

# 紅外線循跡 自走車實習

# 4

- 4-1 認識紅外線
- 4-2 認識紅外線循跡模組
- 4-3 認識紅外線循跡自走車
- 4-4 自造紅外線循跡自走車



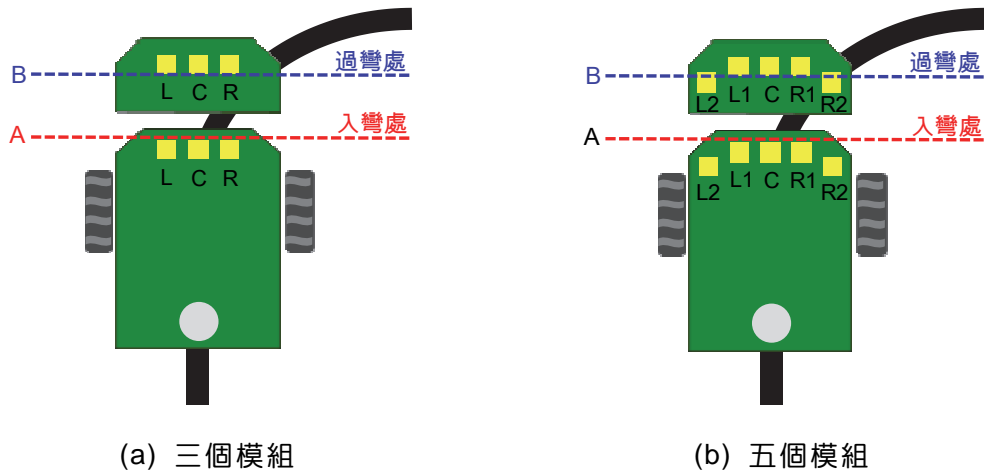


圖 4-10 紅外線模組數目

如圖 4-10(a)所示使用三個紅外線模組，自走車進入軌道 A 點入彎處，紅外線感應到轉彎軌道，回傳至微控制器驅動左、右輪馬達使自走車右轉。但若車速太快，紅外線模組將會來不及感應，自走車直線前進至軌道 B 點過彎處而衝出軌道，無法順利轉彎。三組紅外線循跡模組的優點是**成本低**，缺點是**車速慢**。

如圖 4-10(b)所示使用五個紅外線模組，自走車進入軌道 A 點入彎處，紅外線感應到轉彎軌道，回傳至微控制器驅動左、右輪馬達使自走車右轉。但若車速太快，R1 紅外線模組將會來不及感應，自走車直線前進至軌道 B 點過彎處，R2 紅外線模組仍可感應到轉彎軌道，使自走車能順利轉彎。五組紅外線循跡模組的優點是**車速快**，缺點是**成本高**。

#### 4-2-5 紅外線模組排列間距

紅外線模組的排列間距會影響自走車轉彎的準確度，如圖 4-11(a)所示模組的間距太小時，雖然在軌道 A 點入彎處就能感應到軌道轉彎路徑，但若車速太快、彎角太小，很容易衝出軌道，而且模組間距太小也容易相互干擾，造成誤動作。

如圖 4-11(b)所示模組的間距太大時，直到軌道 B 點過彎處才能感應到轉彎路徑，但反應時間過短，自走車很容易衝出軌道。循跡自走車競賽的軌道大多選用 1.9 公分寬的黑色或白色電工膠帶，因此紅外線循跡模組的排列間距只要大於 1.9/2 公分即可，**建議值為 1.5~2 公分**。

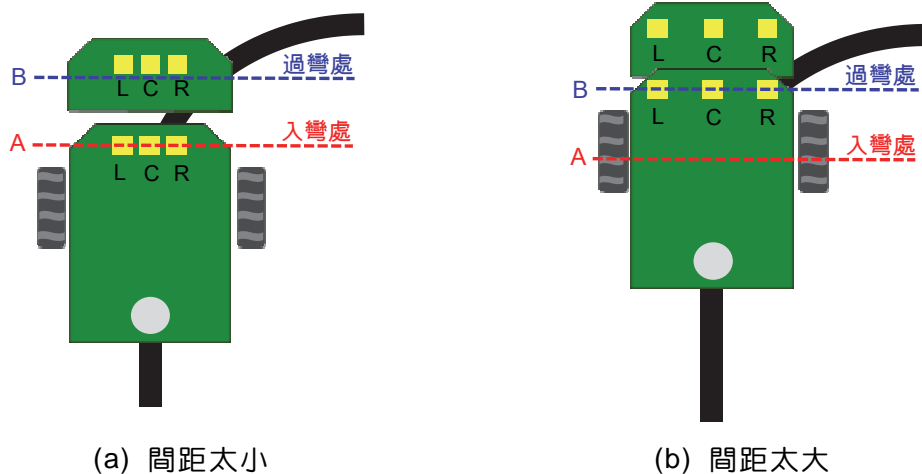


圖 4-11 紅外線模組排列間距

### 4-3 認識紅外線循跡自走車

所謂紅外線循跡自走車（line-following robot）是指自走車可以自動運行在預先規畫的黑色軌道上。其工作原理是利用紅外線發射器發射紅外線訊號至地面軌道，經由紅外線受光電晶體感應反射光的強弱並且轉換成電壓值。經由微控制器比較並修正自走車的行進方向，使自走車能自動運行在軌道上。不同顏色對光的反射程度不同，**黑色吸光反射率最低**，模組輸出高電位（邏輯 1），**白色反光反射率最高**，模組輸出低電位（邏輯 0）。如表 4-3 所示為使用三個紅外線循跡模組的紅外線循跡自走車運行方向的控制策略，其運行情形說明如下：

表 4-3 紅外線循跡自走車運行方向的控制策略

紅外線模組 L	紅外線模組 C	紅外線模組 R	控制策略	左輪	右輪
0	0	0	前進	反轉	正轉
0	0	1	快速右轉	反轉	反轉
0	1	0	前進	反轉	正轉
0	1	1	慢速右轉	反轉	停止
1	0	0	快速左轉	正轉	正轉
1	0	1	不會發生	停止	停止
1	1	0	慢速左轉	停止	正轉
1	1	1	停止	停止	停止



如圖 4-12 所示為紅外線循跡自走車的運行情形，使用左（left，簡記 L）、中（center，簡記 C）、右（right，簡記 R）三組紅外線模組。當紅外線模組感應到黑色軌道時，黑色吸光不反射，紅外線模組輸出高電位（high potential，簡記 H 或邏輯 1）。反之，當紅外線模組沒有感應到黑色軌道時，會有一定程度的反射，紅外線輸出低電位（low potential，簡記 L 或邏輯 0）。

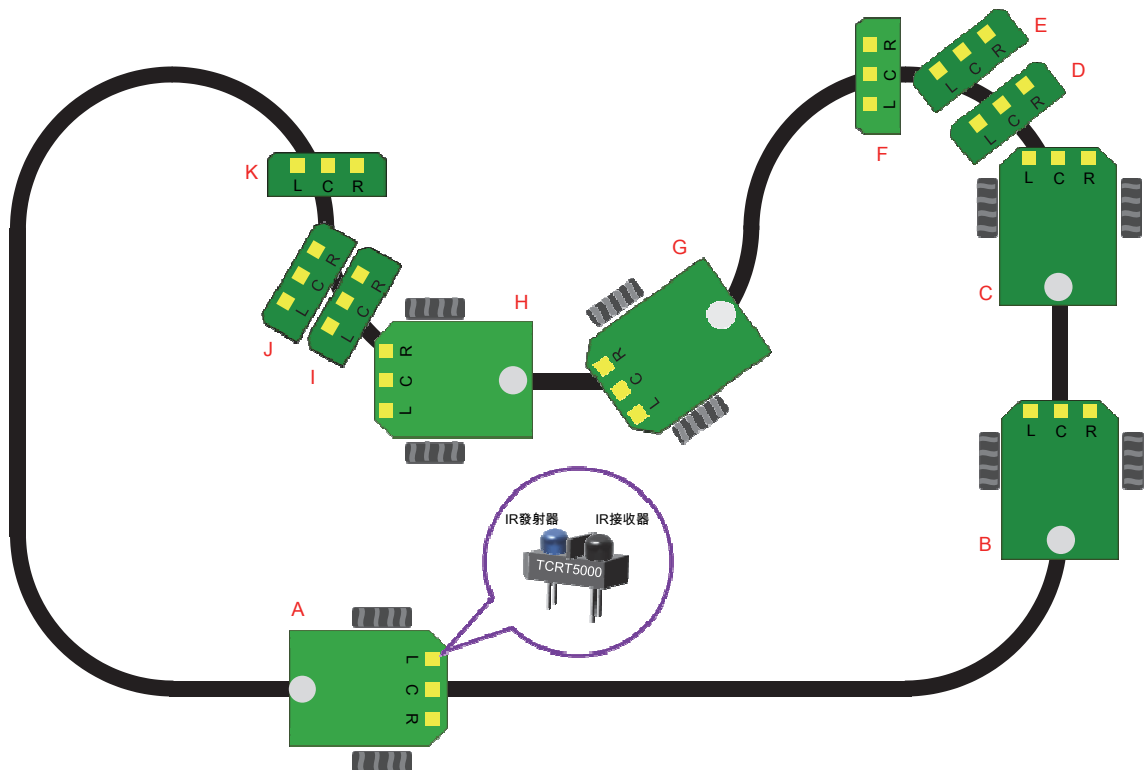


圖 4-12 紅外線循跡自走車的運行情形

當自走車行進至位置 A 及 B 時，模組 LCR 狀態為 010，自走車繼續**前進**。當自走車行進至位置 C 時，模組 LCR 狀態為 110，自走車偏離在軌道右方，必須**左轉彎**。當自走車行進至位置 E 時，模組 LCR 狀態為 100，自走車嚴重偏離在軌道右方，必須**快速左轉彎**修正運行路線，否則自走車會衝出軌道。當自走車行進至位置 G 時，模組 LCR 狀態為 011，自走車偏離在軌道左方，必須**右轉彎**。當自走車行進至位置 H 時，模組 LCR 狀態為 001，自走車嚴重偏離在軌道左方，必須**快速右轉彎**修正運行路線，否則自走車會衝出軌道。



## 4-4 自造紅外線循跡自走車

如圖 4-13 所示紅外線循跡自走車電路接線圖，包含紅外線循跡模組、Arduino 控制板、馬達驅動模組、馬達組件及電源電路等五個部份。

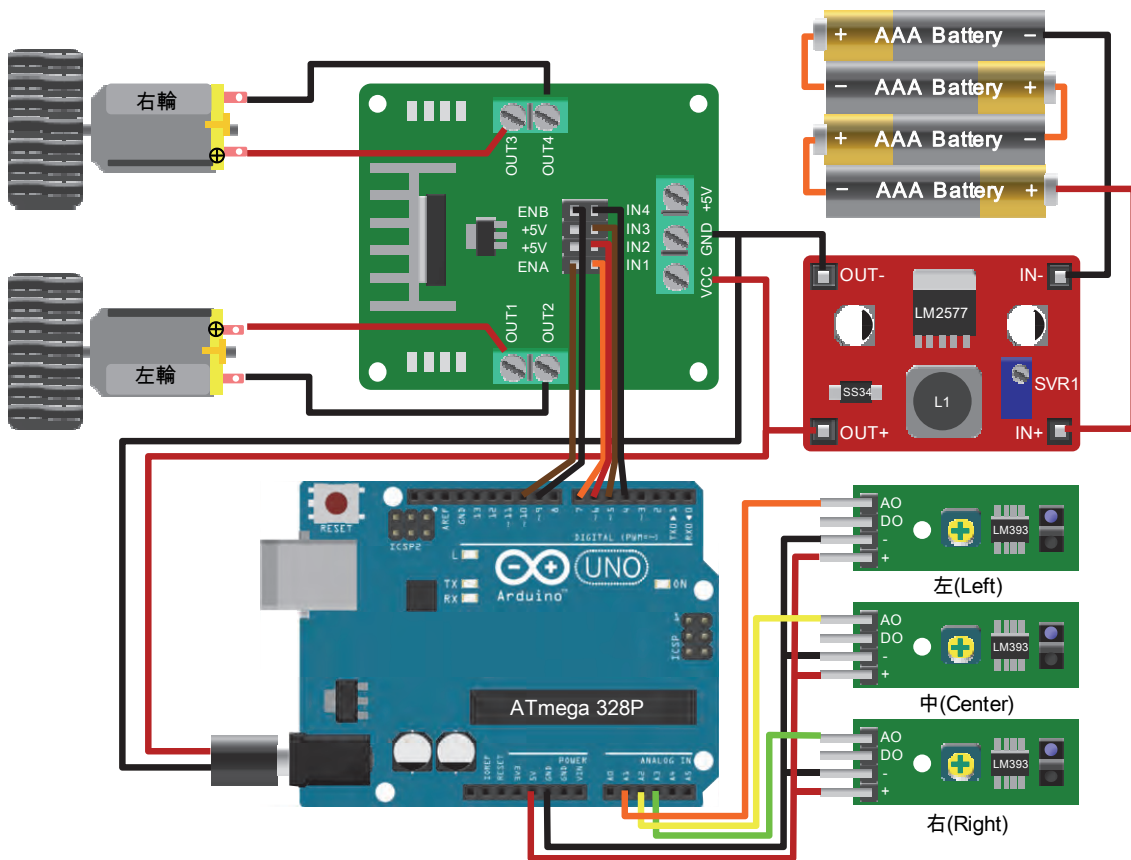


圖 4-13 紅外線循跡自走車電路接線圖

### 紅外線循跡模組

使用如圖 4-9(a)所示四線式紅外線循跡模組，將左、中、右等三組類位輸出 AO 值分別連接至類比輸入 A1、A2、A3 等接腳，由 Arduino 板的+5V 供電給紅外線循跡模組，再以 `analogRead(Pin)`指令來讀取狀態，其中 Pin 為類比輸入 A0~A5。如果使用如圖 4-8(a)所示三線式紅外線循跡模組只有數位輸出 OUT，必須以 `digitalRead(Pin)`指令來讀取狀態，其中 Pin 為數位輸入 0~13。如果為了節省成本，也可以依圖 4-7 所示紅外線感測電路圖自行製作並組合所需要的模組數目。





## Arduino 控制板

Arduino 控制板為控制中心，檢測左、中、右等三組紅外線循跡模組的類比輸出 AO 值，並依表 4-3 所示紅外線循跡自走車運行方向的控制策略，來驅動左、右兩組減速直流馬達的運轉方向，使車子能正確行進在軌道上。

## 馬達驅動模組

馬達驅動模組使用 L298 驅動 IC 來控制兩組減速直流馬達，其中 IN1、IN2 輸入訊號控制左輪轉向，而 IN3、IN4 輸入訊號控制右輪轉向。另外，Arduino 控制板輸出兩組 PWM 訊號連接至馬達驅動模組的 ENA 及 ENB 腳，分別控制左輪及右輪的轉速。因為馬達有最小的啟動轉矩電壓，所輸出的 PWM 訊號平均值不可太小，以免無法驅動馬達轉動。PWM 訊號只能微調馬達轉速，如果需要較低的轉速，可以改用較大減速比的減速直流馬達。

## 馬達組件

馬達組件包含兩組 300rpm/min（測試條件：6V）的金屬減速直流馬達、兩個固定座、兩個 D 型接頭 43mm 橡皮車輪及一個萬向輪，橡皮材質輪子比塑膠材質磨擦力大而且控制容易。

## 電源電路

電源模組包含四個 1.5V 一次電池或四個 1.2V 充電電池及 DC-DC 升壓模組，調整 DC-DC 升壓模組中的 SVR1 可變電阻，使輸出升壓至 9V，再將其連接供電給 Arduino 控制板及馬達驅動模組。如果是使用兩個 3.7V 的 18650 鋰電池，可以不用再使用 DC-DC 升壓模組。每個容量 2000mAh 的 1.2V 鎳氫電池約 90 元，每個容量 3000mAh 的 18650 鋰電池約 250 元。

### □ 功能說明：

使紅外線循跡自走車能夠自動運行在預先規畫的**黑色軌道**上。請依實際狀況調整自走車的行進速度，以免車速過快而衝出軌道。一般地面顏色都不是與黑色軌道對比強烈的白色，會降低紅外線的反射率，使低準位輸出電壓過高。可以利用調整比較電壓值來提高感應靈敏度，以免產生誤動作。

如果是使用三線式紅外線模組或四線式紅外線模組的 DO 輸出，必須連接至 Arduino 板的數位輸入腳，並且調整紅外線模組上的可變電阻改變比較器電



壓，以提高紅外線循跡自走車對軌道的感應靈敏度。如果是使用四線式紅外線模組的 AO 輸出，必須連接至 Arduino 板的類比輸入腳，並且調整 analogRead() 函式所讀取轉換的比較值，以提高紅外線循跡自走車對軌道的感應靈敏度。



#### 程式：ch4\_1.ino

```
const int negR=4;           //右輪馬達負極。
const int posR=5;           //右輪馬達正極。
const int negL=6;           //左輪馬達負極。
const int posL=7;           //左輪馬達正極。
const int pwmR=9;           //右輪馬達轉速控制。
const int pwmL=10;          //左輪馬達轉速控制。
const int irD1=A1;          //左(left)紅外線循跡模組。
const int irD2=A2;          //中(center)紅外線循跡模組。
const int irD3=A3;          //右(right)紅外線循跡模組。
const int Rspeed=200;       //右馬達轉速控制初值。
const int Lspeed=200;       //左馬達轉速控制初值。
byte IRstatus=0;            //紅外線循跡模組感應值。
//初值設定
void setup()
{
    pinMode(negR,OUTPUT);    //設定數位腳 4 為輸出腳。
    pinMode(posR,OUTPUT);    //設定數位腳 5 為輸出腳。
    pinMode(negL,OUTPUT);    //設定數位腳 6 為輸出腳。
    pinMode(posL,OUTPUT);    //設定數位腳 7 為輸出腳。
    pinMode(irD1,INPUT_PULLUP); //設定類比腳 A1 為含提升電阻的輸入腳。
    pinMode(irD2,INPUT_PULLUP); //設定類比腳 A2 為含提升電阻的輸入腳。
    pinMode(irD3,INPUT_PULLUP); //設定類比腳 A3 為含提升電阻的輸入腳。
}
//主迴圈
void loop()
{
    int val;                 //輸入類比信號值。
    IRstatus=0;              //清除紅外線循跡模組感應值。
    val=analogRead(irD1);    //讀取「左 L」紅外線循跡模組感應值。
    if(val>=150)              //感應到黑色軌道?
        IRstatus=(IRstatus+4); //設定感應值位元 2 為 1。
    val=analogRead(irD2);    //讀取「中 C」紅外線循跡模組感應值。
```



## CHAPTER

# 超音波避障 自走車實習

# 10

- 10-1 認識超音波
- 10-2 認識超音波模組
- 10-3 認識超音波避障自走車
- 10-4 自造超音波避障自走車







## 10-1 認識超音波

聲音是一種**波動**，聲音的振動會引起空氣分子有節奏的振動，使周圍的空氣產生疏密變化，形成疏密相間的縱波，因而產生了聲波，人耳可以聽到的聲音頻率範圍在 20Hz~20kHz 之間。

所謂超音波（ultrasound）是指任何聲波或振動，其頻率超過人耳可以聽到的範圍。若超音波的頻率太低則雜音增加；反之若超音波的頻率太高則衰減增加，會降低可到達的距離，在可以測量的距離範圍內，應儘可能提高測量頻率，才能準確測量反射波，以得到較高的距離解析度。一般常用的**超音波頻率範圍在 20kHz~40kHz 之間**。超音波直線發射出去後，會不斷擴大而造成擴散損失，距離愈遠則損失愈大。另外，部份超音波會被傳播介質吸收而造成波動能力損失。一般超音波可以使用的測量距離在 10 公尺以內，常用的**超音波模組最大測量距離以 2~5 公尺居多**。

超音波測距電路是利用超音波模組來測量物體的距離，其工作原理是利用超音波發射器向待測距的物體發射超音波，並且在發射的同時開始計時。超音波在空氣中傳播，遇到障礙物後就會被反射回來，當超音波接收器接收到反射波時即停止計時，此時所測得的時間差即為超音波模組與物體之間的來回時間  $t$ 。因為超音波在空氣中的傳播速度大約為  $v=340$  公尺/秒，所以超音波模組與物體間的距離  $s$  等於  $vt / 2$  公尺。

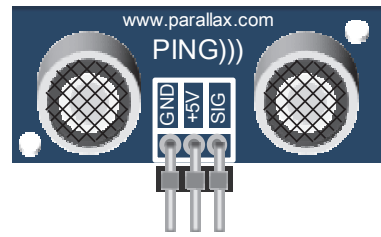
**超音波的應用相當廣泛，在海洋方面，如超音波聲納、魚群探勘、海底探勘等。在醫療設備如超音波熱療、超音波影像掃描、超音波碎石機等。在訊號感測上如超音波壓力感測、超音波膜厚感測、超波震動感測。在工業加工方面如超音波金屬焊接、超音波洗淨機、超音波霧化器等。**

## 10-2 認識超音波模組

如圖 10-1 所示為 Parallax 公司所生產的 PING)))™ 超音波模組（#28015），有 SIG、+5V、GND 等三支腳，工作電壓+5V，工作電流 30mA，工作溫度範圍 0~70°C。PING)))™ 超音波模組的**有效測量距離在 2 公分到 3 公尺之間**。當物體在 0 公分到 2 公分的範圍內時無法測量，傳回值皆為 2 公分。PING)))™ 超音波模組具有 TTL/CMOS 介面，可以直接使用 Arduino 控制板來控制。



(a) 模組外觀

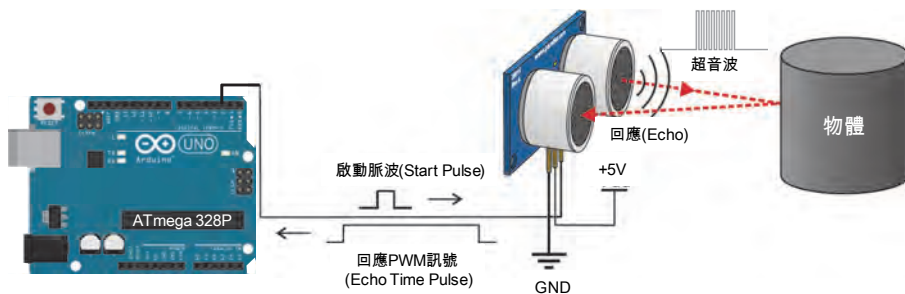


(b) 接腳圖

圖 10-1 PING)))™ 超音波模組

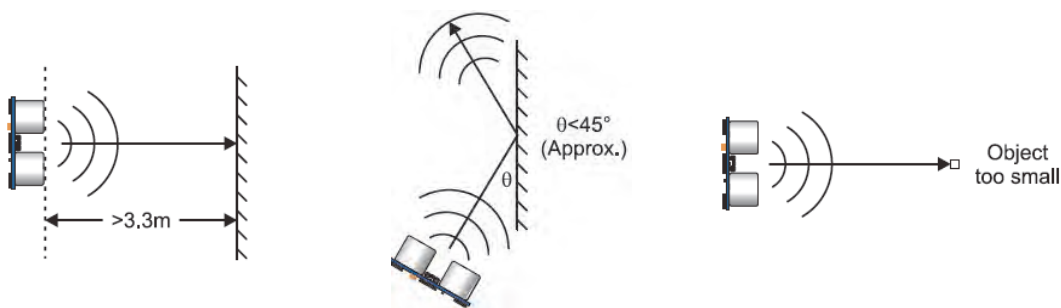
### 10-2-1 工作原理

如圖 10-2 所示 PING)))™ 超音波模組的工作原理，首先 Arduino 控制板必須先產生至少維持 2 微秒（典型值 5 微秒）高電位的啟動脈波至 PING)))™ 超音波模組的 SIG 腳，當超音波模組接收到啟動脈波後，會發射  $200\mu\text{s}@40\text{kHz}$  的超音波訊號至物體，所謂  $200\mu\text{s}@40\text{kHz}$  是指頻率為 40kHz 的脈波連續發射  $200\mu\text{s}$ 。當超音波訊號經由物體反射回到超音波模組時，感測器會由 SIG 腳再回傳一個 PWM 訊號給 Arduino 控制板，所回應 PWM 訊號的脈寬時間與超音波傳遞的來回距離成正比，最小值 115 微秒，最大值 18500 微秒。因為音波速度每秒 340 公尺，約等於每公分 29 微秒。因此，物體與超音波模組的距離=脈寬時間/29/2 公分。

圖 10-2 PING)))™ 超音波模組的工作原理（圖片來源：[www.parallax.com](http://www.parallax.com)）

### 10-2-2 物體定位

有時候待測物體的位置也會影響到 PING)))™ 超音波感測器的測量正確性。如圖 10-3 所示三種超音波模組無法定位物體距離的情形。在此三種情形下超音波模組不會接收到超音波訊號，因此無法正確測量物體的距離。



(a) 物體距離超過 3.3m      (b) 發射角度 $\theta$ 小於 45 度      (c) 物體太小

圖 10-3 三種超音波模組無法定位物體距離的情形 (圖片來源：[www.parallax.com](http://www.parallax.com))

圖 10-3(a)所示為待測物體距離超過 3.3 公尺，已超過 PING)))™ 超音波模組可以測量的範圍。圖 10-3(b)所示為超音波進入物體的角度小於 45 度，超音波無法反射回至超音波模組。圖 10-3(c)所示為物體太小，超音波模組接收不到反射訊號。

### 10-3 認識超音波避障自走車

所謂超音波避障自走車是指自走車可以**自動運行前進**，而且不會碰撞到任何障礙物。為了讓自走車可以自動避開障礙物，可以使用三個超音波模組分別放置於自走車的車頭右方、前方及左方等三個位置，偵測右方、前方及左方等三個方向的障礙物距離。如果考慮成本，也可以使用如圖 10-4 所示伺服馬達與超音波模組的組合，利用伺服馬達**自動轉向** 45 度、90 度及 135 度來偵測右方、前方及左方的障礙物距離。

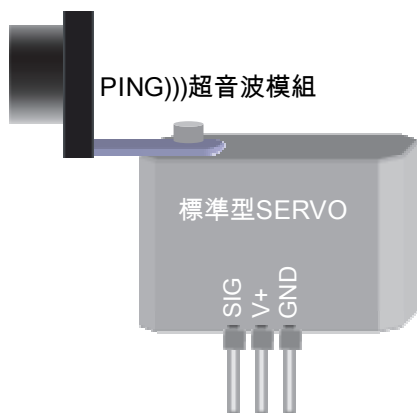


圖 10-4 伺服馬達與超音波模組的組合



### 10-3-1 工作原理

如圖 10-5 所示超音波避障自走車轉動角度與偵測方向的關係，正常情形下自走車自動運行前進。當自走車遇到前方有障礙物且距離小於 25 公分時（可視實際情形調整），自走車立即停止，伺服馬達轉動超音波模組偵測右方（45 度）及左方（135 度）障礙物距離並且回傳給 Arduino 控制板。Arduino 控制板依據前方、右方及左方障礙物的距離，判斷一條可以安全前進的路徑，避開障礙物後再回正繼續前進運行。

必須注意的是當伺服馬達在轉動時，超音波模組是無法正確偵測到障礙物的距離，必須等待伺服馬達停止轉動且穩定一段時間（約 0.5 秒）後，才能偵測到障礙物的正確距離。



(a) 偵測前方障礙物

(b) 偵測右方障礙物

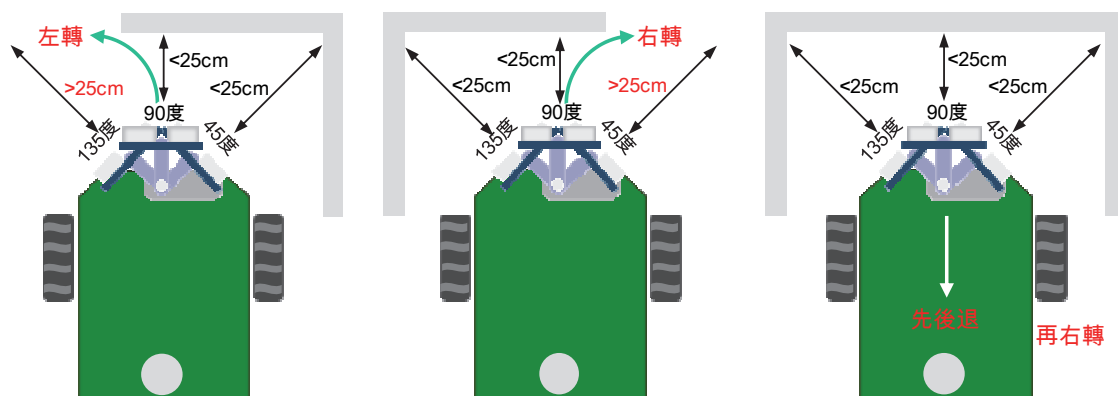
(c) 偵測左方障礙物

圖 10-5 超音波避障自走車轉動角度與偵測方向的關係

### 10-3-2 運行策略

超音波避障自走車正常情形為直線前進，當前方有障礙物時，自走車先停止，開始偵測右方及左方障礙物的距離，選擇障礙物距離較遠的方向為安全的行進路線。

如圖 10-6 所示為超音波避障自走車的行進路線判斷，圖 10-6(a)所示為前方及右方皆有距離小於 25cm 障礙物時，自走車偵測到左方近端無障礙物，左轉運行 0.5 秒避開障礙物後，再回正直行。圖 10-6(b)所示為前方及左方皆有距離小於 25cm 的障礙物時，自走車偵測到右方近端無障礙物，右轉運行 0.5 秒避開障礙物後，再回正直行。圖 10-6(c)所示為前方、右方及左方皆有距離小於 25cm 障礙物時，自走車偵測到右方及左方近端皆有障礙物，先後退運行 2 秒避開障礙物，再右轉運行 0.5 秒後回正直行。



(a) 前方及右方有障礙物 (b) 前方及左方有障礙物 (c) 前、右及左方有障礙物

圖 10-6 超音波避障自走車的行進路線判斷

如表 10-1 所示為超音波避障自走車運行的控制策略，自走車依超音波模組所感測到左方、前方及右方等三個方向的障礙物距離，選擇障礙物距離大於 25cm 的方向前進，如果三個方向的障礙物距離皆小於 25cm，則自走車先後退運行 2 秒、再右轉運行 0.5 秒離開障礙物，之後再回正直行。

表 10-1 超音波避障自走車運行的控制策略

左方障礙物	前方障礙物	右方障礙物	控制策略	左輪	右輪
無	無	無	前進	反轉	正轉
無	<25cm	<25cm	左轉	停止	正轉
<25cm	<25cm	無	右轉	反轉	停止
<25cm	<25cm	<25cm	後退	正轉	反轉
			右轉	反轉	停止
			前進	反轉	正轉

## 10-4 自造超音波避障自走車

如圖 10-7 所示超音波避障自走車電路接線圖，包含超音波模組、伺服馬達、Arduino 控制板、馬達驅動模組、馬達組件及電源電路等六個部份。



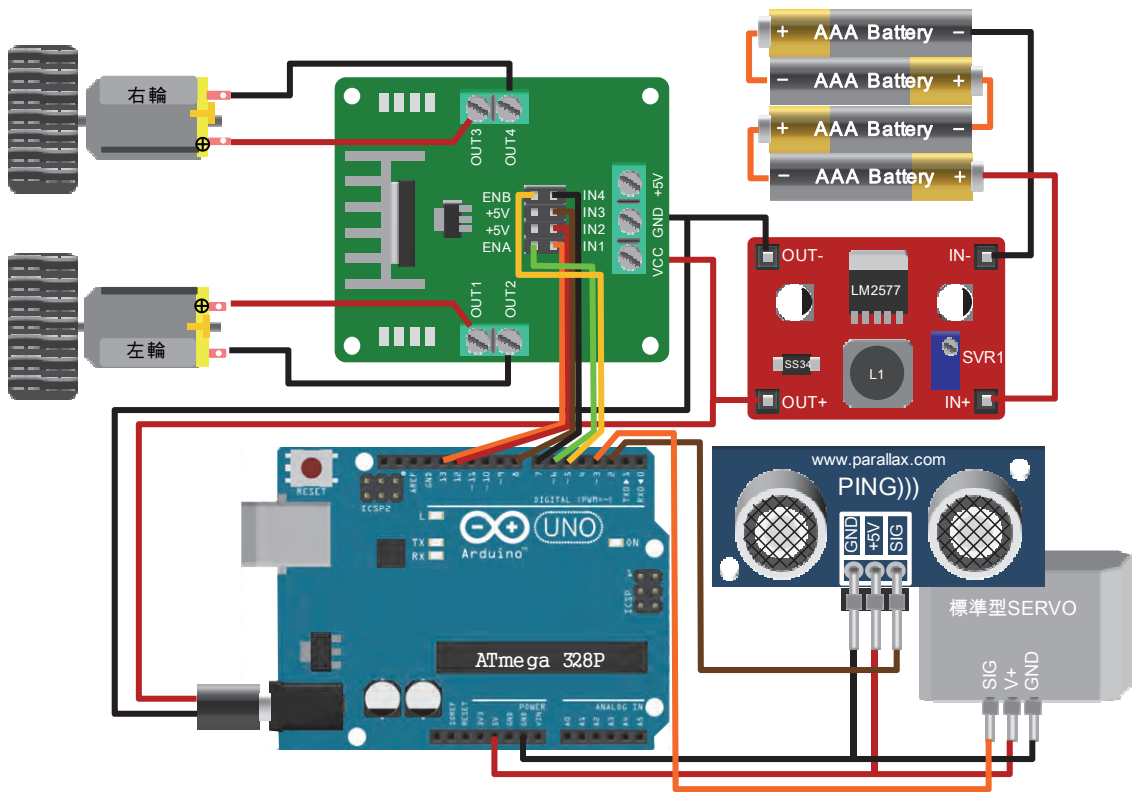


圖 10-7 超音波避障自走車電路接線圖

## 超音波模組

超音波模組與伺服馬達先行組合，由 Arduino 控制板的+5V 供電給超音波模組，並將超音波模組的 SIG 腳連接至 Arduino 控制板的數位腳 2。

## 伺服馬達

伺服馬達與超音波模組先行組合，由 Arduino 控制板的+5V 供電給伺服馬達，並將伺服馬達的 SIG 訊號腳接至 Arduino 控制板的數位腳 3。

## Arduino 控制板

Arduino 控制板為控制中心，判斷超音波模組所偵測到前方（90°位置）、右方（45°位置）及左方（135°位置）等三個方向的障礙物距離來決定自走車的運行方向。依表 10-1 所示超音波避障自走車運行的控制策略，來驅動左、右兩組減速直流馬達，使自走車能自動避開障礙物。



## 馬達驅動模組

馬達驅動模組使用 L298 驅動 IC 來控制兩組減速直流馬達，其中 IN1、IN2 輸入訊號控制左輪轉向，而 IN3、IN4 輸入訊號控制右輪轉向。另外，Arduino 控制板輸出兩組 PWM 訊號連接至 ENA 及 ENB，分別控制左輪及右輪的轉速。因為馬達有最小的啟動轉矩電壓，所輸出的 PWM 訊號平均值不可太小，以免無法驅動馬達轉動。PWM 訊號只能微調馬達轉速，如果需要較低的轉速，可以改用較大減速比的減速直流馬達。

## 馬達組件

馬達組件包含兩組 300rpm/min（測試條件：6V）的金屬減速直流馬達、兩個固定座、兩個 D 型接頭 43mm 橡皮車輪及一個萬向輪，橡皮材質輪子比塑膠材質磨擦力大而且控制容易。

## 電源電路

電源模組包含四個 1.5V 一次電池或四個 1.2V 充電電池及 DC-DC 升壓模組，調整 DC-DC 升壓模組中的 SVR1 可變電阻，使輸出升壓至 9V，再將其連接供電給 Arduino 控制板及馬達驅動模組。如果是使用兩個 3.7V 的 18650 鋰電池，可以不用再使用 DC-DC 升壓模組。每個容量 2000mAh 的 1.2V 鎳氫電池約 90 元，每個容量 3000mAh 的 18650 鋰電池約 250 元。

### □ 功能說明：

自走車自動運行前進，當前方有障礙物時，能夠自動判斷一條可以安全前進的路線，使自走車不會碰撞到任何障礙物。



程式：ch10-1.ino

```
#include <Servo.h> //使用 Servo 函式庫。
Servo Servo; //建立 Servo 資料型態的物件。
const int sig=2; //超音波模組輸出訊號 sig。
const int negR=7; //右輪馬達負極接腳。
const int posR=8; //右輪馬達正極接腳。
const int negL=12; //左輪馬達負極接腳。
const int posL=13; //左輪馬達正極接腳。
const int pwmR=5; //右輪轉速控制腳。
```