投稿類別：資訊類

篇名:智慧型紅綠燈

作者:
王建懿。臺北市立南湖高級中學。三年十班
楊昊軒。臺北市立南湖高級中學。三年十班
朱星翰。臺北市立南湖高級中學。三年十班

指導老師:
涂朝淵老師

壹、前言

一、問題分析

臺北市的學生每天都為通勤煩惱，就算臺北市公共交通建設非常的完善，像是公車和捷運，每天都承載著將近 500 萬的搭乘人數。但是，臺北進入內湖和汐止進入內湖的區段因為受限於地形，加上內湖是整個臺北市電子資訊類工作人數最多的，即使有如此便利的大眾運輸，對於內湖堵塞如此嚴重的情況簡直是杯水車薪，再加上公車系統受限於路上交通車流量，使得內湖區的交通雪上加霜。

二、問題探討

根據『臺北市區瓶頸路口交通問題及其改善策略之研究』(陳增祥，1988)一文中，本組將臺北市交通壅塞原因整理如下：

(一) 目前紅綠燈的紅燈時間過長，無法彈性調配，導致車流堵塞在紅燈前，進而使後方的車輛無法前進，惡性循環下導致交通問題不斷惡化。

(二) 由於目前交通部的設施，紅綠燈秒數由尖峰時段進行調整，而主要幹道的紅燈時間較短來避免堵塞，相對的導致穿越主要幹道的人行道綠燈時間過短，這樣除了人群卡在人行道上造成的另類堵塞，還有過馬路時，有些兒童和老人由於過馬路時間過慢，可能會有交通安全上的疑慮。還有消防車和救護車因為車流堵塞而無法及時趕到現場的案例層出不窮，造成此原因的不外乎是紅燈過長的秒數，還有過多的車流量。

(三) 因目前採用固定秒數的交通號誌，當路口繁忙壅塞時，會導致堵塞狀況繼續惡性循環，無法自行舒緩堵塞狀況，需要大量交警人力即時到現場進行指揮交通，才可以有效的疏散堵塞車輛。

貳、正文

一、現象觀察與問題提出

臺北的交通一直都是交通部的頭痛問題，儘管地下的交通網已經如此發達，但是地面上的交通問題還是沒有改善，所以本組開始就地面上的交通問題去做討論和資料查詢，觀察到臺北的交通問題主要是出現在密度過高的紅綠燈，還有紅綠燈秒數沒有依照塞車路段去做調配導致塞車路段堵塞更嚴重，所以本組希望建立一個有效的系統，彷彿每個路口都有專門的交通警察在指揮，更有效率的解決交通問題。

二、如何建立有效率的紅綠燈交通系統

利用偵測器去偵測車流的堵塞量，再依據車流的堵塞量去調配紅綠燈的秒數速率，讓車輛堵塞的道路，擁有較快的紅燈秒數速率，而車流順暢的道路可以減少綠燈秒數，讓順暢的部分減少秒數來舒緩堵塞的部分，像是有交通警察在指揮交通一樣。

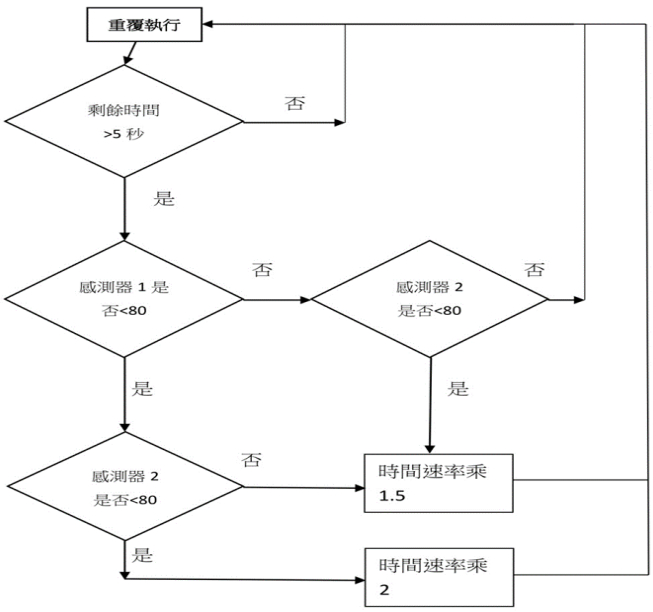
三、創作方法與設備

(一) 使用設備及器材

表一：設備器材表

編號	項目	數量
1	Arduino MEGA 板	1
2	計時器	1
3	七段顯示器	2
4	杜邦線&電線	數條
5	麵包板	2
6	PP 板	2
7	紙板	1
8	超音波感測器	2
9	紙盒	1

(二) 系統流程

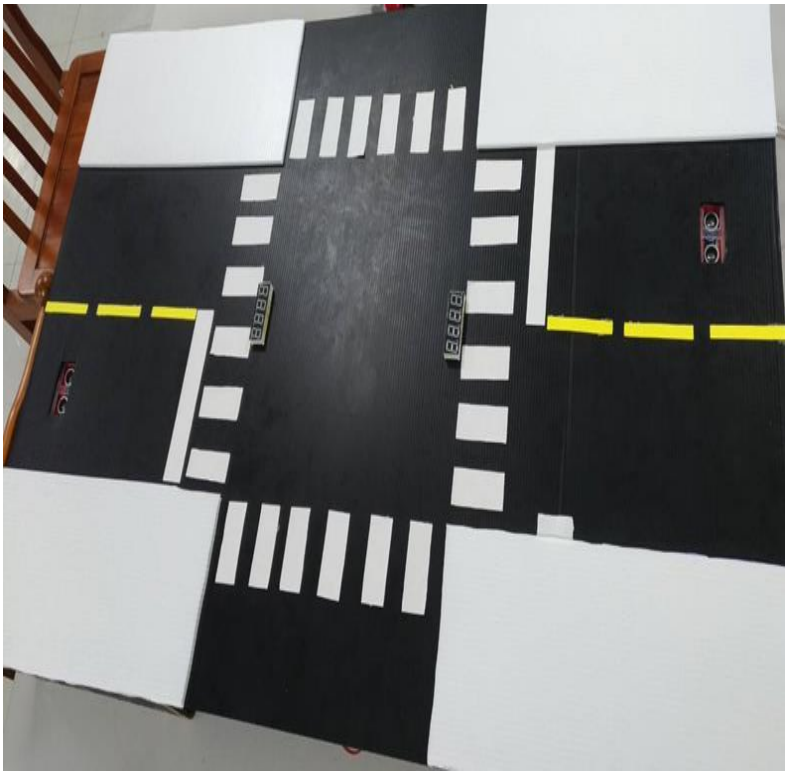


圖一、系統流程圖(資料來源：研究者繪製)

(三) 系統流程說明

(1) 用超音波偵測器，將其埋在馬路正中央的地下約 5 公分，所在位置位於離路口約整條路的 1/2 處，如圖二。

(2) 依據射出的波反彈回來的時間測得距離，若距離小於 20 公分，即表示目前該段的堵塞程度已到達整條路的 1/2 處，即將紅燈秒數速率加快。 (汽車百科:最小離地高度，2018)



圖二、模型俯視圖

(資料來源：研究者拍攝)

(四) 撰寫的程式碼(部分)(楊明豐，2016)

```
float sensor(int);
float sensor(){
  int duration_1;
  digitalWrite(tring_1,HIGH);
  delayMicroseconds(1000000);
  digitalWrite(tring_1,LOW);
  duration_1=pulseIn(echo_1,HIGH);
  float distance_1=(duration_1/2)/29;
  time_1=time_1+inter_time_1;
  delay(inter_time_1);
//  pf("distance_1============%d\n",
  return distance_1;
```

圖三、為額外寫的超音波感測器函數

```
void loop() {
// 經過一秒後就讓number減1

  unsigned long time_now = millis();
//Serial.print("--AA--\n");


int duration_1, duration_2;
digitalWrite(tring_1,HIGH);
delayMicroseconds(1000);
digitalWrite(tring_1,LOW);
duration_1=pulseIn(echo_1,HIGH);

digitalWrite(tring_2,HIGH);
delayMicroseconds(1000);
digitalWrite(tring_2,LOW);
duration_2=pulseIn(echo_2,HIGH);

//pf("---------------duration_1=%d-------

if (number<=5)// (no car)
  {
//    pf("NO_car_Time_now_number=%d\n", t
//    pf("No_car_Time_previous_number=---'
      if(time_now - time_previous<= 150)
        {
          number--;
          time_previous +=1000;
//          pf("number=%d\n", number);
           if(number < 0)
```

圖四、為重複執行，不停重新寫入數據，還有七段顯示器的速率

(資料來源：研究者拍攝)

(五) 撰寫的程式碼(部分)(趙英傑，2017)

```
if (duration_1 > 500 && duration_2>500 && number>5)//
  {
//     pf("NO_car_Time_now_number=%d\n", time_now);
//     pf("No_car_Time_previous_number=---%d\n\n\n\n\n"
       if(time_now - time_previous<= 150)
         {
           number--;
           time_previous +=1000;
//           pf("number=%d\n", number);
            if(number < 0)
              resetNumber(99);
         }
  }

if (duration_1 < 500 && duration_2 > 500 && number>5)
  {
//     pf("Has_car_Time_now_number=%d\n", time_now);
//     pf("Has_car_Time_previous_number=---%d\n\n\n\n\n'
      if(time_now - time_previous<= 150)
       {
         number--;
         time_previous +=600;
//         pf("number=%d\n", number);
          if(number < 0)
          resetNumber(99);
       }
  }
```

圖五、判斷感測器處是否塞車，還有時間是否大於五秒

```
 if (duration_1 > 500 && duration_2 < 500 && number>5)
  {
//     pf("Has_car_Time_now_number=%d\n", time_now);
//     pf("Has_car_Time_previous_number=---%d\n\n\n\n\n",
      if(time_now - time_previous<= 150)
       {
         number--;
         time_previous +=600;
//         pf("number=%d\n", number);
          if(number < 0)
          resetNumber(99);
       }
  }

 if (duration_1 < 500 && duration_2 < 500 && number>5)
  {
//     pf("Has_car_Time_now_number=%d\n", time_now);
//     pf("Has_car_Time_previous_number=---%d\n\n\n\n\n",
      if(time_now - time_previous<= 150)
       {
         number--;
         time_previous +=300;
//         pf("number=%d\n", number);
          if(number < 0)
          resetNumber(99);
       }
  }

  // 不斷地寫入數字
  setNumber(number);
```

圖六、判斷感測器處是否塞車，還有時間是否大於五秒

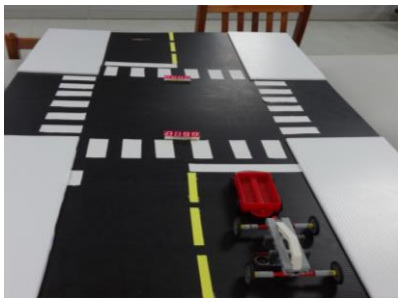
(資料來源：研究者拍攝)

五、模型預演操作情況

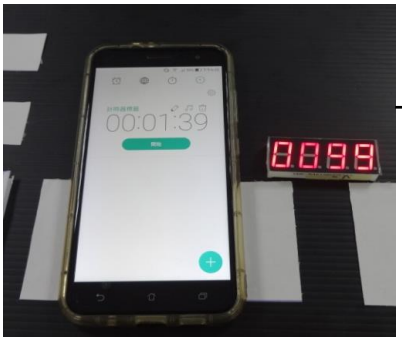
使用原本紅燈等待時間 99 秒(最長兩位數紅綠燈)為基礎(手機計時器代表正常時間)

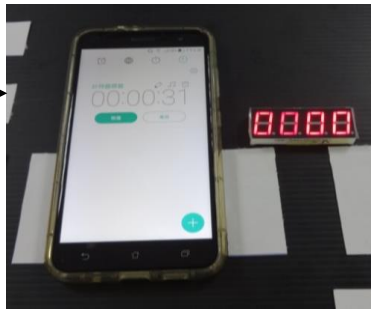
(一) 1 號感測器塞車時，紅綠燈的速度會比原本快 31 秒

(99-5)/1.5+5=68，99-68=31，如下圖七、圖八、圖九。



圖七、 1 號塞車表示圖





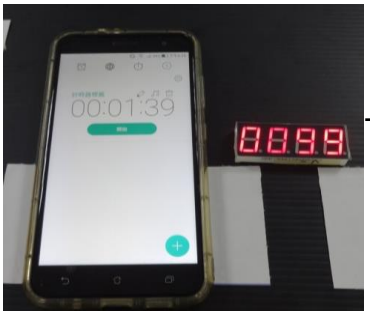
圖八、1 號塞車時間(初)　　圖九、1 號塞車時間(末)

(資料來源：研究者拍攝)

(二) 2 號感測器塞車時，紅綠燈的速度會比原本快 31 秒

(99-5)/1.5+5=68，99-68=31，如下圖十、圖十一、圖十二。

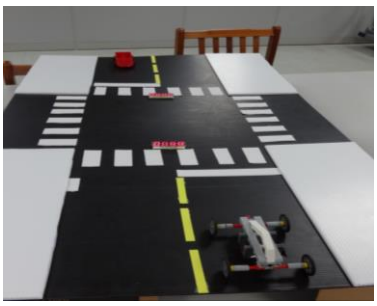
圖十、2 號塞車表示圖



圖十一、2 號塞車時間(初)　　圖十二、2 號塞車時間(末)

(資料來源：研究者拍攝)

(三) 當兩個感測器同時塞車時，紅綠燈的速度會比原本快 47 秒
(99-5)/2+5=52，99-52=47, 如下圖十三、圖十四、圖十五。



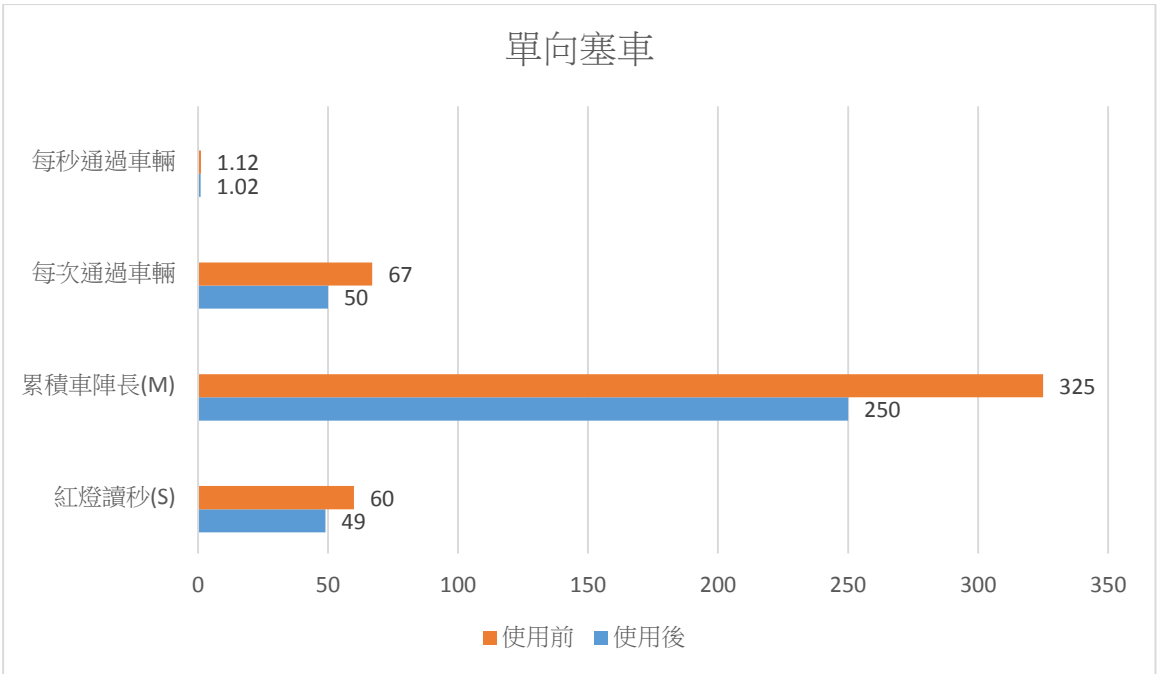
圖十三、1，2 號塞車表示圖

圖十四、1，2 號塞車時間(初)

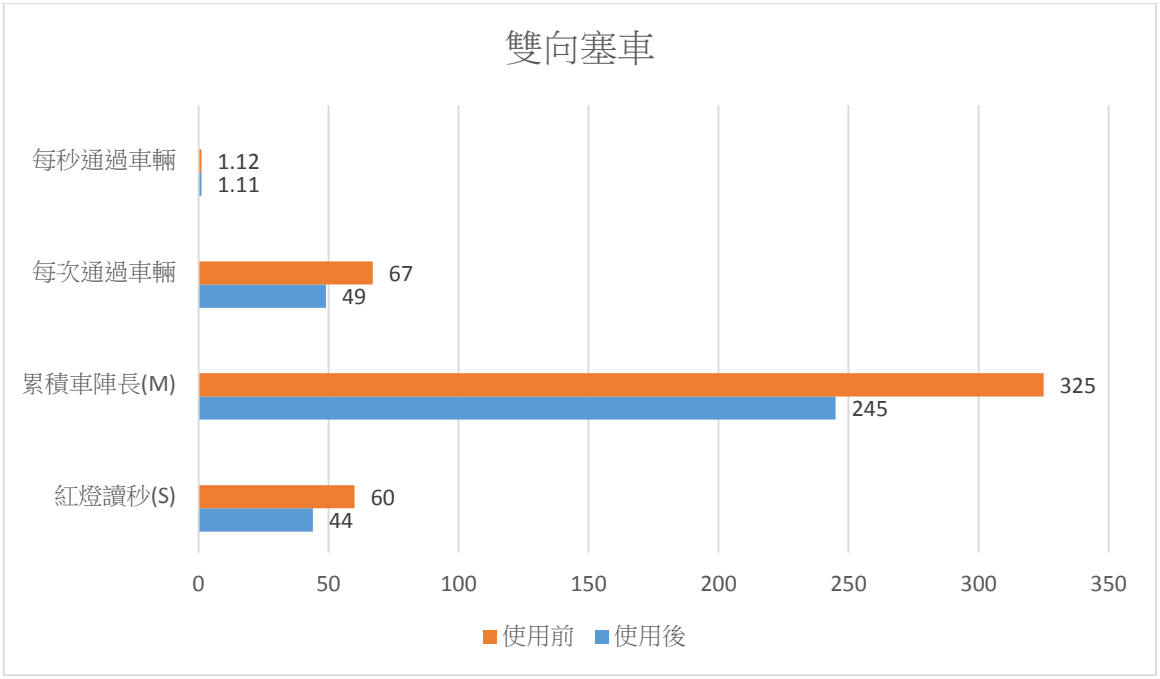
圖十五、1，2 號塞車時間(末)

(資料來源：研究者拍攝)

(四) 使用本裝置前後路口各項數據比較，如圖十六、圖十七。



圖十六、單向塞車時疏通效率(資料來源:研究者繪製)



圖十七、雙向塞車時疏通效率(資料來源:研究者繪製)

參、結論

一、系統執行結果

(一) 當 1 號感測器測得路段 1/2 處有車輛回堵狀況時，會將紅燈秒數速率*1.5，但是最後 5 秒時仍以原速率跑完秒數，得到 (99-5)/1.5+5=68，與原結果 30 秒相差快 8 秒。

(二) 當 2 號感測器測得路段 1/2 處有車輛回堵狀況時，會將紅燈秒數速率*1.5，但是最後 5 秒時仍以原速率跑完秒數，得到(99-5)/1.5+5=68，與原結果 30 秒相差快 8 秒。

(三) 當 1、2 號感測器皆測得路段 1/2 處有車輛回堵狀況時，會將紅燈秒數速率*2，但是最後 5 秒時仍以原速率跑完秒數，得到(99-5)/2+5=52，與原結果 30 秒相差快 12 秒。

由以上三點得知比起完全不做任何處理的紅綠燈秒數快 8 至 12 秒解決塞車問題。

二、遇到的困難

(一) 原本預計使用六個超音波感測器(於 1/4 處一個，為輕度堵塞;1/2 處一個，為中度堵塞;3/4 處一個，為嚴重堵塞，共兩個車道，每個車道三個感應器)來更精準

地定義同個車道，不同的塞車情形)但是感測器數量較多，流程圖因此太複雜畫不下，還有硬體的處理上過於複雜。

(二) 還有如果超音波感測器有其中一個完全被蓋住，則 ECHO 因為收不到訊號，會導致距離變得非常大，導致程式計算上的誤差。

三、解決方式

(一) 現階段先只使用兩個感應器(每個車道各一個)來定義塞車情形。

(二) 將超音波感測器埋在地下 5 公分的位置，避免感應器完全被蓋住導致音波無法回傳的問題。

四、未來展望

(一) 更精準地偵測目前的壅塞程度

我們只有使用一個車道一個的感應器，如果使用兩個以上，可以更精準的定義塞車的程度。

(二) 配合網際網路顯示平面道路路況

將主電路板接上網路後，可以依據塞車程度不同的路段，透過 APP 找到通往目的地最快且最流暢的路徑，還可以知道車流量。

(三) 搭配公家機關救援網節省救援時間(TVBS 新聞，2017)

在救護車或是消防車即將通過時及早利用秒速率改變，疏通即將行駛路徑的車流，使其能夠更快速的通過，達到保障人民生命安全的目的。

(四) 搭配學生上下課人潮避免意外發生(蘋果日報，2017)

搭配中小學的上下課時間，配置出人行道綠燈秒數較長的紅綠燈，這樣不但可以讓學生較安全的通過馬路，也可以減少指揮交通的人力。

肆、引註資料

一、楊明豐(2016)。**ARDUINO 最佳入門與應用：打造互動設計輕鬆學**。臺北市：碁峯資訊。

二、**中國時報。**陳燕珩(2017)臺北最塞路段。2018年5月20日，取自(http://www.chinatimes.com/newspapers/20171201000572-260107)

三、**TVBS。**葉怡君（2017)怕吃罰單，救護車卡在半路。2018年5月20日，取自(https://news.tvbs.com.tw/local/732280)

四、**蘋果日報。**石明啟(2017)國小男童放學遭車撞飛不治。2018 年 5 月 20 日，取自(https://tw.appledaily.com/new/realtime/20171129/1250151/)

五、陳增祥(1988)。臺北市區瓶頸路口交通問題及其改善策略之研究。國立交通大學交通運輸工程研究所:碩士論文

六、範例程式:超音波感測器。2018 年 5 月 20 日，取自(http://ming-shian.blogspot.tw/2013/09/arduino-hc-sr04.html)

七、範例程式:七段顯示器。2018 年 5 月 20 日，取自(http://yehnan.blogspot.tw/2013/08/arduino_26.html)

八、Cooper Maa—arduino 教學。2018 年 5 月 20 日，取自(http://coopermaa2nd.blogspot.tw/2011/05/arduino.html)

九、汽車百科:最小離地高度。2018 年 5 月 20 日，取自(https://c.8891.com.tw/pedia/4/tid/322)

十、趙英傑(2017)。**超圖解 Arduino 互動設計入門第 3 版**。臺北市：旗標科技股份有限公司