

中華民國第 60 屆中小學科學展覽會 作品說明書

高級中等學校組 工程學(一)科

052305

最後的防線-平交道柵欄預警系統

學校名稱：四維學校財團法人花蓮縣四維高級中學

作者： 高二 謝尚均 高二 吳沁霖 高二 吳浩愷	指導老師： 方智暘 許益敏
---	-----------------------------

關鍵詞：交通安全、Arduino、光電感測器

摘要

平交道是我每天上下學的必經之路，而在 108 年九月六日花蓮中華路平交道發生了一啟死亡事故，事發當時，沒有人按下警鈴錯過了黃金救援時間而發生憾事。這啟事件不禁讓我對平交道的安全性打上了一個問號。所以我們希望運用在我們電子資訊學程中，學習到的程式撰寫技能製作，搭配 Arduino uno 板製作出一套能夠自動通報的系統，增加通過平交道時的安全性。

壹、研究動機

對於擁有環島鐵路的台灣人來說，火車是許多人來往各縣市的不二選擇。住在花蓮的我們來說，更是能體會火車對我們的重要性。火車的便利性對大眾來說，是無可取代的。但在便利之餘也衍生出了安全隱患，我們認為現在平交道的安全系統仍然不夠完善。

108 年九月六日，花蓮吉安發生一啟平交道死亡事故，小客車經過時柵欄正好放下，車子卡了 42 秒，直接被太魯閣列車撞翻，而事故發生當下沒有人去按下警鈴，錯過了黃金救援時間而導致憾事發生。要是能有個自動化的預警系統就能避免這類的憾事。平交道是我們每天通勤必經道路。我們希望能夠改善這類問題，將風險降到最低，我們決定做出一套自動預警系統來改善此問題。保護用路人安全。



(圖 1)



(圖 2)

貳、研究目的

近日平交道事故頻繁，生死只在一瞬間，但人往往在緊張時會忘記行動，這時就需要有個自動化的柵欄預警系統，來提醒用路人和列車應有的反應動作：比如地震警報系統，就是在人因地震驚慌時作動以提醒應變措施，便能降低災害的傷亡，而本研究目的在於：

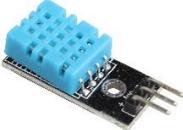
- 一、創造一個有效降低平交道事故發生的預警系統。
- 二、精進電子焊接技術與網路架設、程式撰寫之能力。
- 三、學習團隊合作、溝通，共同達成目標。

參、研究設備及器材

材料	數量	材料	數量
Arduino uno	2	D1 mini	1
超音波感測器	2	溫溼度感測器	2
光電感測器	6	蜂鳴器	2
OLED 螢幕	1	電腦	2
SG-90	2	CCD(針孔攝影機)	1

材料介紹

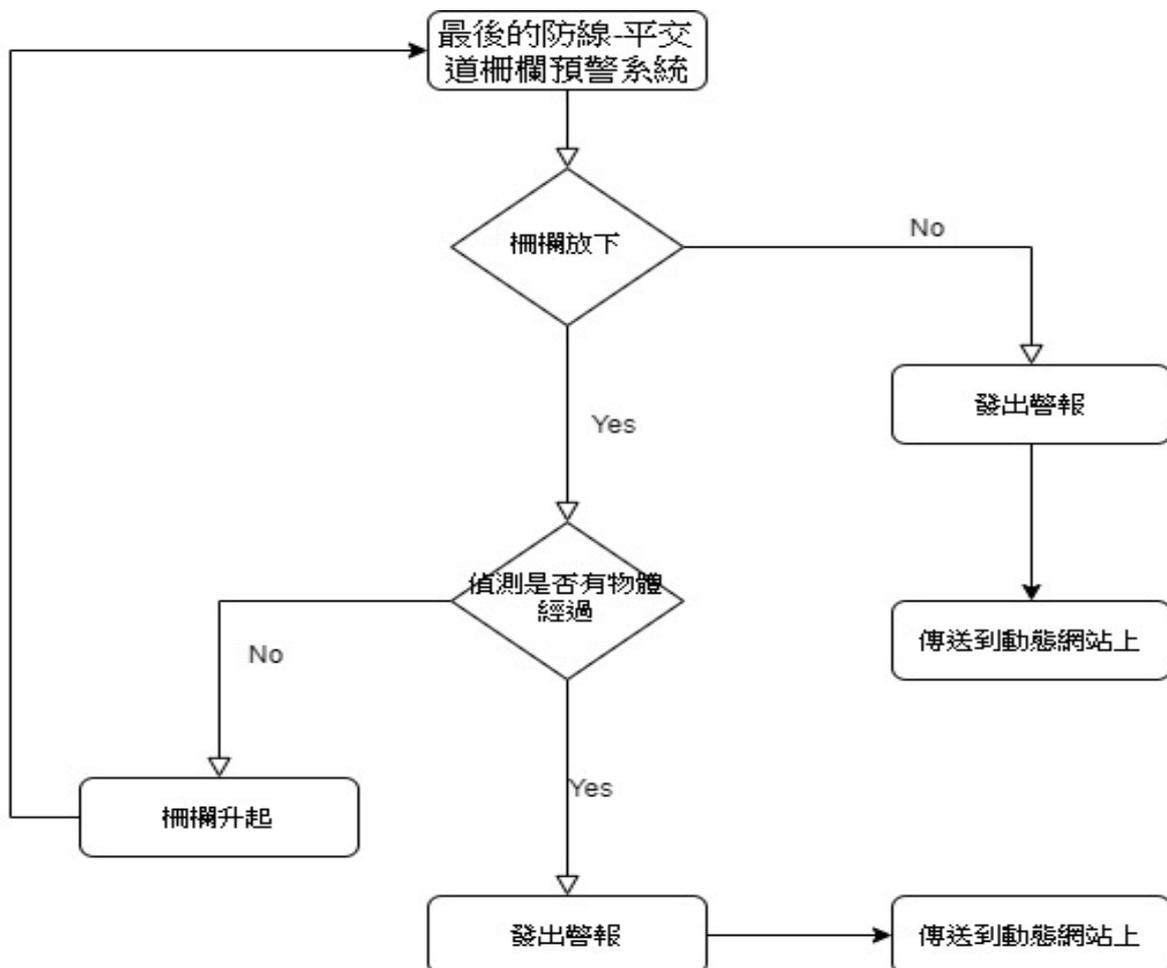
		
名稱：Arduino Uno 板	名稱：OLED 螢幕	名稱：蜂鳴器
型號：UNO	晶片型號：SSD1306	型號：MH-FMD
		
名稱：超音波感測器	名稱：對照式光電感測器	名稱：伺服馬達
型號：HC-SR04	型號：FC-33	型號：SG-90

	
名稱：迷你 Wi-Fi 開發板	名稱：溫溼度感測器
型號：D1 MINI	型號：DHT11

肆、研究過程或方法

一、構思：

鐵路為花蓮不可或缺的運輸工具，只要有意外事故發生影響大眾生活甚鉅，因此我們和指導老師分析了最近的時事，將主題定下，再藉由每人的特長分配工作，將主題發展成研究計畫。



二、程式：

程式設計是最令人頭疼的一部份，一個初觸程式的隊伍往往都需要經過網路查詢，再請益老師分析功能的來龍去脈，最後統整畫出流程圖，這才有了些頭緒但也因為不同觀點的思考邏輯造成組員間的激烈討論。

第一次成品

感測器程式：

我們的障礙物感測器是在光電感測器偵測到柵欄放下時開始啟動偵測，一但偵測到障礙物將會傳送一個 5V 的電壓給主控。然而在寫感測器的程式時，遇到了許多的 BUG 例如在柵欄放下時未偵測到障礙物蜂鳴器就做動了，以及兩個障礙物感測器同時偵測到障礙物才會有反應等等…都讓我們費盡心思的除 BUG

```
#define bz 9 //蜂鳴器接腳
//-----i2c
#include <Wire.h>
//-----伺服馬達
#include <Servo.h>
Servo servo;
int angle = 83;
byte servo_flag = 1;
//-----紅外線感測器
const int IR_rec = 2; // 紅外線接收器
byte IR_flag = 0;
//-----超音波感測器
#include <NewPing.h>
#define TRIG_PIN 12 // trigger Pin
#define ECHO_PIN 11 // Echo Pin
#define MAX_DISTANCE 200 // Maximum distance we want to ping for (in centimeters). Maximum sensor distance is rated at 400-500cm.
NewPing sonar(TRIG_PIN, ECHO_PIN, MAX_DISTANCE); // 設定 NewPing 物件·並給與最遠測試距離
int cm = 0;
//-----光遮斷
int buttonpin = 3; // 定義光遮斷器感測腳位為3
int buttonpin_val ;// 定義val
byte buttonpin_flag = 0;

void setup() {
  pinMode(5, OUTPUT);
  digitalWrite(5, LOW);
  //-----加入 i2c:2
  Serial.begin(9600);
  Wire.begin(2); //arduino join i2c bus on address 2
  Wire.onRequest(requestEvent); //when Request message, run requestEvent
  Wire.onReceive(receiveEvent); //when Receive message, run receiveEvent
  //-----蜂鳴器
  pinMode(bz, OUTPUT); //定義蜂鳴器為輸出端
  //-----伺服馬達
  servo.attach(8);
  servo.write(angle);
  //-----紅外線感測器
  pinMode(IR_rec, INPUT); // 把 irReceiver 接腳設置為 INPUT
  //-----光遮斷
  pinMode(buttonpin, INPUT); // 定義buttonpin為輸入腳位
}

//-----紅外線感測器
void IR() //OK!
{
  int IR_status = digitalRead(IR_rec); // 讀取 irReceiver 的狀態
  if (IR_status == 0) {
    byte IR_flag = 0;
    bz_up();
    // Serial.println("紅外線遮到");
  }
  else {
    byte IR_flag = 1;
    // Serial.println("紅外線沒遮到");
  }
}

//-----超音波感測器
```

```

void Ultrasound() //OK!
{
  unsigned int uS = sonar.ping(); // 送出 ping，並取得微秒 microseconds(uS) 時間
  // Serial.println(sonar.convert_cm(uS));
  cm = sonar.convert_cm(uS); // 換算時間為公分，如顯示 0，表示超出距離範圍
  if (cm < 5)
  {
    bz_up();
  }
}
//=====蜂鳴器響起
void bz_up() //OK!
{
  for (int i = 0; i <= 50; i++)
  {
    digitalWrite(9, 1);
    delay(1);
    digitalWrite(9, 0);
    delay(1);
  }
  digitalWrite(5, HIGH);
}
//=====光遮斷
void button() //OK!
{
  buttonpin_val = digitalRead (buttonpin); // 讀取buttonpin值
  if (buttonpin_val == HIGH) // 如果光遮斷器訊號被中斷
  {
    buttonpin_flag = 1;
    // bz_up();
    // Serial.println("遮到");
  }
  else
  {
    // Serial.println("沒遮到");
    buttonpin_flag = 0;
  }
}
//=====伺服馬達接收
void receiveEvent (int bytes)
{
  servo_flag = Wire.read(); // read one character from the I2C
}
//=====狀況回傳
void requestEvent () { //送資料
  int level = 1;
  Wire.write(level); //send level to master
}
//=====主程式
void loop () {
  digitalWrite(5, LOW);
  if (servo_flag == 1)
  {
    button();
    servo.write(180);
    if (buttonpin_flag == 1)
    {
      Ultrasound();
      IR();
    }
  }
}
}

```

主控程式：

在內部傳送我們使用 i2c 協定將判斷旗號以串列傳送做為溝通的橋樑與外部溝通我們則使用 d1 mini 以 wifi 協定將我們的資料部署至雲端伺服器，而在內部紀錄時間我們則使用 ntp client 將 linux 時間抓取下來，並且用係數分離做處理，然後再將處理完的時間單位，以陣列的方式用串列傳送部署到雲端伺服器。

```

//master_sender.ino
// Refer to the "slave_receiver" example for use with this
#include <Wire.h>
int send_event_start = 1;
int send_event_stop = 0;
int send_event_coming = 0;
//=====
//計時中斷
#include <Timer.h>
Timer tcb;
//=====
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <ESP8266mDNS.h>
#include <WiFiUdp.h>
#ifdef STASSID
#define STASSID "forpc"
#define STAPSK "123789456"
#endif
const char *ssid = STASSID;
const char *password = STAPSK;
ESP8266WebServer server(80);

unsigned int localPort = 2390;
IPAddress timeServerIP;
const char* ntpServerName = "time.nist.gov";
const int NTP_PACKET_SIZE = 48;
byte packetBuffer[NTP_PACKET_SIZE];
WiFiUDP udp;
char IN_s[14] = {          }; //IN:xx:xx(space*14 個)

void handleRoot() {
  char htmlStr[190];
  snprintf(htmlStr, 190,
           "<html><head><meta charset='utf-8' /><meta

           if ((epoch % 3600) / 60 < 10) {
             Serial.print('0');
           }
           Serial.print((epoch % 3600) / 60);
           Serial.print(':');
           if ((epoch % 60) < 10) {
             Serial.print('0');
           }
           Serial.println(epoch % 60); // print the second
           ***也許需要開放才能正常工作***
           */
           IN_s[0] = '|'; IN_s[1] = 'N'; IN_s[2] = '|'; IN_s[3] = ((epoch % 86400) /
           3600) / 10 + 0x30; IN_s[4] = ((epoch % 86400) / 3600) % 10 + 0x30;
           IN_s[5] = '|'; IN_s[6] = ((epoch % 3600) / 60) / 10 + 0x30; IN_s[7] =
           ((epoch % 3600) / 60) % 10 + 0x30; IN_s[8] = '|'; IN_s[9] = (epoch % 60) /
           10 + 0x30; IN_s[10] = (epoch % 60) % 10 + 0x30;
           }
           delay(1000);
         }
         }

void sendNTPpacket(IPAddress& address) {
  memset(packetBuffer, 0, NTP_PACKET_SIZE);
  packetBuffer[0] = 0b11100011;
  packetBuffer[1] = 0;
  packetBuffer[2] = 6;
  packetBuffer[3] = 0xEC;
  packetBuffer[12] = 49;
  packetBuffer[13] = 0x4E;
  packetBuffer[14] = 49;
  packetBuffer[15] = 52;
  udp.beginPacket(address, 123);
  udp.write(packetBuffer, NTP_PACKET_SIZE);
  udp.endPacket();
}

void writeToSerial_Start() {
  Wire.beginTransmission(2); // transmit to device #9
  Wire.write(send_event_start); // sends x
}

http-equiv='refresh' content='1' /><style>body {font-family:
Arial;}</style></head><body><h1>監測
台</h1><h1><h1>%s</h1></body></html>", IN_s);
server.send(200, "text/html", htmlStr);
}

void handleNotFound() {
  String message = "File Not Found\n\n";
  message += "URI: ";
  message += server.uri();
  message += "\nMethod: ";
  message += (server.method() == HTTP_GET) ? "GET" : "POST";
  message += "\nArguments: ";
  message += server.args();
  message += "\n\n";
  for (uint8_t i = 0; i < server.args(); i++) {
    message += " " + server.argName(i) + ": " + server.arg(i) + "\n";
  }
  server.send(404, "text/plain", message);
}

void show_time() {
  WiFi.hostByName(ntpServerName, timeServerIP);
  sendNTPpacket(timeServerIP);
  delay(10);
  int cb = udp.parsePacket();
  if (!cb) {
  } else {
    udp.read(packetBuffer, NTP_PACKET_SIZE);
    unsigned long highWord = word(packetBuffer[40], packetBuffer[41]);
    unsigned long lowWord = word(packetBuffer[42], packetBuffer[43]);
    unsigned long secsSince1900 = highWord << 16 | lowWord;
    const unsigned long seventyYears = 2208988800UL;
    unsigned long epoch = secsSince1900 - seventyYears;
    /*
    Serial.print("The UTC time is ");
    Serial.print((epoch % 86400) / 3600);
    Serial.print(':');
    Wire.endTransmission(); // stop transmitting
    //=====
    Wire.beginTransmission(3); // transmit to device #9
    Wire.write(send_event_start); // sends x
    Wire.endTransmission(); // stop transmitting
  }
}

void writeToSerial_Stop() {
  Wire.beginTransmission(2); // transmit to device #9
  Wire.write(send_event_stop); // sends x
  Wire.endTransmission(); // stop transmitting
  //=====
  Wire.beginTransmission(3); // transmit to device #9
  Wire.write(send_event_stop); // sends x
  Wire.endTransmission(); // stop transmitting
}

void receiveEvent(int bytes) {
  send_event_coming = Wire.read(); // read one character from the
  I2C
}

void setup() {
  Wire.begin(1); //以 Master 身分加入 I2C
  Serial.begin(9600);
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  udp.begin(localPort);
  /*
  若要指定 IP 位址，請自行在此加入 WiFi.config()敘述。
  WiFi.config(IPAddress(192,168,1,50), // IP 位址
              IPAddress(192,168,1,1), // 閘道 (gateway) 位址
              IPAddress(255,255,255,0)); // 網路遮罩 (netmask)
  */
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }
  server.on("/", handleRoot);
}

```

```

server.onNotFound(handleNotFound);+
server.begin();+
tcb.every(10000, writeToSerial_Start); // 每經過 10000 毫秒/10 秒，就
會呼叫 writeToSerial_Start+
tcb.every(20000, writeToSerial_Stop); // 每經過 10000 毫秒/10 秒，就
會呼叫 writeToSerial_Stop+
Wire.onReceive(receiveEvent);+
}+
}+
void loop() {+
server.handleClient();+
tcb.update();+
//=====+
// if(send_event_comeing = '1')+
// {+
//+
// }+
}

```

三、模型製作：

材料：

壓克力板	走線槽
3D 列印(鐵軌)	轉軸

機器

<p>鑽孔機</p> 	<p>線鋸機</p> 
<p>砂輪機</p> 	<p>沙帶機</p> 

鑽孔機：

雖然檢定有使用過，但是在鑽彈簧座的時候就發現問題了，一開始我們試著照畫出來的圓線鑽，但是由於技術不熟，導致位置偏移，於是我們修正將鑽孔機的鑽尾換成最粗的，鑽下去後再用砂紙將邊角打磨後就成功了。

線鋸機：

這是我們第一次使用線鋸機，一開始切割出來的成品真的是歪七扭八，經過指導老師及學長的教導後才開始越切越直，越切越順，只不過因失誤消耗掉一些壓克力板，讓我們覺得有些可惜。

砂輪機：

這也是我們第一次使用砂輪機切割金屬的走線槽，由於砂輪機沒操作好會很危險，所以老師全程都在旁邊指導，但因規劃的尺寸和切割成品有誤差，使我們反覆的操作才達到預定的尺寸。

沙帶機：

使用線鋸機後壓克力板的切割面並不平整，所以使用沙帶機將切割面磨平，以期我們組裝時模型能更美觀，但說的簡單做起來也需要相當的精神才能完成，有幾次的經驗都是不對稱的，在這上也花了許多的時間磨練。

第一次成品：

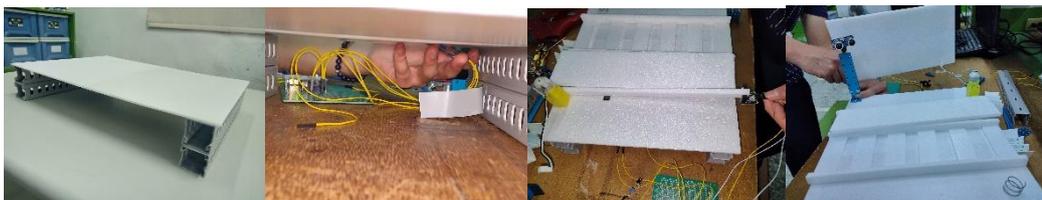
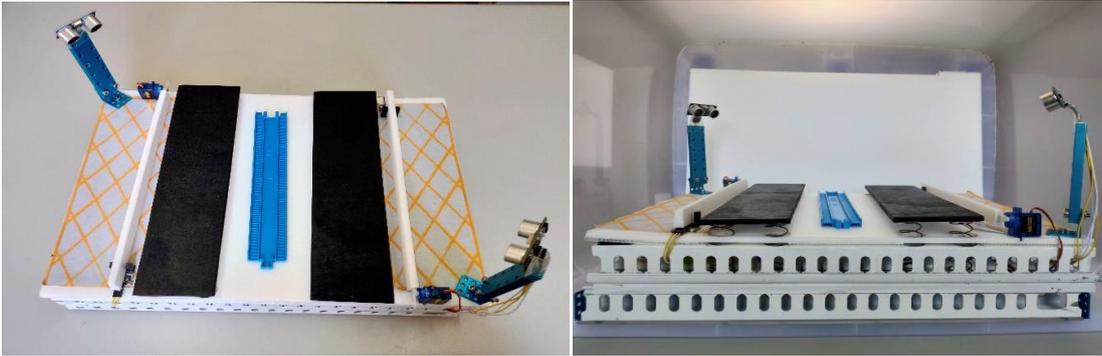
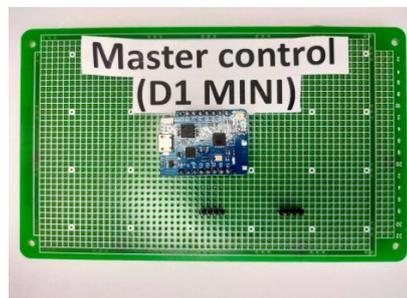
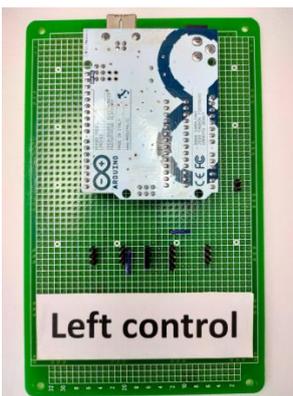


圖 3 模型雛形 圖 4 紅外線避障器架設 圖 5 柵欄架設 圖 6 整體預擺

製作模型腳架時忽略了模型材質載重能力與施力不均的問題，導致模型因過重而產生斷裂，所以我們將腳架改為擺放在長邊。增加了受力面積改善了載重能力不足的；架設紅外線避障器時，因為感測距離過長導致上蓋板放上去就觸發。所以我們就將紅外線避障器改為傾斜擺放，改善蓋板誤觸的問題。



超音波感測器我們採用對角擺放，將超音波打在鐵軌上，預防死角的產生，光電感測器則放在走線槽上面，利用走線槽的空洞將位置固定，而 SG-90 對稱的放在光電感測器的對面，以防止柵欄無法順利下降至感測光遮斷的範圍，而我們將彈簧與彈簧之間的距離設寬為 6.5 cm，長為 29 cm，彈簧壓縮的尺寸為 2cm，而木板則比彈簧長寬各加 1 公分，並將木板漆成黑色以防止紅外線避障器無法準確的感測有無物體經過。



電路板：

我們依照零件的需求，將電路板分成左控制、右控制、中央控制，為了方便安裝，左右的線路完全相同，但起初的電路規畫較為複雜，時常無法分辨位置浪費了許多時間，但經指導老師的調整後，我們重新規劃了電路、腳位、考量供電的問題後，才將電路完成。

第二次成品：

超音波感測器我們原封不動，數據穩定、不易產生誤差、功耗低，但為了能更精準地知道超音波的誤差我們加裝了溫溼度感測器，溫度及濕度會對超音波帶來多大的影響？這是我

們想探討並做出分析的；而紅外線避障器的誤差在我們的意料之外，發射紅外線的穩定度不夠，對接收紅外線的物件也相對嚴苛，我們決定將紅外線避障器換成光電感測器，光電感測器的穩定性相對高，能準確地讓我們知道有無物體經過，便能及時發出警報避免事故發生。而在柵欄旁我們新增了 CCD 即時影像，除了發出警報外也透過圖像讓鐵路局及列車長能依據不同狀況去選擇應變措施。

四、溝通：

人與人之間的溝通往往成為團隊成功的一大要件，身為高中生的我們也不例外，一個程式不同做法、一份設計圖不同想法、一份文書不同看法，即使將程式做得再完善，不經過溝通的淬鍊最終研究也不會完美的呈現。在我們這隊伍中也時常因為意見的不同，將研究放置一旁開始小組會議，雖然不見得有結果但也使我們更了解彼此的想訪及溝通的重要性，在整個研究中我們額外學習到有效的溝通。

伍、研究結果

一、主控程式

在設計主控時，我們所使用的是串列埠傳送，必須在板子之間建立好旗號的偵錯對照表才能使傳送或接收的板子正常動作；我們這次也增加了 OLED 顯示器利用它顯示主控的網路狀況，或是發生狀況的時間點及各種資訊。

二、感測器程式

我們這一次將程式的邏輯重新安排，使得程式運行更加快速同時也讓 CPU 負擔縮小；感測器部分我們也使得他更加貼近現實中的狀況。經過新的程式語法，這一次的感測器的反應速度更加靈敏，在初賽時遇到感測器沒有動作或是反應過度的 BUG 也一並做優化。

陸、討論

Q1：在使用過紅外線感測器後，我們發現效果差強人意，時常會因為光線、溫度、濕度、板子的反光而無法觸發感測器。

A：最終我們使用光電遮斷感測器，因為壓力感測器的感測點須在同一位置才會較為準確需要較多的感測器才能達到更加準確的效果，光遮就相對簡單，只需將遮斷器鑲入板子讓彈簧下壓時能觸發即可，簡單又能有效率的觸發警報，讓鐵路局及列車長能第一時間接收狀況，降低人命及財產的損失。

Q2：為什麼改用全壓克力作為模型設計？

A：為了讓評審更清楚的看見我們親手做的模型及電路板焊接內容。

Q3：為何由原本的動態網站改為靜態網站？

A：由於動態網站的不確定因素太多，例如在連接行動網路時遇到許多 IP 配發的問題，造成無法連線。所以改為固定式 IP 網站讓連線穩定不會有網路衝突。

柒、結論

平交道在我們的生活中極為重要，利用紅外線避障器做出壓力感測器、超音波感測器消除死角、光電感測器偵測柵欄放下，將平交道意外風險降低，但相較於傳統的按鈴通報增加了更多的安全性，讓人就算無法及時下車按下警報器也能夠自動通報，解決了事故發生時沒有人能救援的情況。雖然數據資料須再建置，模型較為陽春，但對於降低風險我們還是有足夠的信心。研究成果固然欣喜但在其中了解到更深層的意義，那這幾個月的努力便值回票價，縱然犧牲假日、放棄休閒娛樂、為全心灌注設計而住在學校，最終努力和成果獲得肯定，如能將研究應用在大眾社會中，那所做的犧牲也微不足道。然而在研究過程中學到的不單單是程式設計、材料選擇、模型組合、數據分析、比較材料的好壞，更是學到團隊合作的重要，如果沒有團隊合作也就沒有這成品的誕生。

參考資料

圖片：

圖一：交通部鐵路管理局關方網站

圖二：<https://www.cna.com.tw/news/超firstnews/201909060228.aspx>

D1MINI [https://www.google.com/url?sa=i&url=https%3A%2F%2Fnettigo.eu%2Fproducts%2Fwemos-d1-](https://www.google.com/url?sa=i&url=https%3A%2F%2Fnettigo.eu%2Fproducts%2Fwemos-d1-mini-v2-wifi-)
mini-v2-wifi-

[https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.elektor.com%2Fblue-0-96-oled-](https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.elektor.com%2Fblue-0-96-oled-display-spi-6-pin&psig=A0vVaw0h7M_fMEHVgkElW0QZ7QtR&ust=1591327943559000&source=images&cd=vfe&ved=2ahUKEwiZ1tLXnOfpAhVO5JQKH76Cg0Qr4kDegUIARCyAg)
display-spi-6-

[https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.elektor.com%2Fblue-0-96-oled-](https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.elektor.com%2Fblue-0-96-oled-display-spi-6-pin&psig=A0vVaw059S2oNpPw6Wry1gspKBZj&ust=1591327571657000&source=images&cd=vfe&ved=0CA0QjhxcFwoTCPjI8L2d5-kCFQAAAAAdAAAAABAD)
display-spi-6-

[https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.elektor.com%2Fblue-0-96-oled-](https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.elektor.com%2Fblue-0-96-oled-display-spi-6-pin&psig=A0vVaw059S2oNpPw6Wry1gspKBZj&ust=1591327571657000&source=images&cd=vfe&ved=0CA0QjhxcFwoTCPjI8L2d5-kCFQAAAAAdAAAAABAD)
display-spi-6-

串列埠測試

<http://yhhuang1966.blogspot.com/2015/09/arduino.html>

零件

紅外線避障器：<https://o4043380.pixnet.net/blog/post/224062414-arduino>

超音波感測器 <https://coopermaa2nd.blogspot.com/2011/01/arduino-lab15.html?m=1>

蜂鳴器 <https://coopermaa2nd.blogspot.com/2011/01/arduino-lab15.html?m=1>

新聞來源

花蓮平交道事故：<https://udn.com/news/story/7320/4032986>

【評語】 052305

1. 本研究製作出一套能夠自動偵測與通報的系統，增加平交道運作的安全性，議題具實用性，值得鼓勵。
2. 實際應用情境方面，可加強錯誤警報的情境設定與預防。
3. 研究應說明相關研究的現況及研究創新點。
4. 目前較無清楚的實驗結果來展示和證明系統運作的狀態和表現。
5. 目前超音波架設方式需要修正。一為兩超音波有可能互相接收到對方發射的超音波而誤判，另一為超音波偵測範圍為 lobe 外形，會傳回偵測範圍內「最短有障礙物的距離」，目前裝設方式有向下傾斜，若障礙物的距離比偵測範圍內最近的地面遠，傳回都是地面的距離，無法判知是否有障礙物在更遠的地方。
6. 一般平交道都具有監視器，可評估以「影像」方式來判定是否異常。藉由比對圖片，就可以知道柵欄是否有放下，以及知道柵欄圍起的空間內是否有異物。

研究動機

火車的便利性對大眾來說是無可取代的，但便利之餘也衍生出安全隱患，期望藉由改善自動化警報系統，將風險降到最低，避免因柵欄未放下的憾事發生，因此決定做出一套自動警報系統來改善此問題，保障用路人安全。

研究目的

1. 創造一個有效降低平交道事故發生的系統。
2. 精進電子焊接技術與資訊撰寫程式之能力。
3. 學習團隊合作、溝通，共同達成目標。
4. 訓練資料收集、整合與表達的技巧。

重大事故新聞



花蓮市中華路平交道今天下午約2點半有輛轎車闖越平交道，遭太魯閣號當場撞翻，駕駛被警消救出時已無生命徵象，送醫急救仍不治。(記者王錦義攝)

108年9月6日，花蓮吉安發生一起平交道死亡事故，小客車經過時柵欄正好放下，車子一動也不動的卡了42秒，直到被太魯閣列車撞翻當下沒有人去按下警鈴，錯過了黃金救援時間而導致憾事發生。

資料來源：<https://udn.com/news/story/7320/4032986>

供電異常 花蓮4平交道號誌故障台鐵搶修中

2020-04-18 14:45 聯合報 / 記者王燕華 / 花蓮即時報導

+ 火車站



花蓮地區火車供電系統今天中午突然發生異常，4個平交道柵欄故障，其中豐川平交道已在下午1時40分修復。記者王燕華 / 攝影

2020年4月18日中午12時50分發現供電異常，從花蓮北埔到吉安，包括豐川、建國路、中華路和宜昌4個平交道號誌故障，柵欄沒辦法放下。王姓市民今天中午路過中華路平交道，正好遇到號誌失靈，導致周邊交通大打結。

資料來源：<https://udn.com/news/story/7328/4501518>

應對方法

使用各項低功耗感測器降低供電不足的可能，並且加上備用電源防止斷電無法正常工作。

假如柵欄在時間內未放下，蜂鳴器將發出警報提醒用路人。

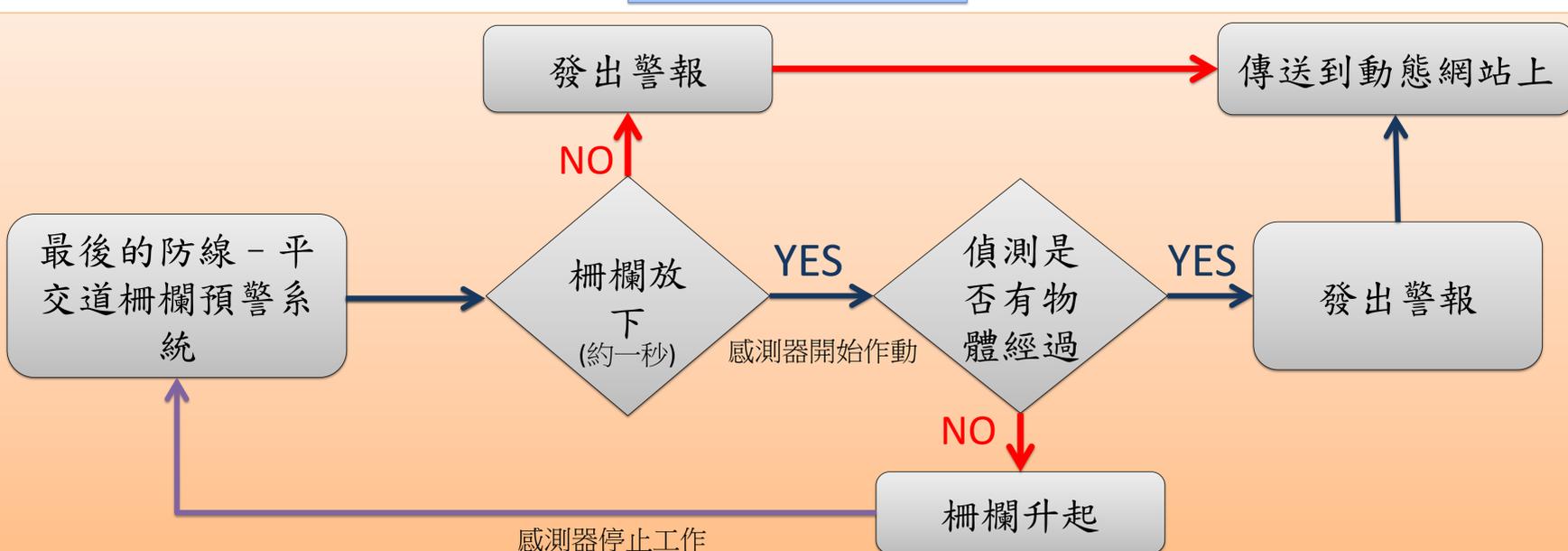
各類火車反應時間

公式： $(1400 - \text{煞車所需距離}) \div \text{列車行駛速度(秒/公尺)}$

資料來源：台鐵工作人員

列車種類	營業最高時速	煞車所需距離	時間反應
普悠瑪號	140	650公尺	19.28s
太魯閣號	140	650公尺	19.28s
自強號	120	800公尺	18.00s
莒光號	110	700公尺	22.9s
區間車	110	700公尺	22.9s

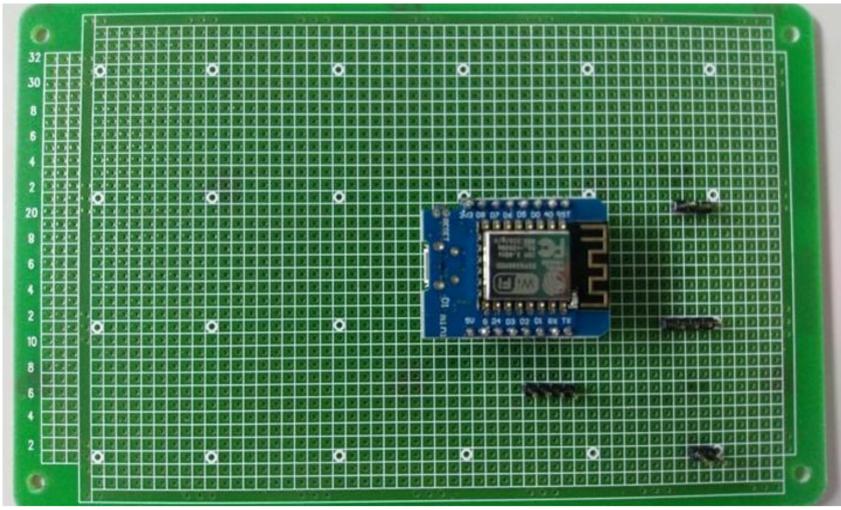
程式流程圖



電路設計

程式運作原理

MASTER

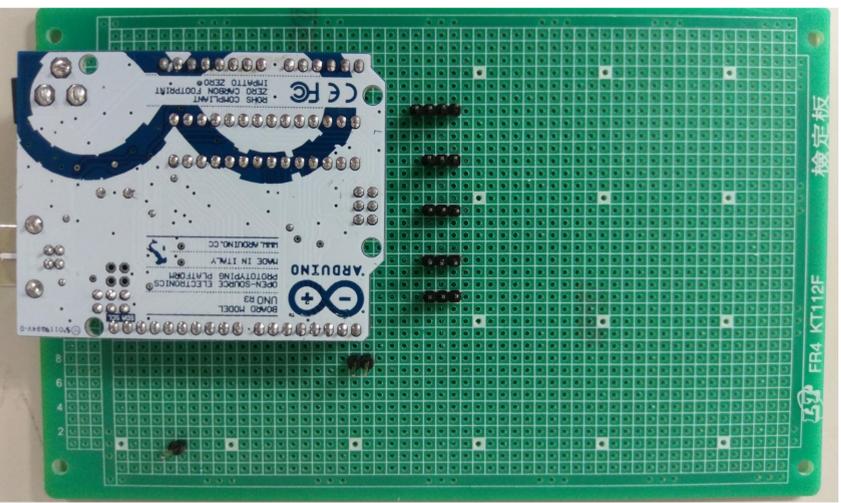


函數定義:
 D1 MINI的函數導入
 呼叫UDP函數
 建置HTML檔
 定義DHT11接腳(3)
 設定OLED初始層面
 建設DHT11副程式
 建設NTP
 設定NTP

初始化D1MINI:
 設定鮑率(115200)
 將DHT11初始化
 初始化WIFI
 判斷是否連線
 將網頁架設在根目錄上
 將伺服器啟動
 初始化OLED
 等待1.5秒後系統正式啟動

迴圈:
 更新網頁
 更新MDNS
 更新溫溼度資料
 檢查CLIENT端有無傳送資料
 有狀況發生時上傳時間與事件到網頁
 EEPROM儲存
 EEPROM讀取上次儲存資料

CLIENT



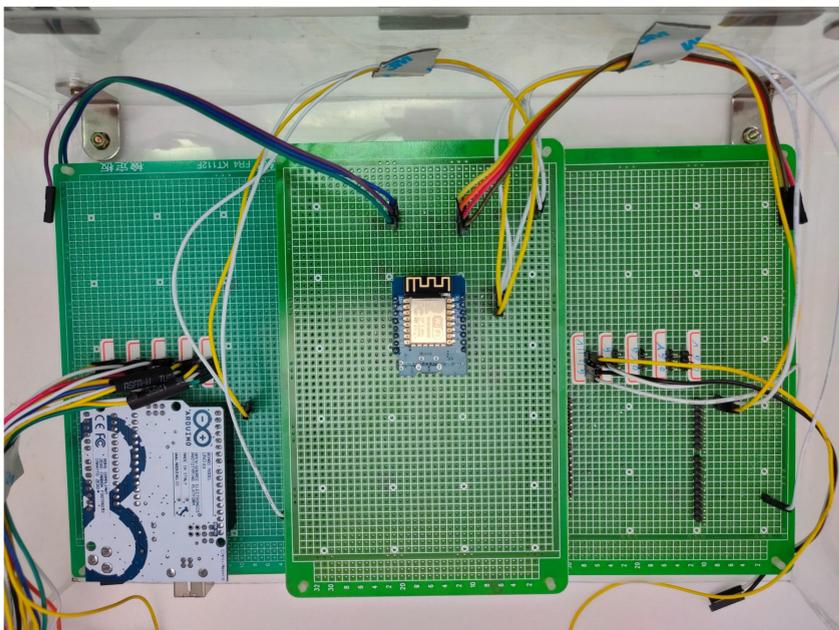
函數定義:
 定義蜂鳴器接腳(9)
 導入伺服馬達程式庫設定初始角度
 導入超音波程式庫
 定義超音波接腳(11)(12)
 設定最大偵測距離
 定義兩個光遮斷接腳(3)(5)
 建置時鐘計時常式(ISR)
 導入TIMER1中斷程式庫
 建置超音波副程式
 建置蜂鳴器副程式
 建置偵測柵欄副程式
 建置壓力感測副程式

初始化:
 設置鮑率為115200
 設定各元件輸出和輸入
 啟動TIMER1

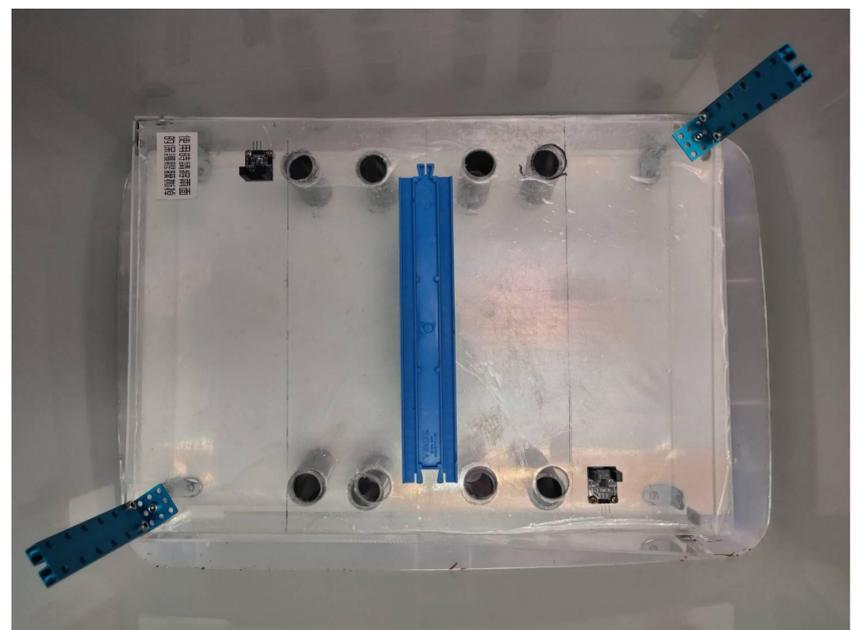
主程式:
 TIMER1更新
 判斷柵欄狀態
 如果沒放下則繼續偵測
 柵欄放下
 超音波和壓力感測器偵測
 如果發生狀況則傳送旗號置D1MINI

程式運作原理

模型建構



CONTROL: Arduino板 我們用電路板焊接將感測器所需的接腳焊上，方便我們測試時可以隨時拆裝，經過初賽的經驗，我們多加設了控制盒，因為在原模型裡走線空間過小無法順利將線路確實固定在接腳上，常有接觸不良的問題，增加了控制盒使我們的走線空間順暢了許多，而在控制盒的上蓋我們使用轉軸設計，能更順利地運用控制盒。



平交道模型:我們的模型一開始是運用所學到的autocad模擬模型，超音波感測器我們採用對角將超音波打在鐵軌上，預防有死角的產生，光電感測器我們則放在水管與水管之間，利用水管的間隙將感測器固定，而SG-90對稱的放在光電感測器的對面，以防止柵欄無法順利地放在光遮斷的範圍，而我們將彈簧與彈簧之間的距離設為寬為6.5，長為29，彈簧壓縮的尺寸為2cm，重量數值比:1:1000，目的是為了模擬車輛重量。此外我們加入了轉軸設計，以方便我們對感測器進行調整。

結論

這次平交道柵欄預警系統的研究讓我們知道雖然台灣的鐵路相較於國外已經相當完善了，卻依然會有許多平交道事故的發生，而在發生狀況的當下通常都無法及時反應，錯過了黃金的救援時間，所以我們試著用一套低成本卻能有效降低意外發生的系統，讓這套系統可以有效運用在平交道上來保障我們這些用路人的安全。

平交道事故頻傳！台鐵砸8億裝偵測器 雷達掃「地面30cm~3km障礙物」



▲台鐵平交道設備老舊。(圖/台鐵局提供)

記者李宜秦／台北報導

交通部台鐵局去年11月22日發生平交道事故，第一時間指稱是因小客車撞闖平交道釀禍，事後才確認是平交道故障。為有效減少事故發生，台鐵局投入8億元經費，安裝「平交道障礙物自動偵測系統」，只要是平交道範圍地面上30公分起到3公尺處，有障礙物都可透過雷達偵測，今年底將先啟用100處，明年底正式上路。

資料來源 <https://www.ettoday.net/news/20200214/1645721.htm>



▲台鐵花費8億裝設平交道偵測器。(圖/台鐵局提供)

我們研發這套系統的目的與台鐵相同，都是為了降低事故發生機率並保障用路人的生命安全，使平交道的危險性降至最低。

目前，台鐵推出了最新的雷達警報系統，雖然這系統在精確度上很優秀，但是造價昂貴、運用範圍不廣，而且也無法輕易移動調整，所以比較適合在固定位子進行偵測；而我們的系統價格低廉、機動性佳、運用範圍廣，適合運用在各類工程在運作施工時方便警示提醒。

未來方向

我們這次製作的系統主要是監測車輛之類較重的物體是否在柵欄放下後闖入或者卡在鐵軌上，對於一些擅闖平交道的行人或腳踏車之類較小較輕的物體無法偵測；針對此問題我們討論出可以增加人體紅外線感測器來偵測是否有人擅闖平交道。除此之外增加了人體紅外線感測器的系統也可以運用在居家警報或者危險區域的偵測上，使其運用範圍更廣闊。

圖片來源: <https://tutorials.webduino.io/images/zh-tw/docs/socket/sensor/pir-01.jpg>

當柵欄放下後感測器開始作動，如有發生狀況，後台人員會收到鐵路不同情況所傳送的不同訊號，要是能再利用台鐵的即時追焦系統讓監測方可以即時監測現場畫面，就可以使其立即做出相對應的反應，或者利用監測畫面對現場狀況進行即時掌控。

圖片來源: http://www.comc.ncku.edu.tw/chinese/a_news/images/mindulle/mindulle_0716.htm



比較圖

	價格	精確度	功率消耗	機動性
台鐵雷達警報系統	昂貴	優	高	良
平交道柵欄預警系統	低廉	良	低	優