

中華民國第 58 屆中小學科學展覽會

作品說明書

高級中等學校組 電腦與資訊學科

佳作

052507

高度決定你的道路

學校名稱：桃園市私立新興高級中學

作者：	指導老師：
高二 林樂濱	蕭煥錫
高二 鄭宗恩	張建國
高二 劉宸佑	

關鍵詞：超音波、Arduino、ESP32

摘要

本研究提出新型動態的垃圾子母車清運模式，此模式將超音波感測器裝在垃圾子車上，感測垃圾子車內的垃圾量，並將感測資料主動傳回伺服端，伺服端經由傳回的資料，即可得知子車內堆積的垃圾量是滿的、半滿的或是空的，並分析哪些子車的垃圾必須及時清運、或可順路清運，哪些可以下次再回收，據此資訊規劃垃圾母車清運垃圾的最佳策略，不但可節省人們等垃圾車的時間，同時縮短垃圾車清運路徑因而減少垃圾車的耗油量與碳排放量，並且解決因垃圾車沿街收垃圾造成的交通阻塞問題，又可避免因垃圾堆積造成環境的清潔衛生問題，有效增加垃圾清運的效益。

壹、研究動機

每次倒垃圾時總是花費不少時間在等垃圾車，圖 1-1 是一般大家都很熟悉的倒垃圾的景象，想想這些等待垃圾車的時間原本可以用來做多少事啊！如果是下雨天的話，倒垃圾花的時間就更久了，而且在垃圾不落地的政策下，上下班時間沿途收垃圾的垃圾車經常造成嚴重的交通阻塞，所以如果能在一些定點放一個大的垃圾子車，大家隨時都可以倒垃圾了，這種垃圾子母車清運方式並不新鮮，但是可能會發生垃圾太多或太久未清運的情況，進而導致髒亂、發臭的環境衛生問題，為了解決此問題，母車必須能夠即時清運子車內的垃圾，因此我們思考如果當垃圾子車的垃圾快滿時，垃圾子車能主動傳訊息讓清潔隊員知道，就能及時派垃圾車清運垃圾，並且可以規劃收垃圾的最短路徑，讓母車出車一趟即可快速收完幾個定點的垃圾，這樣不但可以減少垃圾車的耗油量、倒垃圾花費的時間，也能解決垃圾車所造成的交通問題，大家也不再發生提著垃圾袋滿街追著垃圾車跑的窘境。



(a) 等待垃圾車



(b) 排隊倒垃圾

圖 1-1 倒垃圾的熱鬧景象(資料來源：自由時報)

貳、研究目的

台灣實行垃圾不落地政策後，每個區域都會派垃圾車定時清運一般民眾家中的垃圾，但很多忙碌的上班族常趕不及倒垃圾的時間，讓家中堆滿垃圾，或是當垃圾車沿街收垃圾時，造成交通阻塞，本研究的目的是為了設計出一套清運垃圾的系統，我們規劃運用垃圾子母車來清運垃圾，但與傳統的垃圾子母車模式不同的是垃圾子車將會主動傳回垃圾量，經伺服器分析規劃出垃圾回收路線，再由垃圾母車即時去清運垃圾。此系統預計達成以下的目標：

- (1) 減少人們等待垃圾車的時間
- (2) 解決垃圾車沿街收垃圾時造成的交通壅塞問題
- (3) 減低垃圾車的耗油量與碳排放量
- (4) 避免傳統垃圾子母車清運模式造成的環境髒亂問題

參、研究設備及器材

我們所使用的硬體設備及軟體如表 3-1。重要的硬體與軟體設備我們將分節敘述。

表 3-1 硬體及軟體設備表

硬體	軟體
超音波感測器	CP2102 USB 驅動程式
ESP32 開發板	Arduino IDE
桌上型電腦	Cloud9 IDE
OLED 顯示器	Fritzing
烙鐵	Google Map API
萬用板	
焊錫	
單芯線	
行動電源	

一、超音波感測器

圖 3-1 是超音波測量距離的原理，超音波感測器由超音波發射器(Transmitter)、接收器(Receiver)和控制電路所組成。當超音波感測器被觸發的時會發射一連串 40 kHz 的聲波，並接收從最近的物體反射的回音。因此，測量聲音在感測器與物體之間往返經過的時間可以推算出距離。

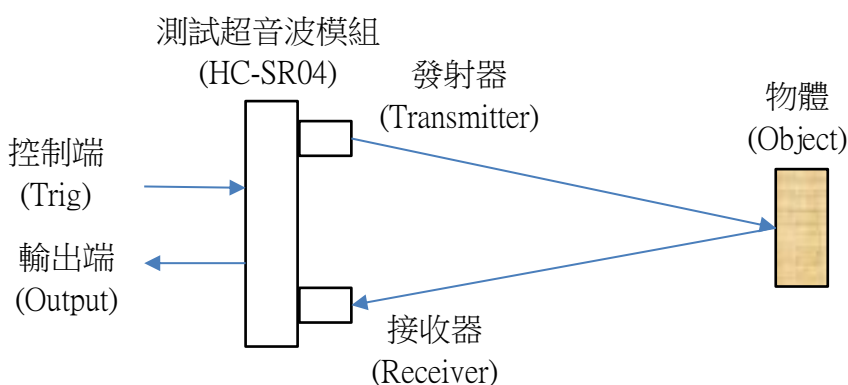


圖 3-1 超音波測量距離的原理

二、ESP32 開發板

ESP32 是一個具有傳統藍牙、低功耗藍牙和 Wi-Fi 的連網功能的微控制器系統。圖 3-2 是 ESP32 開發板腳位圖。



圖 3-2 ESP32 開發板腳位圖

三、OLED 顯示器

圖 3-3 是 OLED 顯示器，特色是超輕薄、自發光、超廣視角、顏色鮮艷且對比更鮮明，便宜又省電。



圖 3-3 OLED 顯示器

四、Arduino 軟體

Arduino 是開放原始碼的單晶片微控制器，採用開放原始碼的軟硬體平台，圖 3-4 是 Arduino IDE 介面，是個簡單好用的 I/O 介面。

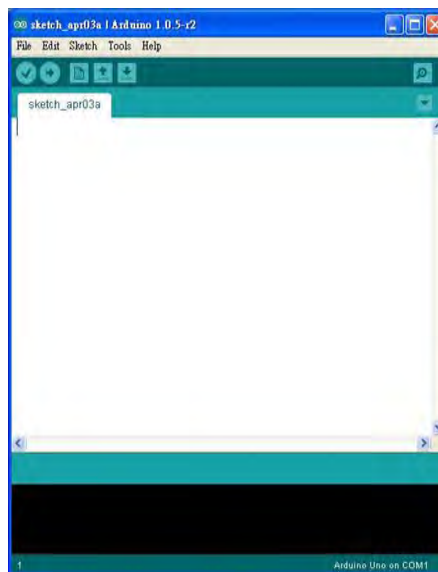
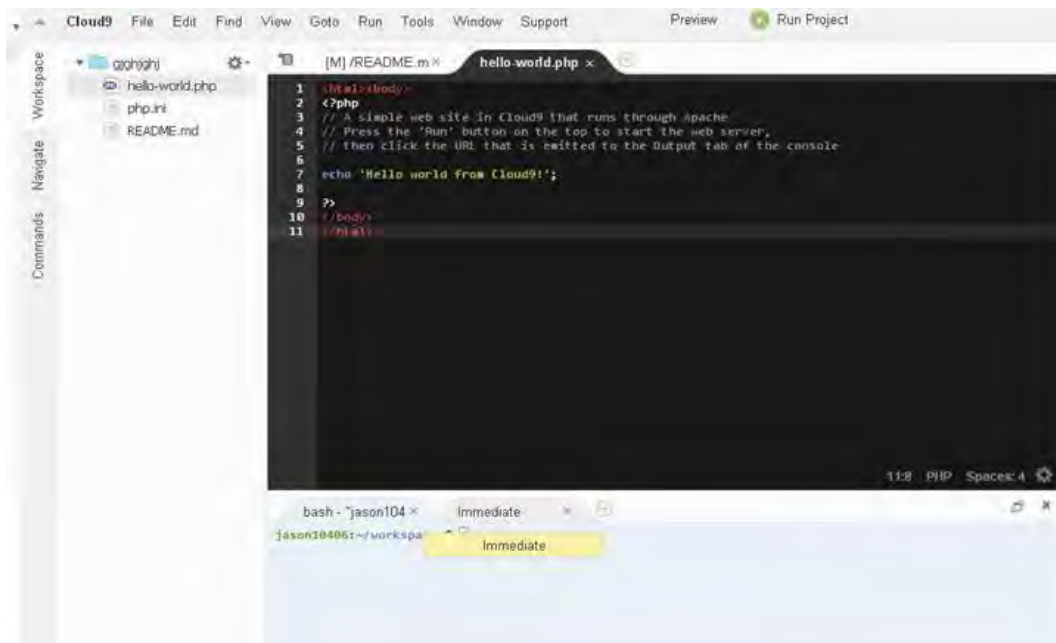


圖 3-4 Arduino IDE 介面

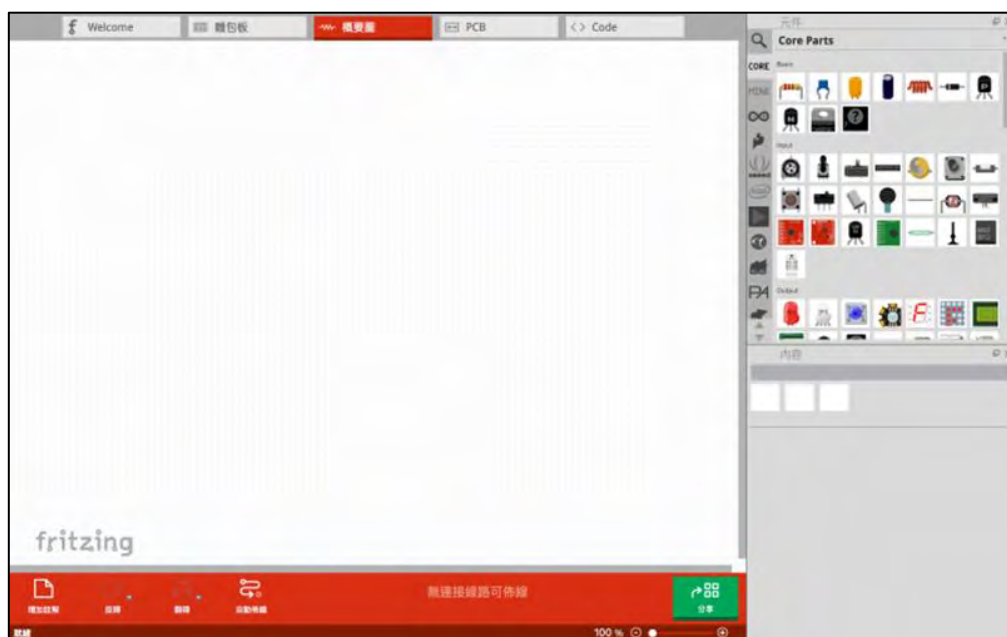
五、Cloud 9 IDE

Cloud 9 是一個雲端整合式開發環境(IDE) (如圖 3-5)，支援超過 40 種以上的程式語言，可以直接透過瀏覽器操作。



六、Fritzing 電路繪圖軟體

Fritzing 軟體可用來設計電子電路(如圖 3-6)，使用介面操作簡單，且已經提供了許多電子元件，還可自行補充其他組件。



七、Google MAP API

Google 公司提供了許多好用的服務，例如 Google Map，我們藉由應用程式介面(Application Programming Interface, API)傳入不同參數，即可呈現客製化的地圖與功能。使用 Google MAP API 首先要申請以便取得金鑰，方法如表 3-2 的步驟。

表 3-2 取得 Google MAP API 金鑰的方法

STEP1	STEP2
	
STEP3	STEP4
	
STEP5	STEP6
	

肆、研究過程或方法

我們設計了一套運用垃圾子母車清運垃圾的系統，但與傳統的垃圾子母車模式不同，垃圾子車將會主動傳回垃圾量，再由垃圾母車經由規劃好的較佳路線，即時去清運數個垃圾子車的垃圾。我們研究的過程與方法詳述於下。

一、測試超音波感測器及 OLED 顯示器

首先用麵包板接 ESP32 開發板測試超音波感測器以及 OLED 顯示器是否有反應，並測量超音波感測器可偵測到的範圍，如圖 4-1 所示。

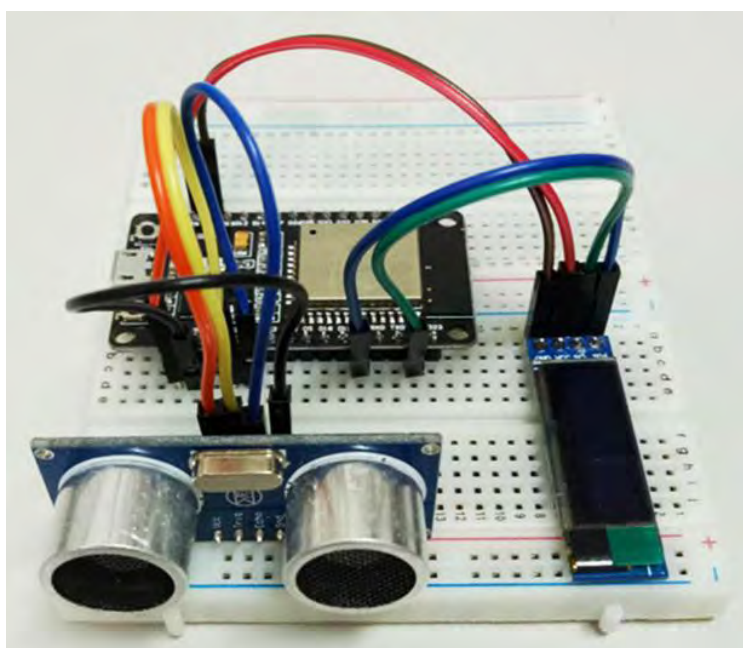
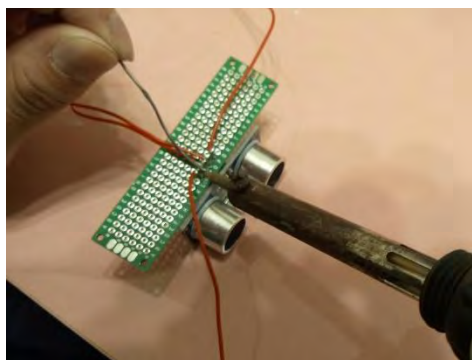


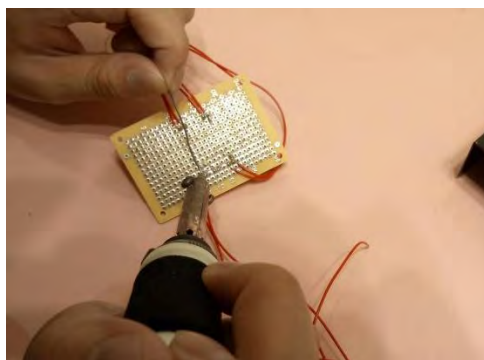
圖 4-1 麵包板測試超音波感測器及 OLED 顯示器

二、設計超音波模組電路

為了測量垃圾子車內的垃圾量，我們利用在學校學到的超音波感測原理，將超音波感測器裝設在垃圾子車上方，可用來測量垃圾子車內的垃圾高度，為了增加準確度，而且垃圾可能被丟入垃圾車的不同角落位置，所以我們設置了五個超音波感測器，分別測量垃圾子車內五個不同位置，圖 4-2 是我們製作垃圾子車的過程，圖 4-3 則是我們設計的超音波模組電路。

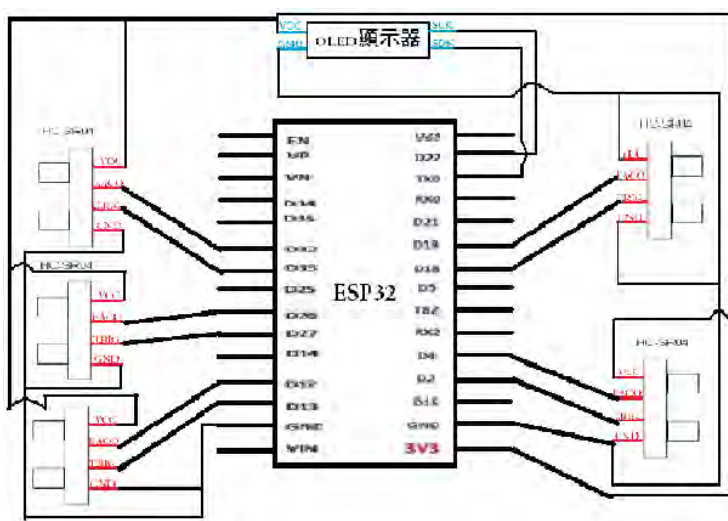


(a)

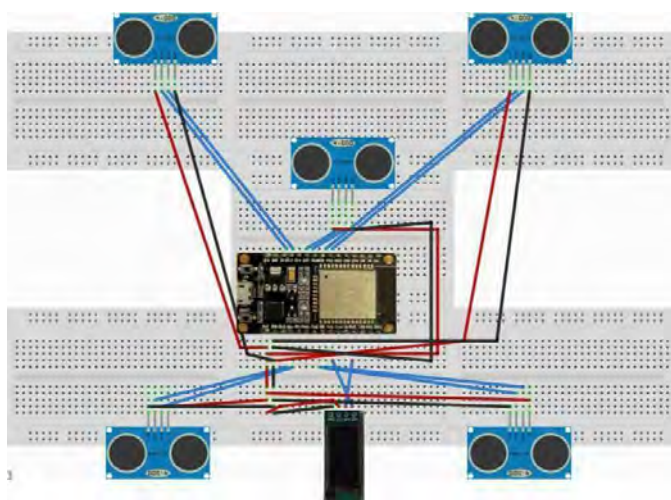


(b)

圖 4-2 製作垃圾子車的過程



(a)



(b)

圖 4-3 超音波模組電路圖

三、設計垃圾清運模式

我們規劃的清運垃圾運作模式垃圾子車必須回傳垃圾量及垃圾子車的位置資料，因此我們設計經由 ESP32 的 Wi-Fi 功能傳送這些資料到伺服端，當伺服端接收到資料就會進行分析，並將分析結果結合 Google Map API 顯示於地圖上，清潔隊員就能即時了解各個垃圾子車的垃圾量，並依據要收垃圾的子車放置地點，伺服端將規劃出最短路徑提供給清潔隊員去收垃圾，來達到減少垃圾車耗油及碳排放量。

垃圾清運模式的研究主要分為兩部分，第一個部分為感測端動作(如圖 4-4)，主要是超音波感測器測量垃圾高度，接著利用 ESP32 傳輸資料，第二部分為伺服端進行分析及顯示結果(如圖 4-5)，實施步驟如下：

步驟一：超音波感測器測量垃圾子車內的垃圾高度

步驟二：ESP32 傳輸資料

步驟三：伺服端進行分析

步驟四：Google Map API 顯示結果於地圖上

步驟五：伺服端規劃最短收垃圾路徑

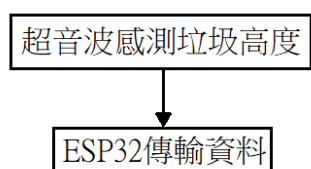


圖 4-4 感測端流程圖

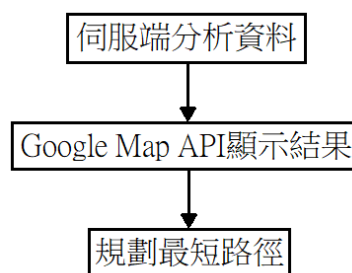


圖 4-5 伺服端流程圖

四、撰寫程式

我們利用 Cloud9 撰寫偵測垃圾量的分析程式，首先平均 5 個超音波傳回的垃圾量值，再根據此平均垃圾量高度決定在 Google MAP API 內垃圾桶圖案顯示的顏色。表 4-1 為垃圾量與顏色的關係，假設垃圾子車的內部高度為 150 公分，當感測值大於等於 70 公分時，垃圾量兩不到一半高度，判斷為垃圾量少的狀態，使用綠色顯示垃圾桶；若感測值小於等於 35 公分，

則垃圾量已經超過 3/4，垃圾量很多，此時使用紅色的垃圾桶；若垃圾高度介於 36~69 公分，則使用黃色的垃圾桶，判定垃圾量為半滿。程式碼部分如圖 4-6。

表 4-1 伺服端判斷垃圾量

分析結果 垃圾量	感測值 ≥ 70 公分	感測值介於 36~69 公分	感測值 ≤ 35 公分
滿			紅色垃圾桶
半滿		黃色垃圾桶	
少	綠色垃圾桶		

```

markers = [];
var data_json = JSON.parse(event.data);
for(var i=0; i<data_json.length; i++){
    var data = JSON.parse(data_json[i]);
    var latlng = new google.maps.LatLng(data.latitude, data.longitude);
    var h0 = parseFloat(data.height0);
    var h1 = parseFloat(data.height1);
    var h2 = parseFloat(data.height2);
    var h3 = parseFloat(data.height3);
    var h4 = parseFloat(data.height4);
    var avg = Math.round((h0+h1+h2+h3+h4)/5.0);

    var icon;
    if(avg >= 70){
        icon = {
            url: "green.png",
            scaledSize: new google.maps.Size(icon_size, icon_size)
        };
    }else if(avg <= 35){
        icon = {
            url: "red.png",
            scaledSize: new google.maps.Size(icon_size, icon_size)
        };
    }else{
        icon = {
            url: "yellow.png",
            scaledSize: new google.maps.Size(icon_size, icon_size)
        };
    }
}

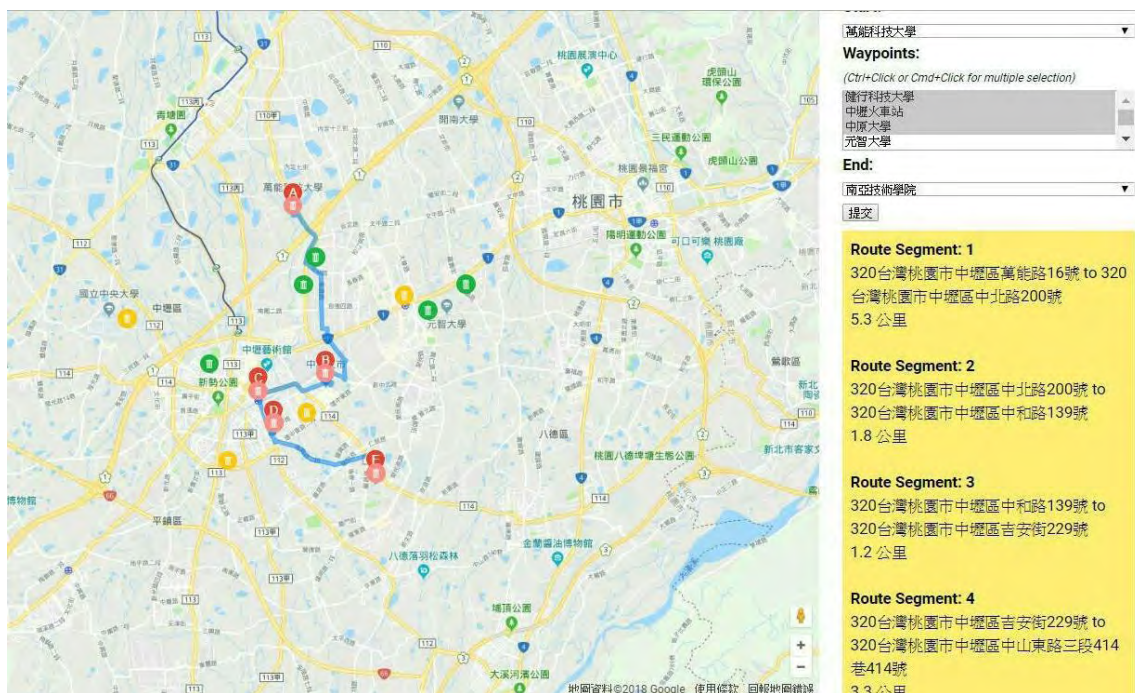
```

偵測垃圾的高度

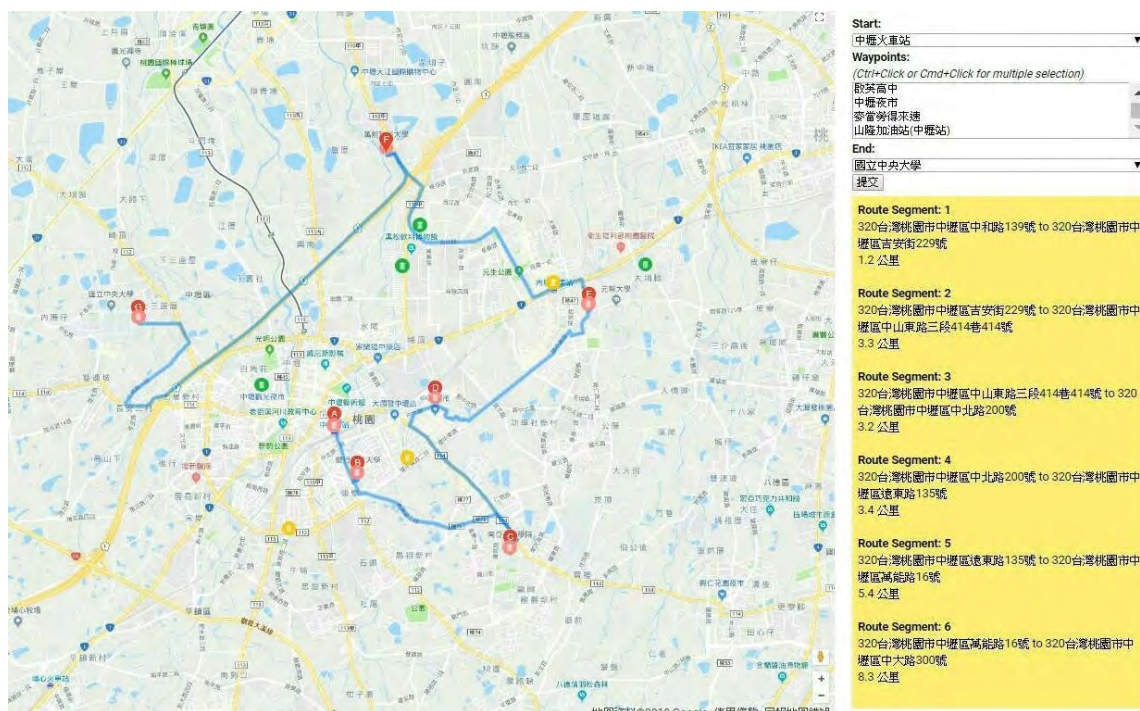
圖 4-6 Cloud9 部分程式圖

五、規劃最短路徑

把偵測到的垃圾量轉換成圖案顯示於 Google MAP 上，並規劃最短路徑(如圖 4-7)。



(a) 規劃最短路徑範例一



(b) 規劃最短路徑範例二

圖 4-7 Google MAP 上顯示最短的路線

伍、研究結果

我們利用紙箱來模擬垃圾子車，如圖 5-1 之綠色紙箱，並將 5 個超音波感測器裝設在紙箱蓋的上方，即模擬垃圾子車上方位置，實作完成的作品如圖 5-2。



圖 5-1 垃圾子車模擬模型

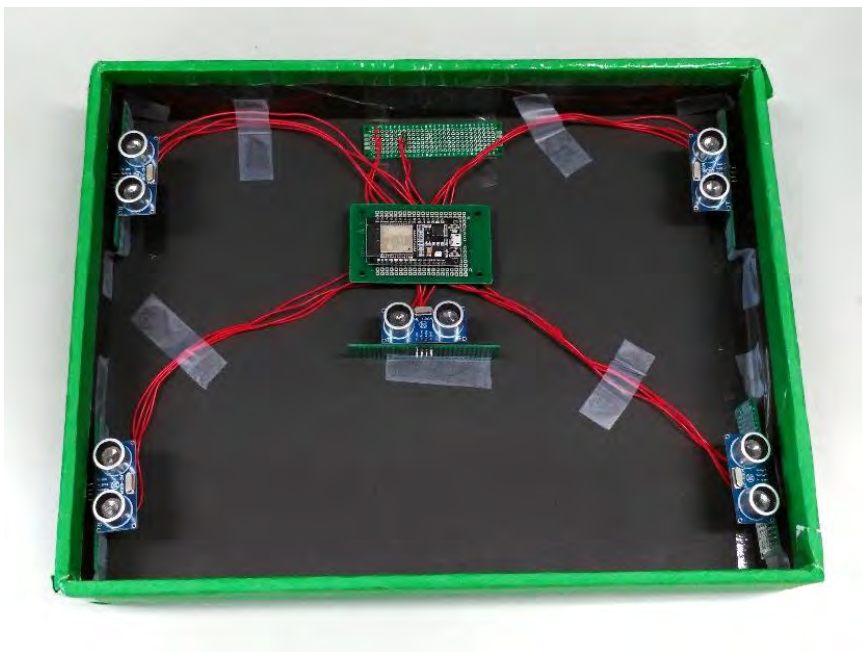


圖 5-2 超音波感測器設置圖

接著我們根據系統實施步驟說明。

一、超音波感測器測量垃圾高度

首先利用 Arduino 程式來處理超音波感測器測得的距離，如圖 5-3。

```
Ultrasonic ultrasonic(TRIGGER_PIN, ECHO_PIN, 3000);
Ultrasonic ultrasonic2(TRIGGER_PIN2, ECHO_PIN2, 3000);
Ultrasonic ultrasonic3(TRIGGER_PIN3, ECHO_PIN3, 3000);
Ultrasonic ultrasonic4(TRIGGER_PIN4, ECHO_PIN4, 3000);
Ultrasonic ultrasonic5(TRIGGER_PIN5, ECHO_PIN5, 3000);

long distance = ultrasonic.Ranging(CM);
Serial.print(distance);
Serial.print("\n");

long distance2 = ultrasonic2.Ranging(CM);
Serial.print(distance2);
Serial.print("\n");

long distance3 = ultrasonic3.Ranging(CM);
Serial.print(distance3);
Serial.print("\n");

long distance4 = ultrasonic4.Ranging(CM);
Serial.print(distance4);
Serial.print("\n");

long distance5 = ultrasonic5.Ranging(CM);
Serial.print(distance5);
Serial.print("\n");
```

圖 5-3 超音波感測器距離程式碼

二、ESP32 傳輸資料

ESP32 將資料利用 JSON 格式傳送至伺服端，傳送的資料包含傳送時間、垃圾子車座標位置及 5 個超音波感測的垃圾高度資料，如圖 5-4。

三、伺服端進行分析

伺服端將接收到的 5 個超音波感測的垃圾高度資料求取平均值後進行分析，並依據分析結果判斷垃圾子車內的垃圾量，當垃圾平均高度綠色圖示代表垃圾子車還很空，黃色圖示代表垃圾子車內垃圾量約為一半，紅色圖示代表垃圾子車快滿了。

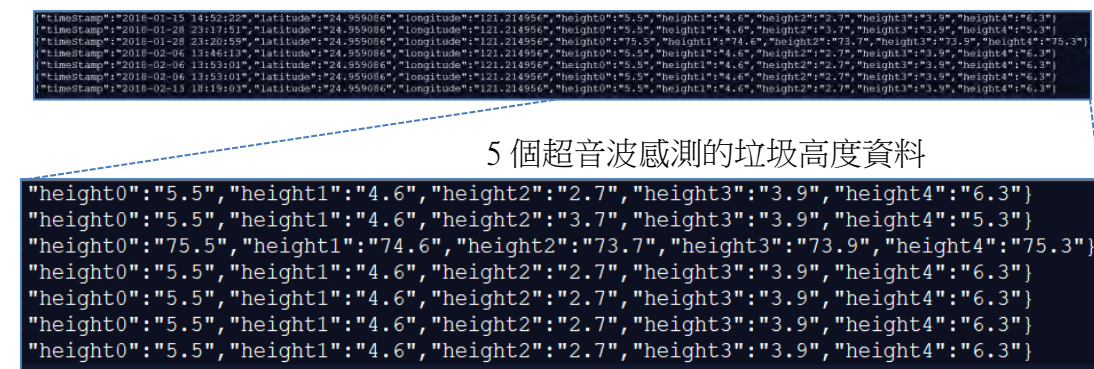
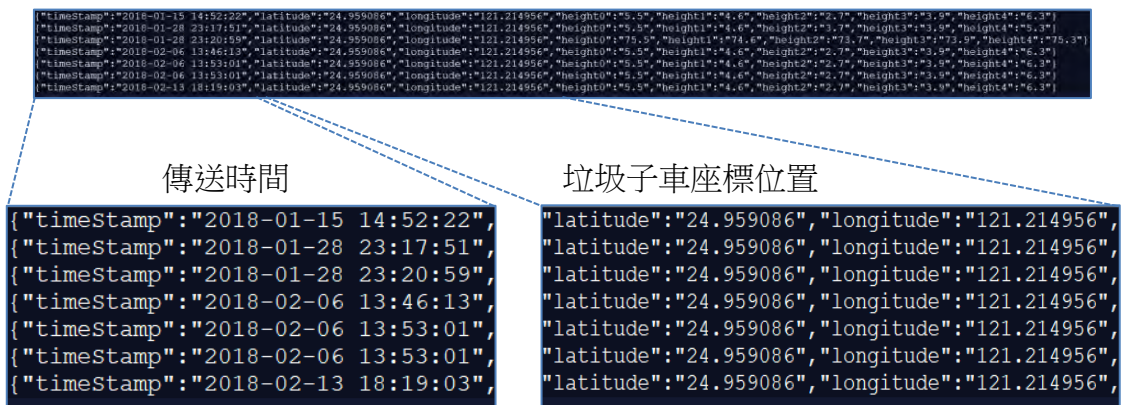


圖 5-4 JSON 格式傳送資料圖

四、Google Map API 顯示結果於地圖上

利用 Google Map API 將上述分析結果顯示在地圖上，如圖 5-5。



圖 5-5 伺服器將分析結果圖

五、伺服端規劃最短收垃圾路徑

清潔隊員根據垃圾桶顏色決定哪些垃圾子車的垃圾，然後伺服端依據需要回收垃圾的地點規劃出最短路徑，如圖 5-6。路線規劃上目前無法自動判定收那幾個，因為可能一台母車收不完，需要派好幾台母車才行，因此讓每台母車的清運人員自行決定清運哪些垃圾子車。

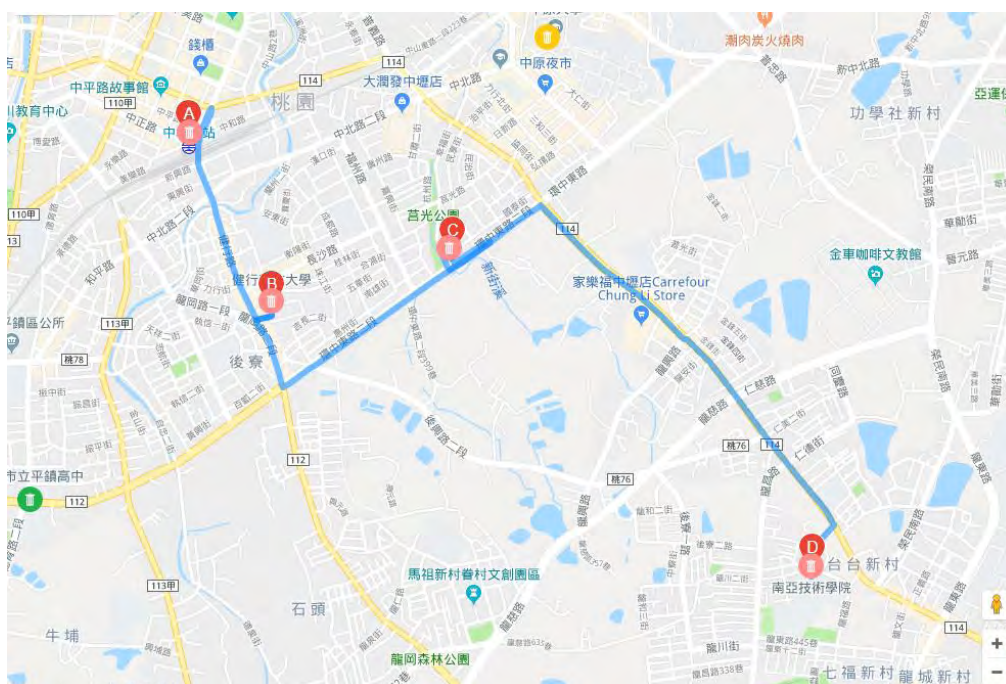


圖 5-6 伺服端規劃最短路徑圖

路徑規劃時需指定啟始點、終點以及經過的點，我們用另一個範例說明。例範起點是中壢火車站、終點是健行科技大學，經過的點包含中原夜市、元智大學與南亞科技大學，最短路徑圖如圖 5-8。我們用 Google MAP 地圖重新分段計算距離和時間，如表 5-1，總距離是 14.2 公里，時間共需花費 40 分鐘。

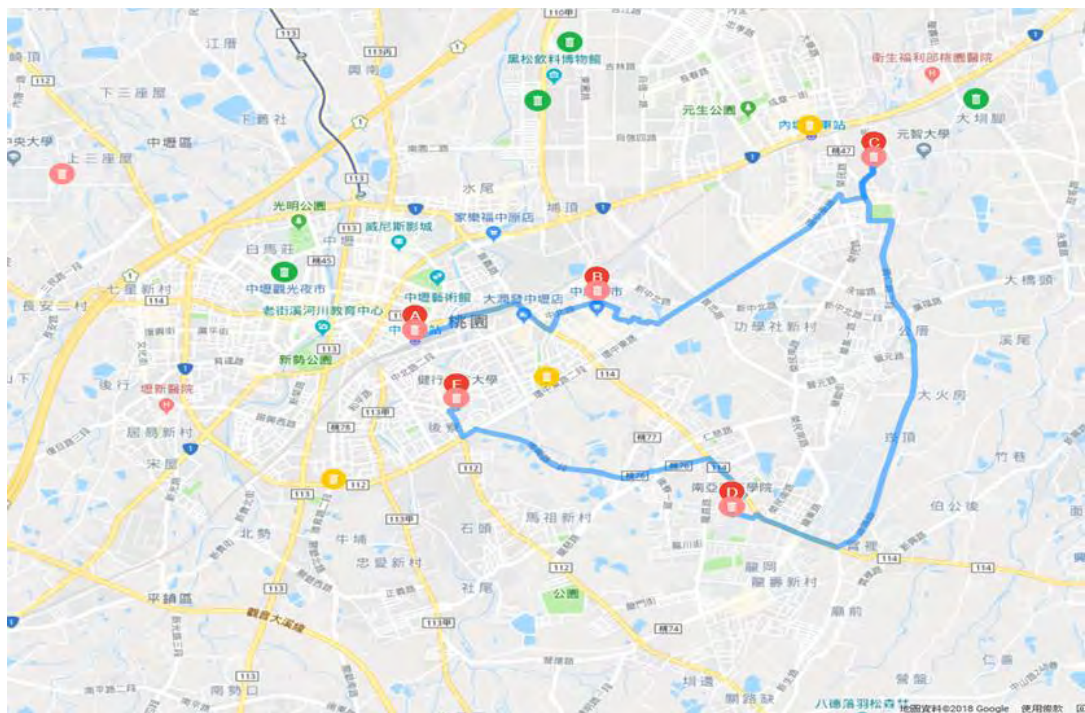
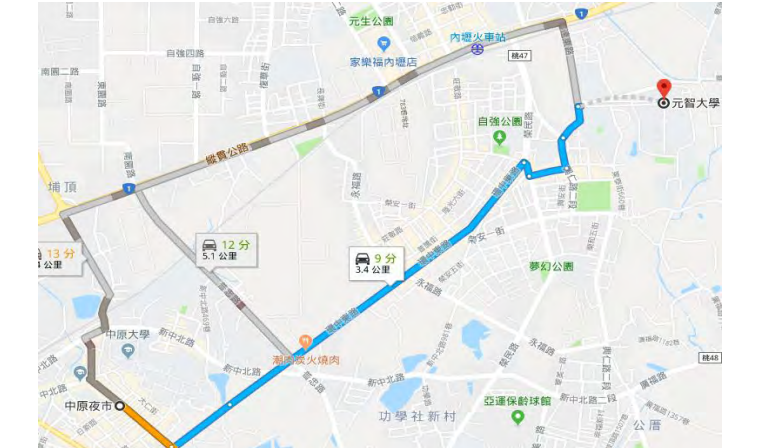
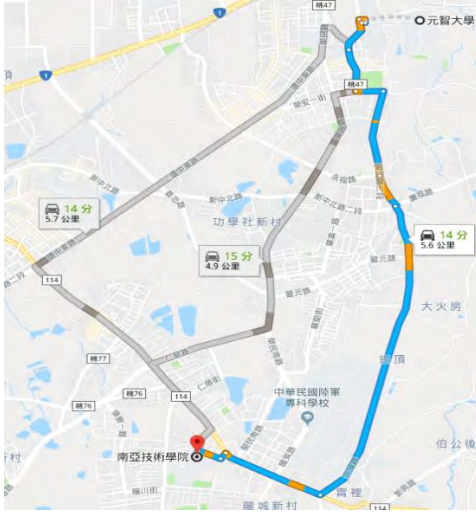
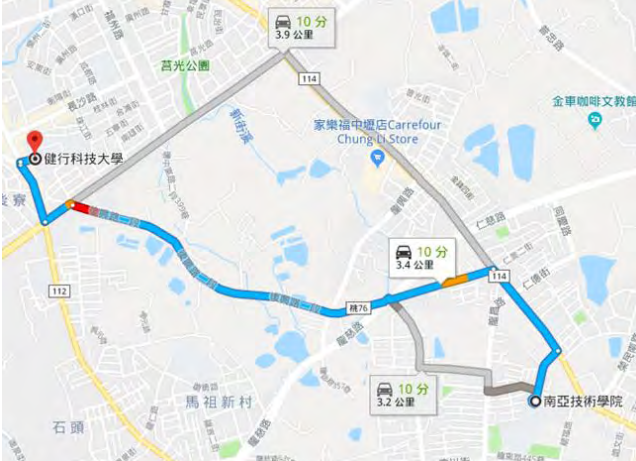


圖 5-7 最短路徑圖

表 5-1 最短路徑距離與時間

地圖與規劃路線	距離與時間
 Map showing the route from Zhongli Station to Zhongli Night Market. The route is highlighted in blue and red, passing through Zhongli University and Zhongli Night Market. The distance is 1.8 km and the time is 8 minutes.	中壢火車站→中原夜市 1.8 公里 8 分鐘
 Map showing the route from Zhongli Night Market to Yuanzhi University. The route is highlighted in blue and red, passing through Zhongli University and Yuanzhi University. The distance is 3.4 km and the time is 9 minutes.	中原夜市→元智大學 3.4 公里 9 分鐘

地圖與規劃路線	距離與時間
	元智大學→南亞科技大學 5.6 公里 13 分鐘
	南亞科技大學→健行科技大學 3.4 公里 10 分鐘
總距離：14.2 公里 總時間：40 分鐘	

陸、討論

一、動態垃圾子母車清運模式成本效益

根據圖 6-1 臺南市東區實施「準時垃圾不落地清運服務」停留點、時間一覽表顯示垃圾車每收一次垃圾平均時間總計約 1 小時 30 分。假設每車每日作業 1 小時 30 分，消耗油量約 5 公升，若油價 1 公升 35 元，表 6-1 計算了一般垃圾車每年每車耗油量 1825 公升、排碳量 476,325 公噸。垃圾子車則較省成本，因每車每日作業路線較短、時間較少，消耗油量以 3 公升計算，表中顯示以全台 5000 輛計算每年省了約 127,723,000 元、省了約 730 公升的油，減少了約 1.905 公噸的排放量，不但清潔人員、倒垃圾的人們省了許多時間，同時也讓我們省了很多的資源。

次序	里別	停 留 點 位 置	起迄時間	
			到達	離開
第一車次				
01	東安	長榮路二段 32 巷 175 號前	15:30	15:35
02	大同	大同路一段 175 巷 111 號	15:50	15:53
03	大同	大同路一段 175 巷 132 弄口	15:55	15:58
04	大同	大同路一段 51 巷尾（府東街交叉口）	16:03	16:06
05	大同	大同路一段 45 巷口	16:07	16:10
06	成大	北門路一段 284 號公路局入口	16:15	16:18
07	成大	北門路 2 段 18 號	16:21	16:23
08	成大	育樂街 57 巷口	16:28	16:30
09	成大	育樂街 58 號（成大活動中心前）	16:31	16:35
10	成大	育樂街 148 巷 17 弄口	16:37	16:40
11	泉南	長榮路 1 段與光華街口	16:46	16:49
12	泉南	光華街 117 號	16:50	16:53
13	泉南	光華街 275 巷與樹林街一段交叉口	16:55	16:58
14	泉南	府連路 234 號（地下道上方轉角）	17:00	17:03
15	泉南	府連路 368 號前（路口）	17:05	17:07

圖 6-1 垃圾車收垃圾時間

表 6-1 一般垃圾車與垃圾子車對照表

	一般垃圾車	垃圾子車
每車每日油費	35 元 × 5 公升 = 175 元/日	35 元 × 約 3 公升 = 105 元/日
每車每年油費	175 元 × 365 天 = 63,875 元/年	105 元 × 365 天 = 38,325 元/年
全台 5000 輛計算	63,875 元 × 5000 台 = 319,375,000 元	38,325 元 × 5000 台 = 191,652,000 元
每車每年耗油量	5 公升 × 365 天 = 1825 公升	3 公升 × 365 天 = 1095 公升
每車每年排碳量	碳 CO ₂ = 2.61 × 1825 ÷ 4.763 公噸	碳 CO ₂ = 2.61 × 1095 ÷ 2.858 公噸
錢每日省了(175 - 105 元) = 70 元 錢每年省了(63,875 元 - 38,325 元) = 25,550 元 5000 輛垃圾車錢省了(319,375,000 元 - 191,652,000 元) = 127,723,000 元 每年省油量(1825 公升 - 1095 公升) = 730 公升 每年減少排碳量(4,763 公噸 - 2.858 公噸) = 1.905 公噸		

二、垃圾回收週期與環境問題

(一) 氣候因素

考慮台灣氣候炎熱，一般垃圾容易腐臭的問題，需要再考量季節氣溫以及垃圾子車中垃圾的留置時間。假設在垃圾桶未滿的情形下，垃圾最長回收週期可為二天，則根據中央氣象局統計 1981-2010 年台中的年均溫為 23.3 度，則設定月均溫每提高 2 度則回收週期縮短 12 小時，則以台中為例，垃圾回收週期如表 6-2 所示。

表 6-2 以台中為例在垃圾桶未滿的情形下垃圾回收週期

地名	一月	二月	三月	四月	五月	六月	七月	八月	九月	十月	十一月	十二月
臺中 月均溫	16.6	17.3	19.6	23.1	26.0	27.6	28.6	28.3	27.4	25.2	21.9	18.1
垃圾回 收週期	2 天				1.5 天	1 天				1.5 天	2 天	

(二) 垃圾量太大的問題

如果每個定點的垃圾子車經常未到設定的回收週期垃圾即滿溢，則需要規劃適時增加該定點的垃圾子車數量，例如平均一天派車清運超過一次，則增加該點之垃圾子車量。

三、其他系統效益探討

(一) 是否要偵測垃圾子車內垃圾的重量？

考量垃圾母車回收垃圾時可以壓縮垃圾，一輛垃圾母車可以清運多少台垃圾子車的垃圾，將與垃圾的密度息息相關，但是如果在垃圾子車上安裝重量感測器，將增加設備成本、耗電量等，重量感測器長期被垃圾重壓的情形下，也易發生故障或失去準確度，此外考量垃圾子車無壓縮垃圾功能，子車的裝載是以容量來決定垃圾是否滿溢，在沒有安裝重量感測器的情形下，目前可以用長期清運的數據統計出一輛垃圾母車大約可以清運多少台垃圾子車的垃圾，如此在節省成本的情形下也能達到較大的垃圾清運效益。

(二) 超音波感測的垃圾高度資料多久傳送一次較為恰當？

偵測的資料傳送間隔可以設定一段長時間，例如 30 分鐘，或者未達 30 分鐘的間隔

時間垃圾高度有了變化才傳送，如此可以減少感測器的耗電量、資料傳輸量以及伺服端的檔案大小，但考量本研究展示時的需求，目前設定每 5 秒傳輸一次感測資料，使展示時方便探查系統是否運作正常。

(三) 順路清運半滿的垃圾子車是否可以增加效益？

假設一台垃圾母車可以清運 10 台垃圾子車，考量如果只有一台已滿的垃圾子車需要立即清運的情況，可以偵測母車到達子車的路線上是否有半滿的垃圾子車，可以順路加以清運，如果母車清運容量未達上限，可以再選出此子車附近的半滿垃圾子車再清運，如此將可增加清運的效益。

(四) 如何派發垃圾母車？

目前本研究之系統利用 Google Map API 垃圾子車的垃圾量分析結果顯示在地圖上，並可計算出母車要清運垃圾的最短路徑。至於一次需要派出多少台垃圾母車數量，可經由過去一台母車可清運的統計數量，計算出應派發的垃圾母車數量，再由清運人員自行討論劃分不重疊的區域，再各自計算清運垃圾的最短路徑。

四、在研究過程中發生的問題與解決方法

(一) 垃圾子車內會有超音波偵測不到的地方嗎？

解決方法：我們在垃圾子車蓋上的四個角落以及中間都有裝設超音波感測器，避免偵測不到的情況。並且將傳回的資料求取平均值。

(二) ESP32 的電源如何供應？

解決方法：我們使用行動電源來供電，在清潔隊員清運垃圾子車的垃圾時，順便更換一顆行動電源。

(三) 在製作垃圾子車模型的過程我們發現 ESP32 有些腳位使超音波感測器無法作用。

解決方法：我們用麵包板把 ESP32 接到超音波感測器，看哪個腳位有作用，發現只

有 2、4；12、13；18、19；26、27；32、33 腳位接到超音波感測器的 Trig 和 Echo 才會有作用。

(四) 要使用 Cloud 9 必須使用信用卡辦帳號？

解決辦法：我們請有使用 Cloud 9 的老師邀請我們，這樣就不用使用信用卡就可以辦帳號了。

柒、結論

我們規劃的垃圾清運模式中垃圾子車可以設置在很多地方，利用感測裝置及時將垃圾子車感測資料傳送至伺服器端，讓清潔隊員清楚地了解各個垃圾子車目前的垃圾量，並派垃圾母車清運垃圾，不會造成垃圾堆積的環境問題，伺服器運用 Google map API 規劃最短路徑，可達到垃圾母車減少耗油量及碳排放量。讓民眾倒垃圾的時間上更有彈性，而且不會產生垃圾車沿路收垃圾時的交通阻塞問題，也不會再發生提著垃圾滿街追著垃圾車跑的窘境。此外，以全台 5000 輛垃圾車計算，每年可省下約 127,723,000 元、省了約 730 公升的油，減少了約 1.905 公噸的排放量，不但清潔人員、人們省了更多時間，同時也讓我們省了很多的資源。

捌、參考資料及其他

- [1] 李容萍(2017-02-01)・跟年假掰掰！桃園家戶一早聞聲「倒垃圾」・自由時報・取自 <http://news.ltn.com.tw/news/life/breakingnews/1962086>
- [2] K.C.M.L. (2017, 3 月 13 日)・如何在 Arduino IDE 上面安裝 ESP32 ?・取自 <https://makerpro.cc/2017/03/how-to-set-up-esp32-on-arduino-ide/>
- [3] ESP32 低功耗 WIFI + 藍芽開發板・華廣電子商城・取自 <http://shop.cpu.com.tw/product/53505/info/>
- [4] Caterpillar (2015 年 6 月 10 日)・mBlock & Arduino (12) 使用超音波感應器測量距離・CodeData・取自 <http://www.codedata.com.tw/social-coding/mblock-arduino-12-ultrasonic-sensor/>
- [5] OLED 的優缺點(2005 年 7 月 15 日)・彰化師大藍光實驗室 OLED 網・取自

http://ykuo.ncue.edu.tw/oled/oled_ad.htm

- [6] G. T. Wang (2015 年 5 月 22 日) • 創辦 Google MAP API 方法 • 取自
<https://blog.gtwang.org/programming/obtaining-api-key-from-google-developers-console/>
- [7] Arduino 軟體介紹 (2018 年 3 月 23 日) • 維基百科 • 取自 <https://zh.wikipedia.org/zh-tw/Arduino>
- [8] Cloud 9 網頁介紹 (2017 年 07 月 26 日) • Michael Talks • 取自
<https://blog.cwchen.tw/programming/2017/07/26/the-brief-intro-on-cloud-9-ide/>
- [9] Fritzing 電路繪圖軟體介紹 (2018 年 3 月 26 日) • 維基百科 • 取自
<https://de.wikipedia.org/wiki/Fritzing#Funktionen>
- [10] 電動壓縮式 Hybrid 油電垃圾車 • 台火開發股份有限公司 • 取自
<https://www.tidehold.com/business/hybrid.html>
- [11] 臺南市東區實施「準時垃圾不落地清運服務」停留點、時間一覽表(2015 年 4 月 6 日) • 取自
<http://www.tnepb.gov.tw/upload/2015040917343125.pdf>
- [12] 1981-2010 年月平均值資料 / 月平均 • 交通部中央氣象局 • 取自
https://www.cwb.gov.tw/V7/climate/monthlyMean/Taiwan_tx.htm

【評語】 052507

本研究提出一新的垃圾子母車清運模式，將超音波感測器裝在垃圾子車上，感測垃圾子車內的垃圾量，並將感測資料主動傳回伺服器端，伺服器端經由傳回的資料，據此資訊規劃垃圾母車清運垃圾的最佳策略，不但可節省人們等垃圾車的時間，同時縮短垃圾車清運路徑因而減少垃圾車的耗油量與碳排放量，並且解決因垃圾車沿街收垃圾造成的交通阻塞問題，又可避免因垃圾堆積造成環境的清潔衛生問題，有效增加垃圾清運的效益。利用感測元件網路解決垃圾問題為一有趣之題目。該研究作品完整包含實作、分析、路徑規劃。

摘要

本研究提出新型動態的垃圾子母車清運模式，此模式運用超音波感測器感測垃圾子車內的垃圾量，並將感測資料主動傳回伺服端，伺服端經由傳回的資料，即可得知子車內堆積的垃圾量是滿的、半滿的或是空的，並分析哪些子車的垃圾必須及時清運、或可順路清運，哪些可以下次再回收，據此資訊規劃垃圾母車清運垃圾的最佳策略，不但可節省人們等垃圾車的時間，同時縮短垃圾車清運路徑因而減少垃圾車的耗油量與碳排放量，並且解決因垃圾車沿街收垃圾造成的交通阻塞問題，又可避免因垃圾堆積造成環境的清潔衛生問題，有效增加垃圾清運的效益。不但清潔人員、人們省了更多時間，同時也讓我們省了很多的資源。

研究動機

每次倒垃圾時總是花費不少時間在等垃圾車，如果是下雨天就更花時間，而且在垃圾不落地的政策下，垃圾車沿途收垃圾時經常造成交通阻塞，如果能在定點放垃圾子車，這樣大家隨時都可以倒垃圾。這種垃圾子母車清運方式並不新鮮，但是可能會發生垃圾太多或太久未清運的情況，進而導致髒亂、發臭的環境衛生問題，為了解決此問題，母車必須能夠即時清運子車內的垃圾，因此我們思考如果當垃圾子車的垃圾快滿時，垃圾子車能主動傳訊息讓清潔隊員知道，就能及時派垃圾車清運垃圾，並且可以規劃收垃圾的最短路徑，讓母車出車一趟即可快速收完幾個定點的垃圾，這樣不但可以減少垃圾車的耗油量、倒垃圾花費的時間，也能解決垃圾車所造成的交通問題，大家也不再發生提著垃圾袋滿街追著垃圾車跑的窘境。

研究目的

1.

減少等待垃圾車的時間
2.

減少收垃圾時造成交通壅塞
3.

減少垃圾車耗油與碳排放量
4.

避免造成環境髒亂問題

台灣實行垃圾不落地政策後，每個區域都會派垃圾車定時清運一般民眾家中的垃圾，但很多忙碌的上班族常趕不及倒垃圾的時間，讓家中堆滿垃圾，或是當垃圾車沿街收垃圾時，造成交通阻塞，本研究的目的是為了設計出一套清運垃圾的系統，我們規劃運用垃圾子母車來清運垃圾，但與傳統的垃圾子母車模式不同的是垃圾子車將會主動傳回垃圾量，經伺服器分析規劃出垃圾回收路線，再由垃圾母車即時去清運垃圾。

研究設備及器材

硬體

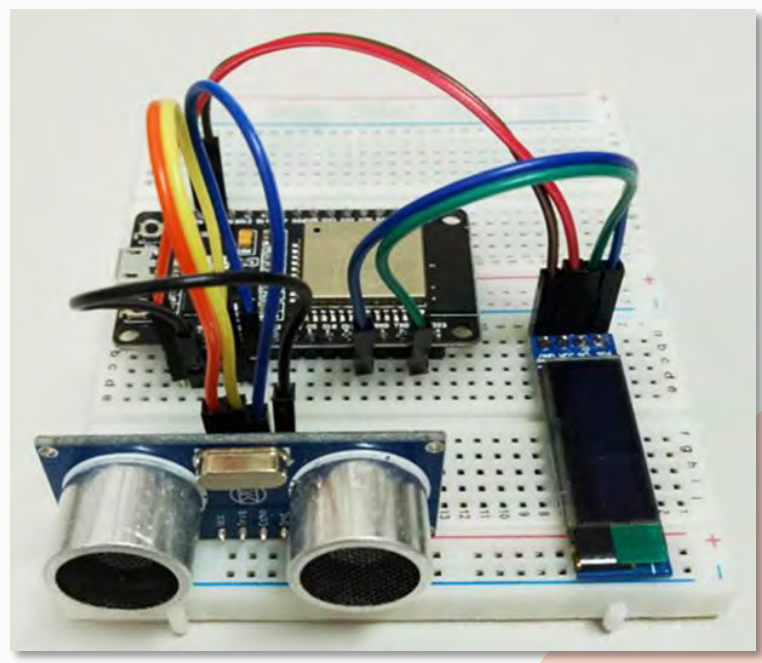
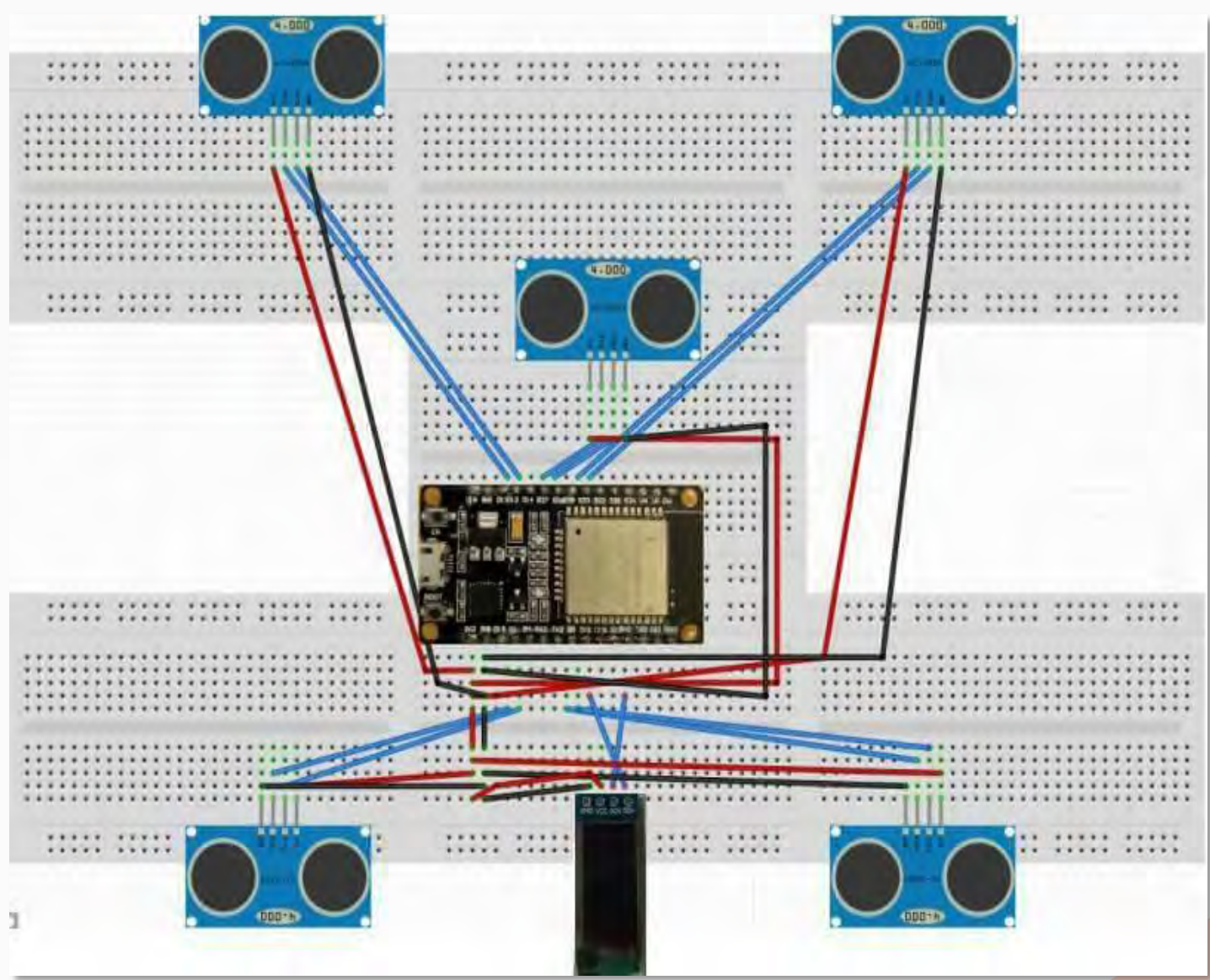
超音波感測器、ESP32開發板、OLED顯示器、桌上型電腦、烙鐵、萬用板、焊錫、單芯線、行動電源

軟體

CP2102 USB 驅動程式、Arduino IDE、Cloud9 IDE、Fritzing(電路繪圖軟體)、Google Map API

研究過程或方法

我們設計了一套運用垃圾子母車清運垃圾的系統，垃圾子車將會主動傳回垃圾量，再由垃圾母車經由規劃好的較佳路線即時去清運數個垃圾子車的垃圾。



設計超音波模組電路

測試超音波感測器及OLED顯示器

設計垃圾清運模式

- 感測端
 - 步驟一：超音波感測器測量垃圾子車內的垃圾高度
 - 步驟二：ESP32傳輸資料
- 伺服端
 - 步驟三：伺服端進行分析
 - 步驟四：Google Map API顯示結果於地圖上
 - 步驟五：伺服端規劃最短收垃圾路徑

撰寫程式模組

規劃最短路徑

■ 超音波感測器程式

```
Ultrasonic ultrasonic(TRIGGER_PIN, ECHO_PIN, 3000);
Ultrasonic ultrasonic2(TRIGGER_PIN2, ECHO_PIN2, 3000);
Ultrasonic ultrasonic3(TRIGGER_PIN3, ECHO_PIN3, 3000);
Ultrasonic ultrasonic4(TRIGGER_PIN4, ECHO_PIN4, 3000);
Ultrasonic ultrasonic5(TRIGGER_PIN5, ECHO_PIN5, 3000);

long distance = ultrasonic.Ranging(CM);
Serial.print(distance);
Serial.print("\n");

long distance2 = ultrasonic2.Ranging(CM);
Serial.print(distance2);
Serial.print("\n");

long distance3 = ultrasonic3.Ranging(CM);
Serial.print(distance3);
Serial.print("\n");

long distance4 = ultrasonic4.Ranging(CM);
Serial.print(distance4);
Serial.print("\n");

long distance5 = ultrasonic5.Ranging(CM);
Serial.print(distance5);
Serial.print("\n");
```

超音波測量距離

■ 傳輸資料運用JSON格式

```
{"timestamp": "2018-05-14 11:30:30", "latitude": "24.96", "longitude": "121.21", "height0": "19.00", "height1": "27.00", "height2": "27.00", "height3": "20.00", "height4": "4.00"}
{"timestamp": "2018-05-14 11:30:36", "latitude": "24.96", "longitude": "121.21", "height0": "19.00", "height1": "28.00", "height2": "27.00", "height3": "20.00", "height4": "4.00"}
{"timestamp": "2018-05-15 15:04:00", "latitude": "24.96", "longitude": "121.21", "height0": "51.00", "height1": "5.00", "height2": "0.00", "height3": "0.00", "height4": "0.00"}
{"timestamp": "2018-05-15 15:04:07", "latitude": "24.96", "longitude": "121.21", "height0": "51.00", "height1": "51.00", "height2": "51.00", "height3": "51.00", "height4": "51.00"}
```

傳送時間

```
{"timestamp": "2018-05-14 11:30:30"}
{"timestamp": "2018-05-14 11:30:36"}
{"timestamp": "2018-05-15 15:04:00"}
{"timestamp": "2018-05-15 15:04:07"}
```

垃圾子車座標位置

```
"latitude": "24.96", "longitude": "121.21"
"latitude": "24.96", "longitude": "121.21"
"latitude": "24.96", "longitude": "121.21"
"latitude": "24.96", "longitude": "121.21"
```

5個超音波感測的垃圾高度資料

```
"height0": "19.00", "height1": "27.00", "height2": "27.00", "height3": "20.00", "height4": "4.00"}
"height0": "19.00", "height1": "28.00", "height2": "27.00", "height3": "20.00", "height4": "4.00"}
"height0": "51.00", "height1": "5.00", "height2": "0.00", "height3": "0.00", "height4": "0.00"}
"height0": "51.00", "height1": "51.00", "height2": "51.00", "height3": "51.00", "height4": "51.00"}
```

■ 伺服端分析垃圾量

垃圾量	感測值 ≥ 70公分	感測值介於 36~69公分	感測值 ≤ 35公分
滿			紅色垃圾桶
半滿		黃色垃圾桶	
少	綠色垃圾桶		

```
markers = [];
var data_json = JSON.parse(event.data);
for(var i=0; i<data_json.length; i++){
  var data = JSON.parse(data_json[i]);
  var latlng = new google.maps.LatLng(data.latitude, data.longitude);
  var h0 = parseFloat(data.height0);
  var h1 = parseFloat(data.height1);
  var h2 = parseFloat(data.height2);
  var h3 = parseFloat(data.height3);
  var h4 = parseFloat(data.height4);
  var avg = Math.round((h0+h1+h2+h3+h4)/5.0);

  var icon;
  if(avg >= 70){
    icon = {
      url: "green.png",
      scaledSize: new google.maps.Size(icon_size, icon_size)
    };
  }else if(avg <= 35){
    icon = {
      url: "red.png",
      scaledSize: new google.maps.Size(icon_size, icon_size)
    };
  }else{
    icon = {
      url: "yellow.png",
      scaledSize: new google.maps.Size(icon_size, icon_size)
    };
  }
}
```

依據感測值判斷垃圾量

■ 運用 Google map API 規劃最短路徑

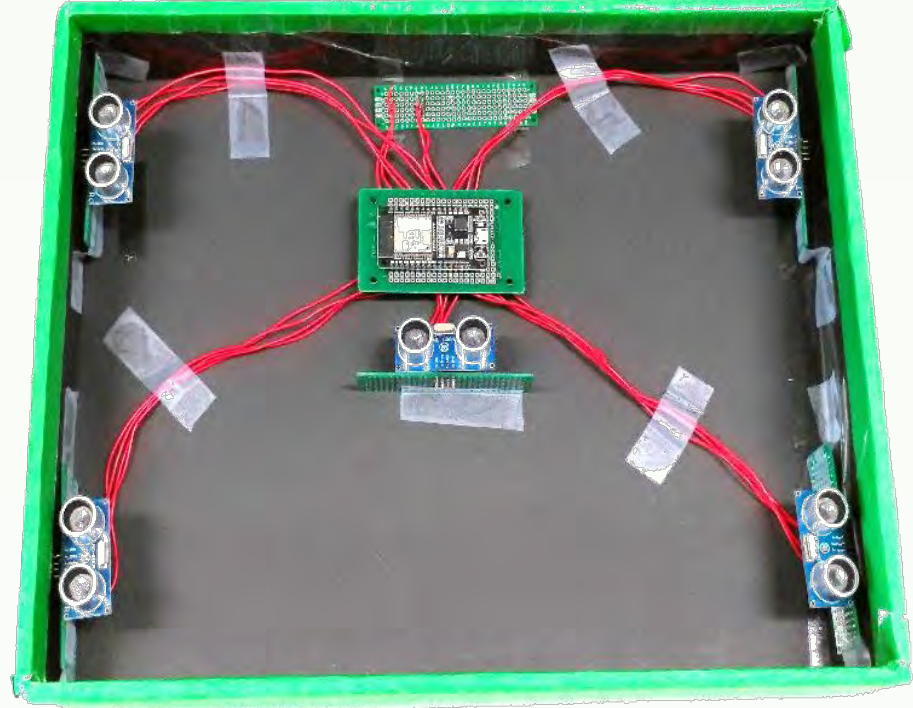


研究結果

■ 垃圾子車模擬模型



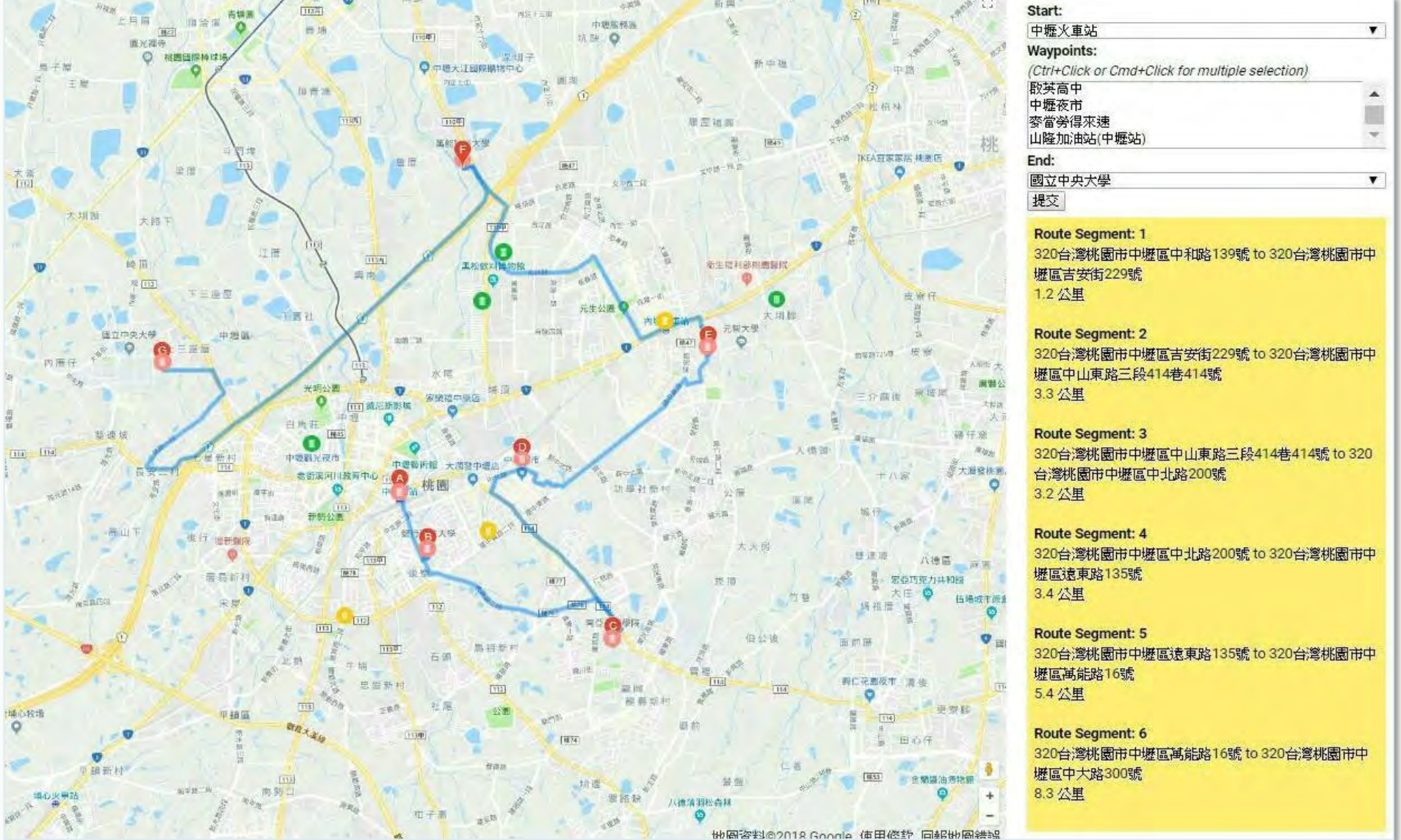
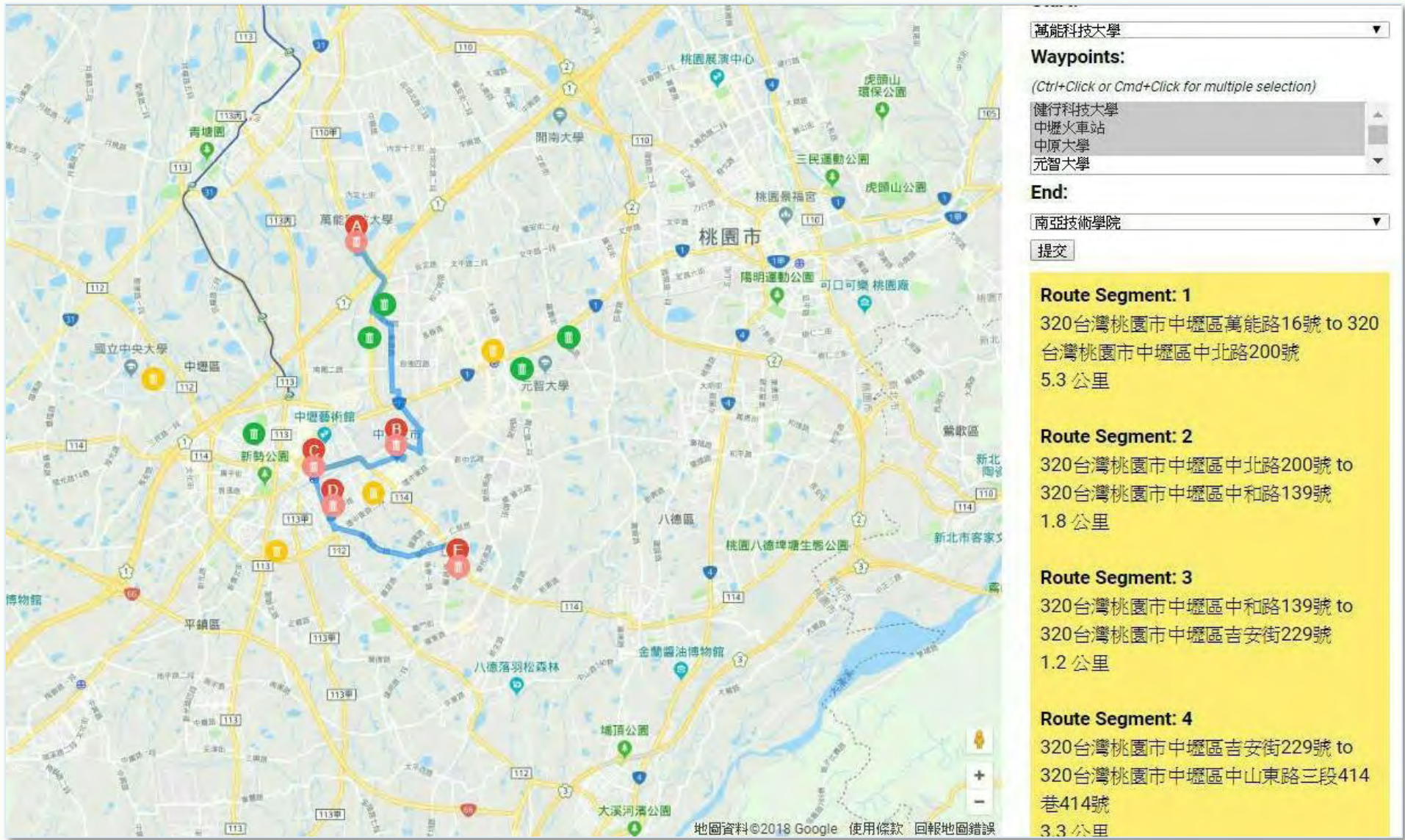
■ 超音波感測器設置圖



■ ESP32傳輸資料

ESP32將資料利用JSON格式傳送至伺服器端，傳送的資料包含傳送時間、垃圾子車座標位置及5個超音波感測的垃圾高度資料。

■ 伺服器端運用Google Map API規劃最短收垃圾路徑範例



■ 伺服器端依據需要回收垃圾的地點規劃出最短路徑。路線規劃上目前無法自動判定收那幾個，因為可能一台車收不完，需要派好幾台，因此讓清潔人員決定清運哪些垃圾子車。

討論與結論

■ 本研究之動態垃圾子母車清運模式成本效益

計算方式	一般垃圾車	垃圾子車
每車每日油費	35元x5公升=175元/日。	35元x約3公升=105元/日。
每車每年油費	175元x365天=63,875元/年。	105元x365天=38,325元/年。
全台5000輛計算	63,875元x5000台=319,375,000元。	38,325元x5000台=191,652,000元。
每車每年耗油量	5公升x365天=1825公升。	3公升x365天=1095公升。
每車每年排碳量	碳CO2=2.61x1825÷4.763公噸。	碳CO2=2.61x1095÷2.858公噸。
■ 錢每日省了(175-105元)=70元 ■ 錢每年省了(63,875元-38,325元)=25,550元 ■ 5000輛垃圾車錢省了(319,375,000元-191,652,000元)=127,723,000元 ■ 每年省油量(1825公升-1095公升)=730公升 ■ 每年減少排碳量(4,763公噸-2.858公噸)=1.905公噸		

■ 垃圾回收週期與環境問題 – 分析垃圾子車清運次數

■ 氣候因素：以台中為例在垃圾桶未滿的情形下垃圾回收週期範例

地名	一月	二月	三月	四月	五月	六月	七月	八月	九月	十月	十一月	十二月
臺中月均溫	16.6	17.3	19.6	23.1	26.0	27.6	28.6	28.3	27.4	25.2	21.9	18.1
垃圾回收週期	2天				1.5天	1天				1.5天	2天	

■ 垃圾量因素：可根據以前垃圾清運週期判斷垃圾累積速度

```
1 <?php
2 require_once('conn.php');
3
4 $sql = "select *, count(*) as c from full_times where time_date='2018-06-06' and device_id='data_test012' group by time_slot;";
5
6 $result = $conn->query($sql);
7
8 $filename = "test.xls";
9 $c_pre = "裝置 \t 日期 \t 時段 \t 滿的次數 \t\n";
10 $c = iconv("UTF-8","BIG5", $c_pre);
11 $contents = $c;
12
13 if ($result->num_rows > 0) {
14     while($row = $result->fetch_assoc()) {
15         $time_slot = checkTime($row['time_slot']);
16         $contents = $contents.$row['device_id']." \t".$row['time_date']." \t".$time_slot." \t".$row['c']." \t\n";
17     }
18 } else {
19     echo "0 results";
20 }
21 $conn->close();
22
23 ob_end_clean();
24 header("Content-type: text/html; charset=utf-8");
25 header('Content-type: application/vnd.ms-excel');
26 header('Content-Disposition: attachment; filename='.$filename);
27 echo $contents;
28
29 function checkTime($pre){
30     switch ($pre) {
31         case 0:
32             return "00:00~00:29";
33     }
34 }
```

查詢特定日期之垃圾子車的垃圾清運次數

	A	B	C	D
1	裝置	日期	時段	滿的次數
2	data_test012	2018/6/6	14:30~14:59	3
3	data_test012	2018/6/6	18:00~18:30	4
4	data_test012	2018/6/6	19:00~19:30	2
5				

查詢結果

時間	感測值	子車狀態	清運次數
timeStamp24	5.5	1	0
timeStamp25	5.5	1	0
timeStamp26	51	0	0
timeStamp27	51	0	0
timeStamp28	3	1	1
timeStamp29	5	1	1
timeStamp30	15	1	1
timeStamp31	51	0	1
timeStamp32	14	1	2
timeStamp33	4.6	1	2
timeStamp34	4.6	1	2
timeStamp35	51	0	2

查詢一段時間之垃圾子車的垃圾量

```
3
4 $sql = "select *, count(*) as c from full_times where time_date < '2018-06-06' and time_date > '2018-05-13' group by device_id,time_date;";
5
```

查詢結果

	A	B	C
1	裝置	日期	滿的次數
2	data_test012	2018/5/14	6
3	data_test012	2018/5/15	3
4	data_test012	2018/5/16	4
5	data_test012	2018/5/30	7
6	data_test012	2018/6/6	9

■ 增加垃圾子車數量

如果每個定點的垃圾子車經常未到設定的回收週期垃圾即滿溢，則需要規劃適時增加該定點的垃圾子車數量，例如當平均一天派車清運超過一次，則可增加該點之垃圾子車量。