

全國高級中等學校專業群科 106 年專題及創意製作競賽
「專題組」



群 別：電機與電子群

作品名稱：水陸兩用探勘車

關 鍵 詞：探勘、水陸兩用、Arduino

目錄

圖目錄	
壹、	摘要..... 1
貳、	研究動機..... 2
參、	研究方法..... 3
一、	硬體介紹..... 3
(一)、	HC-06 藍牙模組..... 3
(二)、	溫溼度感測器..... 4
(三)、	超音波感測器..... 5
(四)、	伺服馬達..... 6
(五)、	攝像頭..... 7
二、	軟體介紹..... 8
(一)、	App inventor 2..... 8
(二)、	Arduino..... 9
(三)、	Altium Designer..... 11
肆、	研究結果..... 13
一、	專題運作方式..... 13
二、	程式設計..... 13
1	連接、螢幕初始化..... 14
2	超音波感測器..... 15
3	伺服馬達..... 16
4	探照燈..... 17
5	直流馬達..... 19
6	溫溼度感測器..... 20
7	切斷..... 21
三	船體部分..... 23
伍、	討論..... 24
陸、	結論..... 25
柒、	參考資料及其他..... 26

圖目錄

圖一	HC-05 藍芽模組及接.....	3
圖二	HC-06 藍牙模組及接腳簡介.....	3
圖三	DHT11 溫溼度感測器.....	4
圖四	超聲波感測器.....	5
圖五	超音坡測距方法.....	6
圖六	伺服馬達.....	6
圖七	攝像頭.....	7
圖八	App inventor 2 操作介面.....	8
圖九	Arduino 硬體- uno 板.....	10
圖十	Arduino 程式設計介面.....	10
圖十一	線路圖.....	11
圖十二	佈線圖.....	12
圖十三	手機控制介面.....	13
圖十四	構想圖.....	23
圖十五	3D 列印成品.....	23

壹、摘要

台灣天災頻繁，當面臨嚴重災難時，頻頻傳出身處第一線的救災人員因公殉職的憾事，但若能以機器取代人工親自進入危險或未知的場所，也許就能減輕救災人員所面臨的風險。這類型的機器也使我们產生興趣而想要更進一步的進行探討。

此專題主要是以藍芽傳輸信號來使我们得以利用手機來控制車體的一切功能，而車體的行駛方向、溫、溼度感測、探照燈以及異物檢測我們則是使用 Arduino 來編寫相關程式。在車體的部分，我們設計成水陸兩用的模式，以便於應付災區多變的地形。

此外，我們也加上了攝像頭，使我们能得知探勘車目前身處的環境。

透過這些裝置，我們將能使用手機控制探勘車，讓我們身處在安全的地點就能夠得知災禍現場的情形，使自身安全有了更大的保障。

貳、研究動機

世上免不了會有天災人禍，用機器取代人力進行救災已成為世界各國的研究焦點，隨著科技的進步，發展出的救災機器人也日新月異，功能一代比一代強，然而，目前各國研製出的救災機器人價格相當高昂，導致不能普及，對於救災效率並沒有大幅的提升，於是我們嘗試做出價格低廉且能夠輔助救災事宜的探勘車，使其能有效地投身於救災工作，挽留住更多寶貴的生命。



下載APP | TVBS官網

首頁 直播 議題 Focus 焦點 ▾ 星聞 食尚 談

TVBS記者邱蘭婷：「像是這種偵測有毒氣體的機器人，他們只要發現一點點不尋常的氣體超過警戒值，就會開始嗶嗶叫。」

高田科技公司經理黃國龍：「它可以立即判定是不是據有爆炸性，或者是揮發性有機氣體的洩漏，它的濃度是不是在一個允許值的範圍內，如果不是的話，應該是立即要作人員的疏散。」

維護民眾和第一線救災人員的生命安危，刻不容緩，不過防災機器人在台灣使用仍不普及，主要是因為進口機器人價錢較貴，一台約250萬台幣，但經由學校研發製造，價錢可以壓到150萬左右。

正修科技大學校長龔瑞璋：「我們如何能夠用我們學校發展出來的技術，轉給我們國內的產業界，來做量產商業化，那當然這個普遍在價錢這個各方面的話，我想可以在救災防災方面會發揮更大的效果。」

高危險性的工作，用機器取代人力，畢竟人命是無法用金錢來衡量，誰也不願意再見到爆炸意外再次發生。

參、研究方法

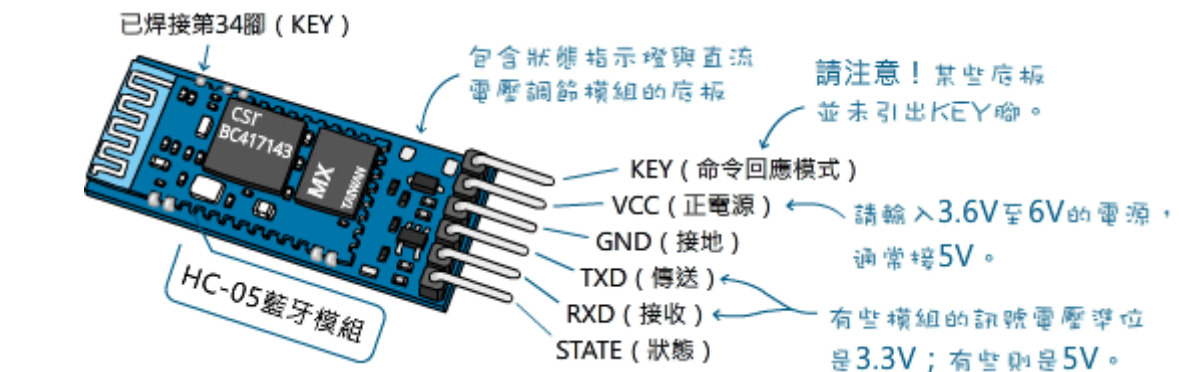
一、硬體介紹

(一)、HC-06 藍牙模組

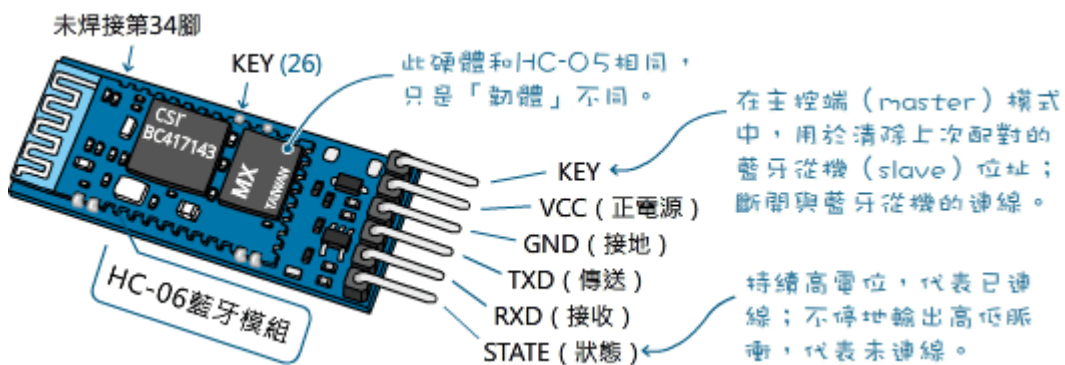
藍牙是一種無線通訊技術標準，用來讓固定與行動裝置，在短距離間交換資料，以形成個人區域網路，其使用短波特高頻(UHF)無線電波，經由 2.4 至 2.485 GHz 的 ISM 頻段來進行通訊。

市面上常見的藍牙有兩種，分別是 HC-05 以及 HC-06，兩種藍牙模組最大的差別在於 **HC-05** 是主/從一體型，出廠預設通常是「從端」模式，但是能自行透過 AT 命令修改，**HC-06** 則是主控端或從端模式，出廠前就設定好，不能更改，通常是「從端」模式。

本專題是採用 HC-06 藍牙模組。



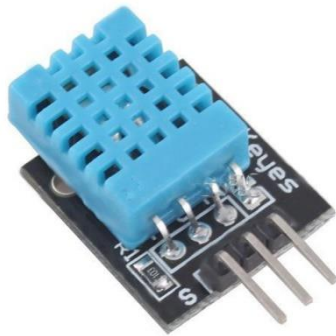
(圖 1) HC-05 藍牙模組及接腳簡介



(圖 2) HC-06 藍牙模組及接腳簡介

「命令回應模式」接腳（通常標示成“KEY”或“EN”），用於啟動 AT 命令模式，調整藍牙模組的設定（例如，調整序列埠的傳輸速率、修改模組的名稱、修改配對密碼...等等）。

(二)、溫溼度感測器



(圖 3) DHT11 溫溼度感測器

(表 1)

型號	DTH11
輸出方式	數位式
溫度量測範圍	0~50 度
精確度	正負 2 度
濕度範圍	20~80%
精確度正負	5%

DHT-11 是一個結合濕度計和測溫元件量測週遭空氣環境，並與一個高性能 8 位元單晶片相連接，將所量測到的溫、濕度資料拆解成為數位訊號，再由 data pin 腳將資料送出。

抓取資料時必須要特別注意時間的掌控，每筆資料的抓取時間間隔要 1~2 秒鐘，不能太快。與 DHT-22 比較，DHT-11 較不精密準確且溫濕度量測範圍不大，傳輸的距離可長 20m 以上，而且比 DHT-22 較小且便宜許多，是十分方便的簡易測試元件。以電阻方式感應濕度，並用 NTC 方式感應溫度(熱敏電阻)，DHT 內本來就有一顆 8 位元 MCU，MCU 的工作就是把感測到的溫度、濕度，以數位信號方式，透過 2 號腳位傳遞出去，所以其實那顆 MCU 也內建類比轉數位功效(ADC)，而溫濕度感測後的相關校準與係數，其實就燒錄在 MCU 的程式內，不過也無法更改了，因為是 OTP (One Time Program)，燒寫一次就不能再更動程式內容。

由於輸出已經是數位型式，所以 2 號接腳與 Arduino 本體，即便相隔一段距離仍可正常傳遞，傳遞過程中仍有一定程度的抗雜訊能力，一般而言能有 20 公尺以上長度。

(三)、 超音波感測器



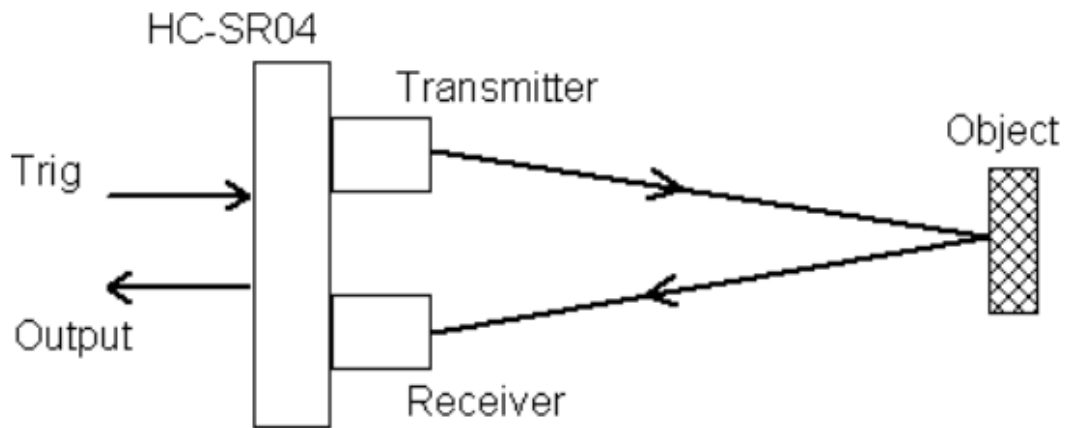
(圖 4)超聲波感測器

(表 2)

型號	HC-SR04
感測距離	2cm-400cm
精度	0.3 cm
感應角度	15 度

超音波感測器是由超音波發射器、接收器和控制電路所組成。當它被觸發的時候，會發射一連串 40 kHz 的聲波並且從離它最近的物體接收回音。超音波是人類耳朵無法聽見的聲音，因為它的頻率很高。

如下圖所示，超音波測量距離的方法，是測量聲音在感測器與物體之間往返經過的時間：



(圖 5)超音坡測距方法

聲音在空氣中的傳播速度大約是每 340 公尺，傳播速度會受溫度影響，溫度愈高，傳播速度愈快。假設以 340 公尺計算， $1000000 / 340 * 100 = 29.4$ microseconds，四捨五入後，可知聲音傳播 1 公分所需的時間為 29microseconds (百萬分之一秒)。

由於超音波從發射到返回是兩段距離，因此在計算時必須將結果除以 2 才是正確的物體距離。所以我們可以利用底下的公式算出物體距離 (距離單位為公分，其中 timing 是測量得到的音波傳播時間)： $\text{timing} / 58$

(四)、 伺服馬達



(圖 6)伺服馬達

(表 3)

型號	SG-90
重量	9g
尺寸	23*12.2*29mm
工作電壓	4.8V
工作溫度	0°C-55°C
轉矩	1.8kg-cm
運轉速度	0.1 秒 /60 度
旋轉角度	0 到 180 度
脈衝持續時間	1.0 ms 到 2.0 ms
脈衝寬度範圍	500~2400μs
PWM 訊號頻率	50 Hz

伺服馬達裡含有直流馬達、齒輪箱、軸柄以及控制電路，一般伺服馬達有三條線，電源（紅色）、接地（黑或棕色）、訊號線（白、黃、橘、藍，甚至是黑色）。透過訊號線傳送 PWM 脈波來控制軸柄的停止位置旋轉角度，這個訊號脈波必須每秒重複 50 次（也就是 50Hz），而脈衝持續時間長短便代表了馬達該將軸柄轉到什麼位置

(五)、 攝像頭



(圖 7)攝像頭

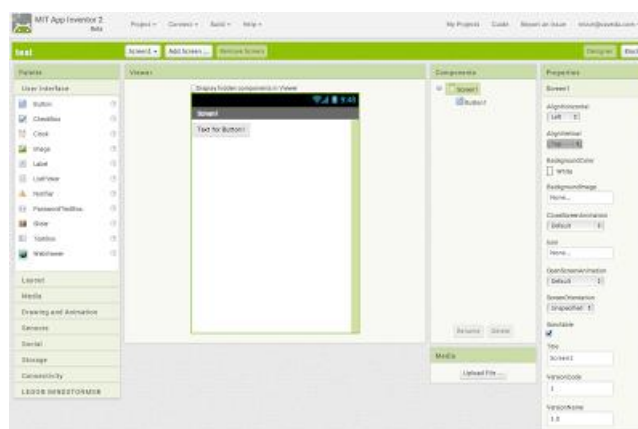
(表 4)

型號	Eachine 1000TVL CCD 迷你 FPV 攝像頭
輸入電壓	5V
耗電	55mA 5V
低照度	0.08Lux / F1.2
感光元件	1/3"CCD
影像格式	NTSC/PAL
解析度	1000TVL
鏡頭	2.8mm IR coated
視角	110°
工作溫度	0 to 40°C
尺寸	28 x 24.5 x 17.5mm
重量	10.4g

一般市售的網路攝影機都有延遲性以及距離的限制，無法達到幾乎同步的影像傳輸，然而空拍機的影像系統能夠解決此類問題，這套系統名為圖傳系統。這套系統的作用範圍約為 300 米到 600 米左右、操作頻率在 5.8GHz 的獨立頻道。在訊號不好的偏遠地區以及災區其效能遠比藍芽、網路傳輸來的更有效率。

二、 軟體介紹

(一)、 App inventor 2



(圖 8) App inventor 2 操作介面

Android 應用開發者是起先由 Google 提供的應用軟體，現在由麻省理工學院維護及營運。

它可以讓任何熟悉或不熟悉程序設計的人來創造基於 Android 作業系統的應用軟體。它使用圖形化界面，非常類似於 Scratch 語言和 StarLogo TNG 用戶界面，並使用拼塊程式 (Blocky) 來撰寫，以圖像方式來呈現讓程式初學者能夠更快上手，同時了解程式設計的脈落及邏輯架構。

App Inventor 所開發的程式可以直接在 Android 手機上執行，也可以安裝 Android 模擬器在電腦上執行。

(二)、 Arduino

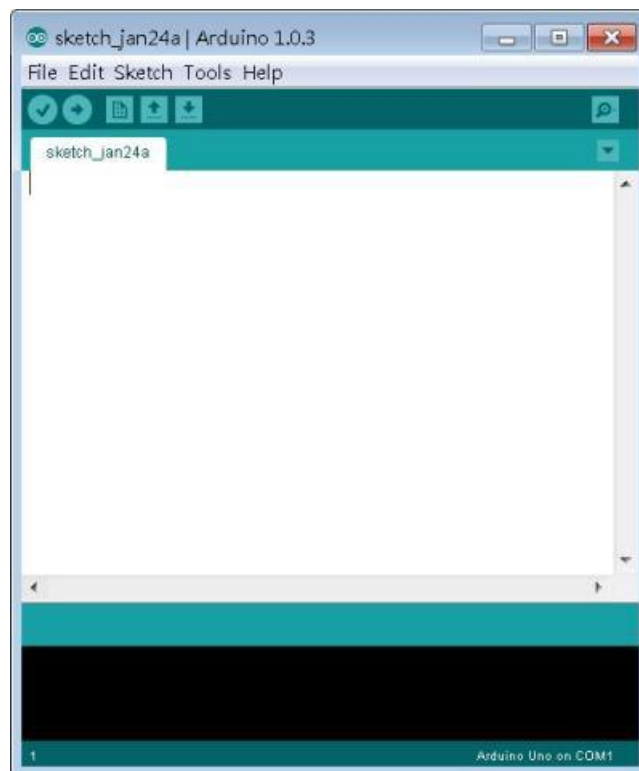
Arduino 是一種開放授權的互動環境開發技術，和 Android 相同的地方在於，兩者皆是使用開放原始碼的軟硬體平台。不同的是，Arduino 提供了簡單好用的 I/O 介面，並具可使用類似於 Java、C 語言等高階語言的開發環境。

以往要處理相關的電子設備時，需要透過工程師，逐一由單一小元件拼湊出整個電路。大多數的設計工具都是為了工程師設計，除了電路外還需要廣泛的知識，才有辦法完成電路。還好微處理器有了長足的進步，除了在使用上變得更為容易，價格上的減少更降低了學習的門檻。不僅軟體是開放源碼，硬體也是開放的。軟體的開發環境可在網上免費下載，而 Arduino 的電路設計圖也可從官方網站自行下載，依據自身之需求進行修，但須要符合創用 CC 授權條款(創用 CC 授權條款)。

開發簡單，參考資料多。在以往的硬體環境中，要開發微控制器的程式，開發者需要具備電子、電機及相關科系的背景，一般人需花費大量時間能有機會進入這個開發環境中。Arduino 學習門檻較為簡單，不需要電子電機相關科系的背景，也可以很容易學會 Arduino 相關互動裝置的開發。由於 Arduino 以公開共享為基礎，多數人都樂於分享自己的的創品，網路上能找的創作案子非常豐富。以此會基礎，有時只需要參考分享者的作品，依據自身的需求行調整，就可以在短時間內完成自己的創作。



(圖 9) Arduino 硬體- uno 板



(圖 10) Arduino 程式設計介面

(三)、Altium Designer

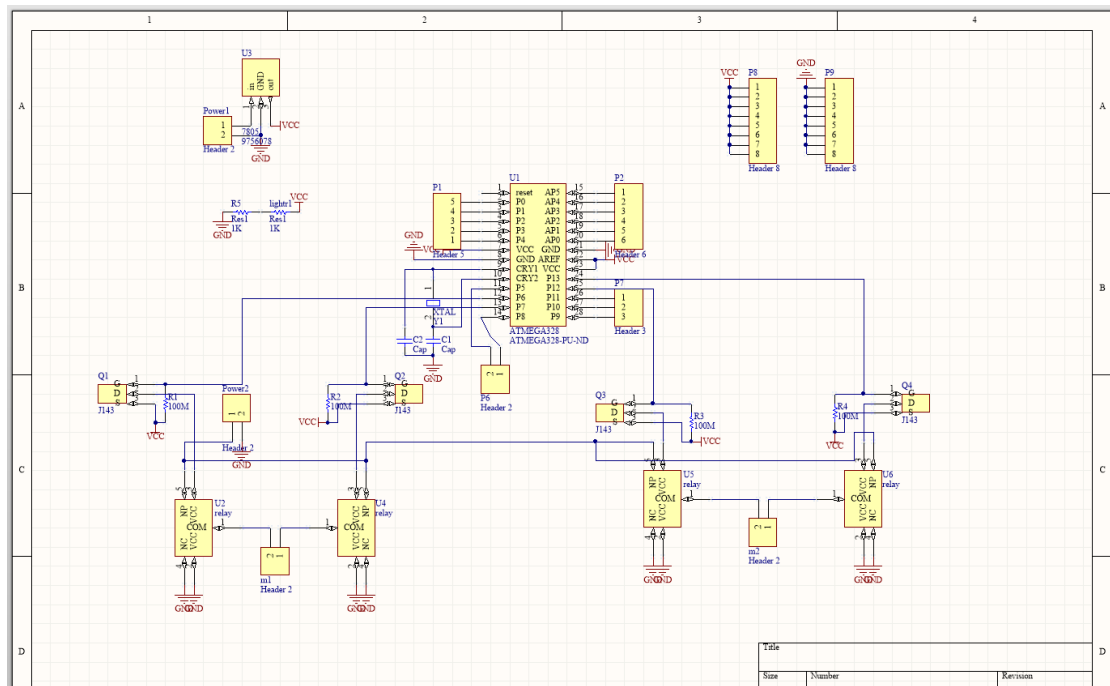
Altium designer 是 altium 公司開發的一款電子設計自動化軟體，主要功能有

- 1、電路圖設計
- 2、印刷電路板設計
- 3、電路類比模擬
- 4、FPGA 及邏輯器件設計
- 5、高級信號完整性分析

我們使用的部分是印刷電路板設計，經由以下幾個步驟即可完成我們所需的自定義電路板

(1) 電路繪製

- 1 對原理圖進行分析和檢查
- 2 建立標準元件庫；建立特殊元器件
- 3 印製板設計文件的建立
- 4 建立網路表



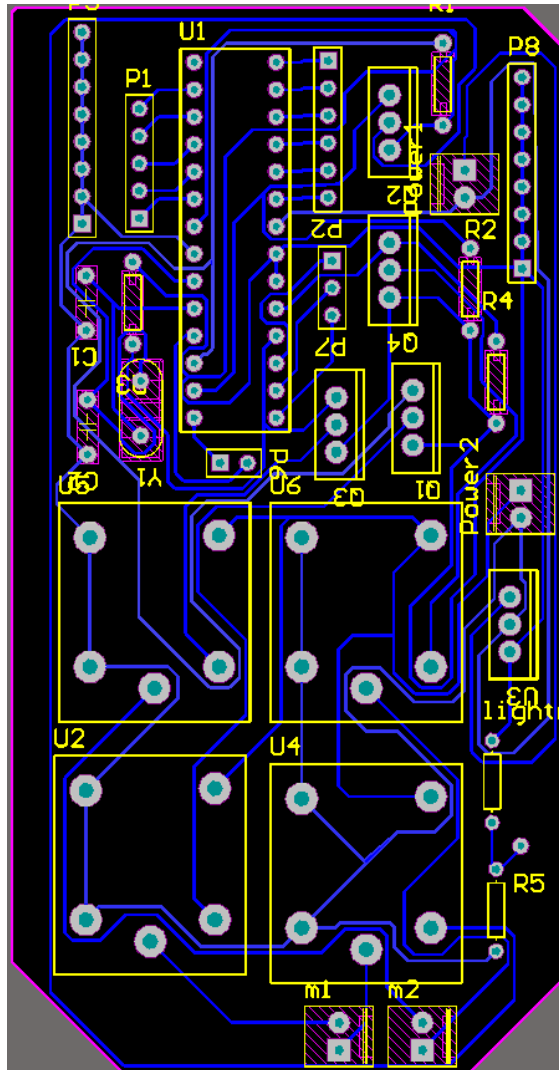
(圖11)線路圖

(2) 電路板佈線及板型設計

- 1 按照成品規格書的要求，將線寬、線距、層定義、過孔、全域參數等相關參數設置好
- 2 根據印製板安裝結構尺寸要求畫出邊框，參照原理圖，結合機構進行佈局，

檢查佈局。

- 3 參照原理圖進行預佈線，檢查佈線是否符合電路模組要求；修改佈線，並符合相應要求。



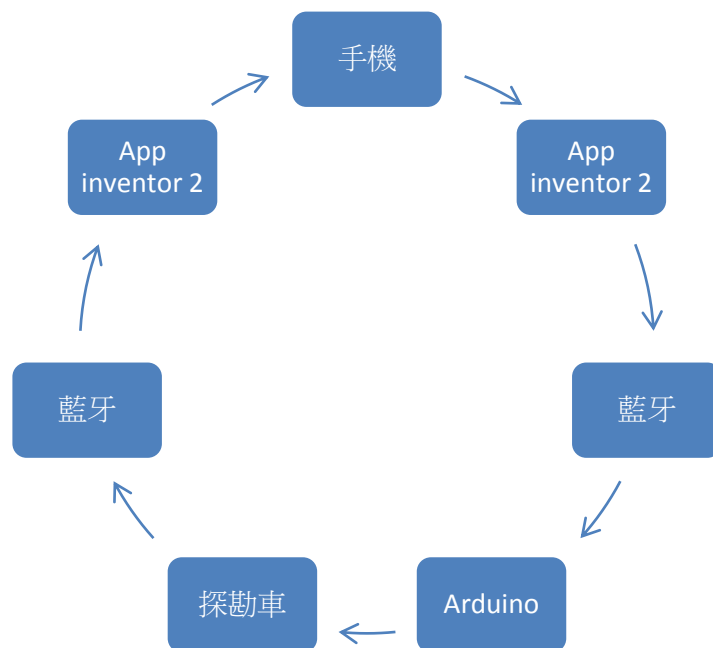
(圖12)佈線圖

(3) 電路板製作

- 1 檢查打樣無誤後，生成底片，BOM表等，設計完成。

肆、研究結果

一、專題運作方式



二、程式設計



(圖 13)手機控制介面

1 連接、螢幕初始化

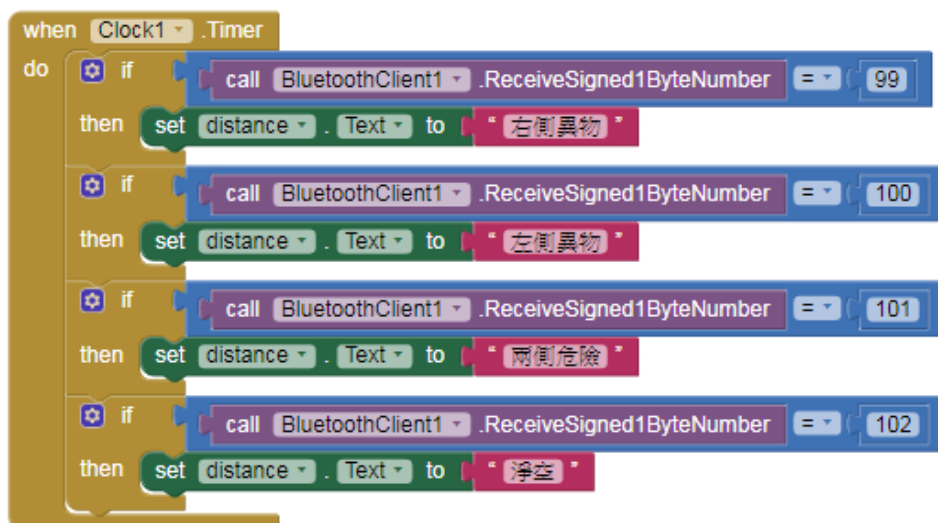
```
when connect .BeforePicking
do set connect . Elements to BluetoothClient1 . AddressesAndNames
```

```
when connect .AfterPicking
do if call BluetoothClient1 . Connect
    address connect . Selection
then set disconnect . Enabled to true
set connect . Enabled to false
set measure . Enabled to true
set Clock1 . TimerEnabled to true
set L . Enabled to true
set R . Enabled to true
set auto . Enabled to true
set on . Enabled to false
set 前 . Enabled to true
set 右 . Enabled to true
set 左 . Enabled to true
set 後 . Enabled to true
```

```
when Screen1 . Initialize
do set measure . Enabled to false
set disconnect . Enabled to false
set connect . Enabled to true
set Clock1 . TimerEnabled to false
set L . Enabled to false
set R . Enabled to false
set auto . Enabled to false
set on . Enabled to false
set 前 . Enabled to false
set 右 . Enabled to false
set 左 . Enabled to false
set 後 . Enabled to false
```

在藍牙連線到裝置前，先將之前配對好的藍芽裝置匯入到準備連接的列表中，選好要連接的藍牙裝置後，將藍牙列表關閉、啟動計時器開始每隔一段時間讀值，並把斷線的按鈕全部設定為可執行。

2 超音波感測器



```
22 unsigned long ping1()
23 {
24   digitalWrite (trigPin1, HIGH); //觸發腳設成高電位
25   delayMicroseconds (5); //持續5微秒
26   digitalWrite (trigPin1, LOW); //觸發腳設成低電位
27   return pulseIn(echoPin1, HIGH); //傳回高脈衝的持續時間
28 }
29
30 unsigned long ping2()
31 {
32   digitalWrite (trigPin2, HIGH);
33   delayMicroseconds (5);
34   digitalWrite (trigPin2, LOW);
35   return pulseIn(echoPin2, HIGH);
36 }
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57 d = ping1() / 58; //換算高脈衝時間成公分單位
58 m = ping2() / 58;
```

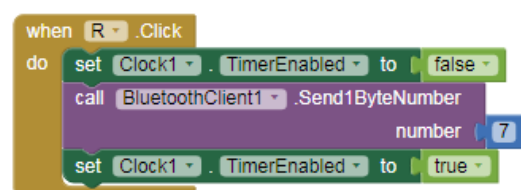
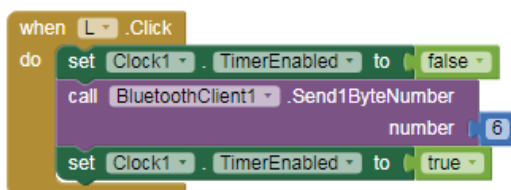
```

171 if( d<15 && m>=15 )
172 {
173   BT.write(99);
174 }
175
176 if( d>=15 && m<15 )
177 {
178   BT.write(100);
179 }
180
181 if( d<15 && m<15 )
182 {
183   BT.write(101);
184 }
185
186 if( d>=15 && m>=15 )
187 {
188   BT.write(102);
189 }

```

設定右側超音波感應器傳回來的距離為 d ，左側為 m ，安全距離為 15 公分
 如果 $d <$ 安全距離， $m >$ 安全距離，傳值 99 給手機，手機顯示右側異物
 如果 $d >$ 安全距離， $m <$ 安全距離，傳值 100 給手機，手機顯示左側異物
 如果 $d <$ 安全距離， $m <$ 安全距離，傳值 101 給手機，手機顯示兩側危險
 如果 $d >$ 安全距離， $m >$ 安全距離，傳值 102 給手機，手機顯示淨空

3 伺服馬達



```

18 int x = 90; //伺服馬達角度控制變數

```

```
115     case 7:
116     x=x-14;
117     if(x<6)
118     {
119         x=20;
120     }
121     myservo.write(x);
122     delay(15);
123     break;
124
125     case 6:
126     x=x+14;
127     if(x>174)
128     {
129         x=160;
130     }
131     myservo.write(x);
132     delay(15);
133     break;
```

把攝像頭裝在伺服馬達上，藉由控制伺服馬達讓攝像頭轉動
設定一變數為 x，使伺服馬達轉動至 x 度
一開始將其設定在 90 度，每按下一次左或右鍵，讓 x 加減 14 度，最大不超過
174 度，最小不低於 6 度。

4 探照燈

```

Initialize global y to 0

when on .Click
do
  set Clock1 . TimerEnabled to false
  set global y to get global y + 1
  if get global y == 1
  then
    set on . Text to 1
    call BluetoothClient1 .Send1ByteNumber
    number 9
  if get global y == 2
  then
    set on . Text to 2
    call BluetoothClient1 .Send1ByteNumber
    number 10
  while test get global y == 2
  do
    set global y to 0
  set Clock1 . TimerEnabled to true

```

```

Initialize global x to 0

when Auto .Click
do
  set Clock1 . TimerEnabled to false
  set global x to get global x + 1
  if get global x == 1
  then
    set Auto . Text to 1
    call BluetoothClient1 .Send1ByteNumber
    number 10
    set on . Enabled to true
  if get global x == 2
  then
    set Auto . Text to 2
    call BluetoothClient1 .Send1ByteNumber
    number 11
    set on . Enabled to false
  while test get global x == 2
  do
    set global x to 0
  set Clock1 . TimerEnabled to true

```

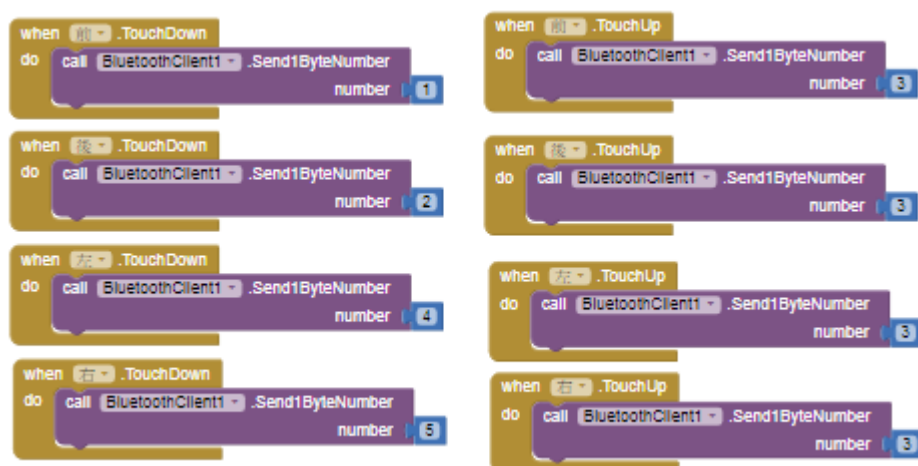
```

20 int a = 2;
141     case 9:
142         a=0;
143         break;
144
145     case 10:
146         a=1;
147         break;
148
149     case 11:
150         a=2;
151         break;
152     }
155 if(a==2)
156 {
157     int pr = analogRead(A2); //讀取光敏電阻的值，會回傳0~1023之間的值
158     digitalWrite(2,pr>pr_min?LOW:HIGH); // 若大於pr_min，熄滅LED，若小於就點亮
159 }
160
161 if(a==1)
162 {
163     digitalWrite(2,LOW);
164 }
165
166 if(a==0)
167 {
168     digitalWrite(2,HIGH);
169 }

```

一開始將探照燈設定在自動模式，按一下手/自動按鈕切換成手動模式後，即可透過旁邊的開/關按鈕來控制燈的明滅
以變數 a 來決定燈的狀態，a 為 2 時，狀態為以光敏電阻控制明滅的自動狀態，a 為 1 或 2 時，狀態為以手機按鈕控制明滅的手動狀態。

5 直流馬達



```

88     case 1 :
89         digitalWrite(6,HIGH);
90         digitalWrite(7,HIGH);
91         break;
92     case 2 :
93         digitalWrite(13,HIGH);
94         digitalWrite(12,HIGH);
95         break;
96     case 3 :
97         digitalWrite(7,LOW);
98         digitalWrite(6,LOW);
99         digitalWrite(12,LOW);
100        digitalWrite(13,LOW);
101        break;
102     case 4 :
103        digitalWrite(6,HIGH);
104        digitalWrite(12,HIGH);
105        break;
106     case 5 :
107        digitalWrite(7,HIGH);
108        digitalWrite(13,HIGH);
109        break;

```

當手機介面上的前、後、左、右鈕被按下時，App inventor 2 會透過藍牙傳一個相對應的值給 Arduino，當 Arduino 收到值時，會驅使晶片 L293D 工作，例如當收到指令為前時，L293D 會使兩顆馬達為順向導通狀態，此時兩顆馬達都會

正轉，使探勘車能向前行駛，而當 Arduino 收到右指令時，L293D 會使其中一顆馬達順向導通，使另一顆馬達逆向導通，此時會出現一顆正轉一顆逆轉的情況，探勘車就能順利執行右轉的指令。

6 溫溼度感測器

```
initialize global text to 0
initialize global A0 to 0
initialize global A1 to 0
initialize global A3 to 0

when measure Click
do
  set measure.Enabled to false
  set Clock1.TimerEnabled to false
  call BluetoothClient1.Send1ByteNumber
  number 69
  set global text to call BluetoothClient1.ReceiveSigned1ByteNumber
  if get global text == 97
  then
    set global A0 to call BluetoothClient1.ReceiveSigned1ByteNumber
    set tem.Text to join get global A0
    " °C "
  set global text to call BluetoothClient1.ReceiveSigned1ByteNumber
  if get global text == 98
  then
    set global A1 to call BluetoothClient1.ReceiveSigned1ByteNumber * 20
    set global text to call BluetoothClient1.ReceiveSigned1ByteNumber
    set global A0 to get global A1 + get global text
    set global A3 to get global A0 - 3 / 20
    set num.Text to join floor get global A3
    " % "
    set global text to 0
  set Clock1.TimerEnabled to true
  set measure.Enabled to true
```

```

59   DHT11.read(Dht11Pin);
60
61   if(BT.available()>0)
62   {
63     int cmd = BT.read();
64     if(cmd==69){
65       int tem = (int)DHT11.temperature;
66       int hum = (int)DHT11.humidity;
67
68       Data[0] = 97;
69       Data[1] = tem;
70       Data[2] = 98;
71       Data[3] = hum;
72       Data[4] = hum/20;
73       Data[5] = hum%20;
74
75       for(int i=0;i<5;i++)
76       {
77         BT.write( Data[i] );
78       }

```

按一下測量鈕後開始傳送溫溼度，如果手機讀到 97，就接續讀取後面資料，這是溫度的數值。反之如果是 98 就是濕度，因為資料傳輸過程中難免會遺漏，因此我們都是先抓頭（就是 97 與 98），確定這是一筆新的資料開頭之後，再去讀取數值，否則可能會發生數值亂跳的情形。另一方面，App Inventor 在藍牙傳輸的延遲狀況較明顯，測量時要多按幾下。

7 切斷


```

when disconnect .Click
do
  call BluetoothClient1 .Send1ByteNumber
  number 8
  set connect . Enabled to true
  set disconnect . Enabled to false
  set measure . Enabled to false
  set Clock1 . TimerEnabled to false
  set distance . Text to "0"
  set tem . Text to "0"
  set hum . Text to "0"
  set auto . Text to "手動"
  set on . Text to "開"
  set on . Enabled to false
  set 前 . Enabled to false
  set L . Enabled to false
  set R . Enabled to false
  set 右 . Enabled to false
  set 左 . Enabled to false
  set 後 . Enabled to false
  call BluetoothClient1 .Disconnect

```

```

59   DHT11.read(Dht11Pin);
60
61   if(BT.available()>0)
62   {
63     int cmd = BT.read();
64     if(cmd==69){
65       int tem = (int)DHT11.temperature;
66       int hum = (int)DHT11.humidity;
67
68       Data[0] = 97;
69       Data[1] = tem;
70       Data[2] = 98;
71       Data[3] = hum;
72       Data[4] = hum/20;
73       Data[5] = hum%20;
74
75       for(int i=0;i<5;i++)
76       {
77         BT.write( Data[i] );
78       }

```

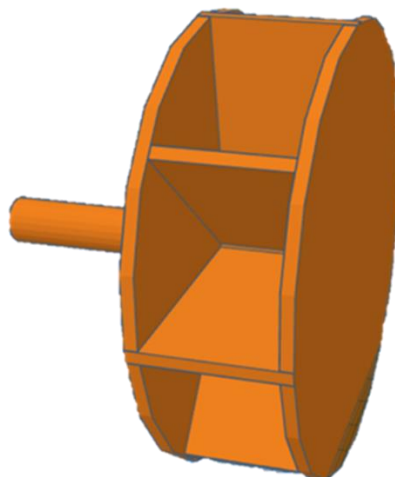
```
129     case 8:  
130     x=90;  
131     myservo.write(x);  
132     delay(15);  
133     break;
```

將伺服馬達的角度歸回正中央，清空溫濕度及距離感測，把所有按鈕除了連接無效化後，斷開藍牙連接。

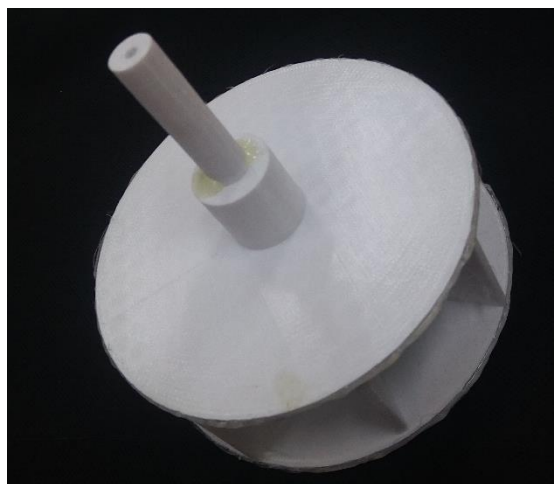
三 船體部分

由於我們想做出一艘水陸兩用的探勘車，所以在輪子的設計上我們並沒有採用傳統的輪胎，而是以水車做為基底，改造而成的(見圖 12)，我們利用 3D 列印來實現我們的構想(見圖 13)，此設計在路上能行使正常，在水上也能藉由打水產生的推進力向前行駛。

在浮力方面，原本的船體並無法勝任，所以我們在船體兩側加上浮瓶，在船尾加裝保麗龍以增加浮力，使的我們的探勘車確實能達到兩棲的作用性。



(圖 14)構想圖



(圖 15)3D 列印成品

伍、討論

在成品方面，溫溼度感測器以及超音波感測器的功能一切正常，但在控制設像頭轉向的伺服馬達部分則有一些小問題，因為偶爾在傳輸信號的過程中會有雜訊的干擾，導致伺服馬達會產生抖動的現象，有時甚至會不受控制的亂轉。而攝像頭的部分則有些許瑕疵，因為我們採用的這款攝像頭必須經由該公司的 APP 才能讓影像傳回手機，所以在控制探勘車時，我們必須透過兩台手機，一台用來操作，一台用來觀看畫面，這部分並不是我們一開始想要的。

而本專題最大的問題是行使方面的問題，我們的探勘車在水上移動是完全沒有問題的，但在路上行駛時只有前進後退的功能屬正常，左轉右轉並不能像在水上時能夠原地自轉 360 度，只能轉非常微小的角度，導致轉向時間非常緩慢，我們推測是因為轉矩不足所造成，我們採用的馬達齒輪組擁有變速效果，所以我們將其轉速改為 38rpm，此時的轉矩達到 2000 多克，雖然能加快轉向速度但在水上時由於轉速過慢導致無法行駛，所以想徹底解決這些問題我們可能要重新採購馬達，這樣才能在有足夠的轉矩時還能保有足夠的轉速。

陸、結論

經過此次的專題，我們認為最大的收穫並不是在學術方面，而是人與人之間的溝通，如何從互相質疑的一盤散沙，進而發展成一支向心力極強，能夠為了同一個目標願意犧牲奉獻的團隊，這是我們最寶貴的經驗，在以後也一定還會用到這種能力。

柒、參考資料及其他

【進階學習】探析 DHT11 溫度感測器

<https://makerpro.cc/2015/05/know-more-about-dht11-temperature-sensor/>

NTC 熱敏電阻溫度量測原理

<http://www.ni.com/white-paper/7112/zht/>

濕度量測技術概述 - 建成科學儀器股份有限公司

http://www.scco.com.tw/General_Eastern/RH-Measurement.htm

NTC 熱敏電阻器是如何工作的?

<http://www.shin-hang.com.tw/index.php?do=product&id=64>

小狐狸事務所: Arduino 溫濕度感測器 DHT11 測試

<http://yhhuang1966.blogspot.tw/2015/08/arduino-dht11.html>

Arduino 範例 19：利用 DHT11 抓取溫度和濕度

<http://ming-shian.blogspot.tw/2014/05/arduino19dht11.html>

Cooper Maa: HC-SR04 超音波感測器介紹

<http://coopermaa2nd.blogspot.tw/2012/09/hc-sr04.html>

【Arduino 教學】實作 07-伺服馬達 SG-90

<https://www.youtube.com/watch?v=cLs5xwUgGkc>

mBlock & Arduino (9) 伺服馬達控制

<https://openhome.cc/Gossip/CodeData/mBlockArduino/mBlockArduino9.html>

Arduino 第十一課(伺服馬達)

<http://blog.ilc.edu.tw/blog/index.php?op=printView&articleId=607043&blogId=4950>

Arduino 練習：伺服馬達以 Tower Pro SG90 為例

<http://yehnan.blogspot.tw/2013/09/arduinotower-pro-sg90.html>

HC-05 與 HC-06 藍牙模組補充說明 (一)

<https://swf.com.tw/?p=693>

HC-05 與 HC-06 藍牙模組補充說明（二）：連接電路

<https://swf.com.tw/?p=705>

葉難: 將 Arduino Uno 當做一台 ISP 線上燒錄器，燒錄 bootloader 或 sketch 到麵包板上的 ATmega328P-PU 晶片

<http://yehnan.blogspot.tw/2013/02/arduino-unoispbootloadersketchatmega328.html>

Arduino 與 AVR 微控制器相關詞彙

<http://yehnan.blogspot.tw/2013/01/arduinoterms.html>

http://newsletter.ascc.sinica.edu.tw/news/read_news.php?nid=2782