

摘要

本專題遙控車是藉由 arduino 遙控零件組裝而成, 為了不浪費電子零件資源及車體輕量化, 車體用的是質量輕且不易折損的尼龍板做為基底, 程式控制板選用的是容易上手且功能強大的 UNO R3 控制板, 此款有許多特色存在, 如藍牙無線模塊, 可以和手機藍牙配對遙智慧小車、可接入外部 7~12V 的電壓。並能搭載多款傳感器模塊, 根據您的想像力實現各種功能... 等等方便且實用的功能。因應現在車子區是邁向智慧化時代, 具有藍芽遙控的功能性, 能在各種危急狀況下, 將傷害降到最低, 於是我們便想出用最基本且最省成本的零件, 組出一台具有功能性的藍芽遙控車。

致 謝

此專題要感謝學校給我們這個機會，讓我們有機會學習自己動手組出一台有創意發想且具有功能性的一台電子遙控車。謝謝父母願意支持我們完成這個專題作品，謝謝系上、指導老師楊政穎老師和楊伯華老師的指導，不管在設計、組裝、程式撰寫及專題進度規劃上都給我們很多幫助，謝謝精準模型店老闆在組裝遙控車上給我們想法上的建議，謝謝今華電子零件批發商願意售出讓我們購得車體上的電子零件，謝謝在此專題上一起努力分工合作的同學。

目 錄

摘 要.....	I
致 謝.....	II
目 錄.....	II
圖目錄.....	V
第 1 章 緒論.....	1
1.1 前言	1
1.2 研究動機	1
1.3 研究方向	1
1.4 研究目的	1
1.5 時間進度管制.....	2
1.6 工作分配	3
第二章 藍芽遙控車理論介紹	4
2.1 車體主要結構	4
2.2 ARDUINO 藍芽接收器	4
2.3 ARDUINO UNO R3 控制板.....	4
2.4 伺服馬達	5
第 3 章 設計與製作.....	8
3.1 使用材料表	8
3.2 製作過程	8
第 4 章 程式與測試.....	11
4.1 成品圖	11
4.2 錯誤! 找不到參照來源。	11
4.3 測試	14
第 5 章 錯誤! 找不到參照來源。	15
5.1 錯誤! 找不到參照來源。	15
5.2 錯誤! 找不到參照來源。	15
第六章 參考文獻.....	15

圖目錄

圖 1 計畫進度管制圖.....	2
圖 2 ARDUINO 藍芽接收器.....	4
圖 3 ARDUINO UNO R3 控制板.....	5
圖 4 GWS S35 STD 伺服馬達.....	5
圖 5 PWM 訊號.....	6
圖 6 PWM 控制伺服馬達的原理.....	7
圖 7 車體底層動力系統面板.....	9
圖 8 隔層立柱.....	9
圖 9 控制系統面板.....	9
圖 10 車體構照完成.....	10
圖 11 單芯線插上在 ARDUINO R3 控制板上.....	10
圖 12 單芯線插上在麵包板上.....	10
圖 13 藍芽遙控車完成圖.....	11

表目錄

表 1 計畫進度管制圖	2
表 2 材料表	8

第1章 緒論

1.1 前言

此專題遙控車為具有藍芽功能性的遙控車, 遙控程式是用 arduino UNO R3 控制板控制, 並運用藍芽感測器連結手機 APP 程式遙控, 來達到方便搖控功能性的效果。

1.2 研究動機

因應現在車子區是邁向智慧化時代, 能自動偵測障礙及具有避障礙的功能性, 能在各種危急狀況下, 將傷害降到最低, 於是我們便想出用最基本且最省成本的零件, 組出一台也具有功能性的藍芽遙控車。

1.3 研究方向

本專題計畫研究方向是自製一台省成本且有藍芽功能的智慧遙控車

1.4 研究目的

1. 探討避障、藍牙遙控多功能實驗
2. 藍牙無線模塊, 可以和手機藍牙配對的應用

1.5 時間進度管制

本專題研究內容共分為資料收集及研究、程式設計與測試、算例規劃、實例計算、材料採買及貨比三家、遙控車車體製作、實驗設備安裝與測試、arduino 程式撰寫及遙控測試、實驗與計算數值之整理分析與比較、結案報告撰寫與製作等 10 項，各工作項目時程進度如下圖所示。

預定進度
 實際進度

月次 工作項目	第 1 月	第 2 月	第 3 月	第 4 月	第 5 月	第 6 月	第 7 月	第 8 月	第 9 月	第 10 月	第 11 月	第 11 月	備註	
1. 資料收集及研究														
2. 程式設計與測試														
3. 算例規劃														
4. 實例計算														
5. 材料採買														
6. 遙控車車體製作														
7. 實驗設備安裝與測試														
8. Arduino 程式撰寫及遙控測試														
9. 實驗與計算數值之整理、分析與比較														
10. 結案報告撰寫與製作														
預定進度 累計百分比	10%	15%	20%	30%	40%	50%	60%	70%	80%	85%	95%	100%		

表 1 計畫進度管制圖

1.6 工作分配

組長:李彥樟	負責藍芽遙控車測試及報告處理
組員:陳昱名	負責車體構造處理及內部線路安裝
組員:曾詠傑	負責車體構造處理及避障礙遙控車測試
組員:劉冠賢	負責所有材料採買及內部線路安裝

第2章 藍芽遙控車理論介紹

2.1 車體主要結構

主要分為控制系統、動力系統及車體

控制系統:在 Arduino 上執行的程式可以使用任何能夠被編譯成 Arduino 機器碼的程式與研編寫。Arduino 計畫提供了 Arduino Software IDE，一套以 Java 編寫的跨平台應用程式。使用 Arduino Software IDE 編寫的程式被稱為 [sketch]

動力系統:伺服馬達的動作特性是進行位置定位控制和動作速動控制，其主要特點是轉速可以精確控制，速度控制範圍廣，可以安定平順等速運轉之外，還可以根據需求隨時變更速度。能迅速做出正轉與逆轉，也能迅速加減速。

車體:以瓦楞板當底分成上下兩層，在下層安裝動力系統的四連電池座及兩個伺服馬達並安裝輪子，以惰論來穩定車子行進間的平衡，在以 7 根髮捲棒來支撐上層的瓦楞板，在上層安裝了控制系統的 Arduino 及麵包板

2.2 Arduino 藍芽接收器

常見的兩種 Arduino 藍芽接收器有:HC-05、HC-06

HC-05:在主控端和被控端中，出廠預設通常是被控端模式，但能自行修改內部程式的參數。

HC-06:在主控端和被控端中，模式已設定好，不能自行更改內部參數。

因為我們不需修改內部參數，所以本專題選用的是 HC-06 接收器。

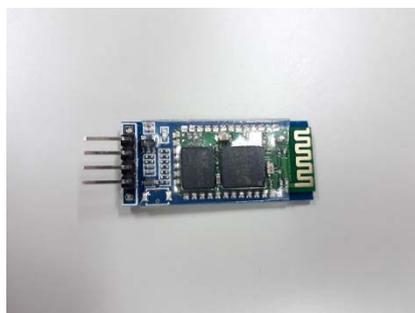


圖 2 Arduino 藍芽接收器

2.3 Arduino UNO R3 控制板



圖 3 arduino UNO R3 控制板

Arduino Uno 的運作直流電壓為 5V，以正確方式提供電源，不只是為了讓控制板正常運作，也為了避免損壞控制板。Arduino Uno 可透過三個管道提供電源：USB 連接埠、電源輸入插座、Vin 腳位。

透過 USB 連接埠提供給控制板的電源必須是 5V 的電壓，最基本的方式，就是透過 Type B USB 連接控制板，而另一頭是 Type A USB 連接個人電腦，這可以提供 5V、500mA 的電源給控制板，這也是本書一開始採取的方式，因為電腦要透過 USB 傳送程式給 Arduino。

如果採藍牙連線或其他無線方式來傳送資料給控制板，不用連接個人電腦與控制板的話，那麼可以使用 USB 電源供應器，像是插頭或者行動電源等，請注意必須是 5V 的直流電壓，至少 2A 電流的輸出。

2.4 伺服馬達

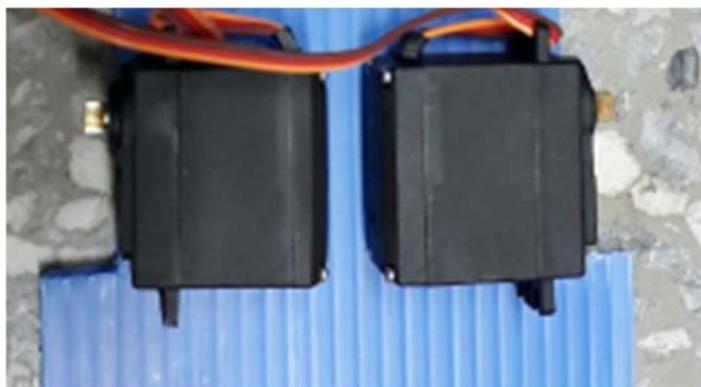


圖 4 GWS S35 STD 伺服馬達

在挑選遙控車的動力系統時，我們相互比較伺服馬達與步進

伺服馬達簡介

伺服馬達為多數小型機器人常用的馬達，它的體積不會很大，重量也輕，且能精確地控制旋轉角度。

一個典型的伺服馬達主要分成四個部分

1. 控制電路晶片
2. 直流馬達
3. 減速齒輪組
4. 可變電阻

伺服馬達裡面也是以一顆直流馬達來做驅動的，不過齒輪變速箱也一起裝在裡面了，一般馬達的力量其實都很小但速度卻很快，所以馬達通常都會再搭配所需的齒輪組，例如像小時候玩的四驅車裡馬達到輪子間的轉動也是經過一組齒輪傳遞的，目的也是為了能將扭力提高，不然其實馬達本身的扭力並不怎麼高，所以一般機器人在用的伺服馬達也都已經將所需的減速齒輪組放進去了，可能有人會問為什麼要特意讓馬達要減速呢？因為馬達的速度和扭力是有相關關係的，同樣的馬達，若變速齒輪組使他的轉速提高的話向對的扭力就會變小，反之亦然兩者的關係剛好成反比。

一般伺服馬達所吃的是 PWM 訊號 (Pulse Width Modulation)

PWM 訊號是像方波一樣 high-low high-low 的擺盪，如圖 5 所示。

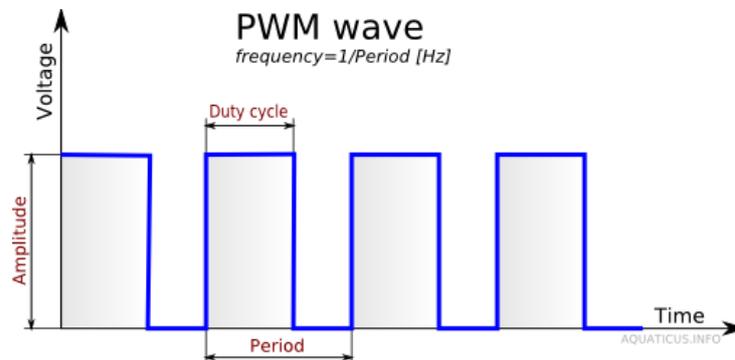


圖 5 PMW 訊號

一般的 PWM 訊號 high 是 5V low 是 0V，而以 PWM 控制伺服馬達的原理大致是以下這樣，如圖 6 所示。

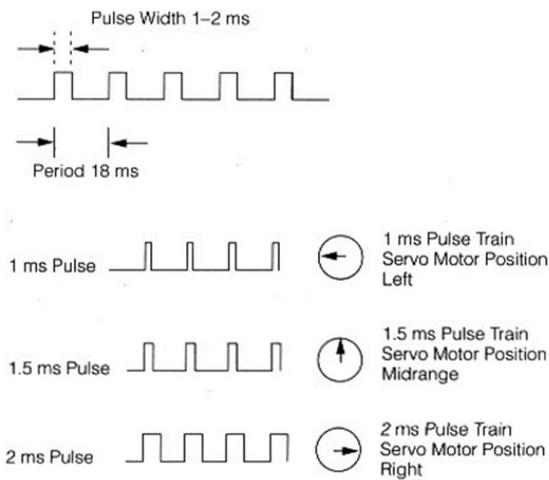


圖 6 PWM 控制伺服馬達的原理

上圖當 18 毫秒為週期的 PWM 訊號中有 1 毫秒是 high 的狀態時，伺服馬達轉向 9 點鐘方向的位子，有 1.5 毫秒是 high 的時候轉到 12 點鐘的位子，2 毫秒是 high 時轉到 3 點鐘的位子，換句話說伺服馬達的轉向則是由 PWM 訊號的一個周期裡 high 所佔的比來控制的，而將接收這樣的訊號並將之轉換給伺服機內的直流馬達就是控制電路晶片在做的工作，而且當馬達轉動到不同的角度時也會跟著帶動伺服機內的一個可變電阻，可變電阻就會因為馬達的角度不同而產稱不同的電阻值，傳給控制電路晶片後就能夠知道馬達現在的角度在什麼地方了。

第3章 設計與製作

3.1 使用材料表(圖 3-1)

材料名稱	數量
Arduino Uno R3	1 個
麵包板	1 個
MG995 伺服馬達 180	2 個
電池 3 號四連電池座	1 個
0.5mm 單芯線	6 條
珍珠板	2 片
髮捲棒	7 根

表 2 材料表

3.2 製作過程

步驟 1: 製作車體底層動力系統面板, 買珍珠板並割好所需形狀及尺寸大小, 再把伺服馬達及電池座黏好在割好的板上 (圖 7)。

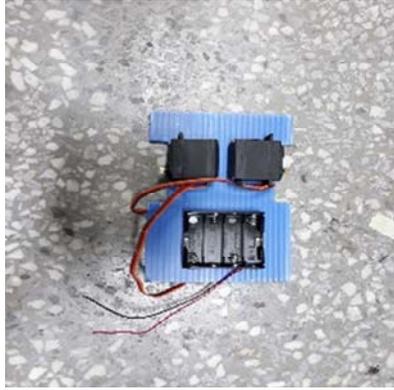


圖 7 車體底層動力系統面板

步驟 2: 在車體底層黏上 7 根立柱做好隔層動作, 並在兩側裝上輪胎 (圖 8)。



圖 8 隔層立柱

步驟 3: 如同第一步驟製作過程, 把第二層控制系統面板做出來 (圖 9)。

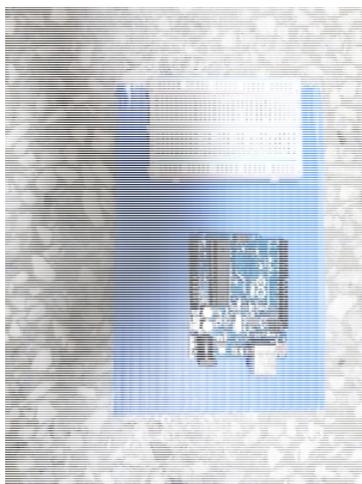


圖 9 控制系統面板

步驟 4: 把兩層面板拼裝組合, 及做好車體構照 (圖 10)。



圖 10 車體構照完成

步驟 5:把單芯線插上在 arduino R3 控制板(圖 11)及麵包板上(圖 12), 完成線路配置。

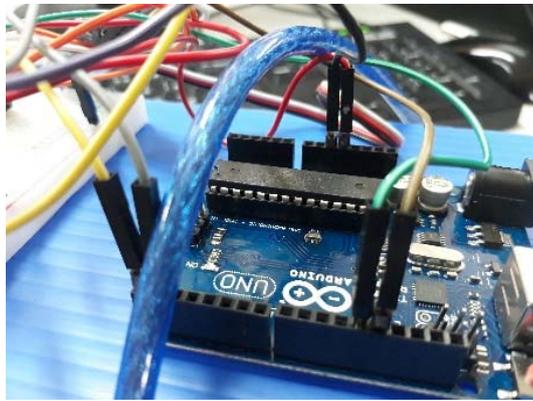


圖 11 單芯線插上在 arduino R3 控制板上

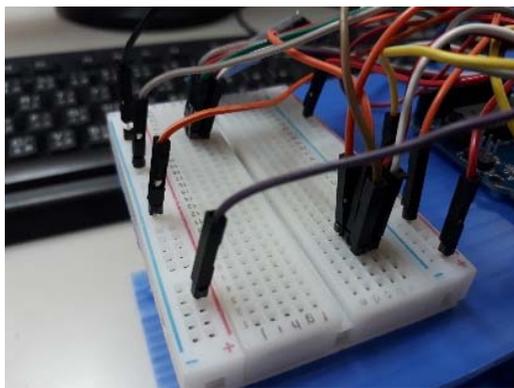


圖 12 單芯線插上在麵包板上

第4章 程式與測試

4.1 成品

本專題成品藍芽遙控車(圖 4-1):



圖 13 藍芽遙控車完成圖

4.2 Arduino 程式

```
#include <Servo.h>
Servo servoRight;
Servo servoLeft;
Servo servoup;
Servo servodown;
void setup()
{
  servoRight.attach(12);
  servoLeft.attach(13);
  servoup.attach(11);
  servodown.attach(10);
  Serial.begin(9600);
}
void loop()
```

```

{
char c;
if(Serial.available () >0)
{
c=Serial.read();
switch (c) {
case '1':
forward(1000);
break;
case '2':
trunLeft(400);
break;
case '3':
trunRight(400);
break;
case '4':
backward(400);
break;
case '5':
bestop(400);
break;
case '6':
up(1000);
break;
case '7':
down(400);
break;

}
}
}
void forward(int time){
servoRight.writeMicroseconds(1300);

```

```

servoLeft.writeMicroseconds(1700);
delay(time);  }
void trunLeft(int time){
servoRight.writeMicroseconds(1700);
servoLeft.writeMicroseconds(1700);
delay(time);  }
void trunRight(int time){
servoRight.writeMicroseconds(1300);
servoLeft.writeMicroseconds(1300);
delay(time);  }

void backward(int time){
servoRight.writeMicroseconds(1700);
servoLeft.writeMicroseconds(1300);
delay(time);  }
void bestop(int time){
servoRight.writeMicroseconds(1500);
servoLeft.writeMicroseconds(1500);
servoup.writeMicroseconds(1500);
servodown.writeMicroseconds(1500);
delay(time);  }
void up(int time){
servoup.writeMicroseconds(1700);
servodown.writeMicroseconds(1300);
delay(time);  }
void down(int time){
servoup.writeMicroseconds(1300);
servodown.writeMicroseconds(1700);
delay(time);
}

```

4.3 測試

輸入完程式後並試著發動，結果沒有依照我們的理想啟動，我們檢查程式後發現一些錯誤，在請教老師及查詢相關資訊後，修正了這些問題。再度啟動後還是沒有做動，我們問了指導老師，老師指出了藍芽控制器並不能直接安裝在麵包板上，要用杜邦線來外接，我們按照老師說的做法來做修改，修改完後終於可以進行移動。

第5章 結論與建議

5.1 結論

歷屆專題上都以太陽能遙控車設計居多,這次我們選擇重複性較少的 arduino 藍芽遙控車當專題,Arduino 這款程式在大二上學期的電子電機學這門課裡有接觸到,那時候有業師來教我們打基本的車子驅動程式,也在學期末時有舉辦成果比賽,雖然只在那學期有學到過,但對此軟體並不陌生,因此這次專題我們就想自己設計一台藍芽遙控車,車體零件都是一手包辦,我們針對此車子的功能性需求來做設計與改裝,包括車體結構設計、伺服馬達、電池選購及應用研究,這些都是經過好幾次的更改設計,並都是靠實際製作後才能發現問題並解決問題。

5.2 建議

若程式參數設定是旋轉 360 度,伺服馬達就要購買能 360 度旋轉的相關型號,若買的是只支援 180 度旋轉的馬達,車子在左右轉操作上會有不會動問題。

參考文獻

1. http://gio781215.blogspot.tw/2012/07/blog-post_18.html
伺服馬達簡介
2. <https://openhome.cc/Gossip/Books/mBlockArduino1-3and1-4.html>
Arduino Uno 功能簡介
3. <http://www.playrobot.com/retire/411-mg995-.html> MG995
伺服馬達
4. <http://shop.cpu.com.tw/product/47191/info/>
ADIO-HC06 藍芽感測器