-SR04，利用程式把它變成只需要一個 IO 就可以控制。經由這個步驟的學習與實驗，我們的 Arduino 控制板在操作 6×5 超音波測距模組陣列雷達系統時，就可以由原來需要 60 個 IO 降到只需要 30 個 IO。
2. 我們使用的 Arduino 控制板只有 13 個 IO，距離 30 個 IO 依然非常的遙遠，經過我們不懈不怠的在網路上搜尋資料後，我們發現 CD74HC4067 是我們的解藥。
3. 要控制 CD74HC4067，我們需要具備二進位的觀念，透過請教老師，我們了解了二進位的基本概念。
4. 當完成作品後，我們針對超音波陣列雷達對平面的偵測效果做了一系列的實驗，並與當初我們在建構理論基礎時的實驗結果作比對，我們真的可以確認了超音波陣列雷達的正面效果：

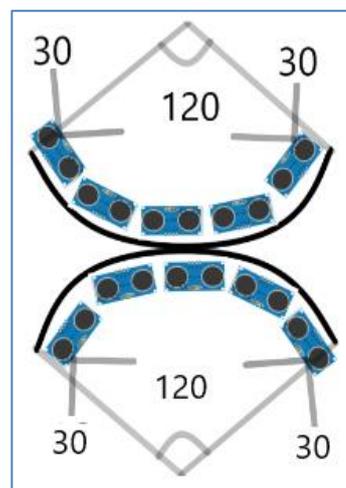
利用超音波測距模組陣列來偵測平面，的確可以把超音波的有效偵測範圍擴大。相對於單一個超音波測距模組而言，實驗的結果表明了，我們可以將平面的有效測範圍直接擴大了 30~40 度，大約一倍的範圍。這個就是我們整個專題的最重要的成果。

陸、討論與結論

1. 在 Arduino 的學習歷程上，我們利用許多時間學習相關知識。例如，我們在四年級時就開始學習 Arduino 的積木程式，我們也在寒暑假時參加一些營隊活動，同時也參加許多智能車的相關課程，這些課程都跟 Arduino 有相關，所以老師在指導我們使用 IDE 設計程式時，我們才能這麼快的進入狀況。
2. 我們使用超音波來實作雷達，是因為聲波與雷達的電磁波都有波的性質，所以我們才會假設用聲波來模擬。現實中我們不可能使用聲波來實作雷達，因為現在飛機的飛行速度都可以輕鬆超過一倍音速、甚至於二倍音速，如果當真使用聲波來實作雷達，那就會有雷達波追不到飛機的現象，根本無法發現敵軍。
3. 我們在實作的過程中發現，電子元件在的安裝與連接電路的過程中，一定要非常確實，以避免組裝系統時因接觸不良而形成斷路，造成程式與電子元件的互動沒有辦法出現我們預期的成果，所以無法察覺錯誤而讓自己懷疑人生。
4. 6x5 超音波陣列，這個名字看起來就非常的專業與科學，但其實很具體，非常容易想像，所以我們很容易了解。但是，當老師在指導我們學習 Arduino 的整數陣列(例如：`int muxChannel[16][4]`)的資料形態時，這的太抽象了，我們確實完全無法理解這是什麼，一直到當我們在老師的指導下，把完整的超音波雷達組裝完成，我們才可以想像 `int muxChannel[16][4]` 這個資料在物理上形態是什麼樣子。
5. 作品完成後的實驗數據顯示，我們的超音波陣列雷達並沒有像我們當初設計時的初衷一樣，可以完整偵測一個旋轉 90 度的平面。究其數據與作品外形後我們歸納可能會造成這種結果的因素是：珍珠板太軟了，造成超音波陣列雷達的骨架被外力(例如：30 個超音波測距模組的重量)影響而造成形變，致使原期望 120 度圓心角的超音波雷達陣列根本沒有 120 度。
6. 在超音波陣列雷達的實驗中，我們只有對正面受測的紙板以同一個方向旋轉不同角度的實驗。我們的作品無法適用全方位 360 度的任意方向旋轉的實驗，因為我們的超音波陣列雷達只實作了一個平面的 120 度圓心角。因此若將我們的超音波陣列雷達實作成一個球面，也許就可以適用一個方向全方位 360 度的任意方向的旋轉。這個目標可以做為我們未來改進的主要方向。



7. 如果我們把兩個超音波陣列雷達背對背安裝在一起，如此一來，我們也可以實現同時能雙向偵測的雷達系統。這種作法初步估計至少需要 6 張 CD74HC4067 模組來實作：以 Arduino 分別控制兩張 CD74HC4067 模組作為第一層，再由第一層的 CD74HC4067 模組分別控制兩張 CD74HC4067 模組來存取 30 個超音波陣列雷達。我們認為這樣的程式撰寫一定很有挑戰性，希望有機會可以嘗試。



柒、參考資料及其他

1. 為甚麼隱形戰機可以隱形
<http://www.hk-phy.org/iq/stealth/stealth.html>
2. 隱形戰機：隱形技術原理在實際中如何運作
<https://kknews.cc/zh-tw/military/xl5nzeg.html>
3. 這些技術幫助隱形飛機蒙蔽了雷達的「眼睛」
<https://kknews.cc/military/yxpp8b.html>
4. 聲波與電磁波有何差別
<https://zhidao.baidu.com/question/680086086798036972.html>
5. 主動式電子掃描雷達(主動式相控雷達)
<https://zh.wikipedia.org/wiki/%E9%9B%BB%E5%AD%90%E6%8E%83%E6%8F%8F%E9%9%A3%E5%88%97%E9%9B%B7%E9%81%94>
6. 雷達系統基本原理
<https://ir.nctu.edu.tw/bitstream/11536/37935/4/250904.pdf>
7. CD74HC4067 16 通道數位/類比分線轉換模組
<https://www.taiwaniot.com.tw/product/analogdigital-mux-breakout-cd74hc4067/>
<https://www.taiwansensor.com.tw/product/sparkfun-analog-digital-mux-breakout-cd74hc4067-16-通道-數位-類比分線轉換模組/>
8. HC-SR04 Ultrasonic module interface with Arduino
<https://beatyourbit.com/arduinointerface/hc-sr04/>
9. Arduino 運用 SSD1306 OLED
<https://blog.jmaker.com.tw/arduino-ssd1306-oled/>
10. 科學 online-反射(Reflection)
<https://highscope.ch.ntu.edu.tw/wordpress/?p=19786>
11. LED 與電阻的必學之術！LED & resistors 101！
<https://www.youtube.com/watch?v=cWEJMusT-hI>
12. Arduino Reference – bitRead()
<https://www.arduino.cc/reference/en/language/functions/bits-and-bytes/bitread/>
13. Arduino Reference – do ... while
<https://www.arduino.cc/reference/en/language/structure/control-structure/dowhile/>