

國立龍潭高級中學學生專題製作競賽

製作專題名稱：地下道自動水位警報警示控制系統

科別：電機科

班級：三年級

學生姓名：蔣冠詠、指導老師：劉哲雄老師

一、現況調查

【東森新聞報導】一整晚暴雨竟然奪走一條寶貴性命，台中市昨天晚間(2014年5月16日)下起滂沱大雨，造成大甲民生地下道淹水，水深將近兩米高，結果有一名母親載著兩個小孩回家，**因為淹水的地下道沒有拉封鎖線**，大雨視線不良誤闖進去車子拋錨，消防隊趕到現場緊急敲破車窗，救出分別10歲和8歲的姊弟，但是媽媽的高跟鞋卻被安全帶卡住，等到吊車將整輛車拉起，母親已經溺水昏迷，經過一整夜的搶救，今天凌晨宣告不治。



看了這則悲慘的意外新聞報導，著實令人難過，據相關交通單位統計全國各地之地下道設有警示警報預警機制系統之地下道寥寥無幾，即使現有之設備系統亦不夠完善，警示、警報、預警機制功能效果不彰，尤其普遍未設有自動柵欄系統及影像監視系統，目前國內市面上完全沒有相關地下道警告警示柵欄自動控系統之產品。

為了能改善地下道行車安全，保障民眾生命安全，心中燃起想以 arduino 微控器設計製作一套具有自動警示、警報、預警功能之外並有自動柵欄控制系統模型及更進一步發展透過網路遠端遙控警示、警報、預警、柵欄系統影像監視的功能模型，符合迎接未來物聯網時代的來臨。此一想法立刻獲得專題製作劉老師的鼓勵與支持，利用本學期專題製作實習課程，在老師指導與科內材料與設備之支援下著手開始研發設計製作製作。

設計動機與目的

1. 三年級上過以 arduino 微控器為主微電腦控制實習，所以有了一些 c/c++ 的程式設基礎，也發覺 arduino 微控器系統功能強大，硬體價格非常便宜，軟體免費、網路資源資料相當多，特別適合高中職學生在不用花費太多金錢下進行開發各種控制系統與專題製作，如果能設計製作出一套具有警示、警報、預警、自動柵欄功能的地下道系統模型做為各地方政府、交通主管單位設置地下道警示警報系統參考，本模型透過與廠商合作，適當轉換成現場真實地下道自動水位警報警示控制系統之建置，可以確保民眾地下道用路安全。
2. 經由此一專題製作強化自己對 arduino 微控器應用能力與學習與組員之間分工合作、集體創作。

二、創意發想歷程



構想流程圖

1. **發現地下道警示系統問題:**經由新聞報導台中地下道因缺乏水位警報警示控制系統，人工方式來不及拉封鎖線或因控制系統警示不夠清晰，造成一位母親誤闖暴雨淹水的地下道意外溺斃而發現全台地下道普遍缺乏控制系統的問題。
2. **構思問題解決辦法:** 結合現有的火車柵欄系統與自來水水塔水位自動控制概念,以 arduino 微控器設計製作一套具有自動警示、警報、預警功能之外並有自動柵欄控制系統及更進一步發展透過網路遠端遙控警示、警報、預警、柵欄系統及影像監視的功能模型。
3. **蒐集資料與購買材料:**由網際網路與圖書館蒐集相關硬體與軟體技術資料，確立硬體設計電路圖經老師審視修正後提出申請，購買材料及模型製作。
4. **程式設計與硬體製作:**取得材料後分組設計程式與硬體組裝。
5. **系統測試與美工:**進行軟硬整合系統測試與修正，確認無誤完成程式設計與硬

體電路後進行外觀美化。

6. 完成。

三、應用潛能分析

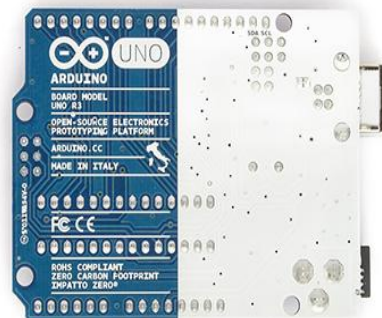
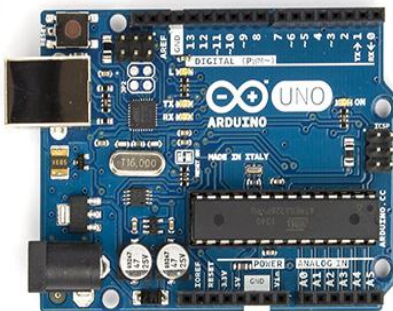
利用 arduino 微控器為主微電腦控制實習及電子學實習中課程所學及老師從旁指導協助加上不斷測試與修正，就可以逐漸打造出理想的成品模型。本專題開發系統 arduino 硬體與軟體設備屬於開放系統與其他為微電腦晶片開發系統價格相較，價格非常低廉，因開發之成品除了具有非常強力的市場價格競爭優勢外，本系統模型透過與廠商合作，適當轉換成現場真實版地下道自動水位警報警示控制系統，各地方政府能以較少的預算普及設置地下道自動水位警報警示控制系統對於地下道用路民眾安全會更進一步保障。

亦能應用於當地下道發生車禍事故或重大天然災變時可以手動控制，使得救災更加方便，是一項多功能的發明。

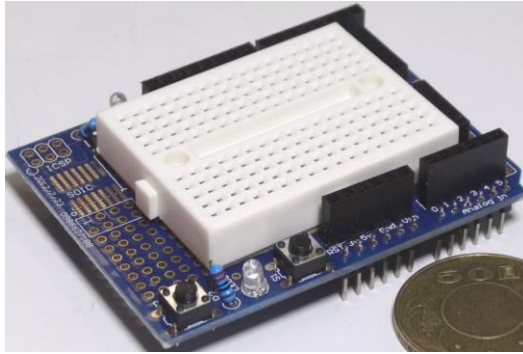
四、成本分析：arduino 微控器開發版、arduino I/O 擴充模組、超音波感測器、ethernet 網路擴充模組、水位感測模組、伺服馬達、繼電器、名片型 led 字幕機、蜂鳴器、電源供應器、LED 燈、輸出電路 (電容、電阻.....)、壓克力模型等。材料及金額表列如下：

(一)、主要零件圖片外觀:

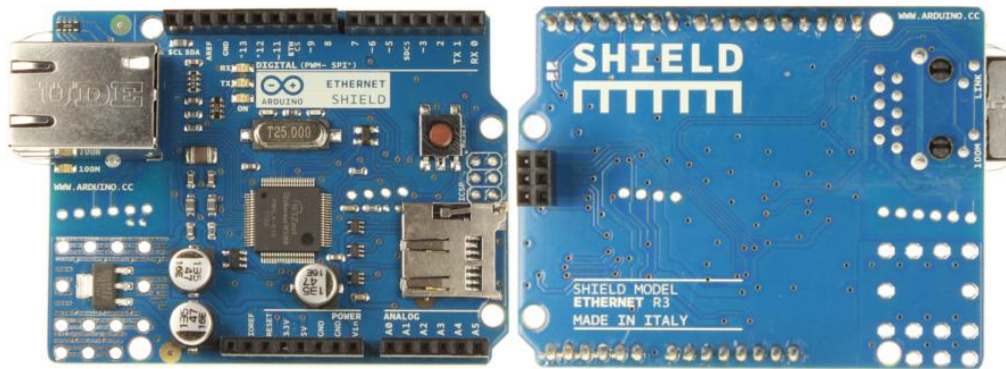
1.arduino 微控器開發版



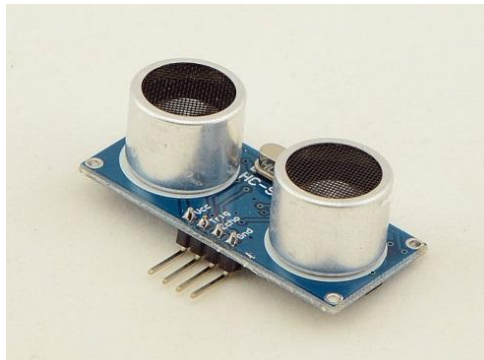
2.arduino 原形擴充模組:



3.ethernet 網路擴充模組:



4.超音波感測器



5. 自製水位感測棒模組



(二)、成本分析

總金額：5475 元。

元 件 名 稱	規 格 說 明	數 量	單 價	合 計
1.arduino 微控器開發版	Mega 256	2	250	500
2.arduino I/O 擴充模組	暫時不購買(進階)	0	1200	0
3.ethernet 網路擴充模組	暫時不購買(進階)	0	800	0
4.超音波感測器	Hc-sr04	2	80	80
5.水位感測模組	Keyes	1	80	80
6. 伺服馬達	Gs995	2	450	900
7.繼電器。	Keyes_srly	2	80	160
8.名片型 led 字幕機	8.*3.5*0.5cm	2	550	550
9. 蜂鳴器		2	30	60
10.電源供應器	9v	1	90	90
11.LED 燈	高亮度(R,G)	12	5	15
12.輸出電路 (電容、電阻)		4	10	40
13. 壓克力模型		1	3000	3000
			總 計	5475

五、作品說明圖說明:

1. 模型結構圖及控制電路：

(1) 模型結構圖

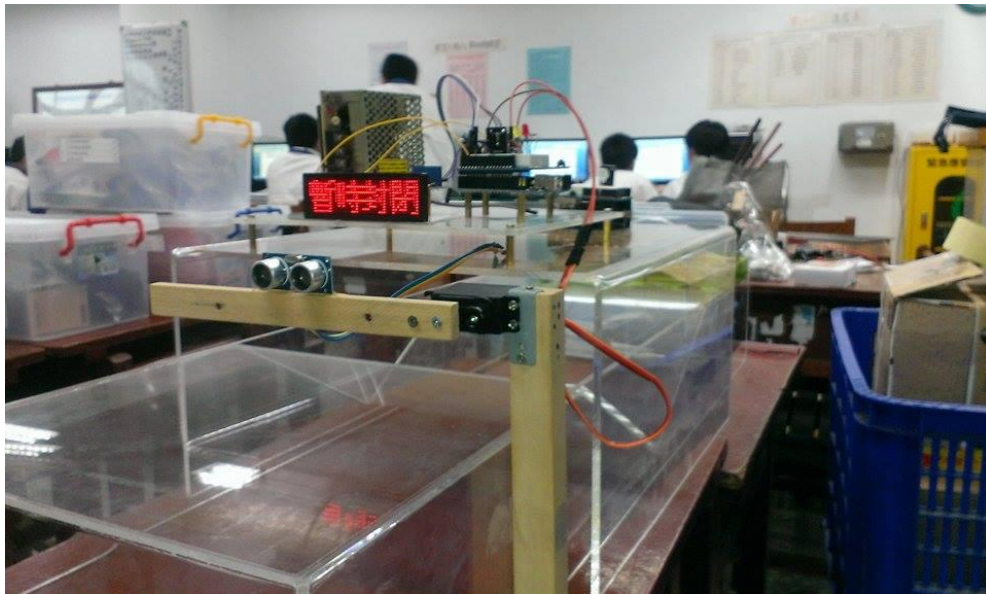
上視圖



側視圖(長)

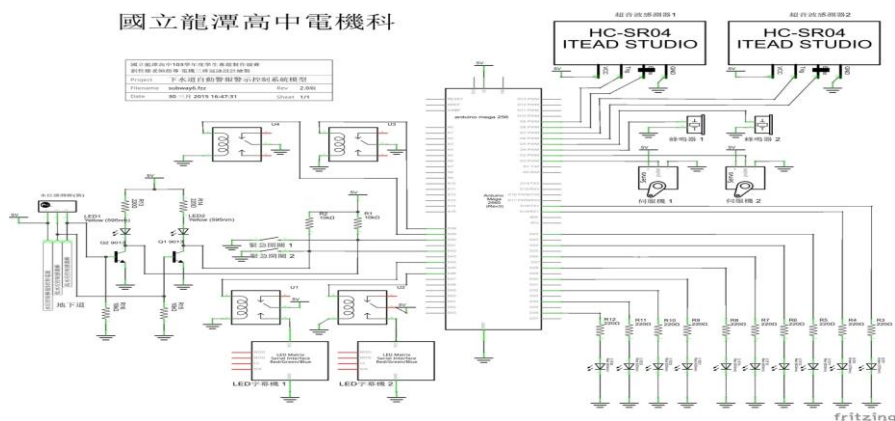


單一入口系統
側視圖(寬)



(2) 控制硬體電路:

① 硬體電路圖:



2. 程式碼註解說明:

功能說明:

利用本控制系統，在地下道的適當位置設置高低水位感測棒。

狀況一:而當地下道內無積水或低於 20 公分時，低水位感測棒偵測到後，arduino 微控器才將柵欄打開(上升)、LED 綠燈亮表示“安全”、LED 字幕機關閉，車輛可正常行駛。

狀況二:當積水高度超過 50 公分高水位感測器時，蜂鳴器響、入口柵欄放下，以管制車輛雙向通行，同時以 LED 字幕機顯示以下警語:『本下水道積水深度已超過 50 公分以上之危險水位,暫時封閉,雙向禁止通行,請改道行駛,切勿強行通過以免發生危險意外』,兩個 LED 紅燈交替閃爍表示提醒“危險”，入口柵欄放下之後蜂鳴器停止聲響、兩個 LED 紅燈亮，LED 綠燈滅。

狀況三:積水高度超過 50 公分高入口柵欄放下之後蜂鳴器停止聲響兩個 LED 紅燈亮，此時若有車輛欲強行通過，當車輛靠近柵欄上之超音波感測器 10 公尺以內時，蜂鳴器開始響起,愈靠近響聲愈急促，同時兩紅燈交替閃爍，以聲響與閃光加強提醒車輛駕駛人勿強行通過。

狀況四:積水高度回降至 20 公分高度以下時，回到**狀況一**

狀況五:而當地下道內無積水或水位低於 20 公分時，地下道發生必須封閉之緊急臨時情況(例如車禍、火災、維修工程、臨時交通管制及緊急避難...等等)，可以手動按下緊急開關，放下柵欄，並可手動切換字幕機預設顯示的地下道封閉警示文字內容(譬如: 車禍、火災、維修工程、臨時交通管制及緊急避難...等等)，柵欄上的紅燈雙閃，超音波如狀況三警報警示功能一樣。

狀況五:緊急狀況解除後，手動切換字幕機恢復為預設顯示的**狀況二**地下道積水封閉警示文字內容，再復歸緊急開關後、紅燈滅、柵欄升起，綠燈亮，回到**狀況一**的自動控制狀態。

以下為程式:

```
//地下道警報系統
//設定水位開關、超音波、蜂鳴器、紅綠 LED、伺服機腳位
#include <Wire.h>
#include <Servo.h>
#include <Ultrasonic.h>
#define ECHO_PIN1      6
#define TRIGGER_PIN1  7
#define ECHO_PIN2      8
#define TRIGGER_PIN2  9
Ultrasonic ultrasonic1(TRIGGER_PIN1, ECHO_PIN1);
```

```

Ultrasonic ultrasonic2(TRIGGER_PIN2, ECHO_PIN2);
Servo servoRight; //宣告右邊伺服馬達
Servo servoLeft; //宣告右邊伺服馬達
const int servoPin1 = 2; //伺服機 1
const int servoPin2 = 3; //伺服機 2
const int Alarm_PIN1 = 14; //蜂鳴器 1
const int Alarm_PIN2 = 16; //蜂鳴器 2
const int greenLed1 = 18; //綠燈 1 pin 左入口
const int greenLed2 = 19; //綠燈 2 pin 右入口
const int redLed11 = 22; //紅燈 11 pin 左入口
const int redLed12 = 24; //紅燈 12 pin 左入口
const int redLed13 = 26; //紅燈 13 pin 左柵欄
const int redLed14 = 28; //紅燈 14 pin 左柵欄
const int redLed21 = 30; //紅燈 21 pin 右入口
const int redLed22 = 32; //紅燈 22 pin 右入口
const int redLed23 = 34; //紅燈 23 pin 右柵欄
const int redLed24 = 36; //紅燈 24 pin 右柵欄
const int switchPin1 = 40; //低水位感測棒
const int switchPin2 = 42; //高水位感測棒
const int ledScreenPin1 = 44; //字幕機電源開關 1
const int ledScreenPin2 = 46; //字幕機電源開關 2
int servoState = 0; //兩伺服馬達轉角狀態值角度歸零
void setup() {
  //Serial.begin(9600);
  pinMode(switchPin1, INPUT); //低水位感測器
  pinMode(switchPin2, INPUT); //高水位感測器
  pinMode(ECHO_PIN1, INPUT); //超音波接收腳 1
  pinMode(ECHO_PIN2, INPUT); //超音波接收腳 2
  pinMode(TRIGGER_PIN1, OUTPUT); //超音波觸發腳 1
  pinMode(TRIGGER_PIN2, OUTPUT); //超音波觸發腳 2
  pinMode(servoPin1, OUTPUT); //伺服機 1
  pinMode(servoPin2, OUTPUT); //伺服機 2
  pinMode(greenLed1, OUTPUT); //綠燈 1
  pinMode(redLed11, OUTPUT); //紅燈 11
  pinMode(redLed12, OUTPUT); //紅燈 12
  pinMode(redLed13, OUTPUT); //紅燈 13
  pinMode(redLed14, OUTPUT); //紅燈 14
  pinMode(greenLed2, OUTPUT); //綠燈 2

```

```

pinMode(redLed21,OUTPUT); //紅燈 21
pinMode(redLed22,OUTPUT); //紅燈 22
pinMode(redLed23,OUTPUT); //紅燈 23
pinMode(redLed24,OUTPUT); //紅燈 24
pinMode(Alarm_PIN1,OUTPUT); //蜂鳴器 1
pinMode(Alarm_PIN2,OUTPUT); //蜂鳴器 2
pinMode(ledScreenPin1,OUTPUT); //字幕機開關 1
pinMode(ledScreenPin2,OUTPUT); //字幕機開關 2
servoRight.attach(servoPin1); //將 servoPin1 指定為伺服馬達驅動信號腳
servoLeft.attach(servoPin2); //將 servoPin2 指定為伺服馬達驅動信號腳
initialservo(); //伺服機角度歸零
}
//水位感測器讀取到達警戒高水位時 0V / 低水位時 5V
void loop() {
int switchState1 = digitalRead(switchPin1);
int switchState2 = digitalRead(switchPin2);
digitalWrite(Alarm_PIN2,LOW);
digitalWrite(Alarm_PIN1,LOW);
//狀況一:當水位感測器偵測到低於低水位以下時，綠色 LED 會發光表示目前為安全
狀態
if (switchState1 == HIGH && switchState2 == HIGH ) {
    digitalWrite(greenLed1,HIGH);
    digitalWrite(greenLed2,HIGH);
    digitalWrite(redLed11,LOW);
    digitalWrite(redLed12,LOW);
    digitalWrite(redLed13,LOW);
    digitalWrite(redLed14,LOW);
    digitalWrite(redLed21,LOW);
    digitalWrite(redLed22,LOW);
    digitalWrite(redLed23,LOW);
    digitalWrite(redLed24,LOW);
    digitalWrite(ledScreenPin1,LOW);
    digitalWrite(ledScreenPin2,LOW);
    initialservo();
    delay(100);
}
//狀況二:當水位高於高水位線，高水位感測器狀態 5V (HIGH)，蜂鳴器會發出警報
聲，

```

//led 字幕機啟動，同時兩紅 LED 燈會不同交替發光閃爍，伺服機轉 90 度放下柵欄後，兩紅燈亮，綠燈熄，。

```
if(switchState1 == LOW && switchState2 == LOW && servoState != 90 ) {
    delay(3000);
    digitalWrite(ledScreenPin1,HIGH);
    digitalWrite(ledScreenPin2,HIGH);
    digitalWrite(greenLed1,LOW);
    digitalWrite(greenLed2,LOW);
    digitalWrite(redLed11,HIGH);
    digitalWrite(redLed12,HIGH);
    digitalWrite(redLed13,HIGH);
    digitalWrite(redLed14,HIGH);
    digitalWrite(redLed21,HIGH);
    digitalWrite(redLed22,HIGH);
    digitalWrite(redLed23,HIGH);
    digitalWrite(redLed24,HIGH);
    int i1=0;
    for(i1=0; i1<=20; i1++){
        //tone(Alarm_PIN1,523);
        //tone(Alarm_PIN2,523);
        digitalWrite(Alarm_PIN2,HIGH);
        digitalWrite(Alarm_PIN1,HIGH);
        digitalWrite(redLed11,HIGH);
        digitalWrite(redLed12,LOW);
        digitalWrite(redLed13,HIGH);
        digitalWrite(redLed14,LOW);
        digitalWrite(redLed21,HIGH);
        digitalWrite(redLed22,LOW);
        digitalWrite(redLed23,HIGH);
        digitalWrite(redLed24,LOW);
        delay(100);
        digitalWrite(Alarm_PIN1,LOW);
        digitalWrite(Alarm_PIN2,LOW);
        digitalWrite(redLed11, LOW);
        digitalWrite(redLed12, HIGH);
        digitalWrite(redLed13, LOW);
        digitalWrite(redLed14, HIGH);
        digitalWrite(redLed21, LOW);
```

```

    digitalWrite(redLed22, HIGH);
    digitalWrite(redLed23, LOW);
    digitalWrite(redLed24, HIGH);
    delay(100);
}
if (servoState != 90) {
    servo90();
    servoState = 90;
}
int i2=0;
for(i2=0; i2<=6; i2++){
    digitalWrite(Alarm_PIN1,HIGH);
    digitalWrite(Alarm_PIN2,HIGH);
    digitalWrite(redLed11,HIGH);
    digitalWrite(redLed12,LOW);
    digitalWrite(redLed13,HIGH);
    digitalWrite(redLed14,LOW);
    digitalWrite(redLed21,HIGH);
    digitalWrite(redLed22,LOW);
    digitalWrite(redLed23,HIGH);
    digitalWrite(redLed24,LOW);
    delay(350);
    //tone(Alarm_PIN1,698);
    //tone(Alarm_PIN2,698);
    digitalWrite(Alarm_PIN1,LOW);
    digitalWrite(Alarm_PIN2,LOW);
    digitalWrite(redLed11, LOW);
    digitalWrite(redLed12, HIGH);
    digitalWrite(redLed13, LOW);
    digitalWrite(redLed14, HIGH);
    digitalWrite(redLed21, LOW);
    digitalWrite(redLed22, HIGH);
    digitalWrite(redLed23, LOW);
    digitalWrite(redLed24, HIGH);
delay(350);
}
digitalWrite(redLed11, HIGH);
digitalWrite(redLed12, HIGH);

```



```

digitalWrite(redLed13, HIGH);
digitalWrite(redLed14, HIGH);
digitalWrite(redLed21, HIGH);
digitalWrite(redLed22, HIGH);
digitalWrite(redLed23, HIGH);
digitalWrite(redLed24, HIGH);
digitalWrite(greenLed1,LOW);
digitalWrite(greenLed2,LOW);
digitalWrite(Alarm_PIN1,LOW);
digitalWrite(Alarm_PIN2,LOW);
}

```

//狀況三:在狀況二下有車輛靠近柵欄超音波測距警報啟動

```

if(switchState2 == LOW && switchState2 == LOW && servoState ==90 ) {
    //digitalWrite(Alarm_PIN1,LOW);
    //digitalWrite(Alarm_PIN2,LOW);
    UltrasonicARLAM1();
    noTone(Alarm_PIN1);
    UltrasonicARLAM2();
    noTone(Alarm_PIN2);
    digitalWrite(redLed11, HIGH);
    digitalWrite(redLed12, HIGH);
    digitalWrite(redLed13, HIGH);
    digitalWrite(redLed14, HIGH);
    digitalWrite(redLed21, HIGH);
    digitalWrite(redLed22, HIGH);
    digitalWrite(redLed23, HIGH);
    digitalWrite(redLed24, HIGH);
}
}

```

//超音波測距警報副程式 1

```

void UltrasonicARLAM1 () {
    delay(30);
    int cmMsec1;
    long microsec1 = ultrasonic1.timing();
    cmMsec1 = ultrasonic1.convert(microsec1, Ultrasonic::CM);
    //Serial.print(" CM1: ");
    //Serial.println(cmMsec1);
    if(cmMsec1 <15) {

```

```

    //Serial.println("too close");
    digitalWrite(Alarm_PIN1,LOW);
    tone(Alarm_PIN1, 523);
    delay(cmMsec1*4+10);
    tone(Alarm_PIN1,698);
    digitalWrite(redLed11,LOW);
    digitalWrite(redLed12,LOW);
    digitalWrite(redLed13,LOW);
    digitalWrite(redLed14,LOW);
    delay(cmMsec1*4+10);
}
digitalWrite(redLed11,HIGH);
digitalWrite(redLed12,HIGH);
digitalWrite(redLed13,HIGH);
digitalWrite(redLed14,HIGH);
noTone(Alarm_PIN1);
noTone(Alarm_PIN2);
}
//超音波測距警報副程式 2
void UltrasonicARLAM2 () {
    delay(20);
    int cmMsec2;
    long microsec2 = ultrasonic2.timing();
    cmMsec2 = ultrasonic2.convert(microsec2, Ultrasonic::CM);
    //Serial.print(" CM2: ");
    //Serial.println(cmMsec2);
    if(cmMsec2 <=15) {
        //Serial.println("too close");
        digitalWrite(Alarm_PIN2,LOW);
        tone(Alarm_PIN2,523);
        digitalWrite(redLed21,LOW);
        digitalWrite(redLed22,HIGH);
        digitalWrite(redLed23,LOW);
        digitalWrite(redLed24,HIGH);
        delay(cmMsec2*3+10);
        tone(Alarm_PIN2,698);
        digitalWrite(redLed21,HIGH);
        digitalWrite(redLed22,LOW);
    }
}

```

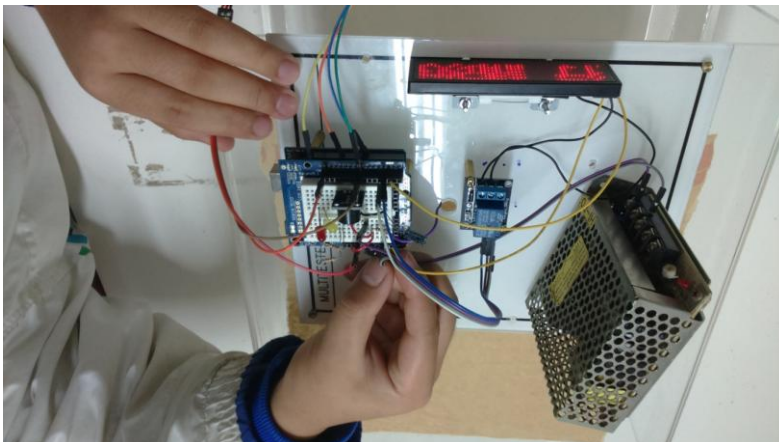
```

    digitalWrite(redLed23,HIGH);
    digitalWrite(redLed24,LOW);
    delay(cmMsec2*3+10);
}
digitalWrite(redLed21,HIGH );
digitalWrite(redLed22,HIGH);
digitalWrite(redLed23,HIGH );
digitalWrite(redLed24,HIGH);
noTone(Alarm_PIN2);
noTone(Alarm_PIN1);
}
//以下 initialservo 為副程式宣告
void initialservo() {
    servoRight.write(0);
    servoLeft.write(0);
    delay(500);
    servoRight.write(0);
    servoLeft.write(0);
    delay(500);
    servoState = 0;
}
//以下 servo90 為副程式宣告
void servo90() {
    int j1=0;
    int j2=0;
    for(j1=0; j1<=90; j1+=5) {
        servoRight.write(j1);
        servoLeft.write(j1);
        delay(2);
        for (j2=0;j2<5;j2++){
            digitalWrite(Alarm_PIN1,HIGH);
            digitalWrite(Alarm_PIN2,HIGH);
            digitalWrite(redLed11,HIGH);
            digitalWrite(redLed12,LOW);
            digitalWrite(redLed13,HIGH);
            digitalWrite(redLed14,LOW);
            digitalWrite(redLed21,HIGH);
            digitalWrite(redLed22,LOW);

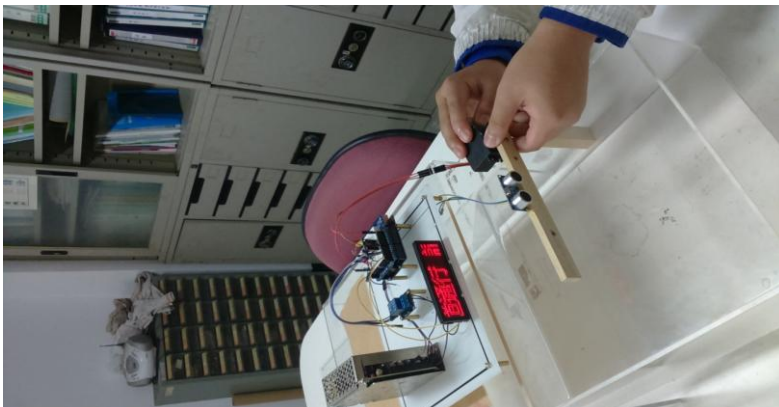
```

```
digitalWrite(redLed23,HIGH);
digitalWrite(redLed24,LOW);
delay(50);
digitalWrite(Alarm_PIN1,LOW);
digitalWrite(Alarm_PIN2,LOW);
digitalWrite(redLed11, LOW);
digitalWrite(redLed12, HIGH);
digitalWrite(redLed13, LOW);
digitalWrite(redLed14, HIGH);
digitalWrite(redLed21, LOW);
digitalWrite(redLed22, HIGH);
digitalWrite(redLed23, LOW);
digitalWrite(redLed24, HIGH);
delay(50);
}
}
servoState = 90;
}
```

電子電路焊接及面板製作圖



電路測試及組裝圖



六、參考資料

- 1.arduino 官方網站 <http://arduino.cc/>
- 2.最簡單的互動設計 Arduino 一試就上手 碁峰 孫駿榮、吳明展、盧聰勇
- 3.超圖解 Arduino 互動設計入門(第二版) 旗標 趙英傑
4. Arduino 互動設計專題與實戰 碁峰 柯博文
5. Arduino 最佳入門與應用 碁峰 楊明豐
6. Arduino 自造指南 碁峰 原文作者：John Boxall 譯者：曾繁勛,陳麒元,趙涵捷