**計算機組織及微算機實驗 期末報告**

|  |  |  |
| --- | --- | --- |
| 班級 | 學號 | 姓名 |
| 資二甲 | 110916015 | 林翡 |

1. 專題主題

了解計算機組織及微算機架構，並透過實作進行驗證與測試。本堂課程將進行循跡避障自走車的實作，並進行實際測試。

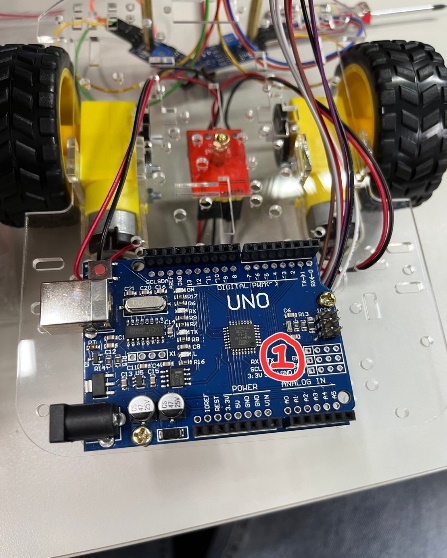
1. 功能規劃簡介

完成循跡避障自走車組裝後，撰寫Arduino程式控制自走車能夠沿著黑線行走，完成循跡的功能。另外，在遇到障礙物時能夠及時繞過障礙物，並繼續前進完成循跡。

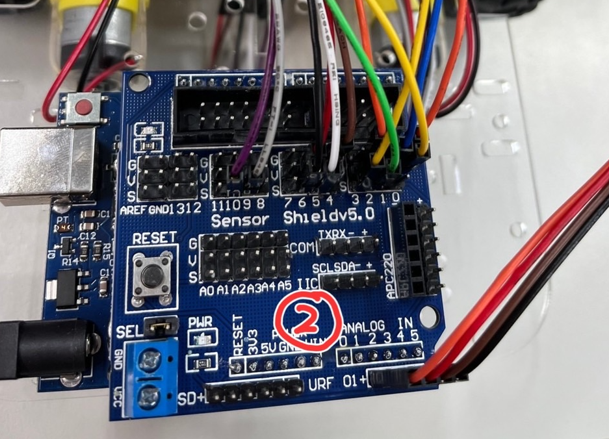
1. 組裝過程說明

所需材料:自走車套件包，一些螺絲、螺帽，10條-彩色杜邦雙頭線(母/母)，螺絲起子，焊錫工具…

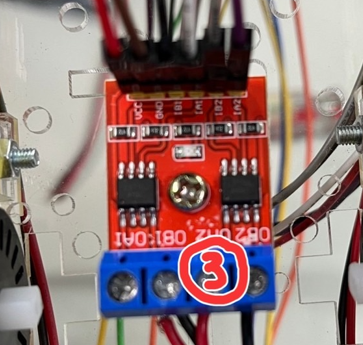
1. Arduino Uno控制板



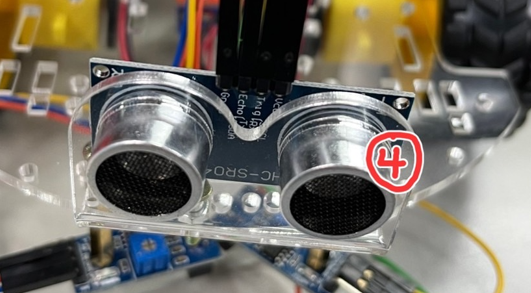
1. 感測器擴充板



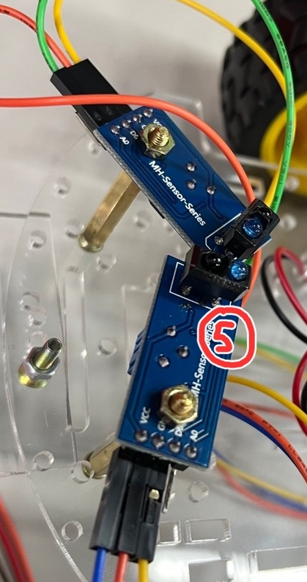
1. 馬達驅動器



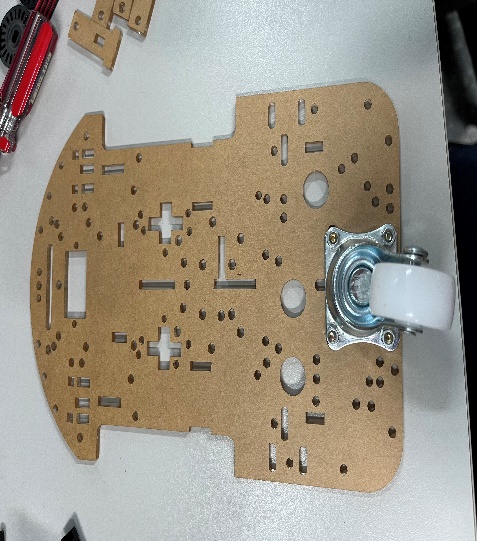
1. 超音波測距模組及支架

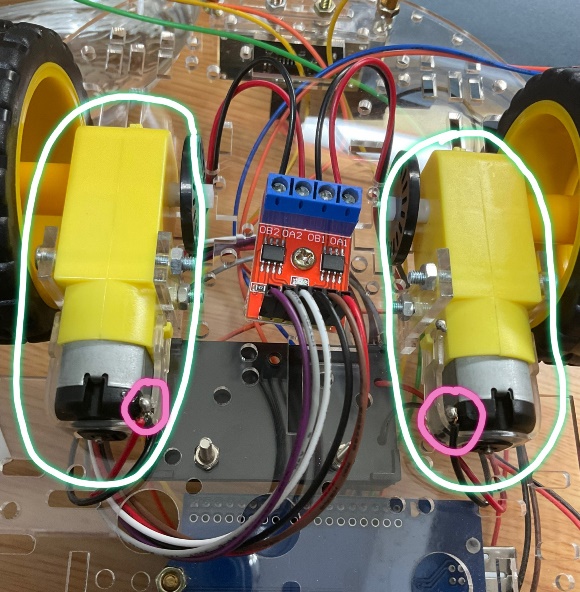


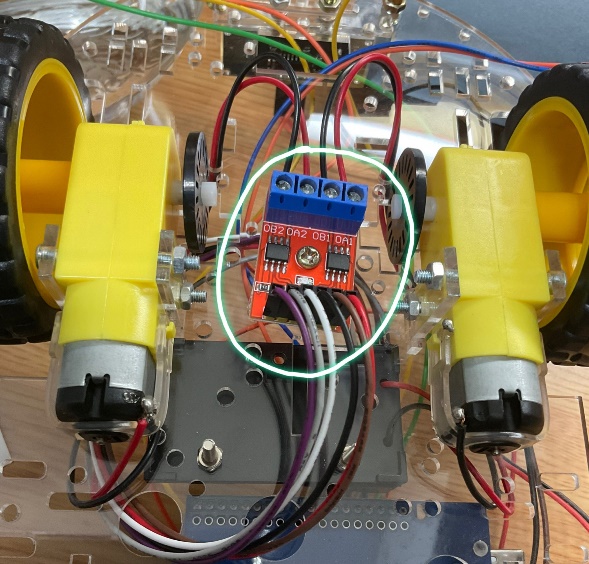
1. 紅外線反射模組\*2

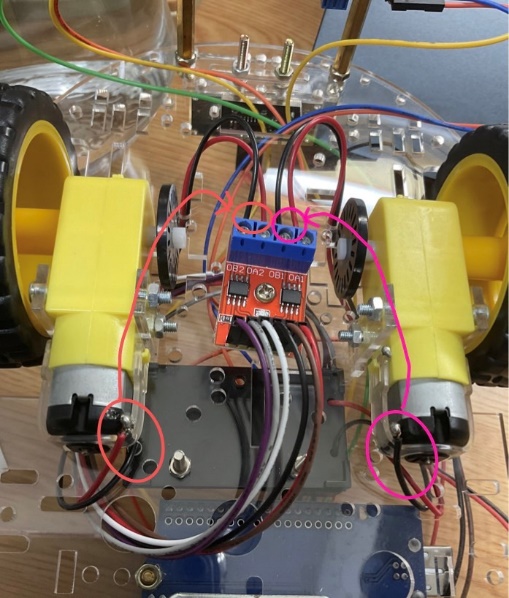


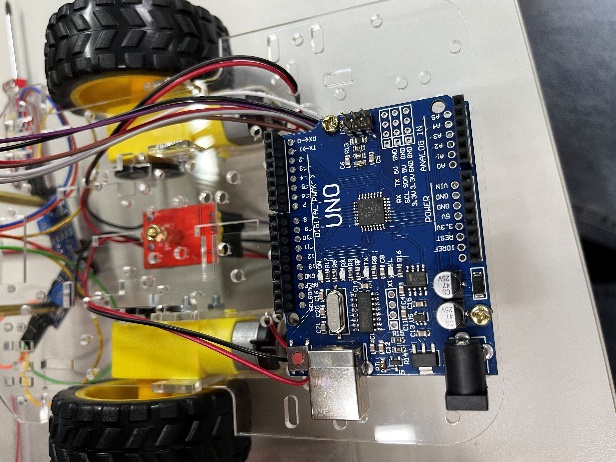
以下為組裝步驟:

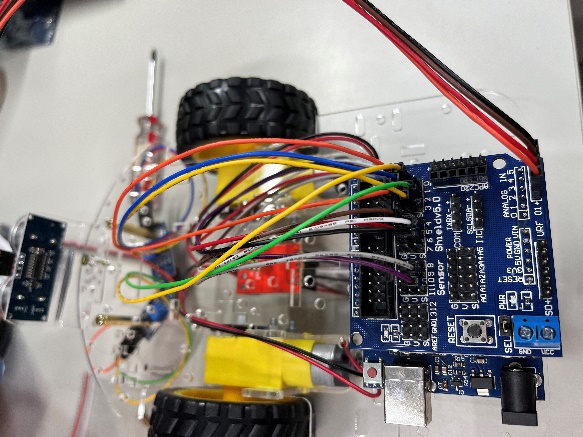
第一步驟:將輔助輪裝在板子上

第二步驟:將兩顆馬達組裝上去(馬達的線要用焊錫的方式接上喔!)

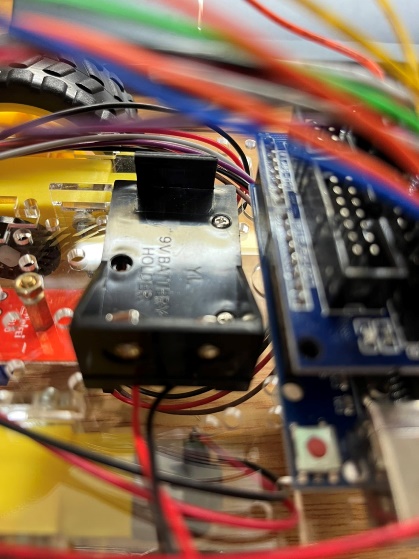
第三步驟:裝上馬達驅動器(將左邊馬達的兩條線接到馬達驅動器上的最左邊兩個孔；右邊馬達的兩條線接到馬達驅動器上的最右邊兩個孔)

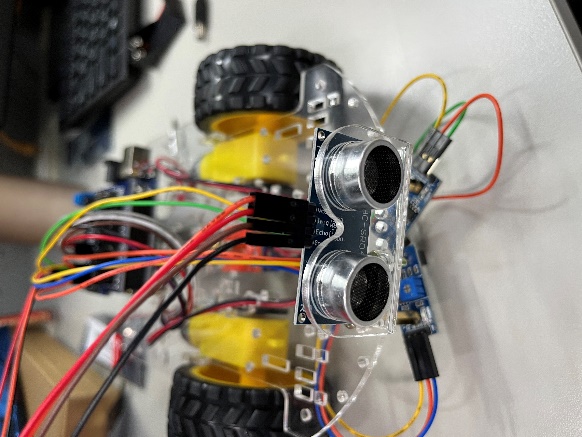


第四步驟:裝上Arduino Uno控制板在馬達的反面那面

第五步驟:裝上感測器擴充板(加在Arduino Uno控制板上面，把它對齊右邊卡上去)

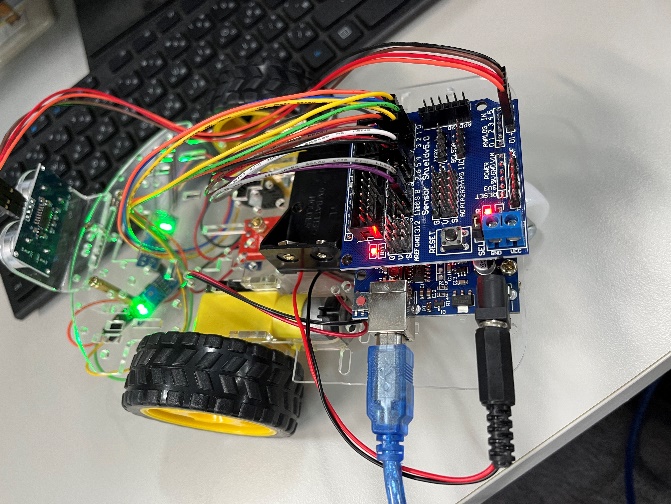
第六步驟:裝上電池盒



第七步驟:裝超音波測距模組及支架

第八步驟:裝上輪胎、紅外線反射模組(右側紅外線反射模組的接線是VCC🡪V，GND🡪G，D0🡪12；左側的接線是VCC🡪V，GND🡪G，D0🡪8)

第九步驟:把杜邦線的一頭接到馬達驅動器，另一頭接到感測器擴充板上(我的接法是1A2🡪9，1**B2🡪10，1A1🡪6，1B1🡪5，GND🡪G，VCC🡪V)**

**第十:完成啦!**

1. Arduino程式碼說明
   1. 循跡行走:

const int In1 = 5;//1B1的腳位

const int In2 = 6;//1A1的腳位

const int In3 = 9;//1A2的腳位

const int In4 = 10;//1B2的腳位

int SensorLeft = 8;//左感測器輸入腳

int SensorRight =12;//右感測器輸入腳

int SL;//左感測器狀態

int SR;//右感測器狀態

void setup() {

Serial.begin(9600);//初始化序列埠

pinMode(In1,OUTPUT);

pinMode(In2,OUTPUT);

pinMode(In3,OUTPUT);

pinMode(In4,OUTPUT);

pinMode(SensorLeft, INPUT);//讀入左感測器狀態

pinMode(SensorRight, INPUT); //讀入右感測器狀態

mstop();

delay(500);

}

void loop(){

SL = digitalRead(SensorLeft); //讀取左感測器狀態值

SR = digitalRead(SensorRight);//讀取右感測器狀態值

if (SL == 1&& SR ==1) {//如果左右感測器的值均為 1，代表左右感測器都看到黑色，車子前進

mstop();

delay(250);

mforward();

delay(50);

}

if (SL == 1 && SR == 0) {//如果左邊感測器的值為1，而右邊感測器的值為0，代表左邊感測器看到黑色，右邊感測器看到白色，車子左轉

mstop();

delay(500);

mleft();

delay(100);

}

if (SL == 0 && SR == 1) {//如果左邊感測器的值為0，而右邊感測器的值為1，代表左邊感測器看到白色，右邊感測器看到黑色，車子右轉

mstop();

delay(500);

mright();

delay (100);

}

if (SL == 0&& SR == 0) { //如果左邊感測器的值為0，而右邊感測器的值為0，代表左右邊感測器都看到白色，車子停止

mstop();

delay (100);

}

}

int mback(){//後退的函式

digitalWrite(In1,LOW);//左邊往前

digitalWrite(In2,HIGH);//左邊往後

digitalWrite(In3,LOW);//右邊往前

digitalWrite(In4, HIGH);//右邊往後

}

void mforward(){//前進的函式

digitalWrite(In1, HIGH);

digitalWrite(In2, LOW);

digitalWrite(In3, HIGH);

digitalWrite(In4, LOW);

}

void mleft(){//左轉的函式

digitalWrite(In1, LOW);

digitalWrite(In2, LOW);

digitalWrite(In3,HIGH);

digitalWrite(In4, LOW);

}

void mright(){//右轉的函式

digitalWrite(In1,HIGH);

digitalWrite(In2, LOW);

digitalWrite(In3, LOW);

digitalWrite(In4, LOW);

}

void mstop(){//停止的函式

digitalWrite(In1,LOW);

digitalWrite(In2,LOW);

digitalWrite(In3,LOW);

digitalWrite(In4,LOW);

}

* 1. 避障:

const int In1 = 5;//1B1的腳位

const int In2 = 6;//1A1的腳位

const int In3 = 9;//1A2的腳位

const int In4 = 10;//1B2的腳位

void mback(){//後退的函式

digitalWrite(In1,LOW);//左邊往前

digitalWrite(In2,HIGH);//左邊往後

digitalWrite(In3,LOW);//右邊往前

digitalWrite(In4,HIGH);//右邊往後

}

void mforward(){//前進的函式

digitalWrite(In1,HIGH);

digitalWrite(In2,LOW);

digitalWrite(In3,HIGH);

digitalWrite(In4,LOW);

}

void mstop(){//停止的函式

digitalWrite(In1,LOW);

digitalWrite(In2,LOW);

digitalWrite(In3,LOW);

digitalWrite(In4,LOW);

}

void mleft(){//左轉的函式

digitalWrite(In1,LOW);

digitalWrite(In2,HIGH);

digitalWrite(In3,HIGH);

digitalWrite(In4,LOW);

}

void mright(){//右轉的函式

digitalWrite(In1,HIGH);

digitalWrite(In2,LOW);

digitalWrite(In3,LOW);

digitalWrite(In4,HIGH);

}

const byte trigPin = 3 ;//超音波模組的觸發腳

const byte echoPin = 2 ;//超音波模組的接收腳

unsigned long d ;//儲存高脈衝的持續時間

unsigned long ping(){

digitalWrite(trigPin,HIGH);//觸發腳設為高電位

delayMicroseconds(10);//持續10微秒

digitalWrite(trigPin,LOW);//觸發腳設為低電位

return pulseIn(echoPin,HIGH,23200);//傳回高脈衝的持續時間

}

int collapseDetect(int d){//自訂距離偵測函數

if( d <= 15 ){

mstop();//碰到障礙物停止

}

else{

mforward();

}

}

void setup() {

pinMode(In1,OUTPUT);

pinMode(In2,OUTPUT);

pinMode(In3,OUTPUT);

pinMode(In4,OUTPUT);

pinMode(trigPin,OUTPUT);//觸發腳設為輸出

pinMode(echoPin,INPUT);//接收腳設為輸入

delay(30);

mforward();

Serial.begin(9600);//初始化序列埠

}

void loop(){

d = ping() / 58 ;//把高脈衝時間值換算成公分單位

Serial.println(String("") + d +" cm");//顯示距離

delay(100);

collapseDetect(d);

}

* 1. 循跡行走+避障:

const int In1 = 5;//1B1的腳位

const int In2 = 6;//1A1的腳位

const int In3 = 9;//1A2的腳位

const int In4 = 10;//1B2的腳位

const byte trigPin = 3 ;//超音波模組的觸發腳

const byte echoPin = 2 ;//超音波模組的接收腳

unsigned long d ;//儲存高脈衝的持續時間

int SensorLeft = 8;//左感測器輸入腳

int SensorRight =12;//右感測器輸入腳

int SL;//左感測器狀態

int SR;//右感測器狀態

void setup() {

Serial.begin(9600);

pinMode(In1,OUTPUT);

pinMode(In2,OUTPUT);

pinMode(In3,OUTPUT);

pinMode(In4,OUTPUT);

pinMode(SensorLeft, INPUT);//讀入左感測器狀態

pinMode(SensorRight, INPUT); //讀入右感測器狀態

pinMode(trigPin,OUTPUT);

pinMode(echoPin,INPUT);

mstop();

delay(500);

}

void loop(){

SL = digitalRead(SensorLeft); //讀取左感測器狀態值

SR = digitalRead(SensorRight); //讀取右感測器狀態值

d = ping() / 58 ;

if( d <=25 ){//當碰到障礙物時停止

mstop();

}

else{//沒障礙物就照著循跡行走

if (SL == 1&& SR ==1) {

mstop();

delay(250);

mforward();

delay(50);

}

if (SL == 1 && SR == 0) {

mstop();

delay(500);

mleft();

delay(100);

}

if (SL == 0 && SR == 1) {

mstop();

delay(500);

mright();

delay (100);

}

if (SL == 0&& SR == 0) {

mstop();

delay (100);

}

}

}

unsigned long ping(){

digitalWrite(trigPin,HIGH);

delayMicroseconds(10);

digitalWrite(trigPin,LOW);

return pulseIn(echoPin,HIGH,23200);

}

void mback(){

digitalWrite(In1,LOW);

digitalWrite(In2,HIGH);

digitalWrite(In3,LOW);

digitalWrite(In4,HIGH);

}

void mforward(){

digitalWrite(In1,HIGH);

digitalWrite(In2,LOW);

digitalWrite(In3,HIGH);

digitalWrite(In4,LOW);

}

void mleft(){

digitalWrite(In1,LOW);

digitalWrite(In2,LOW);

digitalWrite(In3,HIGH);

digitalWrite(In4,LOW);

}

void mright(){

digitalWrite(In1,HIGH);

digitalWrite(In2,LOW);

digitalWrite(In3,LOW);

digitalWrite(In4,LOW);

}

void mstop(){

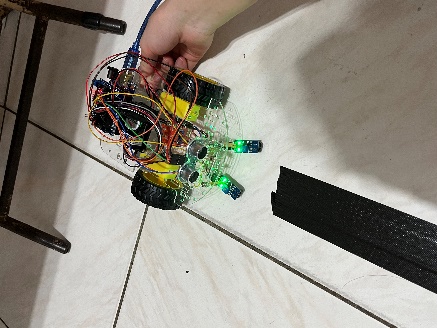
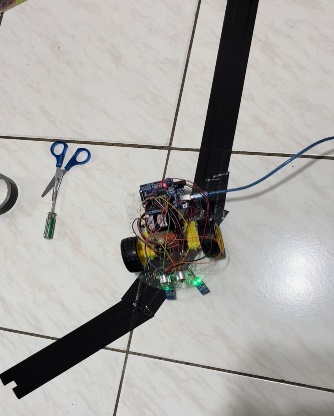
digitalWrite(In1,LOW);

digitalWrite(In2,LOW);

digitalWrite(In3,LOW);

digitalWrite(In4,LOW);

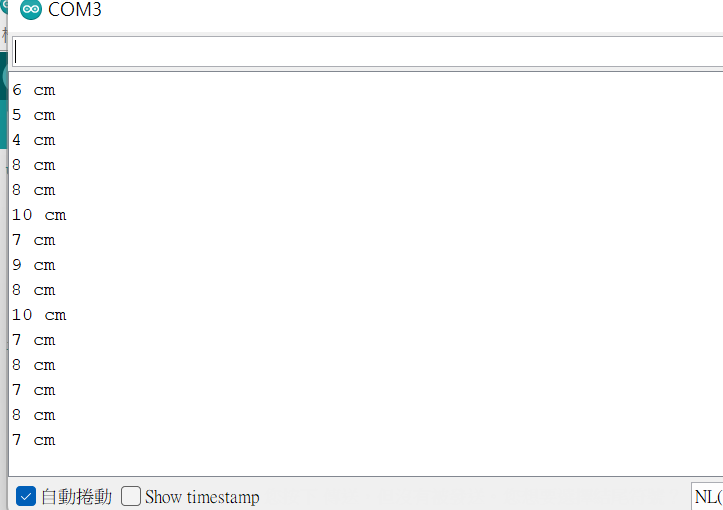
}

1. 實際測試過程
   1.  循跡行走

測試狀況:因為我的紅外線感測器算是蠻不靈敏的，加上它反應黑跟白的時候會有延遲，所以我讓整體行走過程放慢速度，然後在有些時候會有誤差而導致誤判，在測試時最常遇到的狀況就是兩顆紅外線都感應到白色就停止了，但原本應該是要讓車子可以左轉或右轉修正回軌道然後繼續行走，還有我發現我的車子在前進時沒辦法走直線，那時還很納悶為何我黑線是貼直線但車子在前進時就會一直往左偏移，所以才要把車速放慢讓它可以感應並且做修正。在下面影片中可以看到紅外線感測模組的燈其中有一顆會一直亮暗亮暗，那就是偵測的結果，一顆燈亮代表偵測到的是黑色的，紅外線向地板發射訊號而沒有訊號反射回來；兩顆燈亮代表紅外線發射的訊號有反射訊號回來，代表偵測到的是亮面的。

* 1. 避障(加分部分)

超音波偵測障礙物距離:



這是一開始在做超音波測距時讓它可以先讀取與障礙物的距離並印出。



測試狀況:避障功能本來我是想用超音波偵測到障礙物就左轉，那原本車子看到障礙物是會左轉的，但是遇到那種比較稜稜角角的東西它的反應就會比較慢，它可能有偵測到障礙物想左轉了但輪子會卡住，所以我最後就改成偵測到障礙物就停止。然後我一開始距離設10公分，發現超音波感測有時還是會太慢偵測到而撞到障礙物才左轉，這也會讓車子一直卡住，所以最後我就拉長了距離，增加它的反應時間。

* 1. 循跡行走+避障(加分部分)

測試狀況:這部分的循跡行走就跟第一部分的一樣，避障也是會順利完成的，我是用比較簡單的避障方法，就是偵測到障礙物就停止，障礙物移除又繼續循跡行走到終點，這邊的測試讓我有點怕怕的，照片中是車子偵測到障礙物停下來的瞬間，但距離障礙物超近的，那時真的超怕車子會就這樣撞上去，但最後還是有順利完成。

* 循跡行走、避障、循跡行走加避障的測試影片:

(影片連結:<https://www.youtube.com/watch?v=q97ATSf6LuQ&ab_channel=%E6%9E%97%E7%BF%A1>)

1. 實作心得

看看自己這台車在循跡行走過程中真的有點笨笨的感覺，就是沒辦法很順暢地走完，一定要靠停頓，不然車子真的會一下衝太快就跑出黑線外。這台自走車算是我人生中第一次從無到有自己組裝到完成，高中的資訊課我印象中好像也有上過這種自走車，但都沒讓我們自己組裝，只是寫幾行程式讓車子可以跑而已，所以對於自走車的一些原理、套件在一開始還不是那麼了解。還有那時候兩顆馬達的線都要焊錫，焊錫也是我以前都沒做過的，剛開始用時還超級生疏呢!我還記得前幾堂課要開始測試車子能不能前進、後退時我的車子就大碰壁了，因為我的輪子都不會轉，甚至要手動轉它才會轉，而轉速也很慢，馬達有奇怪的聲音，那時還以為是自己拿到壞掉的材料，詢問老師才明白是程式跟線路的問題，後來也順利解決了。後來是那個紅外線感測模組的問題，怎麼我的其中一顆紅外線的燈有兩個燈都會亮，但有一個燈偵測到黑色是比較暗的，那時我以為這樣應該也可以吧，但詢問學姊後，學姐說要一個燈亮一個全暗才是對的，於是我又開始陷入苦惱，後來才發現也是接線的問題，之後就是無線循環的調靈敏度，只能說那真的太難調啦!我甚至有一次調了快兩節課還沒調出來，雖然最後結果還是蠻不靈敏的，但還是能分辨，總之製作自走車的過程也讓我的心起起伏伏的，但不得不說，過程是有趣的，也讓我學習到哪裡有問題就要找出可能是哪邊出問題，並且找到解決辦法，不是一昧的原地轉圈，這其實就跟寫程式的debug很像，這真的是我最需要學習的地方。一開始還以為做不出循跡加行走的功能，最後看到車子能如預期地行走還是蠻感動的，雖然它是真的蠻遲鈍的。