中華民國第 61 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

052313

無人機的消防應用

學校名稱：國立屏東高級工業職業學校

作者：	指導老師：
職三　徐瑞驊	林昭福 王彥貿

關鍵詞：無人機、消防

摘要

科技的日新月異，人們除了在日常娛樂拍攝中結合了空拍機，在各個職業領域為了便利與效率也運用上了空拍機，傳統空拍機的空中攝影技術也逐漸結合各項感測器、機械結構來達到職業上的需求。本專題使用單晶片 Arduino nano、MEGA Mini Pro、ESP-32 和飛行控制器 Pixhawk 為主要控制板打造一台空拍機，並結合熱成像與各感測模組達到消防救援的功效。

壹、 研究動機

現今生活中，空拍機在社會中已經普及化，在電影或是紀錄片中的各種畫面都使用到空拍機來達成高空拍攝的目的，但空拍機除了空拍外難道不能結合其他功能嗎?它擁有靈活小巧的身體，能去到許多人類無法到達的地方，如果能將空拍機裝上各種模組並將數據從空中回傳至地面藉由螢幕顯示各項數據，如此就能減少一些人力跟人身安全的顧慮，人就不需要為了一些數據而冒險到這些危險的地方。火災一直是社會中常見的災害，尤其在中秋節大多數的人都會烤肉、放煙火，火災的發生率也升高，但在一些交通不便或擁擠的地方發生火災會使得消防車無法迅速到達，如果能進化空拍機的影像功能，讓空拍機在空中進行溫度感測如此一來就能從空中快速發現火源與罹難者，並在初期迅速將火源撲滅，也不用擔心交通不便與擁擠的狀況，如此就能減少救援時的時間與危險。

貳、 研究目的

一、實現無人機滅火

與市面上所賣的娛樂型空拍機不同，本研究必須要依照所需的負重、抗風自行打造一台空拍機，並增加投放裝置，在需要一定負重的狀況下同時也要考慮空拍機的攜帶方便性。

二、空拍機與感測模組結合

在空拍機上裝置溫度、氣體(CO_2、CO 等)感測模組來滿足消防救援的需求。

三、空拍機與熱成像結合

利用空拍機空中飛行的性質加裝熱成像的功能，讓消防或其他單位能從火災中迅速找到火源與罹難者。

參、 研究設備與器材

材料名稱	數量	功能
Arduino nano	3	遙控器控制中心
Arduino MEGA Mini Pro	1	空拍機控制中心
ESP-32 開發版	2	熱像儀主機
2.4GHz 功率放大器	1	增加遙控距離
nRF24L01 無線模組	6	傳輸&接收資訊
RX5808 接收模組	2	接收影像
圖傳發射器	1	發送影像
MQ-9 感測器	1	感測一氧化碳
LM35	1	感測溫度
SSD1306 OLED	2	顯示 FPV 控制畫面
HMI 螢幕	1	控制飛行模式
7 吋 LCD	1	顯示 FPV 畫面
TFT 彩屏螢幕	1	顯示熱像素
AMG8833	1	熱像儀模組
H4 680 機架	1	空拍機機架
X4110S 無刷馬達	4	空拍機動力
XRotor 40A PRO 電調	4	控制馬達轉速
Runcam2-4K 像機	1	空拍主畫面
Tarot3D III雲台	1	穩定主畫面
Pixhawk2.4.8 飛控	1	空拍機主控板
3DR Radio 數傳	1	傳輸空拍機上的數據

表一.研究設備與器材

肆、 研究方法及過程

一、 理論討論

(一) 2.4GHz

2.4GHz 頻帶是一個被普遍使用的頻帶，生活中大多數的傳輸設備都有它的蹤影，像是我們的 WiFi、無線滑鼠、鍵盤。因為空拍機的操控主要是傳輸數值資料不需要高傳輸速率，再加上傳輸距離必須要遠與穩定，所以目前常見的空拍機大多都是使用 2.4GHz 來進行操控。

(二) 5GHz

2.4GHz 與 5GHz 的主要差別在於他們的波長，2.4GHz 波長較長，也就比較容易繞射，傳輸距離較遠，5GHz 則波長較短不易繞射，容易被障礙物阻擋，傳輸距離也就比較短，但也因此 5GHz 的傳輸速率比 2.4GHz 來的快。我們常見的空拍圖傳系統因為主要傳輸的資料是影像，如果採用 2.4GHz 那在進行影像傳輸時可能會有延遲，這樣將造成我們在地面所看到的影像與空拍機實際拍到的影像不同步，所以在空拍圖傳上常見是以 5.8GHz 回傳影像。

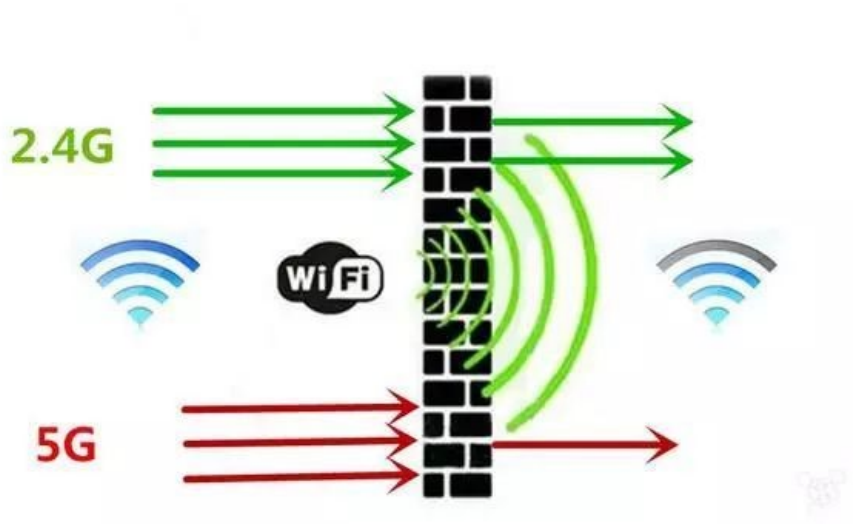


圖 1　2.4G 與 5G

(三) 無人機的等級

1. 低高度、視距內：休閒娛樂用途。
2. 中高度、視距外：公務、研究用途。
3. 高高度、視距外：軍事用途。

這次研究所做出來的無人機比較偏向中高度，用來勘查、獲取影像、高空消防救援，還有獲取溫度、氣體濃度等各項數據。

二、 系統架構

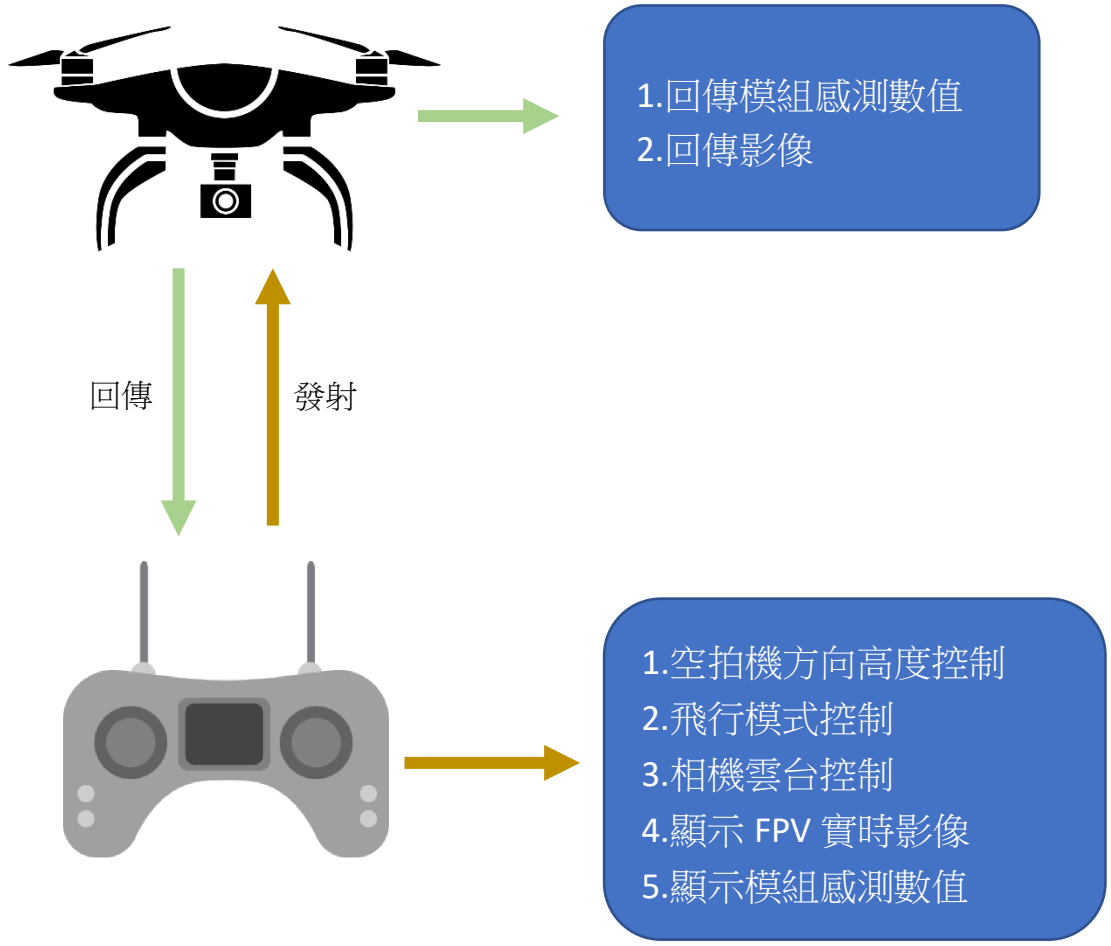


圖 2　系統架構

三、 研究過程

研究過程主要分為四大部分，遙控器製作、空拍機製作、熱像儀製作、模組化功能。

(一) 遙控器製作

1. 遙控器發射機

遙控器發射機主要來控製空拍機的方向與相機雲台的上下左右，所以使用了兩個遊戲搖桿與兩個可變電阻，遊戲搖桿用來控製空拍機方向，可變電組用來控制雲台方向。

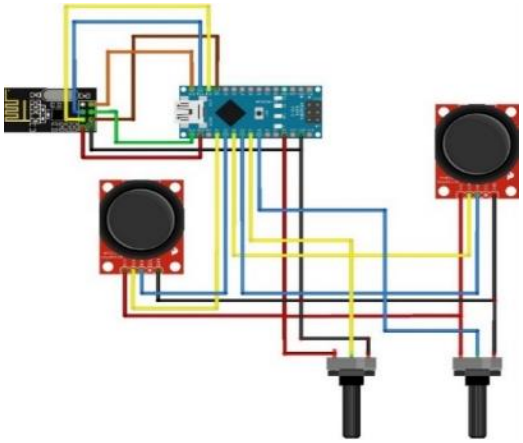


圖 3　發射機線路圖

因為是遙控器所以將 nRF24L01 無線模組先設為發射模式。

```
rf24.stopListening();          // 停止偵聽；設定成發射模式
```

再將 Arduino nano 板由搖桿模組與可變電組感測到的 0~1023 類比訊號轉換成位元 0~255 。

```
data.j1PotY = map(analogRead(A0), 0, 1023, 0, 255);//將0~1023類比訊號轉換成0~255
  data.j2PotX = map(analogRead(A2), 0, 1023, 0, 255);//將0~1023類比訊號轉換成0~2
  data.j2PotY = map(analogRead(A3), 0, 1023, 0, 255);//將0~1023類比訊號轉換成0~2
  data.pot1 = map(analogRead(A7), 0, 1023, 0, 255);//將0~1023類比訊號轉換成0~255
```

最後經由 nRF24L01 無線模組將數值 0~255 發射出去。

```
radio.write(&data, sizeof(Data_Package));//發送0~255數值
```

使用 nRF24L01 無線模組來做雙向溝通，空拍機上裝設兩個 nRF24L01 無線模組，一個回傳空拍機上模組所感測到的數據，一個接收遙控器的控制訊號。

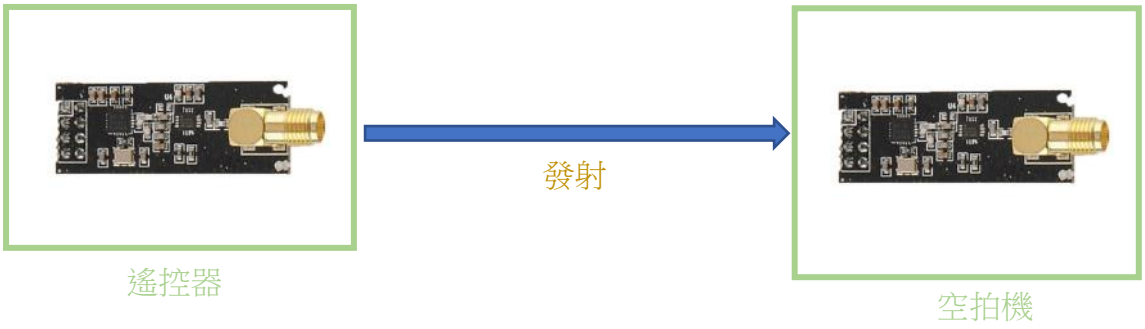


圖 4　接收與發射

官方宣稱 nRF24L01 的傳輸距離能達到 1100 公尺，但實際會有訊號干擾、建築物阻擋訊號等問題，在建築物內實測傳輸距離是一樓到三樓，訊號穿透性大概兩層樓就是極限了，我又在空曠處實際進行距離測試但到了 500 公尺的距離訊號就斷斷續續了，空曠處 500 公尺的傳輸距離對於空拍機來說有些太短，所以決定使用 2.4GHz 功率放大器來增加它的傳輸距離。

圖 5　2.4G 功率放大器

測試方式是如果有訊號，讓 oled 顯示 Hello；沒有訊號則顯示 NO。

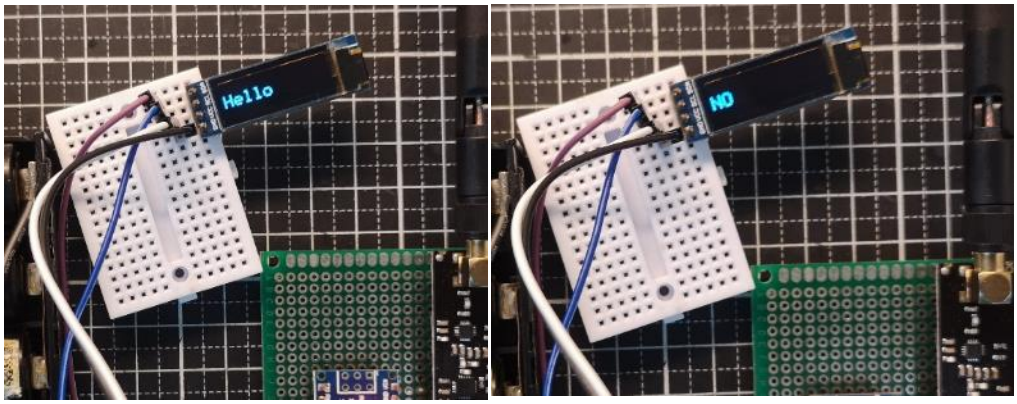


圖 6　2.4G 距離測試

空曠處	無放大器	有放大器
距離	344 米	704 米

表二. 距離測試

2. USART HMI 觸控螢幕

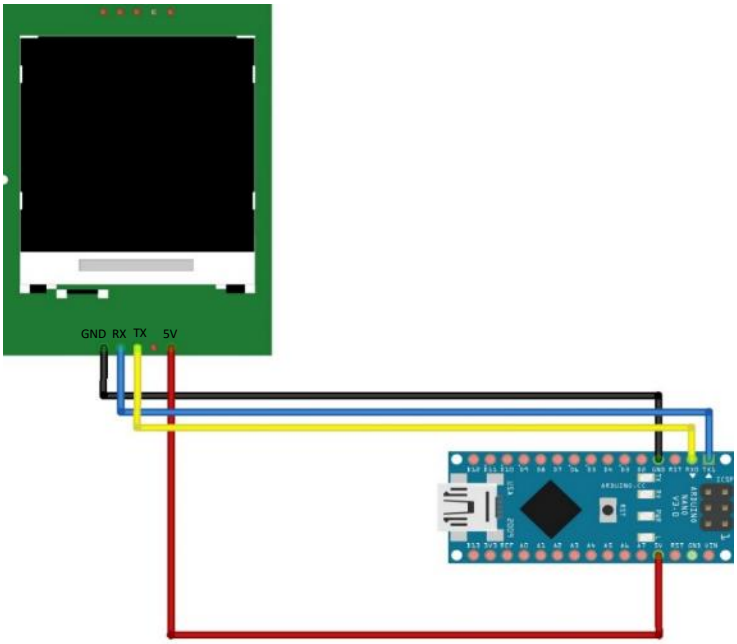


圖 7　HMI 螢幕線路圖 fritzing

使用 USART HMI 設計螢幕背景樣式，並增設幾個觸碰按鈕用來控制空拍機的飛行模式。空拍機除了單純控制其方向之外還能控制其飛行模式，像是定高、定點、與自動返航等。一般的 2.4g 遙控器大多都是利用實體開關來進行飛行模式控制，但遙控器的大小限制所以開關的數量也不能太多，但使用觸碰螢幕來進行控制了話，理論上只要單晶片的輸出接腳夠多，要設置幾個觸碰按鈕都沒問題。

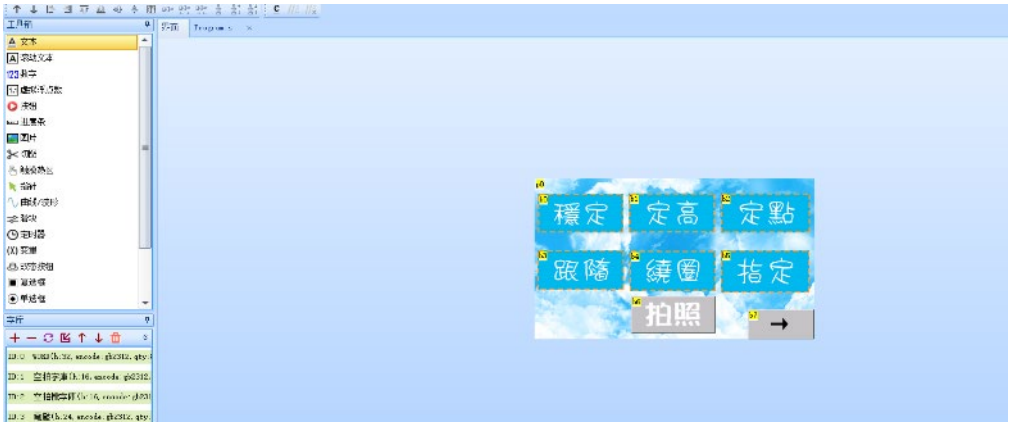


圖 8　HMI 設計

撰寫 HMI 螢幕在 Arduino 上的程式內容。

```
//宣告觸碰按鈕
NexButton bA = NexButton(0, 2, "b0"); //穩定
NexButton bB = NexButton(0, 3, "b1"); //定高
NexButton bC = NexButton(0, 4, "b2"); //定點
NexButton bD = NexButton(0, 5, "b3"); //跟隨
NexButton bE = NexButton(0, 6, "b4"); //繞圈
NexButton bF = NexButton(0, 7, "b5"); //指定
NexButton bG = NexButton(0, 8, "b6"); //快門
//NexButton bH = NexButton(0, 9, "b7"); //進入第二頁
//NexButton bI = NexButton(1, 10,"b8"); //進入第一頁
NexButton bJ = NexButton(1, 7, "b9"); //更新數據


//宣告文字
NexText Volt = NexText(1, 5, "Volt");
NexText Gas1 = NexText(1, 3, "t0");
NexText Gas2 = NexText(1, 4, "t1");
NexText Gas3 = NexText(1, 6, "t3");


//宣告可觸碰物件
NexTouch *nex_listen_list[] = {
  &bA,//穩定
  &bB,//定高
  &bC,//定點
  &bD,//跟隨
  &bE,//繞圈
  &bF,//指定
  &bG,//快門
  //&bH,//第二頁
  //&bI,//第一頁
  &bJ,//更新數據
  NULL
};
```

```
void bAPushCallback(void *ptr) {        //如果按下bA

  data.hmi =1100;                       //data.hmi的PWM訊號為1100us
}

void bBPushCallback(void *ptr) {        //如果按下bB

  data.hmi =1300;                       //data.hmi的PWM訊號為1300us
}

void bCPushCallback(void *ptr) {        //如果按下bC

  data.hmi =1400;                       //data.hmi的PWM訊號為1400us
}

void bDPushCallback(void *ptr) {        //如果按下bD

  data.hmi =1550;                       //data.hmi的PWM訊號為1550us
}

void bEPushCallback(void *ptr) {        //如果按下bE

  data.hmi =1700;                       //data.hmi的PWM訊號為1700us
}

void bFPushCallback(void *ptr) {        //如果按下bF

  data.hmi =1900;                       //data.hmi的PWM訊號為1900us
}

void bGPushCallback(void *ptr) {        //如果按下bG

  data.shu =2000;                       //data.shu的PWM訊號為2000us
}

void bGPopCallback(void *ptr) {
   data.shu =1000;                      //data.shu的PWM訊號為1000us
```

從 Arduino nano 板上的 A0 腳位讀取遙控器電池返回的值 0~1024，再藉由這些數值去換算電池目前的電壓，以 18650 鋰電池來說 3.6v 算快沒電；4.2v 算滿電。

```
          for(int i=0;i<1000;i++){            //取平均值
int Vl = analogRead(A0);          //從A0口讀取電壓數據存入剛剛創建整數型變量Vl，模擬口的電壓測量範圍為0-5V 返回的值為0-1024
float vol = Vl*(4.83 / 1024.0);   //我們將Vl的值換算成實際電壓值存入浮點型變量vol
float v = (vol-3.6+0.06)*(100/0.42); //電壓4.02v 100%,電壓3.6v 0%
v2= v2 + v;
```

最後將換算出來的電量顯示到 HMI 螢幕上，HMI 螢幕的部分就算完成。

```
float v2;
void bJPushCallback(void *ptr) {       //如果按下更新按鈕

            for(int i=0;i<1000;i++){            //取平均值
  int Vl = analogRead(A0);          //從A0口讀取電壓數據存入剛剛創建整
  float vol = Vl*(4.83 / 1024.0);  //我們將Vl的值換算成實際電壓值存入
  float v = (vol-3.6+0.06)*(100/0.42); //電壓4.02v 100%,電壓3.6v 0
  v2= v2 + v;
}

v2=v2/1000;
static char vout[6];
dtostrf(v2, 3, 2, vout);       //將float轉換成char
Volt.setText(vout);
```

3. RX5808 接收模組(圖傳接收製作)

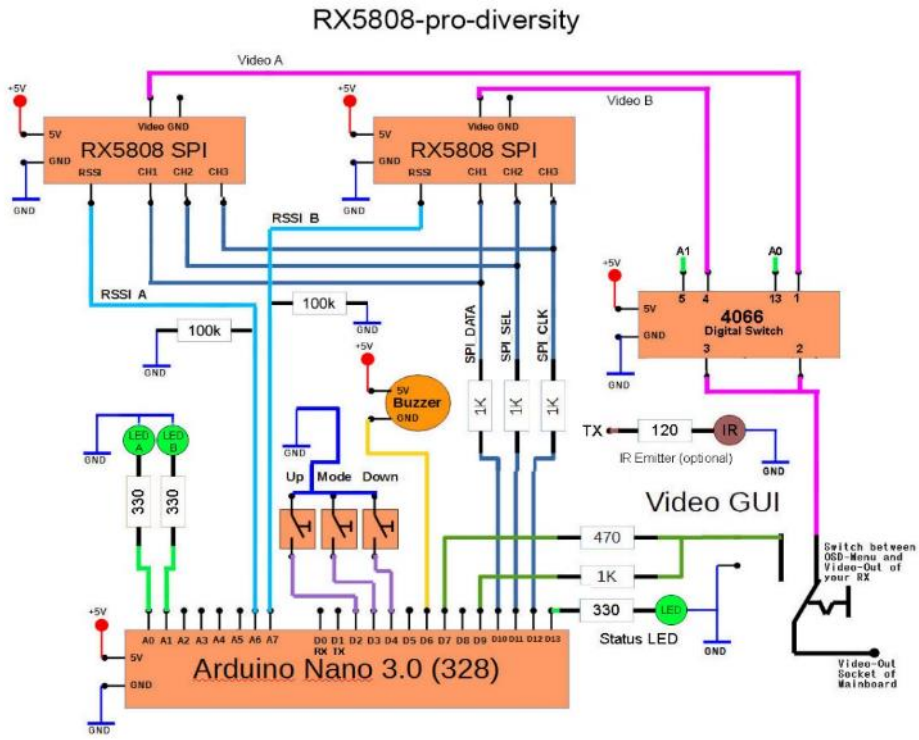


圖 9　RX5808 線路圖

使用 2 個 RX5808 接收模組是為了讓空拍機在飛行中能有最好的 RSSI 值，也就是訊號強度，因為空拍機在空中飛行的範圍很大，使用 2 個 RX5808 接收模組就能有效地增加圖傳的穩定性。

RSSI=Pt+Gr+Gt-Lc-Lbf，其中，Pt 為發射功率，Gr 為接收天線增益，Gt 為發射天線增益，Lc 為電纜和纜頭的損耗，Lbf 為自由空間損耗。

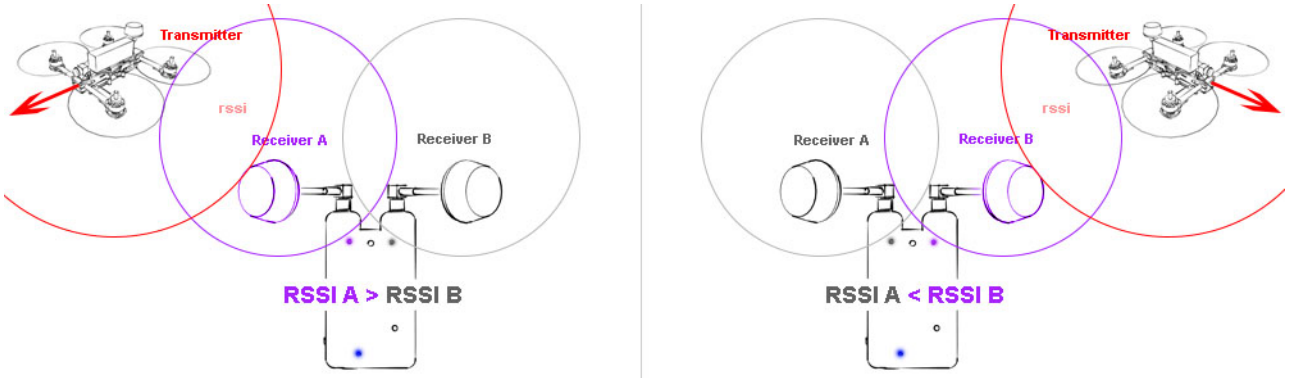


圖 10　RSSI 值

撰寫 RX5808 在 Arduino 上的程式碼。

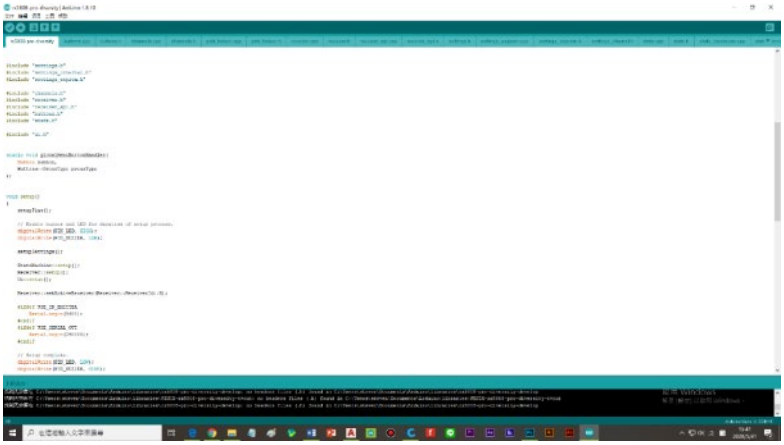


圖 11　RX5808 程式碼

由電路圖可發現線路較複雜，手工銲接線路可能會有接錯與接觸不良的可能，所以製作 RX5808 PRO diversity 圖傳的 PCB 電路板解決這個問題。

圖 12　RX5808 PCB 板

5.8GHz 的天線大致分為兩種，指向性、全向性，指向性傳輸距離遠，但涵蓋範圍較小；全向性傳輸距離較短，但涵蓋範圍較廣。由於我們使用兩個 RX5808 模組，所以兩種天線都能一並裝上，大大增加圖傳的穩定性。

圖 13　指向型天線

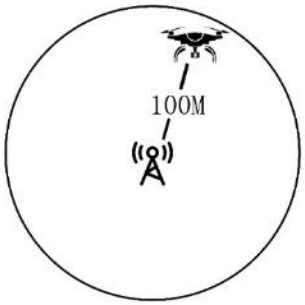


圖 14　全向性天線

4. 外殼製作

使用 AutoCAD 繪製遙控器 3D 模型，並使用 3D 列印機列印外殼。

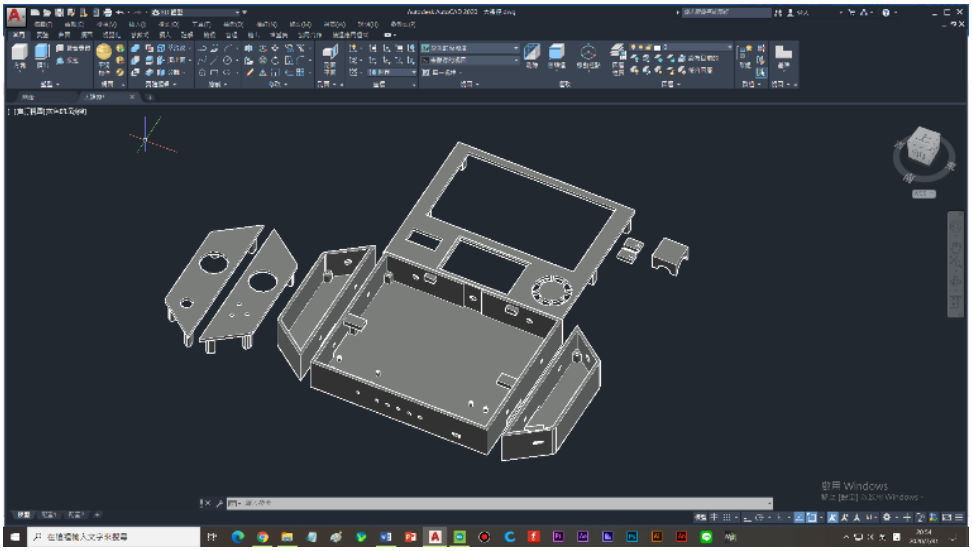


圖 14　繪製 3D 模型

將各零件裝進遙控器內，並配置線路。電源使用 18650 串聯成 12V 來供電給遙控器。

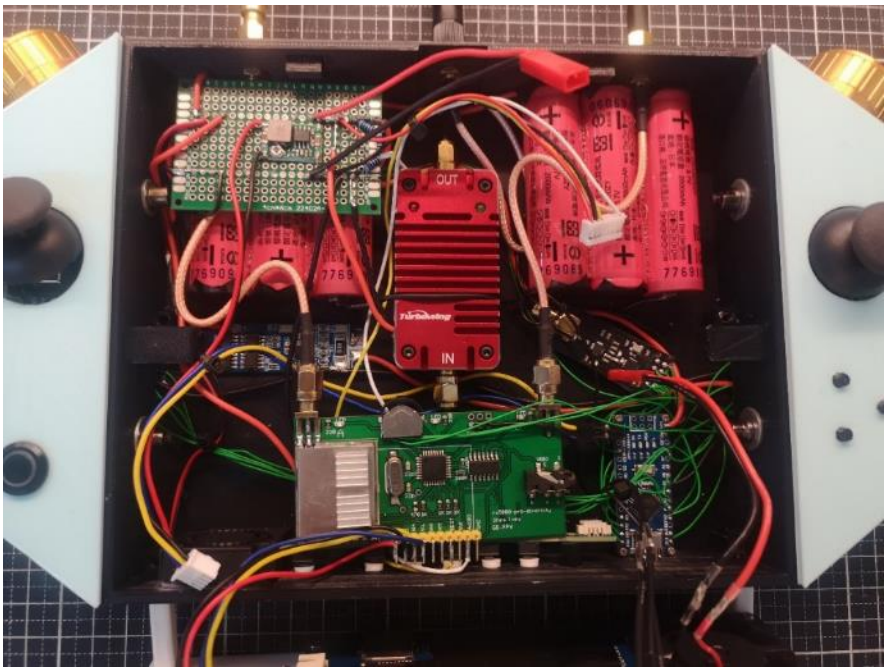


圖 15　配置遙控器線路

遙控器成品，遙控器螢幕由大到小分別為 7 吋 FPV 螢幕、HMI 觸控螢幕、OLED 圖傳控制螢幕。

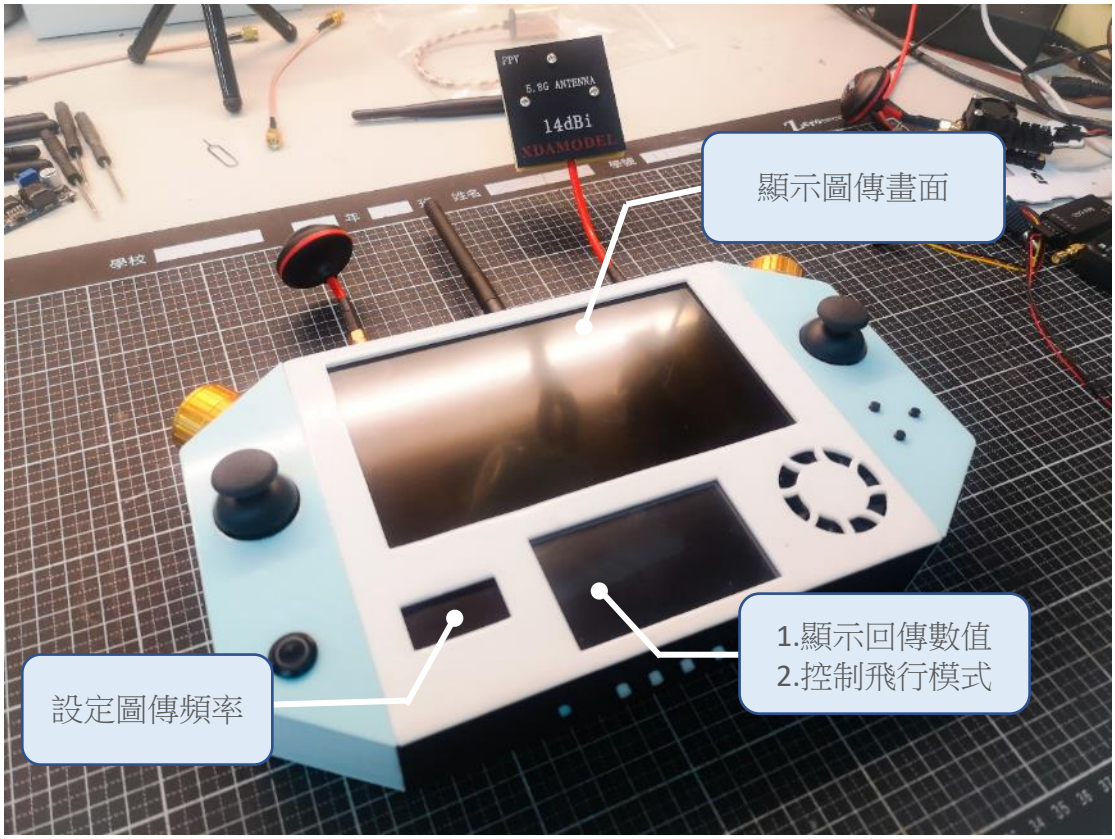


圖 16　遙控器成品

(二)空拍機製作

1. 空拍機接收器

與遙控器發射器的製作方式一樣使用 nRF24L01 模組，不過由於控製空拍機就需要用上 7 隻數位接腳，又因為空拍機上的空間與重量都有限制，所以決定採用 MEGA 2560 Mini Pro 板來製作，它跟 Arduino MEGA 板有一模一樣接腳數量，但體積卻小很多，非常適合用來做空拍機的接收器。

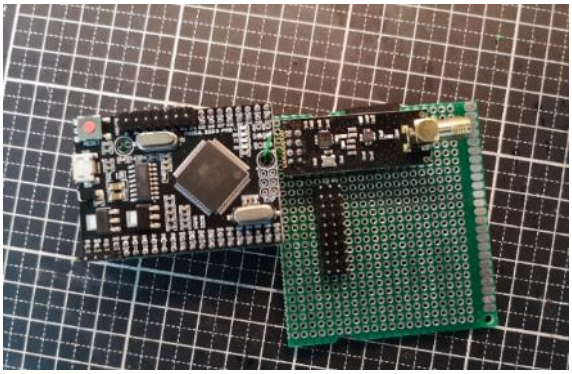


圖 17　MEGA 2560 Mini Pro 板

因為要接收遙控器發射出來的訊號所以將 nRF24L01 模組先設為接收模式。

```
rf24.startListening();  // 開始監聽無線廣播
```

讀取遙控器發射出的 0~255 訊號，並將 0~255 轉換成 1000~2000 的 PWM 訊號來控製空拍機的前進方向。

```
ch_width_1 = map(data.throttle, 0, 255, 1000, 2000);     // pin D2 (PWM signal)
ch_width_2 = map(data.pitch,    0, 255, 1000, 2000);     // pin D3 (PWM signal)
ch_width_3 = map(data.yaw,      0, 255, 1000, 2000);     // pin D4 (PWM signal)
ch_width_4 = map(data.roll,     0, 255, 1000, 2000);     // pin D5 (PWM signal)
ch_width_5 = map(data.aux1,     0, 255, 1000, 2000);     // pin D6 (PWM signal)
ch_width_6 = map(data.aux2,     0, 255, 1000, 2000);     // pin D7 (PWM signal)


ch1.writeMicroseconds(ch_width_3);
ch2.writeMicroseconds(ch_width_4);
ch3.writeMicroseconds(ch_width_2);
ch4.writeMicroseconds(ch_width_1);
ch5.writeMicroseconds(data.hmi);
ch6.writeMicroseconds(ch_width_5);
ch7.writeMicroseconds(ch_width_6);
ch8.writeMicroseconds(data.shu);
```

Pitch：控製空拍機的前後

Roll：控製空拍機的左右

Yaw：控制空拍機的旋轉

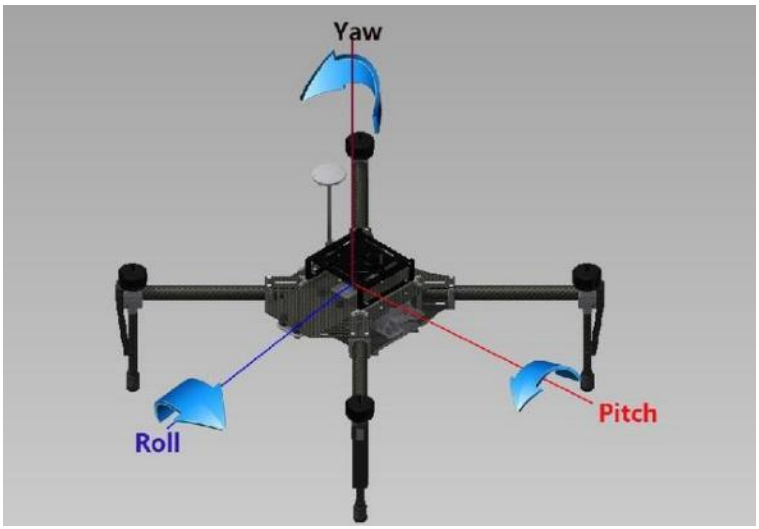


圖 18　空拍機方向

2. 無刷馬達

空拍機所用的電機型號是 x4110s，這是一款外轉式無刷電機，大多的空拍機也都是採用外轉式的無刷電機。

我估算最終完成的無人機重量大約在 3kg 左右，藉由這款電機的力效表可以看出使用 6S(22.2v)電池時搭配 1555 槳在此重量能達到最大的效率，也就是能夠飛得越久。(槳所說的 1555 前面為長度 15 吋，後面為螺距 55)

6S 高效率型

Prop (inch)	Volts (V)	Amps (A)	Thrust(g)	RPM (RPM/Min)	Watts (W)	Efficiency (g/W)
		1	400	2700	22.2	18.01801802
		2	640	3385	44.4	14.41441441
		3	840	3552	66.6	12.61261261
		4	1000	4285	88.8	11.26126126
		5	1130	4577	111	10.18018018
DJI1555	22.2	6	1260	4877	133.2	9.459459459
		7	1390	5125	155.4	8.944658945
		8	1520	5340	177.6	8.558558559
		9	1630	5535	199.8	8.158158158
		10	1740	5745	222	7.837837838
		12	1930	6060	266.4	7.244744745
		13.7	2130	6317	304.14	7.003353719
		1	450	2135	22.2	20.27027027
		2	660	2631	44.4	14.86486486
		3	860	3025	66.6	12.91291291
		4	1020	3342	88.8	11.48648649
		5	1190	3591	111	10.72072072
		6	1330	3791	133.2	9.984984985
1750	22.2	7	1470	3975	155.4	9.459459459
		8	1580	4165	177.6	8.896396396
		9	1680	4295	199.8	8.408408408
		10	1780	4457	222	8.018018018
		12	2000	4688	266.4	7.507507508
		15	2230	4962	333	6.696696697
		19.8	2540	5380	439.56	5.778505779

圖 19　x4110s 無刷馬達

3. 電子變速器

無刷馬達是用變頻的方式去調整它的轉速，在航模中我們把調整速度的元件叫做電子變速器(Electronic Speed Control , ESC)，由它接收 PWM 訊號 1000μs~2000μs，再由它控製無刷馬達的轉速。

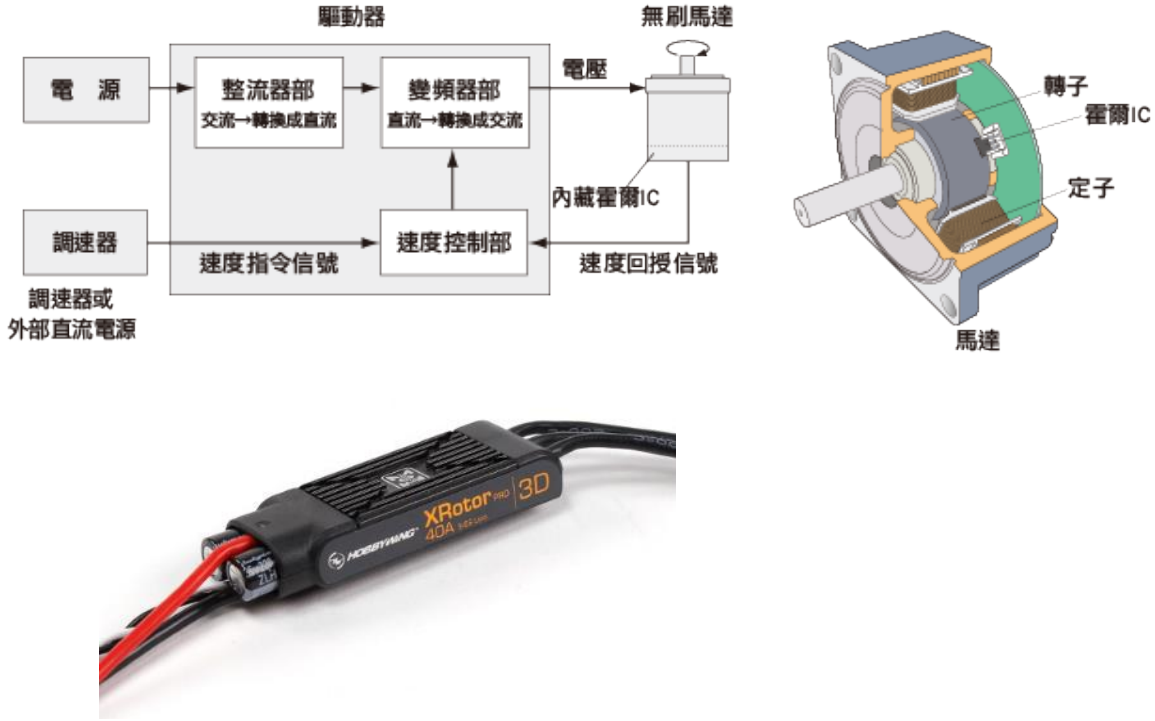


圖 20 電子變速器

4. Pixhawk 飛行控制器

Pixhawk 是一款開源硬體，是最近幾年由蘇黎世理工大學推出來的高性能飛控硬體板。Pixhawk 有內製 MPU6000 三軸加速度計/陀螺儀、L3GD20 16 位陀螺儀、LSM303D 14 位加速度計/磁力計、MS5611 MEAS 氣壓計，藉由這些模組讓空拍機在空中穩定飛行。

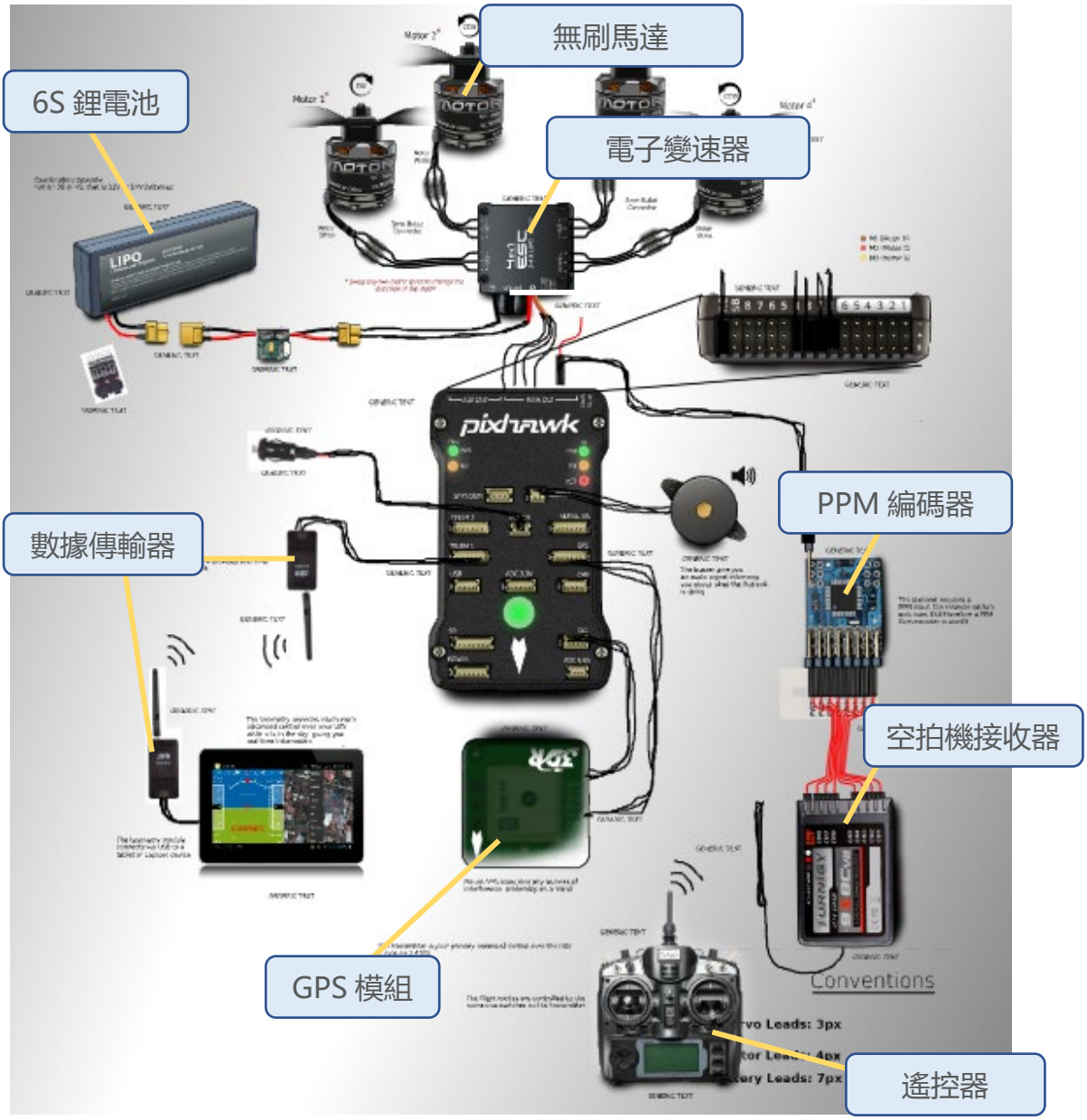


圖 21 Pixhawk 飛行控制器

Pixhawk 可藉由 Mission Planner 這個軟體去進行編程、調整內部參數、PID 等，搭配數傳能在空拍機飛行時顯示空拍機的高度、距離、仰角、速度等各項數據，也能在飛行時去調整空拍機的 PID。

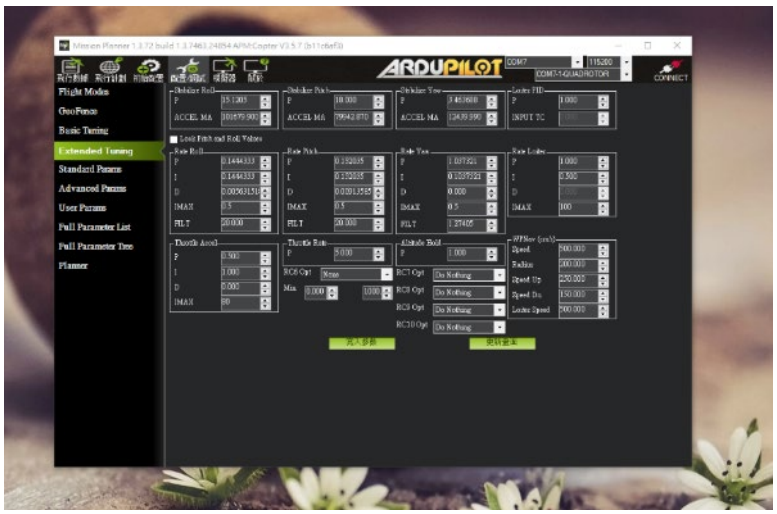


圖 22　PID 調整

(1) 比例(P)控制器：

比例控制考慮當前誤差，誤差值和一個正值的常數 Kp（表示比例）相乘。Kp 是在控制器的輸出和系統的誤差成比例的時候成立。比如說，一個電熱器控制器是在目標溫度和實際溫度差 10°C 時有 100%的輸出，而其目標值是 25°C。那麼它在 15°C 的時候會輸出 100%，在 20°C 的時候會輸出 50%，在 24°C 的時候輸出 10%，注意在誤差是 0 的時候，控制器的輸出也是 0。

因為空拍機在飛行時會有許多外在因素去影響空拍機的飛行，像是空拍機本身的重量不平衡、風與氣流都會造成機身傾斜，此時就需要使用比例控制去修正誤差讓機身穩定。

(2) 積分(I)控制器：

積分控制考慮過去誤差，將誤差值過去一段時間和（誤差和）乘以一個正值的常數 Ki。Ki 從過去的平均誤差值來找到系統的輸出結果和預定值的平均誤差。一個簡單的比例系統會震盪，會在預定值的附近來回變化，因為系統無法消除多餘的糾正。通過加上負的平均誤差值，平均系統誤差值就會漸漸減少。所以，最終這個 PID 迴路系統會在設定值穩定下來。

空拍機在比例控制修正誤差時可能會造成機身來回抖動，這樣會造成相機在拍攝時產生果凍效應，畫面會有波浪狀，此時就必須加大積分控制器的值。

(3) 微分(D)控制器：

微分調節就是偏差值的變化率。能夠實現系統的超前控制。如果輸入偏差值線性變化，則在調節器輸出側疊加一個恆定的調節量。

這可以想成是空拍機的剎車，空拍機在前進時突然停止時因為會有慣性，所以就需要超前去做剎車，這樣空拍機才不會繼續往前。

實際飛行空拍機並不斷調整 PID 直到空拍機能夠穩定飛行

圖 23　實飛空拍機

(4) 設定飛行模式

藉由遙控器上之觸碰螢幕發送一個 PWM 訊號來控制飛行模式，假設發送 1000 空拍機是穩定模式；發送 1300 是定高模式；發送 1500 是懸停模式，在不同環境與用途可以選擇合適的飛行模式。

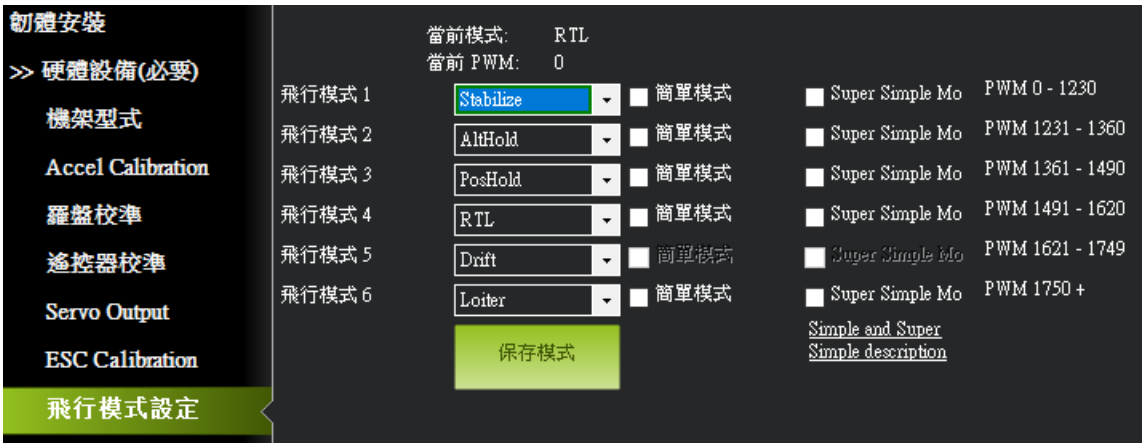


圖 24　飛行模式設定

5. 數據傳輸器

藉由它能將飛行空拍機 Pixhawk 上的各項數據，空拍機位置、時速、電量等數據實時傳輸到手機或電腦上，方便操作者隨時知道空拍機在飛行時的狀況，藉此可以讓空拍機在飛行中由手機直接調整 PID 而不需連接電腦。

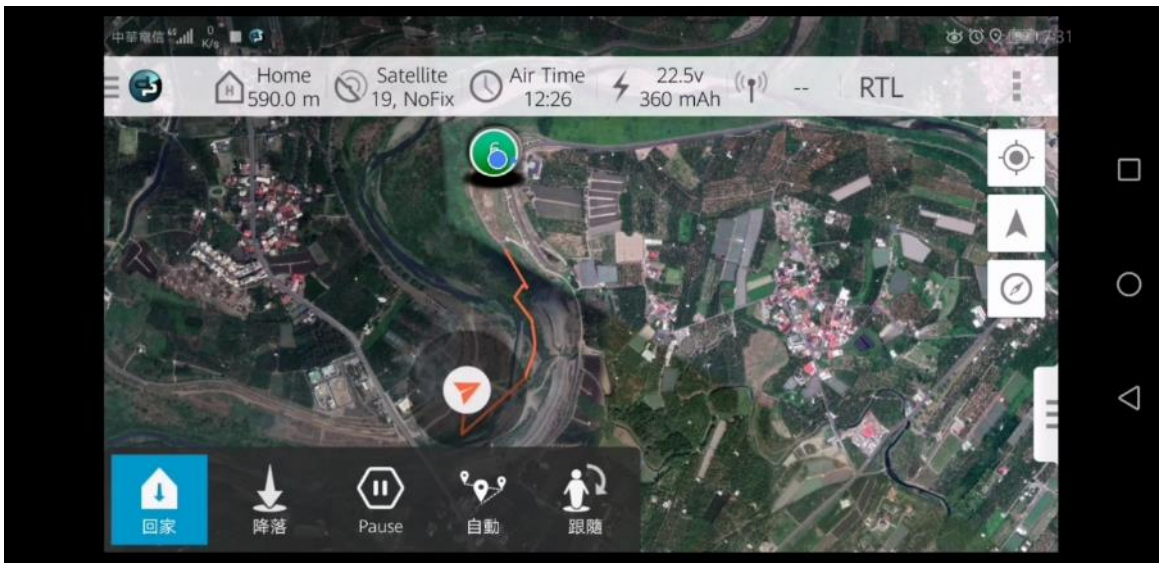


圖 25　數據傳輸器

6. 三軸無刷雲台

空拍機在飛行時，前後左右的操控都會造成機身的傾斜，相機所拍出來的畫面也會前後左右的晃動，為了解決這個問題就必須要裝設雲台來保持相機的平衡，當機身前後左右傾斜時，雲台讓相機保持水平；當相機左右旋轉時，雲台讓相機緩慢跟著轉動來減少畫面晃動。

圖 26　三軸雲台

在空拍時也能藉由 PWM 訊號去控制雲台的上下左右以便觀看四周的景象，我將遙控器兩旁的可變電阻旋鈕設定為控制雲台的方向。

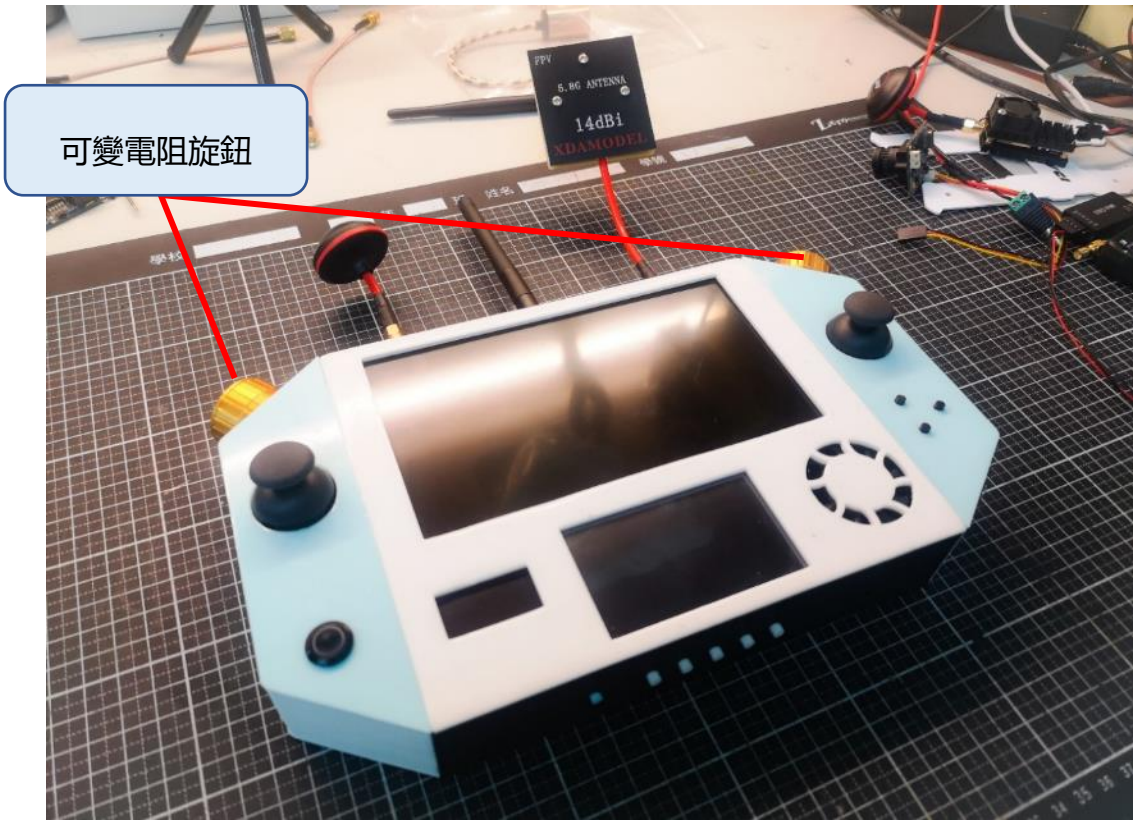


圖 27　可變電組旋鈕

(三) 熱像儀製作

我所使用的熱像儀感測模組為 AMG8833 畫素相較於其他像 MLX90640 或 Dev Kit V2 來的低，但 AMG8833 的優勢就在於價格，AMG8833 的價格跟 MLX90640 差了近 3 倍，跟 Dev Kit V2 更是差了 10 倍，由於我所需要的功能只是用來感測火源，並不需要太高的畫素，所以 AMG8833 8×8 的像素也就夠了，但是使用 AMG8833 迴圈執行的速度必須要快，如果使用一般的 UNO 或 NANO 板了話影像會十分延遲，所以製作熱像儀是使用 ESP32 來當作主板。

	Arduino nano/UNO	ESP 32
時脈(Clock Speed)	16MHz	240MHz

表三　時脈

藉由 AMG8833 讀取一個 8×8 溫度陣列。

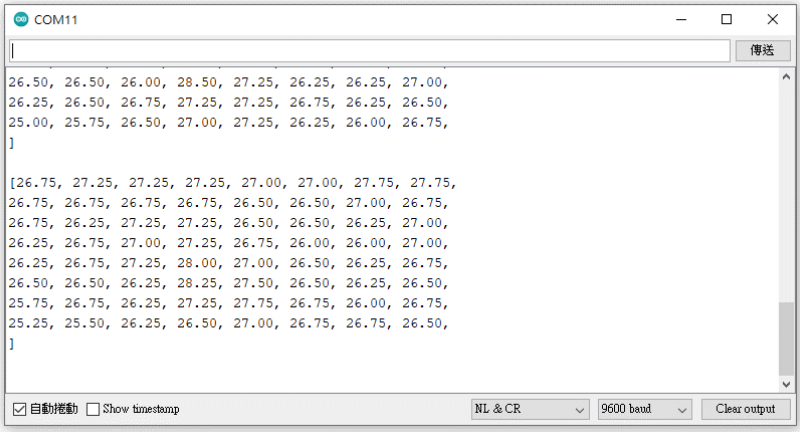


圖 28　AMG8833 陣列

由 nRF24L01 模組將溫度陣列發送出去後並在地面接收陣列最後再用 TFT 螢幕以 8×8 的畫素顯示陣列，由此一來就能在地面上看到空拍機上的熱像儀畫面。

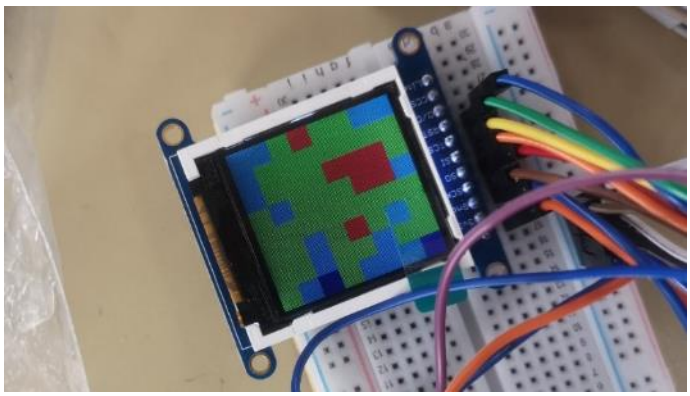


圖 29　AMG8833 影像

熱像儀用來顯示熱像素，但除了熱像素外在火災現場可能還需要測得溫度、二氧化碳濃度、一氧化碳濃度等，所以我打算再使用一組 nRF24L01 來傳輸 LM35 和 MQ 系列感測模組的數據。

作法跟前面都一樣，使用類比接腳讀取 MQ 模組的數值，再由 nRF24L01 發送到地面，最後由 OLED 顯示 MQ 模組感測出來的數值。

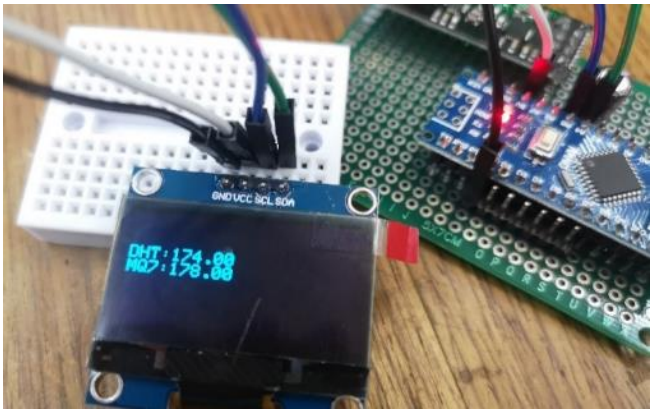


圖 30　OLED 顯示

製作外殼將熱像儀與溫度氣體感測器結合為一體。

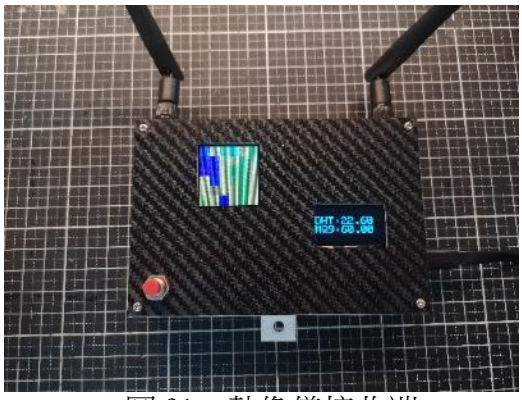


圖 31　熱像儀接收端

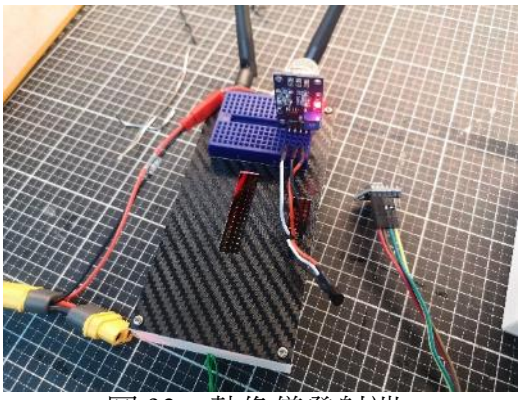


圖 32　熱像儀發射端

(四) 模組化功能

1. 因為要在無人機上懸掛滅火彈來進行滅火，最後我在無人機底部裝置一個掛餌器，一般掛餌器是用來給釣魚人遠距離放餌用的，我這邊將它用來懸掛滅火彈，並使用 PWM 訊號控制伺服機投放。

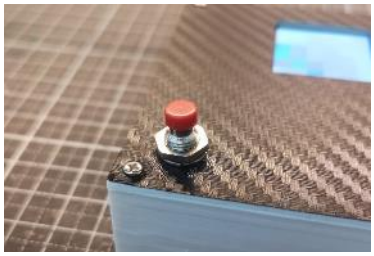


圖 33　投放紐

圖 34　投放裝置

滅火彈就像一顆炸彈一樣，不過它是用來滅火的。將滅火彈丟入火中後滅火彈便會爆炸將內部的乾粉(小蘇打)散布開來達到滅火的效果。

小蘇打在受熱時可生成二氧化碳氣體來滅火，反應式為:$2NaHCO_3 \rightarrow Na2CO_3 + H_2O + CO_2\uparrow$

圖 35 滅火彈

2. 照明燈

在夜晚進行救火時可在無人機上裝設照明燈，使用一個 MOSFET 當作開關再用 Arduino 單晶片控制 LED 燈的開與關。

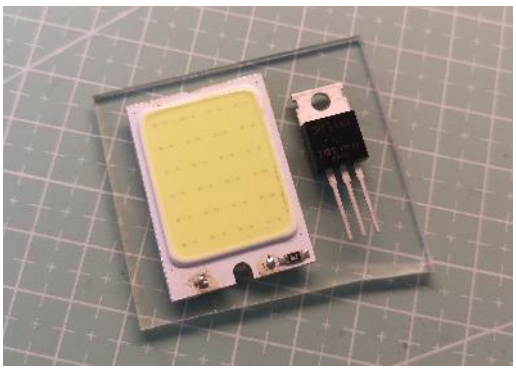


圖 36 照明燈

3.噴水器

除了投擲滅火彈外，我在無人機上裝置了噴水器來就行撲滅 A 類火災，一樣使用 MOSFET 當作開關來控制噴水。

圖 37 噴水器

4.廣播器

我在無人機上裝設了一個喇叭，並用 2.4GHz 模組從地面傳輸聲音到無人機上，如此一來就能從遠處來引導受難者走出火場。

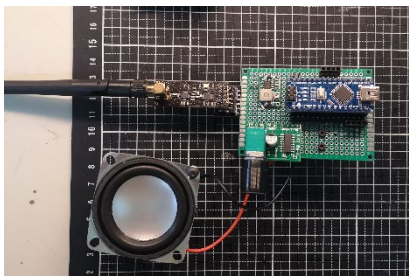


圖 38 廣播器

5.模組化

為了減少不必要的重量，無人機能因應不同的火場更改模組，依據白天或晚上能選擇要不要裝設照明燈，依據 A、B、C、D 類火災可以選擇使用滅火彈或水來滅火，依據人員多寡可以選擇要不要裝設廣播器。

伍、 研究結果

一. 遙控距離

最後測得的遙控距離是 704 米，無人機在 704 米後斷訊，我在無人機斷訊後讓它啟動返航模式它會自動飛回來，所以無須擔心斷訊後無法控制的狀況，704 米的遙控距離在一般的火場以已經足夠了。

空曠處	無放大器	有放大器
距離	344 米	704 米

二. 熱成像

熱成像的畫質由於是使用 AMG8833 較低階的模組，所以只能傳輸 64 像素的畫質，不過這用來判斷熱源也已經足夠了。

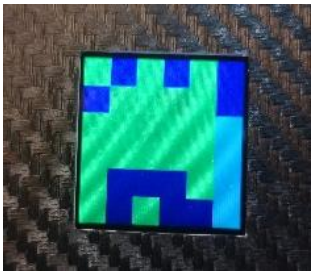


圖 36 無感測到熱源

圖 37 感測到熱源

三. 耗電量

整台無人機的基本重量加上一顆 6000mah 22.2v 的電池是 3 公斤整，再加上熱像儀、二氧化碳等感測模組也大約在 3150g 左右，空拍機在空中懸停時間是 25 分鐘。

四. 載重量

藉由上方的力效表可以看出空拍機一軸最大升力是 2130g，四軸就是 8520g，扣除機身與電池的重量最後升力是 5520g，無人機上滅火彈的重量大約都在 1.1kg 上下，藉由更改投放結構無人機至少可以載重四顆滅火彈。

五. 實際測試

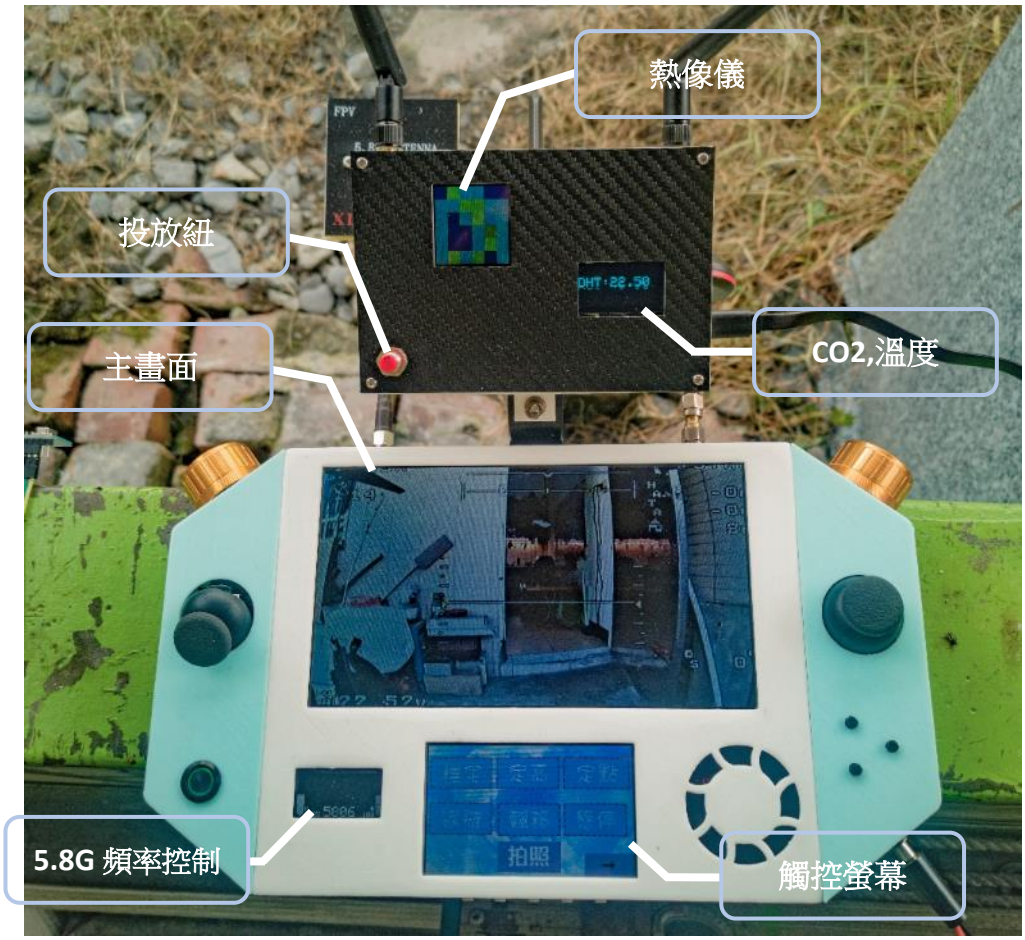


圖 39 遙控器介紹

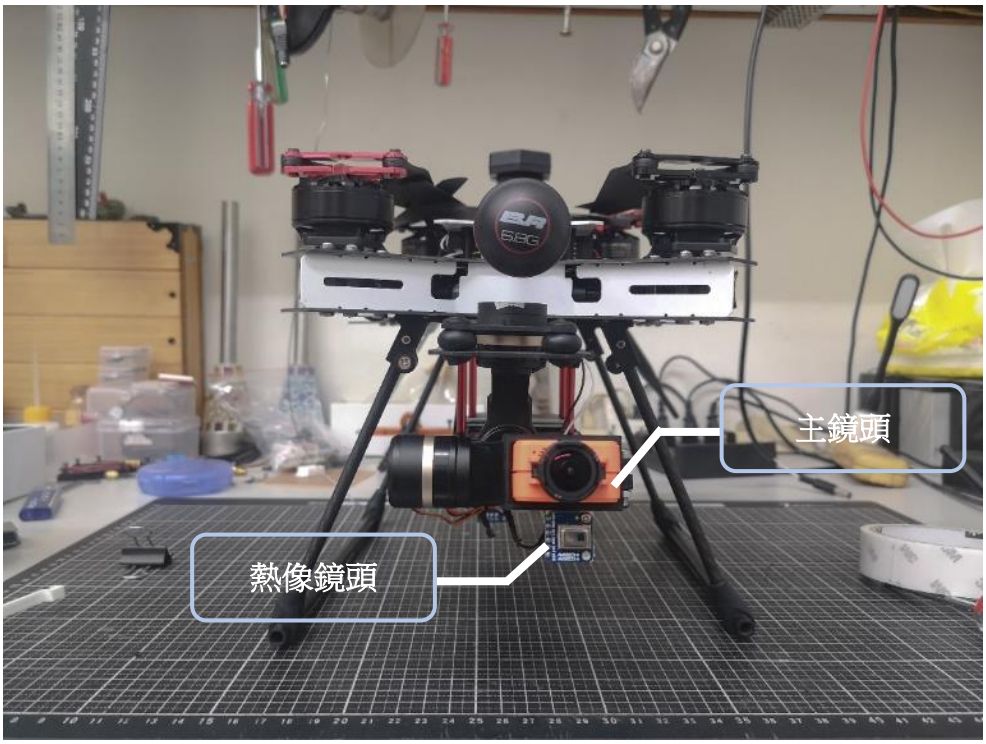


圖 40 無人機正面圖

使用一公升的寶特瓶來代替滅火彈並實際懸掛飛行。

圖 41 實測懸掛

以人體溫度代替火源，不過由於是在室外，熱像儀的效果比在室內的差。

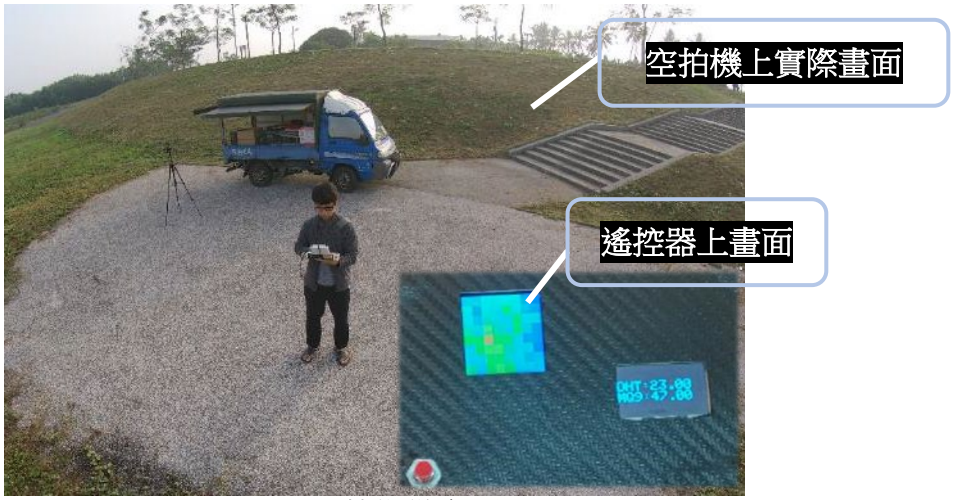


圖 42 熱像儀實測

偵測到火源後按下投放紐，投放寶特瓶(滅火彈)。

圖 43 投放實測

陸、 討論

一. 遙控距離

雖然遙控距離 704 米的距離已經對火災現場夠用了，但我想在未來使用傳輸距離更高的模組繼續增加無人機的遙控距離到 3000 米以上，如此一來就能從距離火災更遠的距離起飛，救災的時間也能再減少。

二. 熱成像

這次使用的模組畫質較低，只能用來判斷熱源，而且在白天的室外效果更差，在後來傳輸熱像素可能就會改使用其它的傳輸模組，熱像儀也使用更高畫素的模組，這樣一來不但火源，人、動物、整個火災現場的架構都能更清晰的顯示出來。

三. 耗電量

25 分鐘對於救援時間我覺得還有些太短了，藉由更改槳葉或換更大效率的無刷馬達或許能讓飛行時間增加到 30 分鐘以上。

柒、 結論

事實上，愈來愈來國家開始注意到無人機救災的發展趨勢，不管是火場、山域或水域，無人機具備極高的便利性及操作性，同時可以擴大廣度與視角，儼然成為救災人員的絕佳幫手。這次研究就是針對無人機結合消防的實際應用與實用性，經

過幾個月的努力終於成功打造一台消防無人機，以下是我從這次研究所得到的經驗與結論。

一、無人機的打造並不難，但無人機必須結合生活應用才能達到最大的效益。

二、傳統的消防設備缺乏機動性，無人機方便的超作性與便利性能加快救災的速度。

三、無人機能在救災前先行勘查地形與進行初步的滅火，最好的結果就是在火災擴大前立刻撲滅火源，遇到太大的火災無法立即將火撲滅也能使用無人機偵查讓消防人員清楚的瞭解災況。

四、本研究使用摺疊機架減小無人機的體積達到輕便好攜帶的效果。

五、無人機空拍影像結合熱成像與搭載滅火彈，經實際測試能準確發現火源與迅速將火源撲滅。

六、無人機能夠因應不同火場而更改感測模組感測不同的數據。

捌、 參考資料及其他

1. How To Make 6 Channel Radio Control For Models. Diy Proportional RC，取自: https://www.rcpano.net/2020/04/09/how-to-make-6-channel-radio-control-for-models-diy-proportional-rc/
2. Nextion Display with Arduino – Getting Started，取自: https://randomnerdtutorials.com/nextion-display-with-arduino-getting-started/
3. Arduino Thermal Camera，取自: https://learn.adafruit.com/adafruit-amg8833-8x8-thermal-camera-sensor/arduino-thermal-camera
4. Pixhawk2.4.8 教程，取自: http://pix.1yuav.com/
5. 空拍機 543 資料整理，取自； https://slidesplayer.com/slide/17127048/

【評語】052313

本作品使用單晶片 Arduino nano、MEGA Mini Pro、ESP-32 和飛行控制器 Pixhawk 打造一台空拍機，並希望結合熱成像與感測模組達到消防救援的功效，研究方法及過程說明詳細，作者經由空拍機的製作，展現機電系統整合的能力，令人讚賞，雖然在初步功能測試方面，有距離不夠遠、熱成像的畫質不夠、及耗電量不足等缺點，日後皆可再改進，作品本身仍是很好的嘗試。

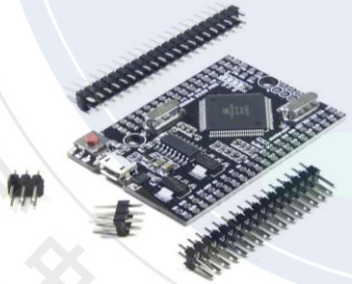
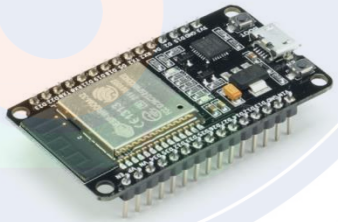
無人機滅火具有明確應用性，尤其在高樓雲梯車伸不到的位置，因此本作品的構想很好，然而旋翼機的設計與製作已成為習知技術，且旋翼機搭載滅火彈也有前例，從科展進行科研的角度，仍希望鼓勵作者在工程之外，著重科研創新。即使為相近系統，仍可探討許多議題，例如：旋翼機比於直昇機或其它定翼機適合帶滅火彈嗎？成本、時間、能耗等的比較為何？各自的優劣為何？針對高樓情境，滅火彈該如何拋射？針對山林火災，又該如何拋射？抑或山林火災由地面拋射即可？以上思考情境提供作者參考，期待本作品能演化出具有更多巧思之二代原型機。

作品簡報

無人機的消防應用

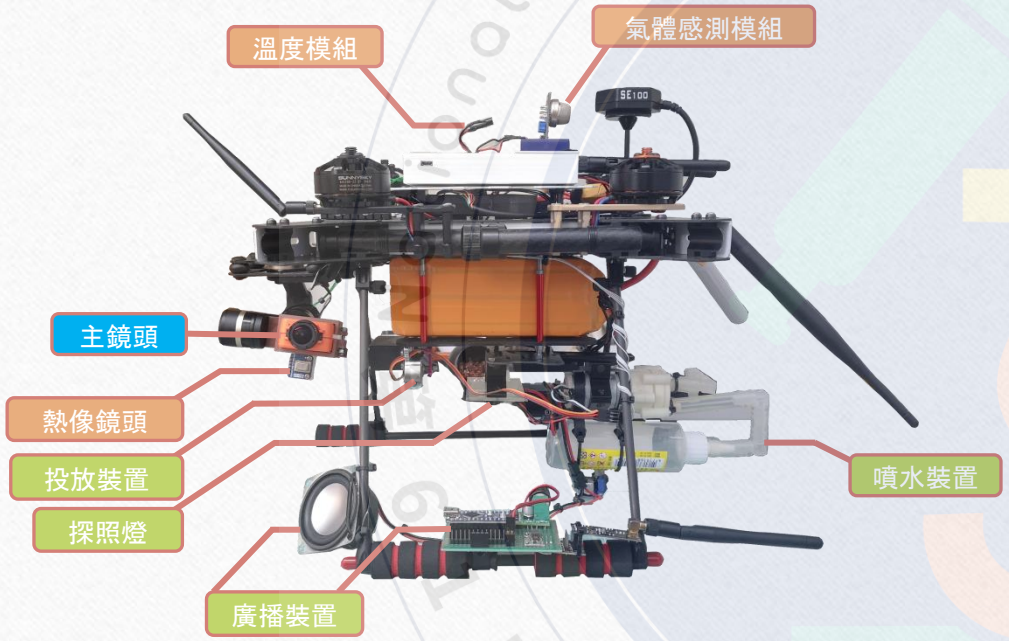
摘要

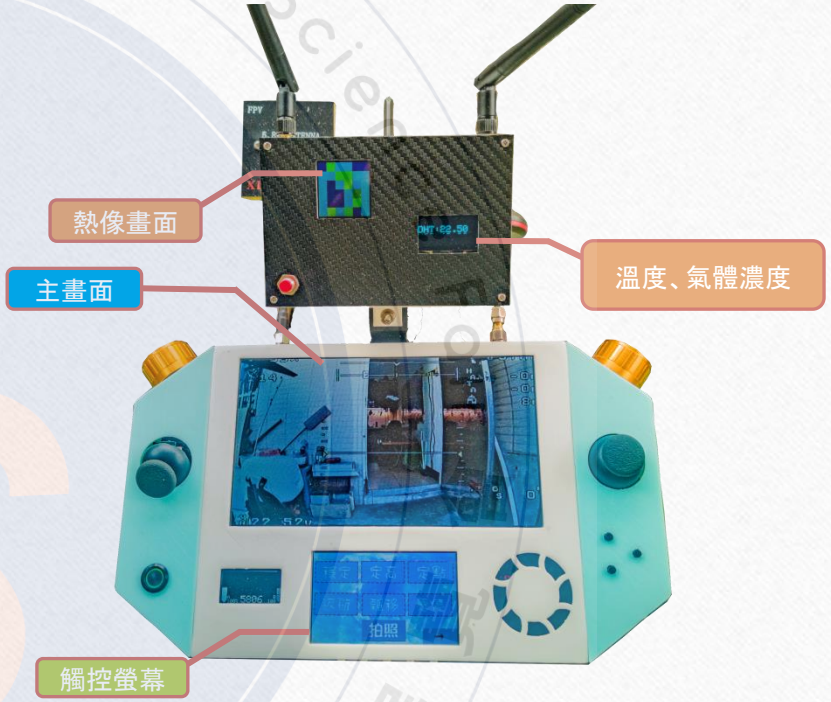
- 本專題使用單晶片Arduino nano、MEGA Mini Pro、ESP-32和飛行控制器Pixhawk為主要控制板打造一台空拍機，並結合熱成像與各感測模組達到消防救援的功效。

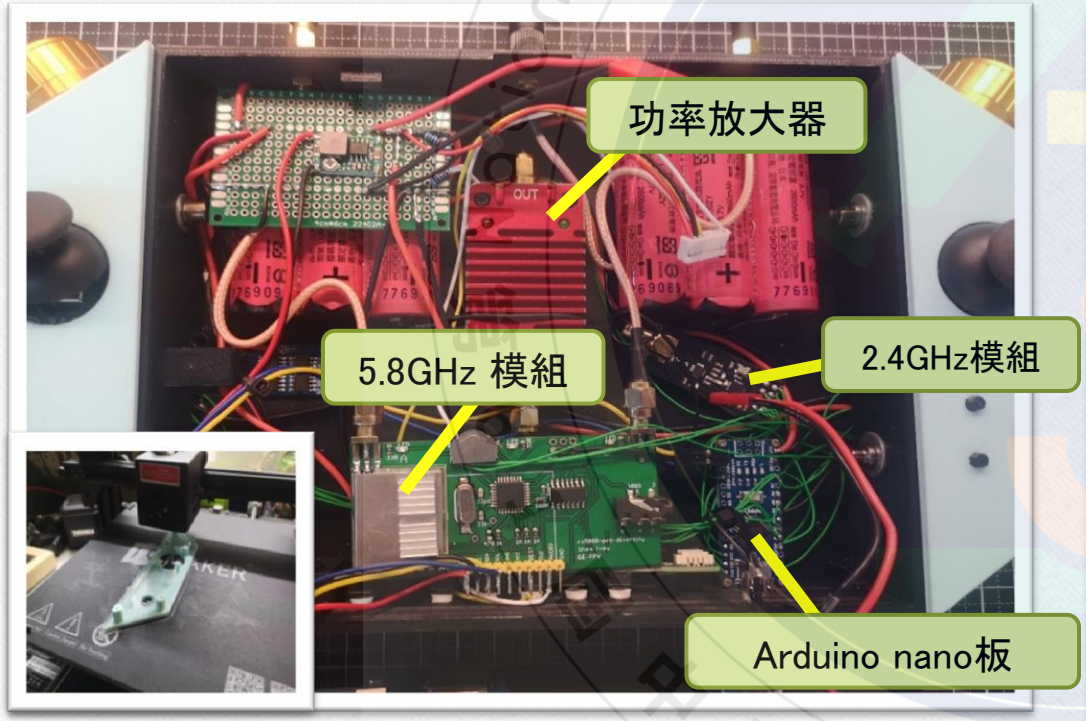

研究目的

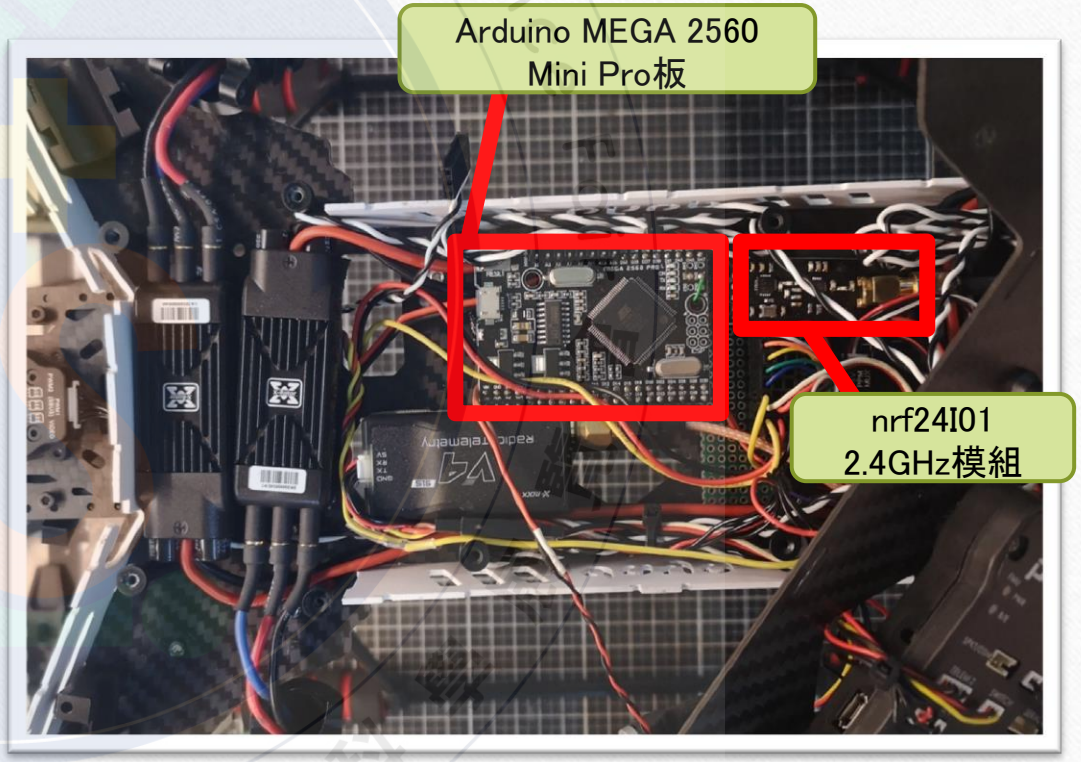
- 在一些交通不便或擁擠的地方發生火災會使得消防車無法迅速到達，消防車強行闖紅燈也容易造成車禍，如果能進化空拍機的影像功能，讓空拍機在空中進行溫度感測如此一來就能從空中快速發現火源與罹難者，並在初期迅速將火源撲滅，也不用擔心交通不便與擁擠的狀況，如此就能減少救援時的時間與危險。


溫度模組
氣體感測模組
主鏡頭
熱像鏡頭
投放裝置
探照燈
廣播裝置
噴水裝置

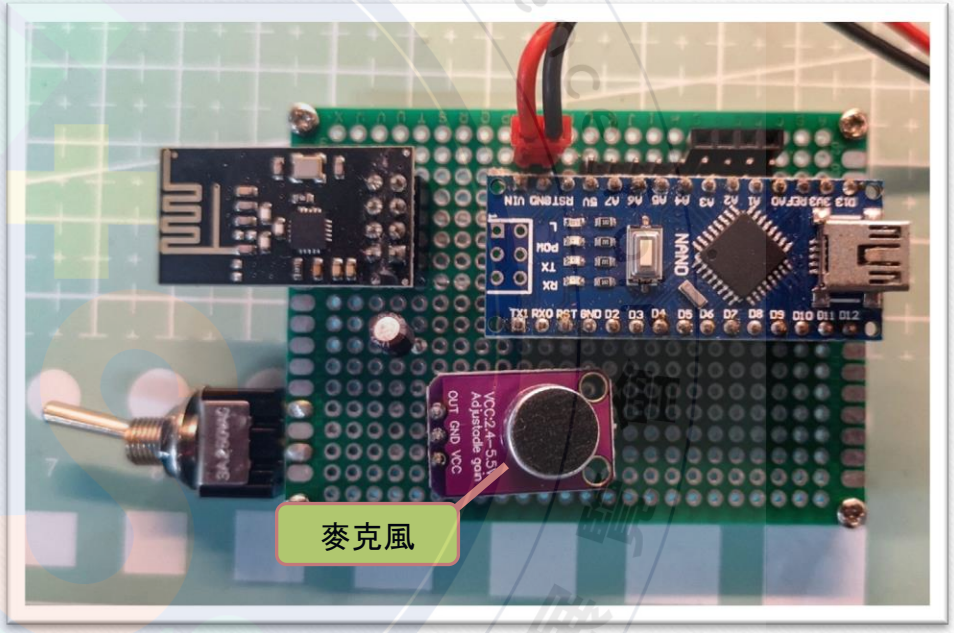

熱像畫面
主畫面
溫度、氣體濃度
觸控螢幕

空曠處	無放大器	有放大器
距離	344米	704米






72.02
LED ON
LED OFF
WAT ON
WAT OFF
投放
自動調參
自動調參
更新
穩定
定高
定點
返航
飄移
懸停
拍照

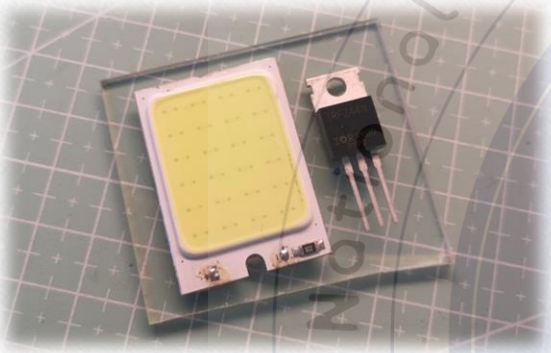

麥克風

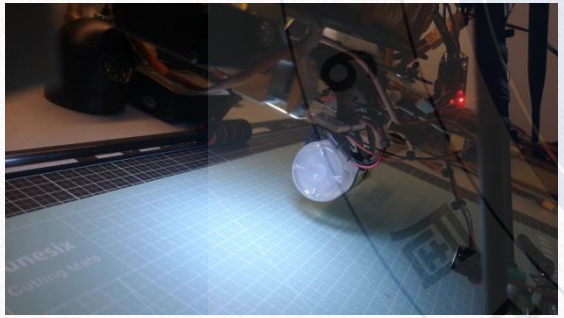
投放裝置

投放器

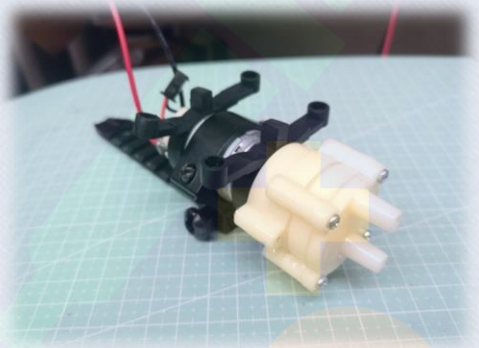
滅火彈就像一顆炸彈一樣，不過它是用來滅火的。將滅火彈丟入火中後滅火彈便會爆炸將內部的乾粉(小蘇打)散布開來達到滅火的效果。

照明燈

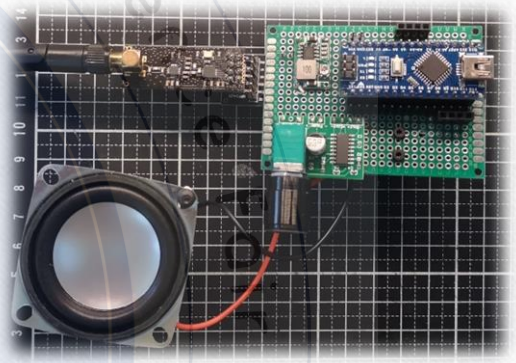




噴水器



廣播器



使用2.4GHz模組從地面傳輸聲音到無人機上，如此一來就能從遠處來引導受難者走出火場。

數據傳輸模組

藉由它能將空拍機**位置、時速、電量**等數據實時傳輸到手機或電腦上。

還有一個功能就是能直接用**手機規劃飛行路線**，讓無人機**自動飛行到目的地**。

技術規格

重量	3150g(Min)/3550g(Max)
軸距	650mm
螺旋槳型號	dji1555
電池	6000mah 22.2V
飛行時間	25分鐘
抗風性	5~6級風
速度	70km/h(Max)
載重	4公斤
遙控距離	704米

無人機測試

結論

- 目前的無人機是以消防做為基礎所打造的，但在製作的過程中也出現無人機更多的可能，噴水系統能用在農業噴藥，影像也能用在農作物的勘查；廣播器則能用在大型活動現場的交通指揮；投放裝置也能用在水難投放救生衣等，所以在未來我想以多功能的無人機為目標進行改造升級。