

逢 甲 大 學  
航 太 與 系 統 工 程 學 系  
碩 士 論 文

多軸旋翼機之導航系統及  
地面監控站之實現

Realization of the Navigation System and  
Ground Control Station for a Multi-axis  
Rotorcraft

指 導 教 授：陳啟川

研 究 生：吳錦炘

中 華 民 國 一 百 零 三 年 七 月

逢甲大學  
航太與系統工程學系碩士學位論文

多軸旋翼機之導航系統及  
地面監控站之實現

Realization of the Navigation System and Ground  
Control Station for a Multi-axis Rotorcraft

研究型論文

實務型論文

研究生：吳錦炘

經碩士學位考試合格特此證明

評審委員

賴勇誌

黃振文

陳啟川

指導教授

陳啟川

主任

黃振文

考試日期：中華民國 103 年 7 月 24 日

## 誌謝

本篇論文能夠順利的完成，必須感謝指導教授 陳啟川博士在大學以及研究所生涯中的指導，訓練我在生活上以及待人處事上的能力，感謝 賴盈誌博士在學生論文遇到瓶頸及困難時，給予極大的協助，幫助學生能順利克服難關，感謝 黃振鴻博士給了許多課程上及生活上的建議，並以人生經驗給了我許多啟發。

感謝學長姊柏龍、怡榮、景懋、姍昀在畢業後仍協助和關心我們的進展，給予我解決問題的建議及方向；感謝同學睿恆、哲瑜、政宏、守豪、子喬，在課業上互相勉勵；感謝學弟妹潘瑜、至兼、羽哲、易聰，在生活及論文上的協助。

感謝我的父母和家人們，因為你們的支持及鼓勵，讓我能專心於課業和研究工作，順利完成碩士學業，最後感謝曾經協助過我的各位高中朋友智浩、偉豪、展耀、廷州，陪伴我紓解壓力及給予我勉勵，有了眾多人的協助，我才得以順利完成學業，在此僅以此篇論文獻給所有關心我愛護我的人。

吳錦炘 謹誌於

逢甲大學工學院

中華民國103年8月

## 摘要

近年來無人飛行載具 UAV (Unmanned Aerial Vehicle) 的使用普及，並且已從遙控飛行發展到自主飛行，應用領域也涵蓋軍事及民生。由於多軸飛行器的高穩定性，使得其在空拍及勘查領域越來越受到重視。

論文主要以 Arduino 做為發展自主導航之控制核心。藉由 GPS 接收機所接收之訊號並使用 Arduino 對多軸飛行器之位置做判斷後，以「向量導航法」來實現多軸飛行器之導航，再依達到目標位置所需之條件對飛行控制板進行 PWM 訊號控制，而多軸飛行器穩定部分則交由原本之飛行控制板處理。另外，再結合以 LabVIEW 做為系統設計程式以開發地面站之人機介面。其主要功能為即時追蹤載具之位置，並使用飛行儀表來顯示多軸飛行器目前的飛行狀態，且藉由此介面可簡易的輸入並傳送新指定之目標導航點。接著利用無線傳輸模組來進行多軸飛行器與地面監控站之間的數據連接，藉此利於監控人員能直接判讀多軸飛行器的姿態以及是否有正確地執行任務。

關鍵字：旋翼機、向量導航

## Abstract

In recent years, UAV (Unmanned Aerial Vehicle) is widely used and has been transited from the remote control style to autonomous flight. UAV was also applied to the military and the civilian usages; especially, the Multi-axis UAV which has drawn more attention in the field of aerial photography and exploration due to its high stability.

This paper is mainly to put to use Arduino as the core for developing the augmented autonomous navigation and control system. After manipulating and calculating the data with the signals from GPS receiver and the waypoints or the target location, the Arduino precedes the navigation by means of sending the PWM commands to the on-board flight control system of the Multi-axis UAV; in the meantime, the stability of this vehicle is still controlled by its original on-board system. Besides, a ground control system (GCS) which serves as monitoring interface and control device has also been developed with LabVIEW. The mission related information, including the new waypoints and the health status such as position, velocity, and attitude of the UAV, can also be uploaded and downloaded by above GCS through the wireless transceivers.

Keyword: Multi-axis UAV, Autonomous navigation ,Arduino, LabVIEW

# 目錄

誌謝.....	i
摘要.....	ii
Abstract.....	iii
目錄.....	iv
圖目錄.....	vii
表目錄.....	x
第一章.....	1
1.1 前言.....	1
1.2 文獻回顧.....	2
1.3 各章概述.....	3
第二章.....	5
2.1 機載電腦.....	5
2.1.1 Arduino.....	5
2.1.2 AQ50D 飛行控制板.....	7
2.2 硬體架構.....	8
2.2.1 YON DON 多軸機架.....	8
2.2.2 馬達.....	10
2.2.3 螺旋槳.....	10

2.2.4 電子變速器 .....	11
2.2.5 3DR Radio Telemetry kit-433Mhz 無線傳輸模組 .....	12
2.2.6 GPS 接收器 .....	14
2.2.7 電力系統 .....	15
2.2.8 遙控器及遙控接收器 .....	17
第三章 .....	19
3.1 導航方法 .....	19
3.1.1 向量導航法 .....	20
3.1.2 轉向條件 .....	23
3.1.3 轉向控制方式 .....	26
3.1.4 到達目標點判定 .....	27
3.2 導航程序 .....	29
3.2.1 導航流程 .....	29
3.2.2 安全設置 .....	31
3.3 地面控制站設計 .....	32
3.3.1 地面監控站人機介面設計 .....	33
3.3.2 ActiveX 外掛程式 .....	35
第四章 .....	37
4.1 系統測試 .....	37

4.1.1 導控硬體整合測試 .....	37
4.1.2 地面控制站測試 .....	45
4.2 地面測試 .....	46
4.2.1 手動遙控模式測試 .....	47
4.2.2 自動導航模式測試： .....	52
4.2.3 測試結果 .....	56
第五章.....	57
參考文獻.....	59
附錄 A.....	61





## 圖目錄

圖 2.1 Arduino Mega2560 控制板 .....	6
圖 2.2 AQ50D 飛行控制板 .....	8
圖 2.3 YONDON 多軸機架實體圖 .....	9
圖 2.4 A2208 無刷馬達 .....	10
圖 2.5 雙槳實體圖 .....	11
圖 2.6 B4515 電子變速器實體圖 .....	11
圖 2.7 無線傳輸模組實體圖 .....	12
圖 2.8 無線傳輸模組軟體介面 .....	13
圖 2.9 Arduino NEO-6M GPS 接收器 .....	14
圖 2.10 GPS 測試軟體 .....	15
圖 2.11 Desire Power 電池實體圖 .....	16
圖 2.12 遙控及接收器圖 .....	18
圖 2.13 遙控接收器圖 .....	18
圖 3.1 航行與目標航向之向量夾角示意圖 .....	20
圖 3.2 參考點假設未更新示意圖 .....	23
圖 3.3 取得即時更新位置點流程圖 .....	24
圖 3.4 前往目標點及修正流程圖 .....	25

圖 3.5 導航程式流程圖 .....	30
圖 3.6 安全設置流程圖 .....	31
圖 3.7 地面控制站運作流程 .....	33
圖 3.8 地面監控站人機介面圖 .....	34
圖 3.9 飛行儀表範例 .....	35
圖 4.1 硬體整合圖 .....	38
圖 4.2 系統整合流程圖 .....	38
圖 4.3 Arduino 擷取 GPS 訊號圖(GPGGA).....	39
圖 4.4 Arduino 擷取 GPS 訊號圖(GPRMC).....	39
圖 4.5 俯仰角於 LabVIEW 讀數比對圖(一).....	40
圖 4.6 俯仰角於 LabVIEW 讀數比對圖(二).....	41
圖 4.7 滾轉角於 LabVIEW 讀數比對圖(一).....	41
圖 4.8 滾轉角於 LabVIEW 讀數比對圖(二).....	42
圖 4.9 Arduino 讀取接收機訊號圖 .....	42
圖 4.10 Arduino 輸出訊號圖 .....	43
圖 4.11 導航切換控制器測試(一).....	43
圖 4.12 導航切換控制器測試(二).....	44
圖 4.13 無線傳輸模組傳送導航點測試(一) .....	44
圖 4.14 無線傳輸模組傳送導航點測試(二) .....	45

圖 4.15 地面站圖控介面測試 .....	46
圖 4.16 各位置置中 .....	48
圖 4.17 遙控器於俯角位置 .....	48
圖 4.18 遙控器仰角位置 .....	49
圖 4.19 遙控器於右偏航位置 .....	49
圖 4.20 遙控器於左偏航位置 .....	50
圖 4.21 遙控器於油門最大 .....	50
圖 4.22 遙控器於油門最小位置 .....	51
圖 4.23 遙控器於右滾轉位置 .....	51
圖 4.24 遙控器於左滾轉位置 .....	52
圖 4.25 導航運作流程圖 .....	53
圖 4.26 多軸爬升數據圖 .....	54
圖 4.27 多軸直航數據圖 .....	54
圖 4.28 多軸轉向數據圖 .....	55
圖 4.29 多軸往目標點直航數據圖 .....	55
圖 4.30 多軸停懸數據圖 .....	56

## 表目錄

表 2.1 Arduino 規格表 .....	7
表 2.2 多軸機架規格 .....	9
表 2.3 馬達規格表 .....	10
表 2.4 電子變速器規格表 .....	12
表 2.5 電池規格表 .....	16
表 2.6 導控系統的用電情況表 .....	16
表 3.1 儀錶板樣式 .....	36
表 4.1 各硬體重量概算表 .....	47
表 A.1 無線傳輸模組規格表 .....	61

# 第一章

## 緒論

### 1.1 前言

多軸旋翼機，也是無人飛行載具的其中一種。無人飛行載具 (Unmanned Aerial Vehicle)，簡稱為 UAV，它的控制方式大致上可以歸納為兩種，一種是由人在地面上以目視進行遠端的遙控，而另外一種方法則是由導控電腦作自主性的導航控制。

在國內，無人飛行載具之發展已有相當程度的基礎，但是相較於國外廣泛運用於各領域，我們卻像是剛起步而已。無人飛行載具又細分為許多種類，依據任務目的不同，有不同的外型及特點，如本文所選用的多軸旋翼機，其最大特點便是穩定、可停懸，主要用於範圍不大的定點空拍，其他也有像大範圍搜尋所使用的定翼機等等。雖然國

外已經有公司販售套裝的軟硬體設備，例如飛行載具、飛控電腦及地面工作站……等等，雖然目前市面上各產品功能尚稱完善，但若採用直接購買相關的系統，也就等於被這些公司所侷限住，即無法進行更詳細的參數微調或維修。於是如何自己發展一套完整的軟硬體設備對於多軸旋翼機的自主導航來說是相當重要的。

本文裝載在多軸旋翼機上的是發跡於義大利的一個開放原始碼單晶片微電腦—Arduino，主要是用來處理多軸旋翼機導航所需的相關資料，例如 GPS 接收器、飛行姿態……等等，向量導航的運算及對飛行姿態進行控制，判斷無人飛行載具目前位置和目標位置的距離及方向，再進一步利用 Arduino 透過 PWM 腳位告知飛行控制板所應該對應的動作，進而改變各馬達的轉速以達到轉向的目的，往目標點航行，到達目標點後保持穩定滯空姿態再等待地面控制站送出下一組目標點後，接著重複向量導航的運算。

## 1.2 文獻回顧

本節將針對參考過之論文作一些簡單的概述。

吾人是由邱永賢 學長的「應用 LabVIEW PDA 模組於導航系統之建構」【1】有了初步自主導航設計之概念，利用 GPS 接收器得到目前飛行載具的位置，並擷取高度計、加速度計等資料以及撰寫的程式藉由 Mini SSCII 伺服控制器控制飛行載具之操縱面以修正飛行載

具航向及姿態，直至到達目標點為止。

進而再參考張進欽 學長的「應用 LabVIEW 虛擬儀控於機載控制系統之建構」【2】主要是利用 ARK-3381 為機載電腦，整合其他感測硬體次系統為導航系統的基本架構，雖未實機測試，但亦提供了於實機飛行時使其更加完善之處理方案。

還有翁子傑 學長的「嵌入式系統於無人飛行載具自主導航之實現」【3】，他亦是利用 ARK-3381 為機載電腦，整合各感測儀器次系統，如 GPS 接收器、高度計及電子羅盤，再經由導航程式做判斷轉彎與否及巡行狀態的確認，並整合地面監控站，由地面監控站發送目標點位置，達到即時導航並監控之目的。

由於須要監控載具的姿態以及對多軸旋翼機下達導航目標點位置，所以地面控制站的建構是參考劉祐瑜 學長的「無人飛行載具地面監控站平台之建構」【4】，利用 GMS 公司所出的飛行儀表板軟體及地圖軟體，使地面控制站的人機介面變得更美觀而且更方便監測飛行載具的姿態。

### 1.3 各章概述

本文旨在讓機載電腦透過程式設計與其他感測儀器結合後，可以達到無人自主導航的目標，而本節將簡述本論文各章節著重之重點，作一些簡單的摘要，希望讓讀者能夠更了解本文的主旨所在。論文內

容簡述如下：

### 第一章 緒論：

介紹多軸無人飛行載具之背景與發展，以及本文所參考的文獻回顧簡介。

### 第二章 系統架構：

多軸旋翼機的軟硬體設備及系統架構的概略介紹。

### 第三章 自主導航系統設計：

介紹導航方法與流程，以及應用 GPS 訊號資料和其它數據，再經由機載電腦(Arduino)作運算進而配合飛行控制板控制無刷馬達，達成無人自主導航的目標。

### 第四章 系統整合與自主導航測試：

本章介紹多軸旋翼機的系統與次系統(Arduino)的整合，且在地面進行模擬測試與實際測試，對整體系統測試之結果加以討論與分析。

### 第五章 結論與未來展望：


此章節將對本文實際測試的結果及需改進的部分做最後的討論，以及未來的期許。



## 第二章

### 系統架構

#### 2.1 機載電腦



無人飛行載具裝載之導航系統的主要核心就是 Arduino，它負責接收所有使用者遙控器的資訊以及計算是否應該轉彎等判斷，而功能強大的 Arduino 板子正好適合我們的需求。且目前市面上關於 Arduino 的資源廣大，取得容易且價格相較於其他控制器低廉，故我們選用 Arduino 作為我們主要的機載電腦控制核心。

##### 2.1.1 Arduino

本文採用的 Arduino 機板是由義大利生產的開放式電路板，視為一個開放原始碼的單晶片微電腦，使用了 Atmel AVR 單片機，並採

用了基於開放原始碼的軟硬體平台，構建於開放原始碼 simple I/O 介面，並且具有使用類似 Java，C 語言的開發環境。Arduino 也可以獨立運作成為一個可以跟軟體溝通的介面。優點是可以簡單地與感測器，各式各樣的電子元件連接，並支援多樣的互動程式，且使用低價格的微處理器。

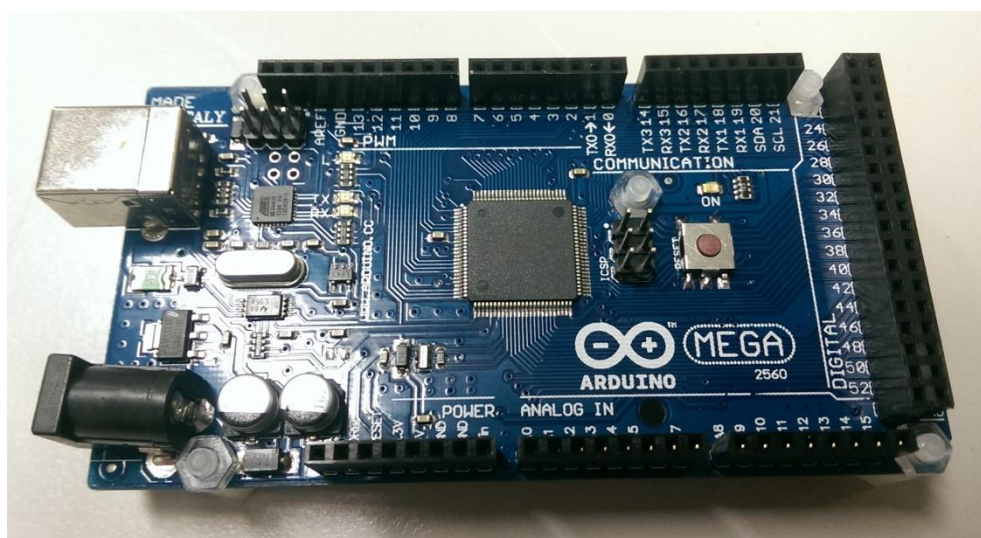


圖 2.1 Arduino Mega2560 控制板

Arduino 使用具有類似 Java，C 語言的良好 Processing/Wiring 開發環境，本文主要使用 C 語言作為其輔助開發語言。亦可與 Macromedia Flash、Processing、Max/MSP、Pure Data 和 SuperCollider 等軟體，結合電子元件，例如開關或感測器或其他控制器件、LED、步進馬達或其他輸出裝置，作出互動作品。Arduino 也可以獨立運作成為一個可以跟軟體溝通的介面。【5】

表 2.1 Arduino 規格表

Arduino 規格	
Microcontroller	ATmega2560
Flash Memory	256 KB (ATmega2560) of which 8 KB used by bootloader
Digital I/O	54(of which 15 provide PWM output)
Analog Input	16
輸入電壓	6V~20V

### 2.1.2 AQ50D 飛行控制板

AQ50D 為一台灣公司 TMF 出產之飛行控制板，專門對於多軸旋翼機做輔助控制，擁有自我穩定、返航、定 GPS 點飛行等多項輔助功能。多軸旋翼機是透過遙控器進行遠距離操縱的飛行載具，且利用多組馬達直接提供其飛行所需之升力與飛行控制。控制板上有陀螺儀(感測轉動速度)和加速度計(感測加速度或重力)，這些感測器可以幫助操縱員穩定載具系統。亦可利用遙控器上的兩隻搖桿來操控多軸旋翼機，達到俯仰、滾轉、航向、油門等動作，因此至少需要四通道的遙控器與接收機。



圖 2.2 AQ500 飛行控制板

## 2.2 硬體架構

### 2.2.1 YON DON 多軸機架

市面上多軸旋翼機之款式不勝枚舉，本文選擇了 YON DON 的多軸機架，其最大特色為通用三軸、四軸(十字型或 X 型)、六軸飛行載具。在此，我們選擇了將此組機架組裝為一六軸飛行器，由於其孔位亦通用市面上多數飛行控制板，我們不必再另外鑽孔，相當方便。此機架的中間圓盤採用碳纖維，機臂的部分則是使用鋁合金中空方型管，擁有較輕的重量及優秀的機體剛性。

表 2.2 多軸機架規格

YONNOD 多軸機架規格	
鋁條	6 條
碳纖圓板	2 片
腳架	1 組
組裝後高	21cm
組裝後軸距	59cm
中心盤直徑	15cm



圖 2.3YONDON 多軸機架實體圖

## 2.2.2 馬達

「新西達 A2208-800KV」為外轉無刷馬達，係利用電子變速器訊號控制，進行能量轉換。運用電生磁原理產生旋轉動力。無刷馬達之特性為高效率、壽命長，無磨損消耗。

表 2.3 馬達規格表

外轉子無刷電動馬達	電池(個數)		KV	最大效率 率電流	無負載電流	最大電流	最大效率	軸徑 (mm)	重量	電阻	尺寸 (mm)
	Ni-MH	Li-Poly									
A2208	8~10	2~3	KV 800	4~7A (>74%)	0.4 A	8 A/60 s	80%	3.17 mm	36 g	225 mΩ	27.5x26



圖 2.4A2208 無刷馬達

## 2.2.3 螺旋槳

此螺旋槳為 TMF 0938、0938R(反槳)塑膠材質，直徑九吋、Pitch 為 38；具有翼翹設計，減少翼尖渦流之高效率螺旋槳。擁有極佳的

動態平衡，能有效降低震動。此螺旋槳能快速反應馬達的扭力與轉速，配合本論文採用的馬達相當合適。



圖 2.5 雙槳實體圖

#### 2.2.4 電子變速器



圖 2.6 B4515 電子變速器實體圖

電子變速器，又稱為遙控用無刷馬達速度控制器。此款電子變速器 NOVATECH B4515 25A 有三大特點。第一：低電壓保護及馬達相

位進角等繁瑣的多項設定已自動偵測設定，完全智慧自動設定，大量簡化以往舊款電變繁瑣的設定流程。第二：內建三種飛機、直升機、滑翔機三種模式，可透括遙控器簡單切換。第三：適用多款無碳刷馬達，且馬達從靜止加速到全速耗時最短，加速曲線表現優秀。

表 2.4 電子變速器規格表【6】

型號	連續/瞬間電流	電池數/伺服機數	重量	尺寸 (mm)
B4515	25/35(A)	2/4,3/2~3,4/2	22 g	38x24 x6

### 2.2.5 3DR Radio Telemetry kit-433Mhz 無線傳輸模組

本文所使用的無線傳輸模組為 3DR Radio Telemetry kit-433Mhz，其主要功能為將飛行控制板 AQ50D 上的感測器訊號經由無線傳輸模組傳送到地面站以得知目前多軸旋翼機的姿態資訊及 GPS 經緯度資料；



圖 2.7 無線傳輸模組實體圖



此外，從地面控制站下達改變目標導航點的指令，亦是經由此無線傳輸模組傳送給 Arduino，以達到監控及改變導航位置的目的。實體圖如圖 2.7 所示，其詳細規格表請參閱附錄 A。

無線傳輸模組的設定可以經由它附的軟體去改變韌體設定，連接後可以直接顯示出目前的設定，其軟體介面如圖 2.8。

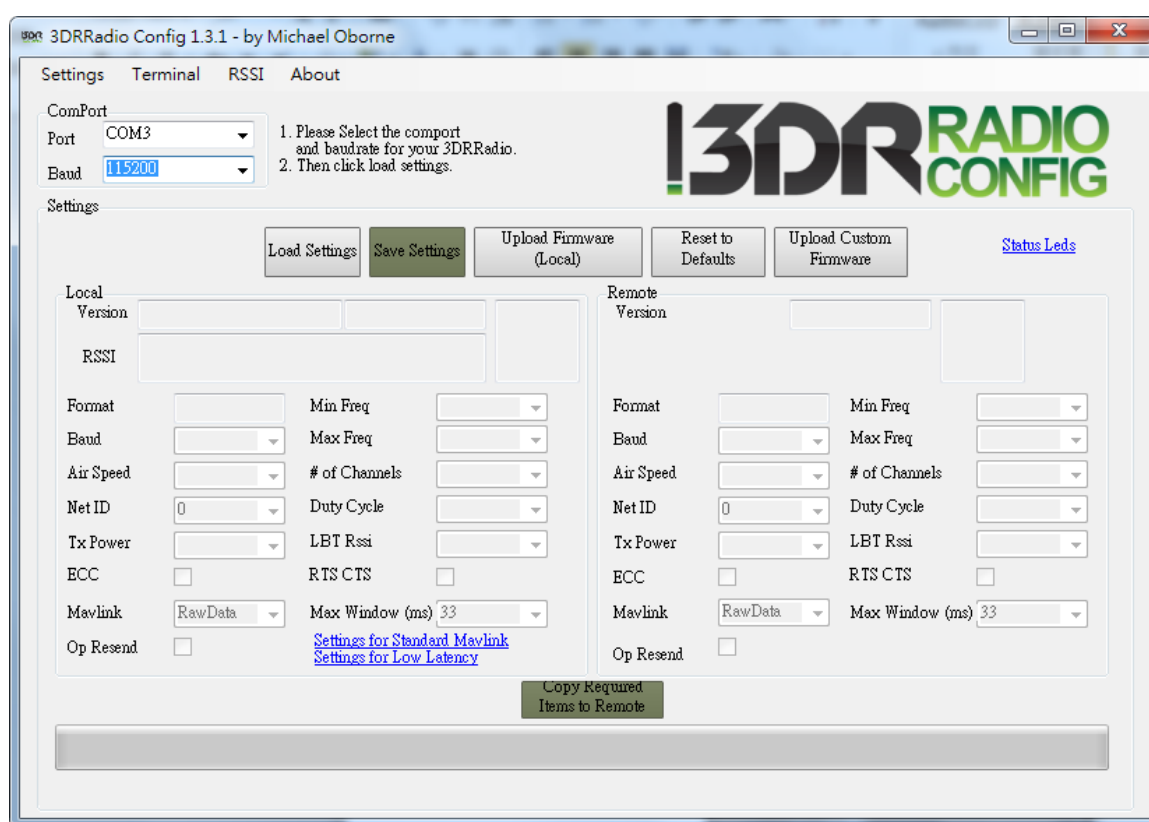


圖 2.8 無線傳輸模組軟體介面

為了配合 AQ50D 的 Baud Rate，我們將此設定為 115200，且地面站的 Baud Rate 亦統一調整為 115200，以方便各系統間的訊號傳輸無誤。

## 2.2.6 GPS 接收器

航行過程中，得知多軸機目前位置是極其重要的資訊之一，可藉由 GPS 接收器其中的\$GPRMC 格式來取得目前的經緯度，或者是採用\$GPGGA 格式來取得高度訊號，而 GPS 接收器的選擇可以依照不同任務的需求去選擇符合的型號。

本文所使用的 GPS 接收器是由 u-blox 公司所製造的 NEO-6 系列模組，其優點是重量輕而且體積小，且精度在本文可接受範圍內，具有低功耗和低成本的特點。

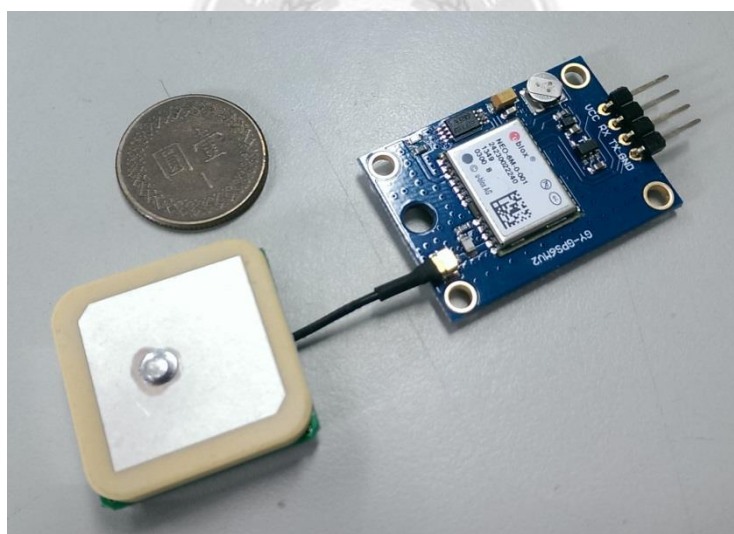


圖 2.9 Arduino NEO-6M GPS 接收器

圖 2.10 為此 GPS 的設定及調校軟體圖，此為進入程式後所顯示之前設定好的參數資訊。

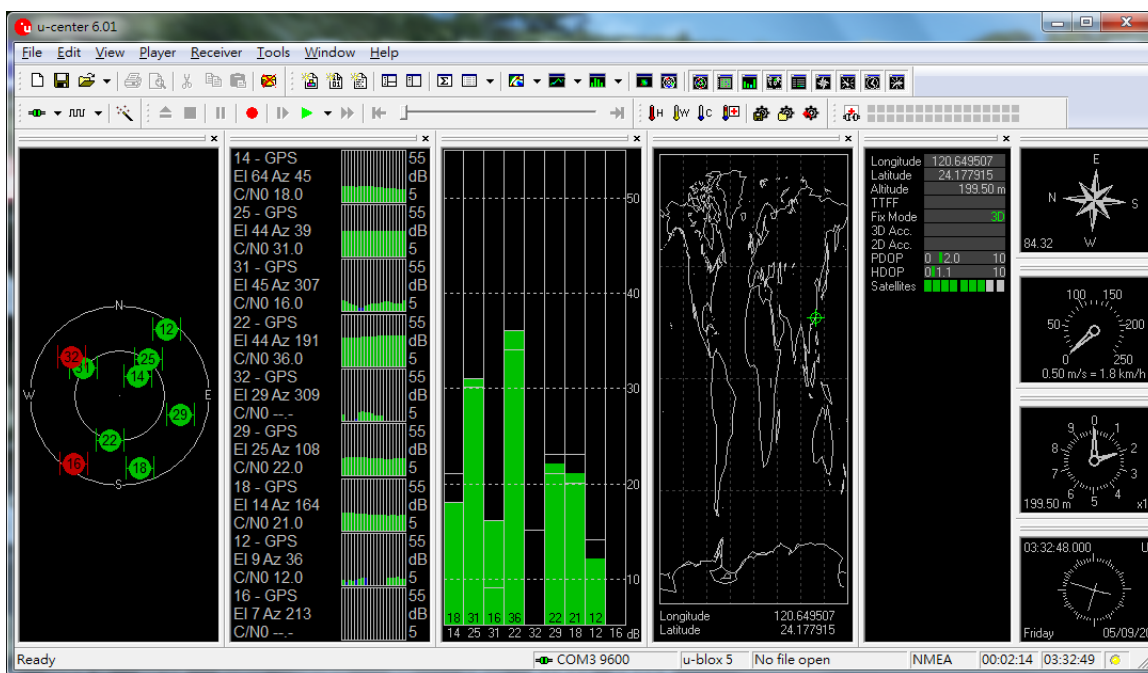


圖 2.10 GPS 測試軟體【7】

由軟體中可以查看目前 GPS 運作狀態，衛星收訊狀況及分布情形與目前定位點等多樣資訊。

### 2.2.7 電力系統

本文選選用 Desire Power 公司的遙控模型用鋰電池供給各個硬體次系統使用，採用最新奈米技術(Nanotech)製造，可承受高達 5C 的充電電流可大幅縮短充電時間，且具有高放電均值電壓(High MPV)，高循環壽命等特色，它提供 11.1 V 的電壓和 3800 mAh 的電流，讓多軸載具有足夠的電源供給。如下圖 2.11 所示。



圖 2.11 Desire Power 電池實體圖

表 2.5 電池規格表

長 x 寬 x 高	138 x 46 x 25(mm)
重量	307g
最大連續電流	152A(40C)
最大瞬時電流	304A(80C)
最大充電電流	19A(5C)

表 2.6 導控系統的用電情況表

設備	工作電壓	平均耗電流	數量	平均總耗電流	功率
GPS 接收器	5(V)	80(mA)	1	80(mA)	0.4(W)
無線傳輸模組	5(V)	25(mA)	1	25(mA)	0.125(W)
Arduino	5(V)	480(mA)	1	480(mA)	2.4 (W)
遙控器接收機	5(V)	40(mA)	1	40(mA)	0.02(W)
TMF AQ50D	5(V)	480(mA)	1	480(mA)	2.4(W)
馬達+電子變速器	11(V)	1800(mA)	6	10800(mA)	118.8(W)

經由表 2.6 可得知導控系統所消耗的功率約為 124W，而吾人所選用的 DesirePower 高電壓電池可提供 42.18W 持續一小時的供電，因此若 124W 之功率，系統約可執行 20 分鐘的時間，因考慮各馬達做較大變化之動會提高電量之消耗，故預估飛行時間約為 10 分鐘。

## 2.2.8 遙控器及遙控接收器【8】

遙控器由地面操控人員所手動操作，接收機則是放置於機身中間部分，兩條天線呈 90 度垂直朝下伸出，以利接收避免死角。本文之六軸載具，需先使用遙控器起飛至理想高度後，才能開始啟用自動導航，並將控制權交與機載電腦(Arduino)。在導航任務完成後，亦須依賴此遙控器取回控制權，做接下來的操控或是降落的動作。也就是說，除了正在執行導航任務時是交由機載電腦(Arduino)操控，其餘的部分還是由手動遙控完成。

本儀器由 Futaba 公司出品，遙控器型號為 T9CHP、接收機型號為 R617FS。遙控器為一組有 9 動訊號發射器的遙控模組，是 FASST 2.4Ghz 的系統，接收器則是為 7 動的接收器，與發射器使用相同的 FASST 2.4Ghz 系統。



圖 2.12 遙控及接收器圖

下圖 2.13 遙控接收器的部分，CH1~CH6 分別為副翼(滾轉)、升降舵(前後)、油門(上下)、方向舵(偏航)、導航模式切換、自動返航。而 CH7 則接 5V 電源。

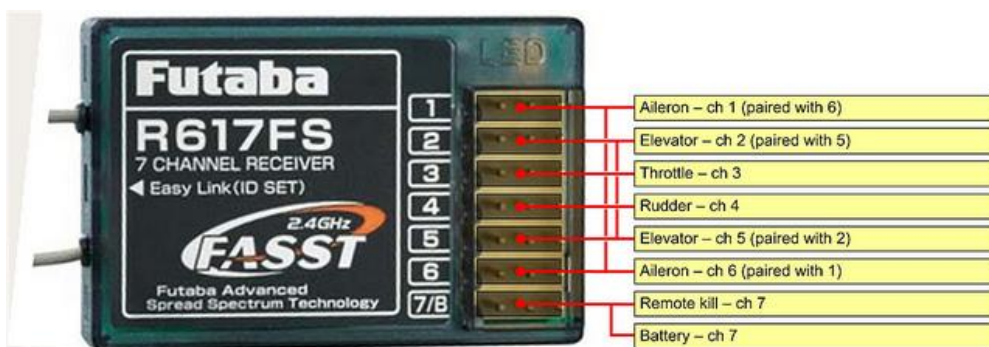


圖 2.13 遙控接收器圖

## 第三章

# 自主導航系統設計

### 3.1 導航方法

目前已有許多的導航方法，例如有：慣性導航系統，這種方式使用價格昂貴的高級設備(至少包含處理器、加速度計、陀螺儀、其他運動感測模組)去計算出最有效率之路徑，其優點是給定了初始條件後，會自動更新當前位置及速度，缺點是小誤差會隨時間累積成大誤差。但如果考慮到經費及缺點的話就較適合其他類型的導航方法，而本文主要利用 GPS 全球定位系統來獲取多軸旋翼機的位置資訊，再進一步計算出到達目標點的飛行路徑及所需要之飛行參數，以達成多軸旋翼機之自主導航的目的。

### 3.1.1 向量導航法【3】

所謂的向量導航法，是利用已知的三個點，分別是參考位置點、目前位置點以及目標位置點，然後以目前位置作為基準點與參考位置點及目標位置點所畫出之兩條直線的夾角，以此夾角的角度大小為依據從而判斷轉向與否。

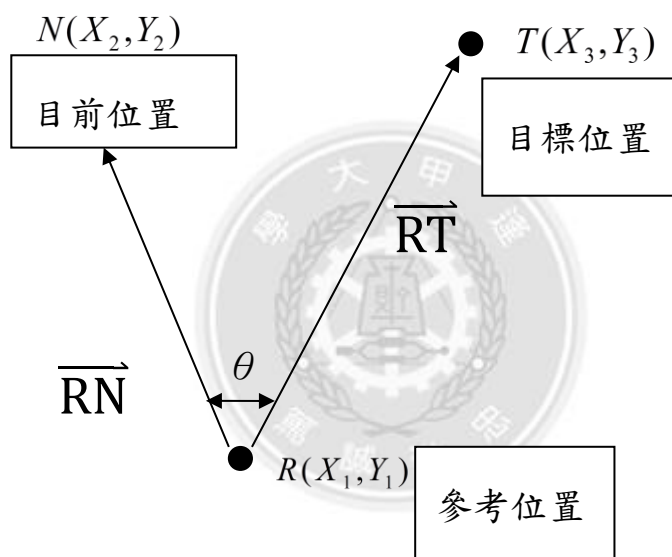


圖 3.1 航行與目標航向之向量夾角示意圖

而各點位置座標來源皆是由 GPS 接收器所提供的經、緯度資訊。目前市面上的 GPS 的通訊協定幾乎都以 NMEA-0183 為主，NMEA 的封包裡有包含許多種格式，而本文主要使用 NMEA 封包當中以【\$GPRMC】和【\$GPGGA】開頭的兩種格式，而以【\$GPRMC】為開頭的格式，優點是可以經由其中一項定位狀態資料來判斷此筆資



料可用與否，A 代表資料可用與 V 代表資料不可用，藉此來判定此筆資料的可用性；而【\$GPGGA】為開頭的格式，優點則是接收到的資料封包中，有 GPS 接收器目前高度的資訊，我們利用此高度訊號來跟高度計所取得高度訊號來做比對。

我們首先擷取 GPS 的訊號至 Arduino 中，然後 Arduino 程式擷取到封包中以【\$GPRMC】開頭的資料，接著就可以從此封包中擷取得到經、緯度的座標點，起飛前所取得的經、緯度  $X_0$  及  $Y_0$ ，將直接由 TMF 飛行控制板定義為多軸旋翼機起始的位置座標點  $I(X_0, Y_0)$ ，也就是將地面控制站的所在位置設定為地面起始位置，要結束任務時還可以進一步的設定為返航點。

多軸旋翼機在自主導航時所擷取到的 GPS 訊號，就是在空中飛行時當下位置點的經、緯度座標資訊，此經、緯度可當成是空間中的一個座標  $N(X_2, Y_2)$ ，因為 GPS 訊號是每秒更新，我們將三秒前 GPS 的經、緯度值當作是參考點座標  $R(X_1, Y_1)$ ，而利用  $R(X_1, Y_1) - N(X_2, Y_2)$  可構成一個向量  $\overrightarrow{RN}$ ，而目標點  $T(X_3, Y_3)$  與參考點  $R(X_1, Y_1)$  相減則可得到另一個向量  $\overrightarrow{RT}$ ，這兩組向量值間有一個夾角  $\theta$ ，本文藉此夾角  $\theta$  的大小來判斷多軸旋翼機的航向是否必須轉向。圖 3-1 為航行向量與目標航向向量夾角示意圖。

其中， $\theta$  的角度大小，是利用反餘弦定理求得。

反餘弦定理：

$$\theta = \cos^{-1} \frac{\overline{RN} \cdot \overline{RT}}{|\overline{RN}| |\overline{RT}|} \quad (3.1.1)$$

但是經由計算反餘弦所求得的數值，只能得到角度值，並無法判斷目標位置是相對於目前位置的左邊或是右邊，所以此時必須再藉由向量外積的結果來判別。本文目前所計算之向量並沒有使用到高度的資訊，即 Z 軸的資訊，故把其當作二維座標使用向量外積，求出的值就有正負號，由此正負值將可以判斷目標點在目前點的左邊或是右邊。

外積公式：

$$\left| \overline{RT} \times \overline{RN} \right| = \det \begin{vmatrix} (X_T - X_R) & (Y_T - Y_R) \\ (X_N - X_R) & (Y_N - Y_R) \end{vmatrix} \quad (3.1.2)$$

假設算出之外積值為正值，本文將此定義為：此時目標的位置在多軸旋翼機航向 $\overline{RN}$ 的右側；反之若外積值為負值，也就是表示目標位置在多軸旋翼機航向 $\overline{RN}$ 的左側。

接著再藉由將 GPS 截取到的經緯度資訊輸入到機載電腦

(Arduino)，經由事先燒錄於 Arduino 控制板上的計算程式，判斷其應作的轉向，便可以達到無人飛行載具自主導航的目標。

### 3.1.2 轉向條件

本文的 Arduino 程式之參考點位置是以三秒前的目前位置做為參考點位置，並以持續更新的方式來給定，其目的在使多軸旋翼機能依照目前計算出來數值去判斷應旋轉多少角度以及這角度是左轉或右轉。利用此種不斷更新參考點的方式來避免參考點假設過於遠，所造成的轉向判斷特例。如圖 3.2 所示，圖 3.2 所表示的就是目前點與目標點的位置是相差甚遠，但參考點卻沒有更新，導致判斷特例而無法到達目標點。

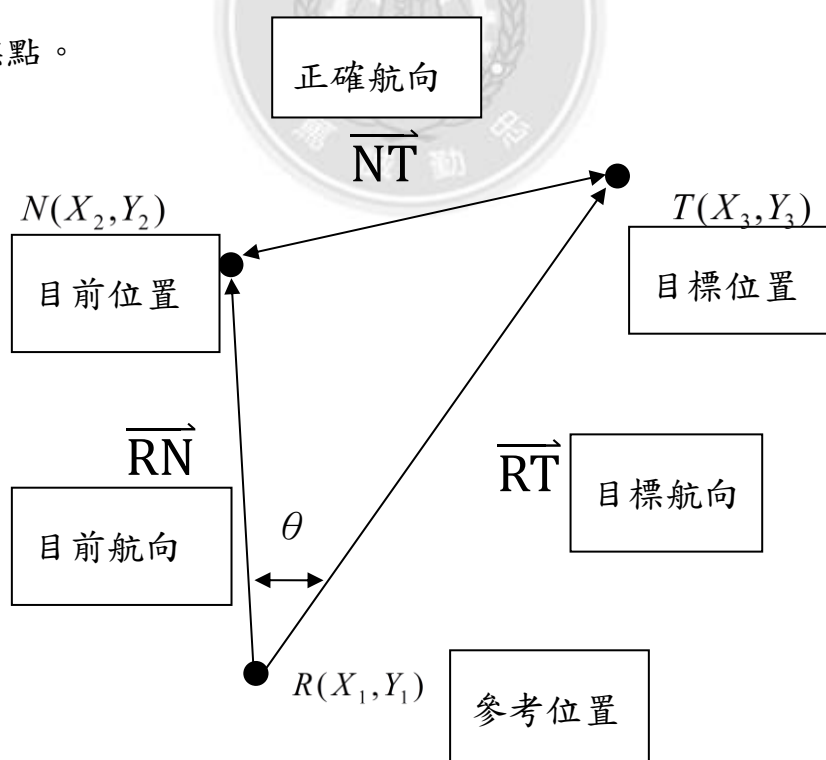


圖 3.2 參考點假設未更新示意圖

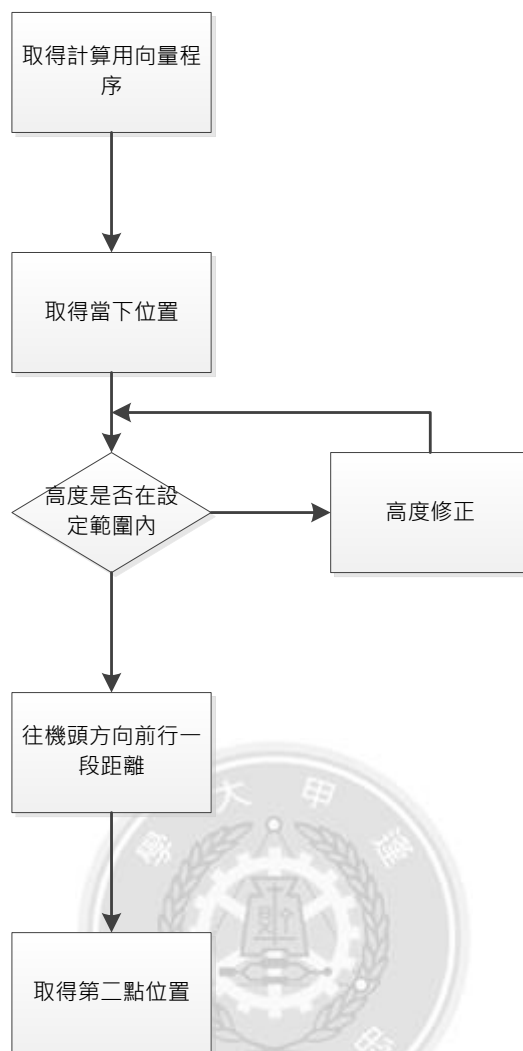


圖 3.3 取得即時更新位置點流程圖

當內積夾角  $\theta$  大於吾人所設定之角度時，此時再藉由目標航向及目前航向的外積正負值判斷左右轉，當外積值為負值，其判斷目標航向在多軸旋翼機之左邊，此時開始進行向右轉向的動作，即遙控器訊號打 Yaw 向右，等待多軸機轉向到我們要求的位置後再將其打回中間值。由於本文所使用的無人載具為多軸旋翼機，它可以停旋轉向，可以有效避免如定翼機轉彎過度的問題，而當修正至  $\theta$  值小於設定角度時，便可以直接控制 Pitch 向前使機身呈前傾姿態，直線往目標點

飛行。若多軸旋翼機在直線前行的過程中再度偏離航向，則必須再重複上面所述的姿態控制的動作。在此過程中，飛行控制板上所提供的姿態、高度計和電子羅盤的數據也透過無線傳輸模組送往地面站作詳細的飛航資訊確認。

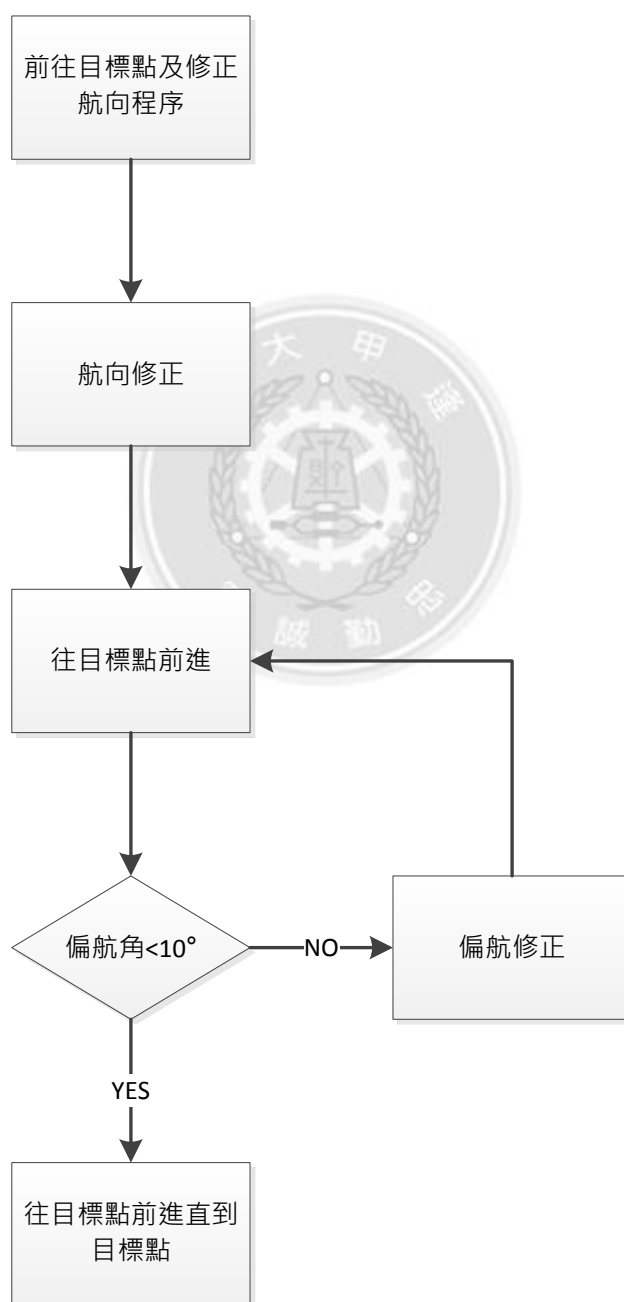


圖 3.4 前往目標點及修正流程圖

由於本文目前所做的向量計算只有考慮於平面，所以關於高度方面本文採用的是固定高度的方式來航行，所以自主導航執行的另外一個條件是所在高度必須在設定的高度範圍內，吾人定了一個高度誤差範圍為上下 30 公尺，當傳送預先設定好的高度給導航電腦(Arduino)後，若多軸旋翼機的所在高度不足抑或是超過此設定範圍時，則會於切換為導航模式的位置點持續以八成油門加速或兩成油門減速以進入本文設定的高度範圍內。假如內積夾角  $\theta$  值大於設定角度，自主導航程式就會開始運作，直到導航電腦(Arduino)判斷無須轉向時，就再度進入修正高度的程序；在高度範圍內而且又判斷無須轉向時，則完成導航並停懸於目前位置。流程圖如圖 3.4。

### 3.1.3 轉向控制方式

機載電腦 Arduino 接收到 GPS 訊號、遙控器訊號後，擷取 GPS 訊號中的經、緯度，再經由 Arduino 程式中的向量導航程式運算後，再透過 Arduino 的 PWM 輸出腳位送出與遙控器相同的 PWM 訊號給 TMF 飛行控制板，再由飛行控制板輸出訊號給電子變速器來達到控制馬達轉速的目的。

飛行控制板傳送給 Noavtech B4515 電子變速器的指令中主要為 PWM 訊號。吾人使用的 TMF 飛行控制板最多可以支援到八軸旋翼

機，所以編號為 0~7，而本文使用到的馬達總共有 6 顆，其同時負責滾轉、方向舵和油門。而多軸旋翼機轉向時，全依靠六顆馬達利用轉速差所產生之力矩達到 Yaw 的效果。

若多軸旋翼機的航向與目標位置的夾角大於設定角度時，此時機載電腦(Arduino)就會判斷已經偏離目標航向，接著進行向量外積運算以確定目標位置是相對於目前位置的右邊或左邊以及角度，接著多軸旋翼機就可以朝判斷後的方向飛行，以機頭對著目標位置的姿態前進。而多軸旋翼機的偏航轉向方式是依靠提高其中三個馬達的轉速，並降低另外三個馬達的轉速，來獲得轉速差所產生之旋轉力矩，且不會失去原有的昇力，藉此以達到改變機頭之方向，且高度不會下降的目的；而滾轉則是利用左右兩邊各三個馬達轉速的不同，來獲得的兩邊昇力不平均，進而控制機身姿態滾轉到特定角度。

### 3.1.4 到達目標點判定【3】

由於 GPS 本身就有著 5-10 公尺的誤差範圍，因此多軸旋翼機的飛行無法十分精準的到達並停留於目標點，所以本文設定一個範圍來判定是否有到達目標點，以避免到達的判定太過於困難，而導致持續修正無法完成任務的情況，並且以此為導航點的依據【2】、【10】。

本文假設地球為一正圓球體，計算後得到經度一秒約為 31 公尺、

緯度一秒約為 31 公尺，而對於兩點之間的距離計算，本文則是採用先將經緯度的單位(度、分、秒)統一換算為 NTU 單位，NTU 為度乘以十萬，先不考慮遠距離，且為初略估算，故使用畢氏定理即可。其算式如下：

$$\text{地球半徑} : R \cong 6.3782 \times 10^6 \text{ m}$$

$$\text{經度每秒} : 2\pi R \div (360 \times 60 \times 60) = 30.92 \text{ m}$$

$$\text{緯度每秒} : 2\pi R \div (360 \times 60 \times 60) = 30.92 \text{ m}$$

$$\text{NTU 單位} : \text{度} + \frac{\text{分}}{60} + \frac{\text{秒}}{3600}$$

利用誤差半徑的方式，由多軸旋翼機當時的目前位置點與目標點所構成向量的絕對值，如果小於或是等於誤差半徑時，就判定多軸旋翼機為已到達目標點【2】、【10】。

本文設定之誤差半徑為 5 公尺，因為若誤差半徑設定較大，多軸旋翼機只要接近誤差半徑的邊緣時就會判定為到達導航點，如此一來對於多軸旋翼機到達目標點的判定準確性將有所誤差；但若誤差半徑設定過小的話，多軸旋翼機可能會超越目標點(GPS 有 5-10 公尺的誤差)，即永遠無法判定到達目標點，也就是說多軸旋翼機目前的位置和目標所構成向量的絕對值，小於或等於 5 公尺時，便判定到達目標點，然後就維持停懸狀態等待地面控制站或遙控器送出新的命令。



## 3.2 導航程序

### 3.2.1 導航流程

機載電腦 Arduino 為整個多軸旋翼機的導航控制中心，藉由吾人撰寫的 Arduino 導航程式來完成自主飛行的任務。首先程式將會分為兩個部分，一部分為遙控訊號，此部分會將遙控器的 PWM 訊號不做處理，直接轉傳送給 TMF 飛行控制板，另一部分則為導航控制程式。而導航程式的部分，將會持續更新多軸旋翼機的目前點及參考點，返航點則是利用 TMF 飛行控制板本身所內建的功能，此飛行控制板會自動將起飛點設定為返航點。而機載電腦(Arduino)將接收 GPS 訊號訊號，並擷取可用的 GPS 訊號(\$GPRMC 格式中第二段訊號為 A)中的經、緯度後，由向量導航法的運算來決定轉向方向與角度大小，也用來判斷是否達目標點。倘若啟動導航程式時，並未在本文所設定的目標點誤差範圍內，則會先判斷多軸旋翼機是否在所吾人所設定的高度範圍內，接著再判斷多軸旋翼機的目前點與目標點的夾角  $\theta$ ，再藉由向量外積所產生的正負值來判斷目標點在多軸旋翼機的左側或右側，接著再進入多軸旋翼機轉向的程序。當多軸旋翼機達到目標航點且機身呈水平姿態後，則執行停旋程序保持停旋狀態，直到遙控器或地面站有下一步指命；若夾角  $\theta$  又大於設定值，則在進入導航迴圈，

直到完成飛行任務為止。導航程式流程圖如圖 3.5 所示。



圖 3.5 導航程式流程圖

### 3.2.2 安全設置

本程式各階段皆有設置安全開關，以防止於導航程式中之突發狀況情形。此安全開關可以於突發狀況要發生時，可即時切換，導控電腦便會在結束當下的動作程序後，省略後續該執行之程序動作，並結束此次導航任務。此設置為安全考量，於此狀況結束任務後，便直接由地面操控人員接手飛行控制，不會進入一般任務結束後的停懸等待階段。

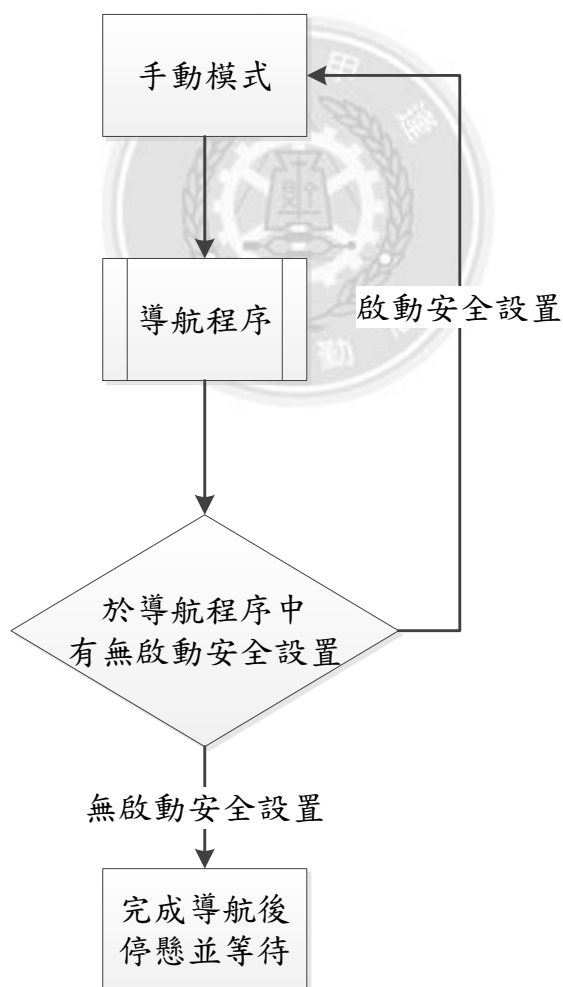


圖 3.6 安全設置流程圖

此安全開關設置於第五動，與導航任務開始之切換為同一動，故當導航時發生突發狀況時，只需將第五動開關立即接換回手動模式，便是直接進入安全模式，便可以直接由地面人員操控來應付突發狀況，待狀況解除後，即可重新開始導航任務。

### 3.3 地面控制站設計

地面控制站顧名思義就是把在多軸旋翼機上的飛航資訊經過整合包裝後利用無線傳輸模組的傳送端，藉由無線傳輸的方式把資料傳送到地面的接收端，並且在接收到這些封包資料後，利用 LabVIEW 程式撰寫成便於閱讀的人機介面，以利於能夠直接得知多軸旋翼機的飛航狀態，包括 GPS 資料、高度和姿態角度的變化……等等，而且也可以將目前多軸旋翼機的位置呈現於人機介面上的地圖上，也就更能簡單且明瞭的直接了解目前多軸旋翼機的位置；地面控制站除了可以得知目前的資訊之外，也可以直接傳送更新後的導航點給多軸旋翼機，透過這個功能讓多軸旋翼機可以在一次飛行任務當中接收新的指令並更換導航點；這樣就不會受限於一次性的任務限制，亦即要返航降落後才能再給予新的任務，然後再重覆執行起飛的動作之不便。其運作流程圖如圖 3.7 所示。

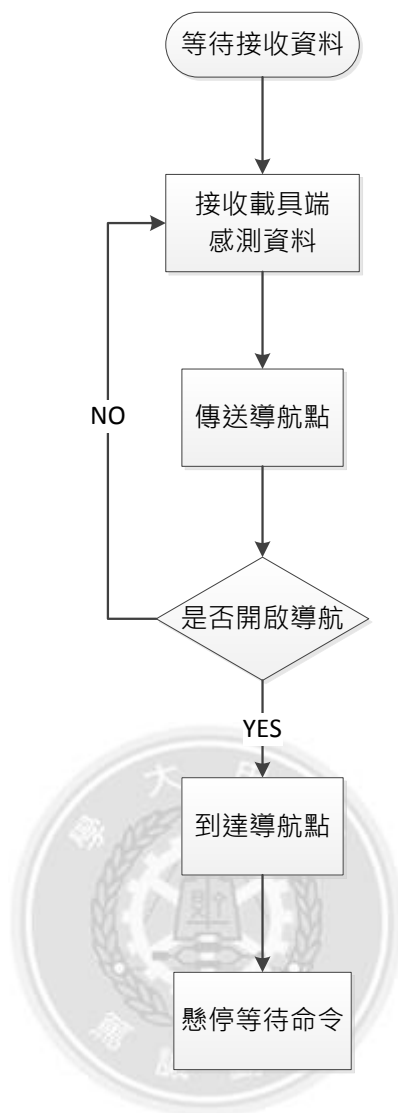


圖 3.7 地面控制站運作流程

### 3.3.1 地面監控站人機介面設計

吾人以 LabVIEW 為主要程式的基本架構，程式介面如圖 3.8，有方便的地圖顯示，飛行姿態的訊息，亦可以輕鬆的藉由滑鼠點擊的圖來輸入導航經緯度座標並傳送至機載電腦(Arduino)。

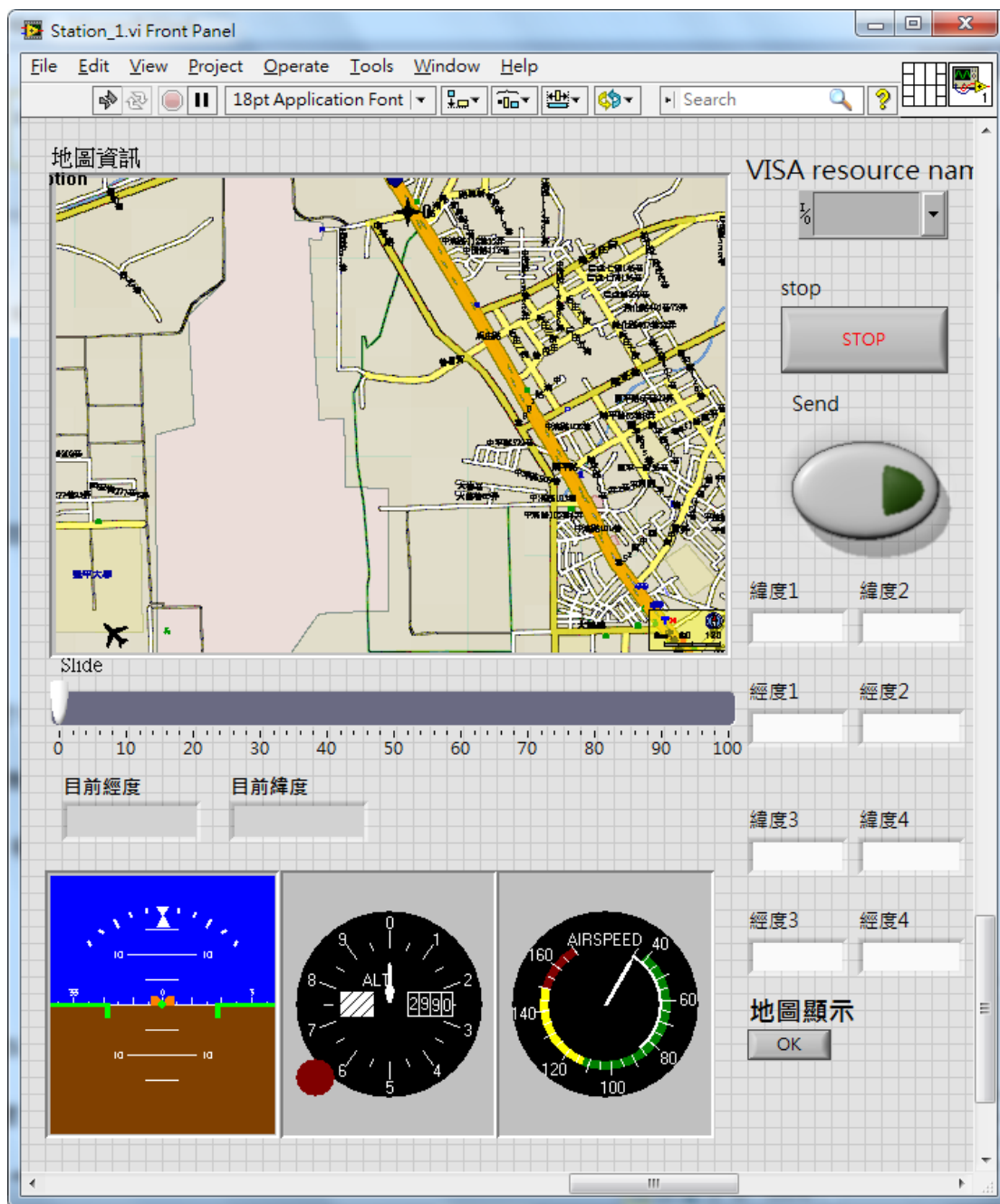


圖 3.8 地面監控站人機介面圖

接收的數據除了包含 TMF 飛行控制板的感測器資訊，以及經過向量計算後判定要作左轉或右轉的動作回傳，所以可以很清楚的知道多軸旋翼機正在進行的動作，以及是否有如預期的飛往指定的目標點。

### 3.3.2 ActiveX 外掛程式

由 Global Majic Software (GMS) 公司所開發的 Aircraft Instruments 以及 MMAP 這兩款軟體，可以與 LabVIEW 相容，是市面上通用的飛行儀表和電子地圖軟體，TMF 飛行控制板上感測器的資料可以藉由飛行儀表來呈現目前的狀態如圖 3.9【4】。

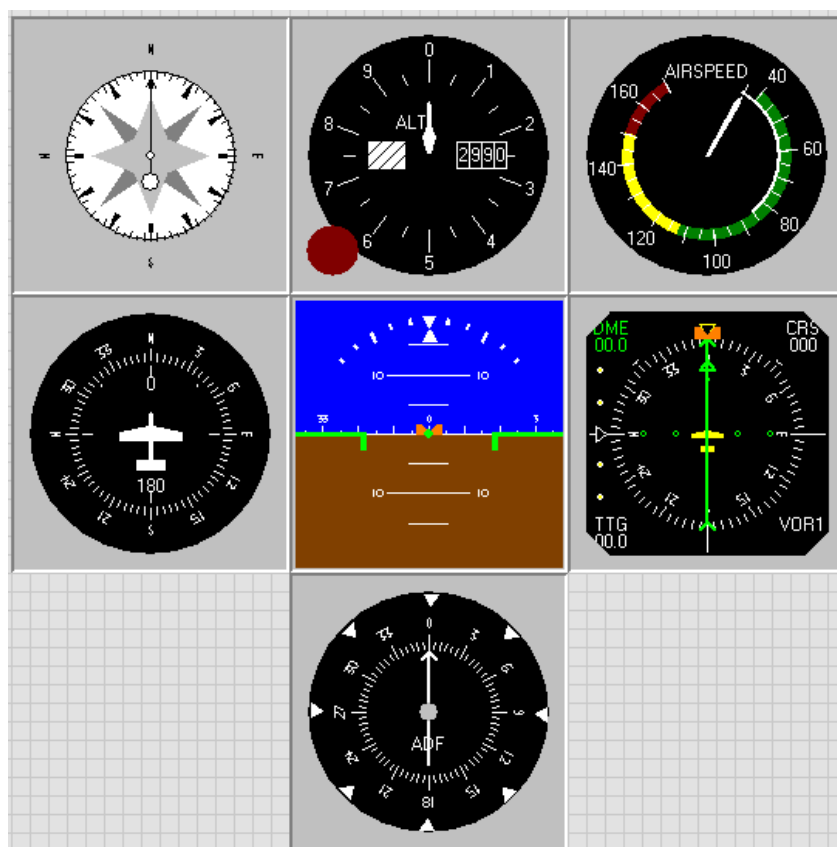


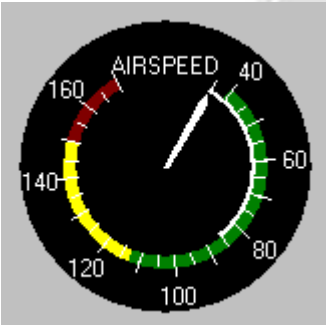


圖 3.9 飛行儀表範例

本文使用的飛行儀表簡介如表 3.1，吾人只簡介本程式有使用的飛行儀表，其他沒有說明的儀表請參閱參考資料。

表 3.1 儀錶板樣式

儀表樣式	說明
 An attitude indicator instrument with a blue sky background and a brown ground background. A white horizon line separates the two. A white aircraft symbol is centered on the horizon. The instrument has markings for roll and pitch angles.	此儀表可顯示飛行器的滾轉角與俯仰角。
 A circular altimeter instrument with a black face and white markings. The needle points to 0. A digital display shows '2990'. The word 'ALT' is visible. There is a red warning light at the bottom left.	此儀表可顯示目前飛行器的高度。
 A circular airspeed indicator instrument with a black face and white markings. The needle points to approximately 100. The word 'AIRSPEED' is at the top. The scale has markings at 40, 60, 80, 100, 120, 140, and 160.	此儀表用來表示目前飛行器的空速。



## 第四章

# 系統整合與無人自主導航測試

### 4.1 系統測試

裝上多軸旋翼機實機應用飛行之前，本文將所有軟硬體整合並組裝完成後，藉由多個測試項目以確保程式正常運作，本文列舉幾項較為重要的測試如：軟硬體整合測試、地面測試以及模擬測試，一一按照所列之項目做測試及調整，這些測試的目的在於減少直接飛行時的錯誤直接造成損害。

#### 4.1.1 導控硬體整合測試

在進行完整的地面測試以前，目前最重要的就是各硬體的運作跟軟體的結合是否正常。圖 4.1 為各硬體整合圖。

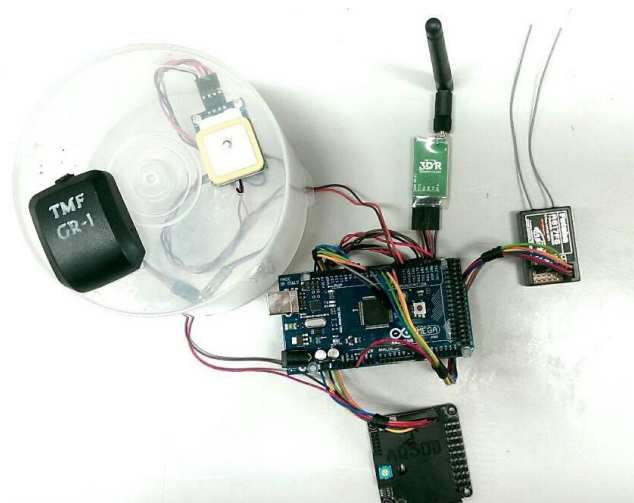


圖 4.1 硬體整合圖

首先開啟測試用的程式，並連接各硬體後，逐一測試並檢查各硬體所讀到的數據資料的正確性，以及是否有正常運作。找出有無不能正常運作功能之程式，並修正之。圖 4.2 為系統流程整合圖。

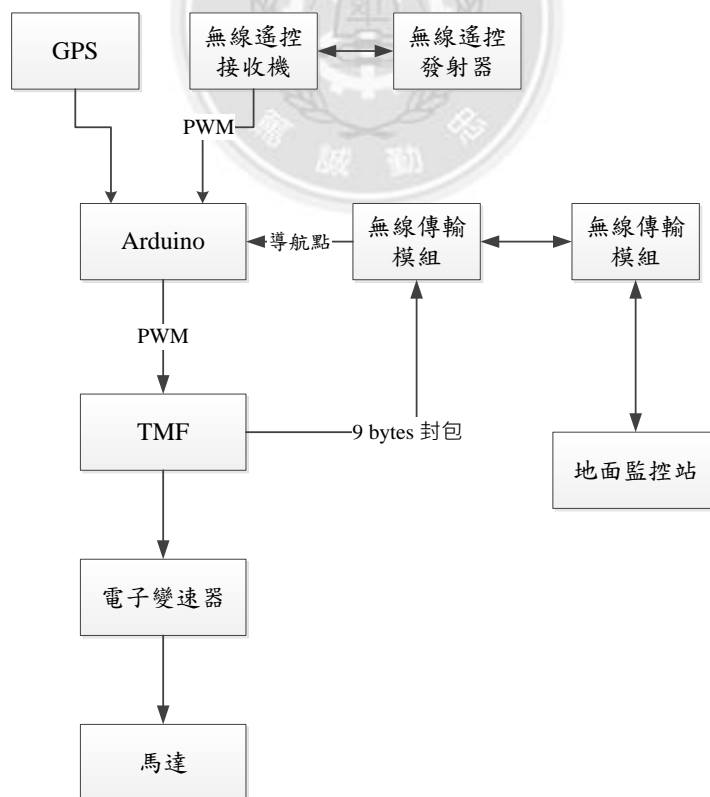


圖 4.2 系統整合流程圖

### 各次系統訊號測試：

GPS 的測試，可直接使用 Arduino 擷取數據，並藉由 Arduino 內建的 Serial Monitor 功能來顯示目前 GPS 的狀態，如圖 4.3。

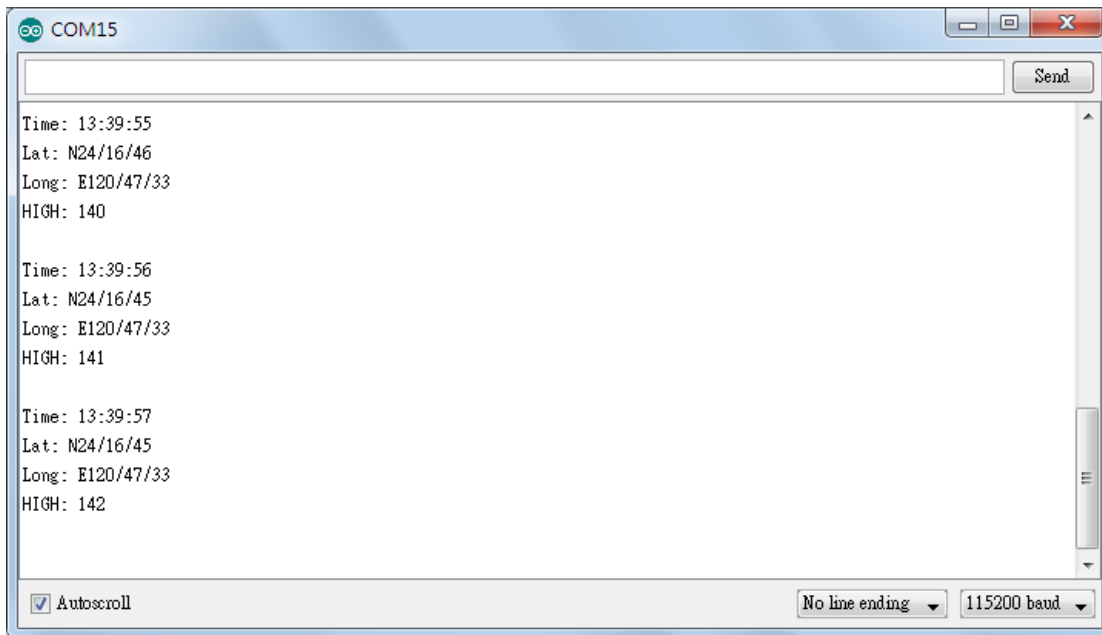


圖 4.3 Arduino 擷取 GPS 訊號圖(GPGGA)

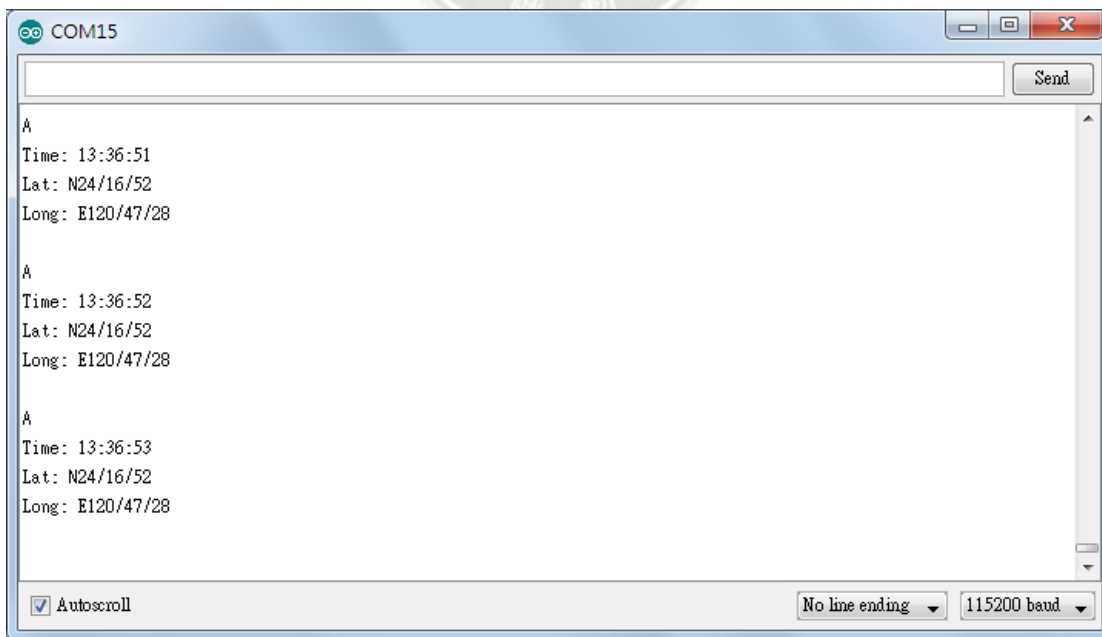


圖 4.4 Arduino 擷取 GPS 訊號圖(GPRMC)

三軸姿態儀的測試，由於我們是使用飛行控制板內建的三軸姿態儀，所以這個部分我們使用已撰寫好之 LabVIEW 程式，來讀取飛行控制板上所輸出的三軸姿態資料，再將其解碼為較為容易理解的顯示讀數，這邊利用量角器與螢幕上的顯示讀數來做比對，這邊分為兩個部分，分別是俯仰角(圖 4.5、圖 4.6)、滾轉角(圖 4.7、圖 4.8)。

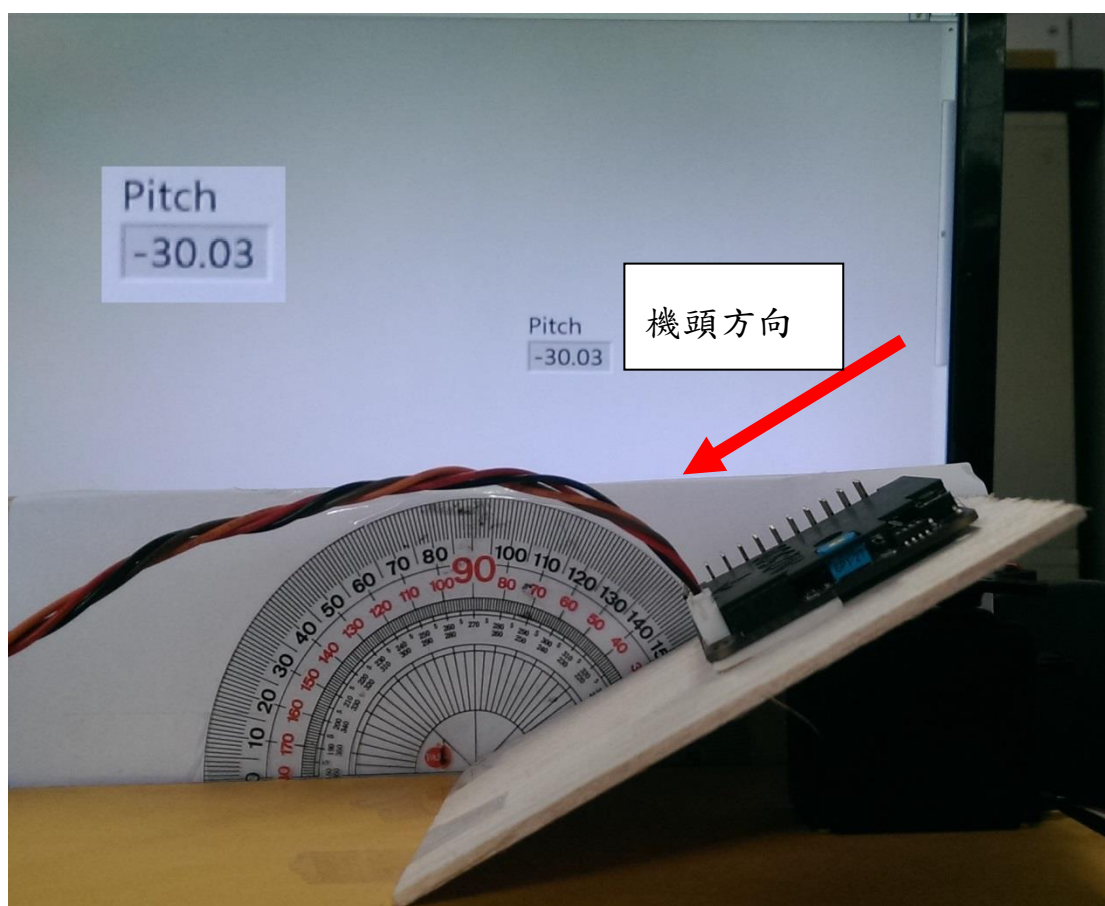


圖 4.5 俯仰角於 LabVIEW 讀數比對圖(一)

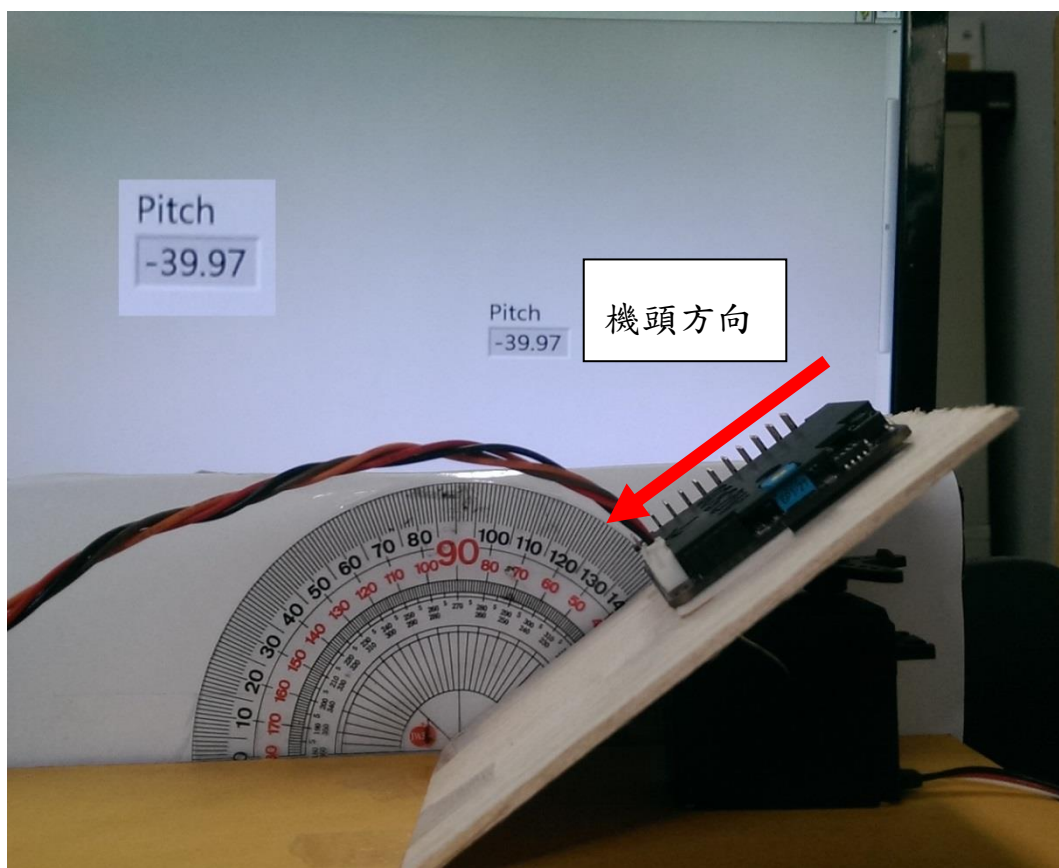


圖 4.6 俯仰角於 LabVIEW 讀數比對圖(二)

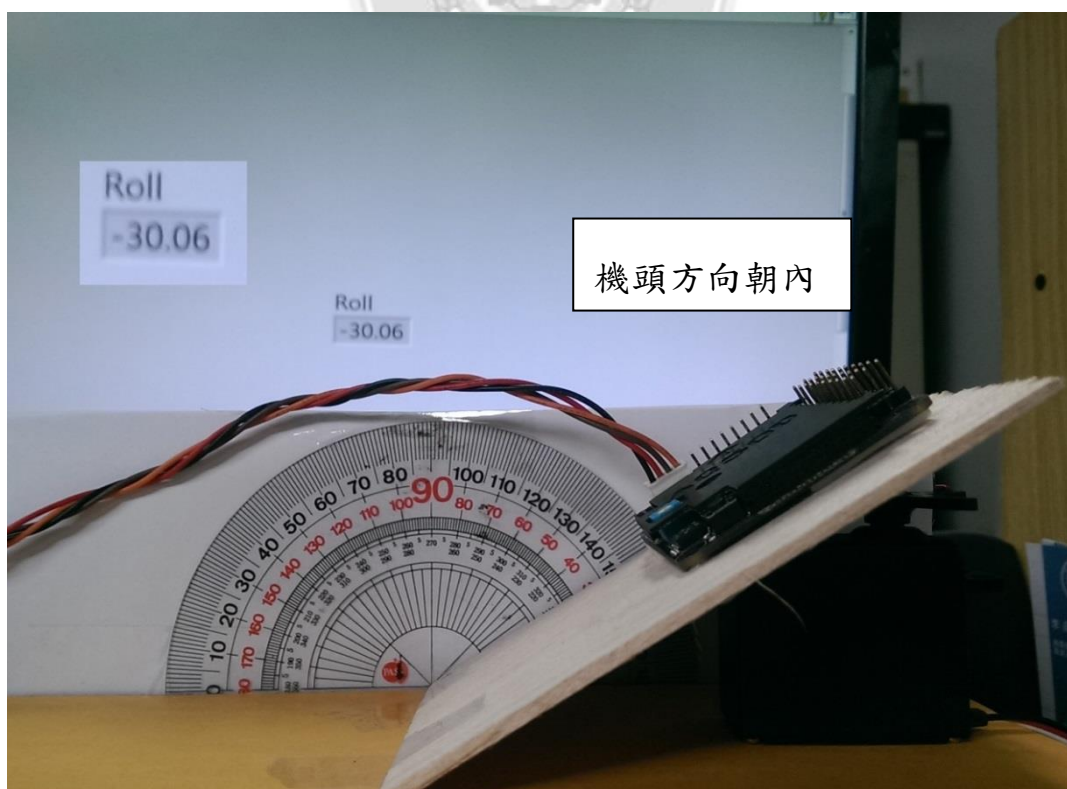


圖 4.7 滾轉角於 LabVIEW 讀數比對圖(一)

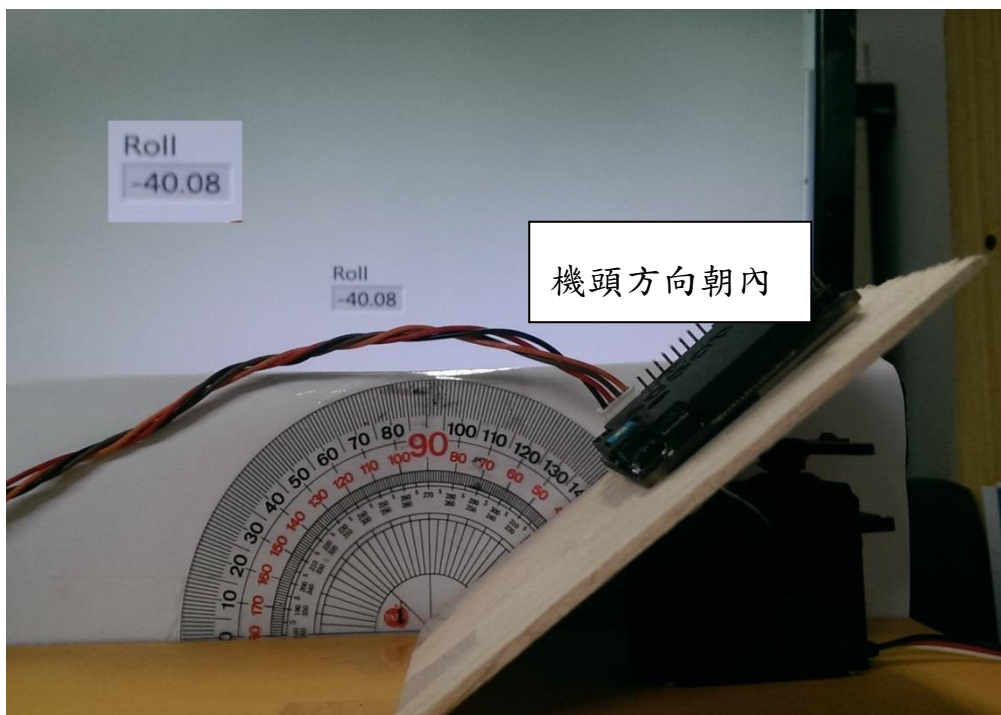


圖 4.8 滾轉角於 LabVIEW 讀數比對圖(二)

飛行控制板的測試告一段落後，接著做遙控器訊號的測試，先使用 Arduino 確認讀取到接收機的訊號如圖 4.9，再比對經處理後 Arduino 所輸出之訊號如圖 4.10，檢查是否一致。

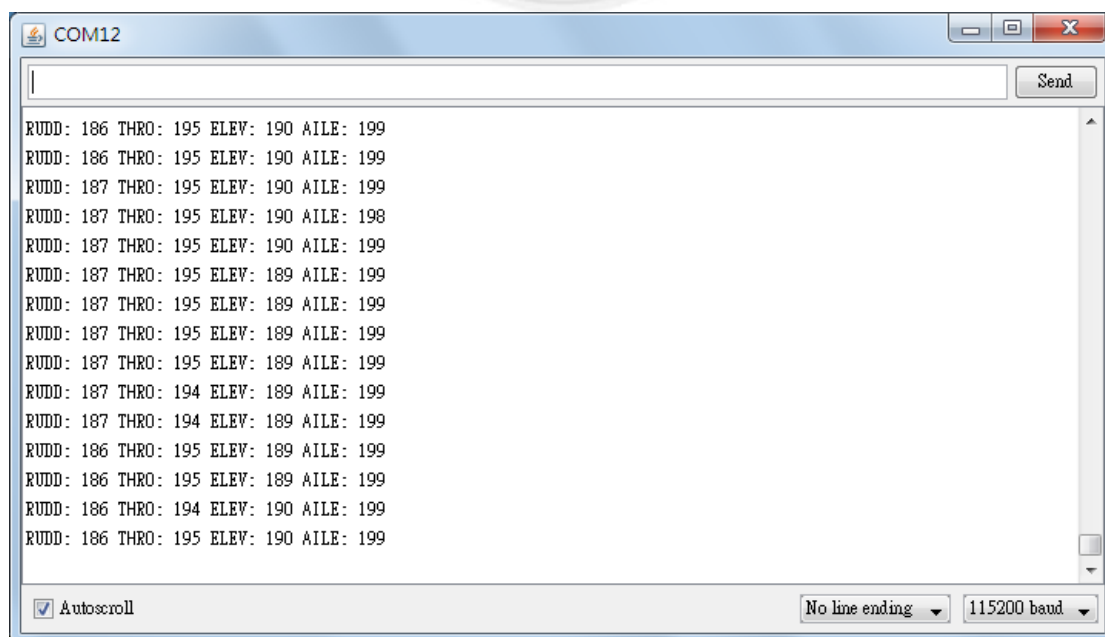


圖 4.9 Arduino 讀取接收機訊號圖

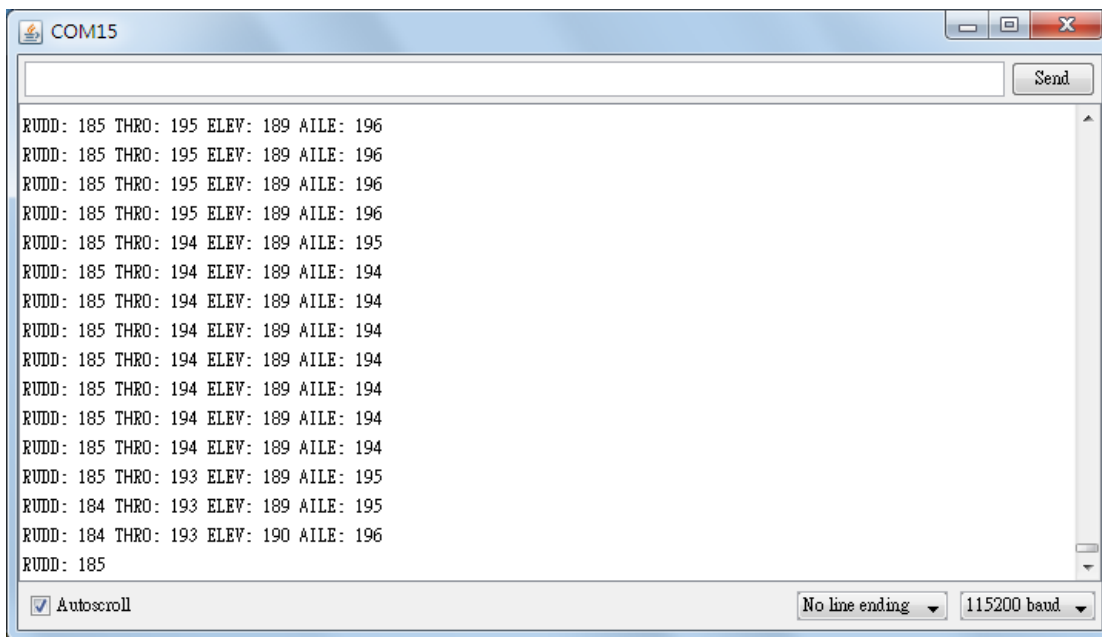


圖 4.10 Arduino 輸出訊號圖

我們將遙控器的第五動作設置為導航的切換開關，在 Arduino 程式裡利用第五棟的讀數來判斷切換至導航模式與否。



圖 4.11 導航切換控制器測試(一)



圖 4.12 導航切換控制器測試(二)

無線傳輸模組的測試，我們係利用地面站端的 LabVIEW 電腦程式，藉由 3DR 無線傳輸模組傳送導航點資料給 Arduino 板子，所以這邊做無線傳輸模組的測試，以確定傳送的資料無誤。

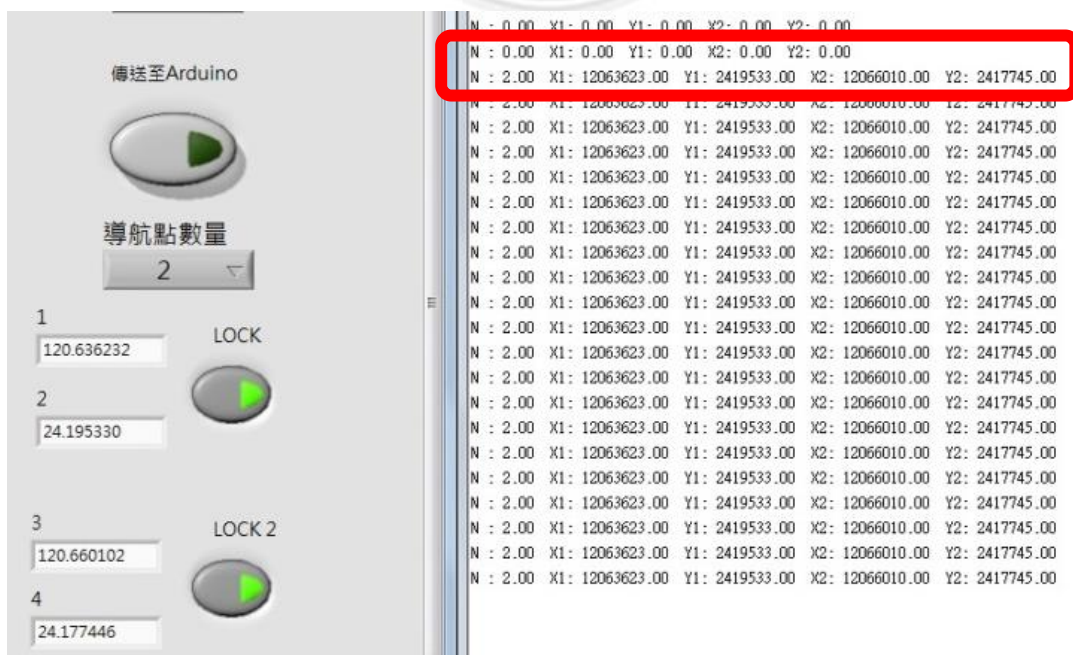


圖 4.13 無線傳輸模組傳送導航點測試(一)



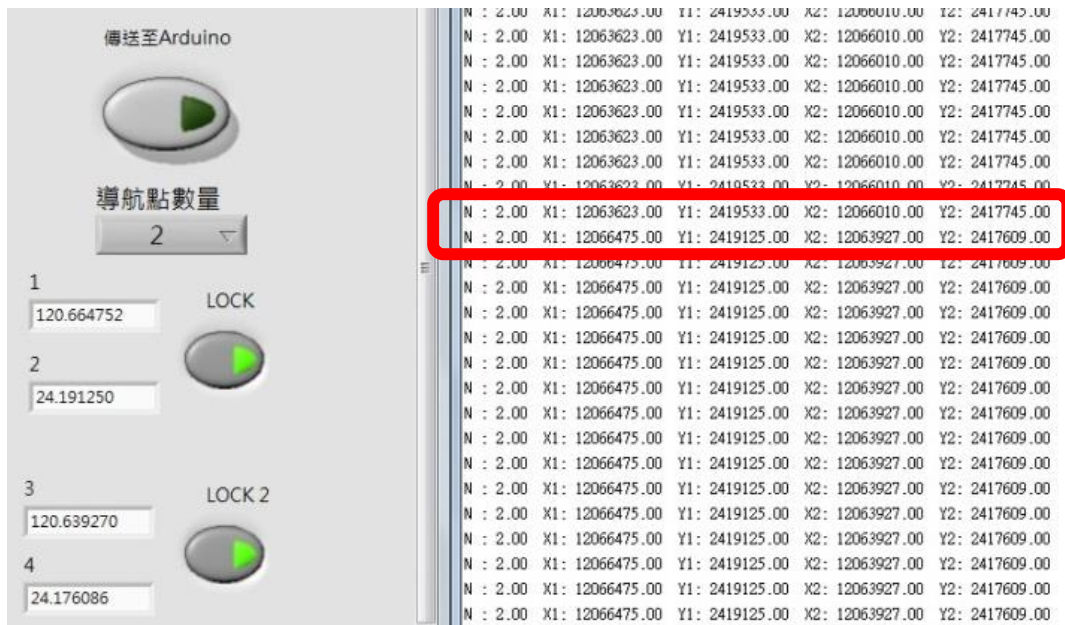


圖 4.14 無線傳輸模組傳送導航點測試(二)

各次系統都測試並且無誤後，再用另一台 Arduino 做起飛前的地面確認，確認所輸出給 TMF 飛行控制板的訊號是否正確，至此硬體整合測試告一段落。

#### 4.1.2 地面控制站測試

開啟地面端程式，確定 ActiveX 控制項可以正常動作，即地圖跟飛行儀控軟體都能夠正常顯示，接著也是確定是否有任何不能運作的程式功能，找出並加以修正。



圖 4.15 地面站圖控介面測試

## 4.2 地面測試

地面測試的方式是開啟導航電腦跟地面控制站，觀看導航資訊是否可以從導航電腦經由無線傳輸模組傳送給地面控制站，然後地面站再藉由地圖及飛行儀控軟體直接呈現目前的導航資訊，接著再測試從地面站是否可以傳送導航點的資訊給導航電腦，有傳送到的話再觀察是否可以正常執行命令。由於須考慮到載具的起飛重量，所以將硬體裝入多軸載具前，必須先詳細的計算並確認外掛模組的重量，下表 4.1 列出外掛模組中各硬體之重量級總和。

表 4.1 各硬體重量概算表

硬體	數量	重量
Arduino	1	28.7g
Arduino 擴充版	1	27g
GPS 接收器	1	17.7g
3DR 無線傳輸模組	1	14.5g
總和	4	87.9g

故吾人所建構之外掛模組，重量不到 88 公克，所以可以掛載於其他多軸飛行載具上，不至於造成太大的影響。

#### 4.2.1 手動遙控模式測試

首先將整合好後的系統，組裝完成並接上電腦。利用電腦所讀取到的模擬遙控器數據，來測試並確認導航程序及手動模式是否正確。本文撰寫之手動遙控模式，是在起飛及降落程序時，可以依照操控人員隨意操控配合動作，一般使用在普通飛行及無特殊任務需求時，通常為可視範圍內的飛行。接著依照以下順序做測試，依序為各位置置中、俯仰、偏航、油門、滾轉。

此時搖桿置中，RUDD 訊號亦置中，約 185。THRO 置中，約 186。  
ELEV 置中，約 189。AILE 置中，約 195。



圖 4.16 各位置置中

此時搖桿於最大俯角位置，故 ELEV 為最大值約 238。



圖 4.17 遙控器於俯角位置

此時搖桿位於最大仰角位置，故 ELEV 為最小值約 136。



圖 4.18 遙控器仰角位置

此時搖桿位於右偏航位置，故 RUDD 為最小值約 128。



圖 4.19 搖控器於右偏航位置

此時搖桿位於左偏航位置，故 RUDD 為最大值約 240。



圖 4.20 遙控器於左偏航位置

此時搖桿位於最大油門位置，故 THRO 為最大值約 239。



圖 4.21 遙控器於油門最大

此時搖桿位於最小油門位置，故 THRO 為最小值約 141。

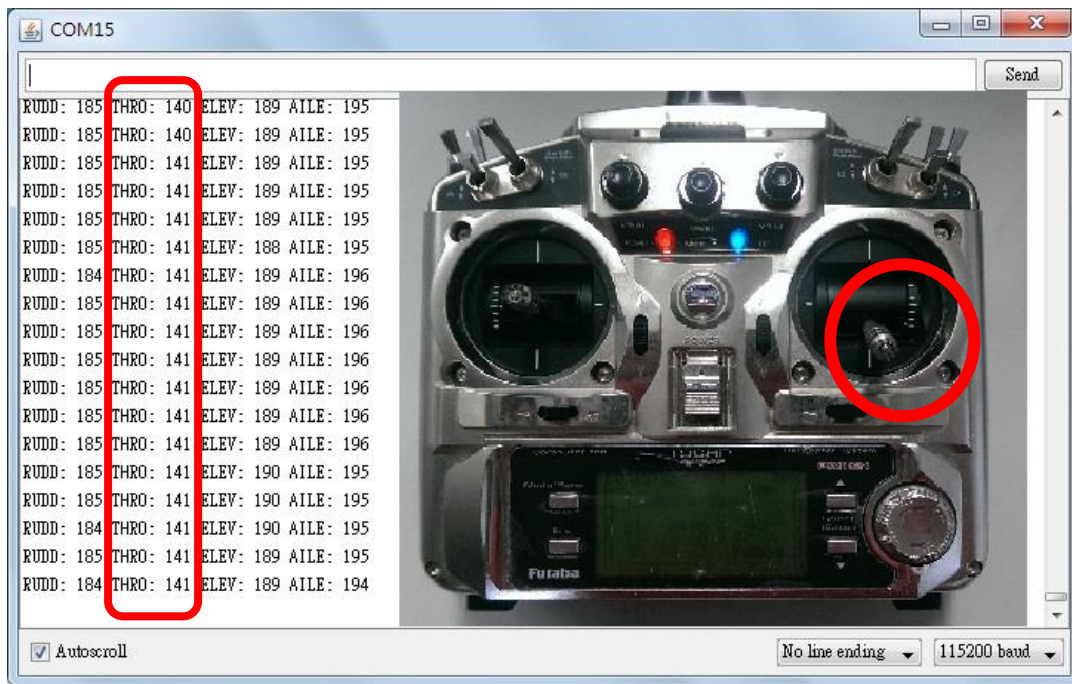


圖 4.22 遙控器於油門最小位置

此時搖桿位於右滾轉位置，故 AILE 為最小值約 141。

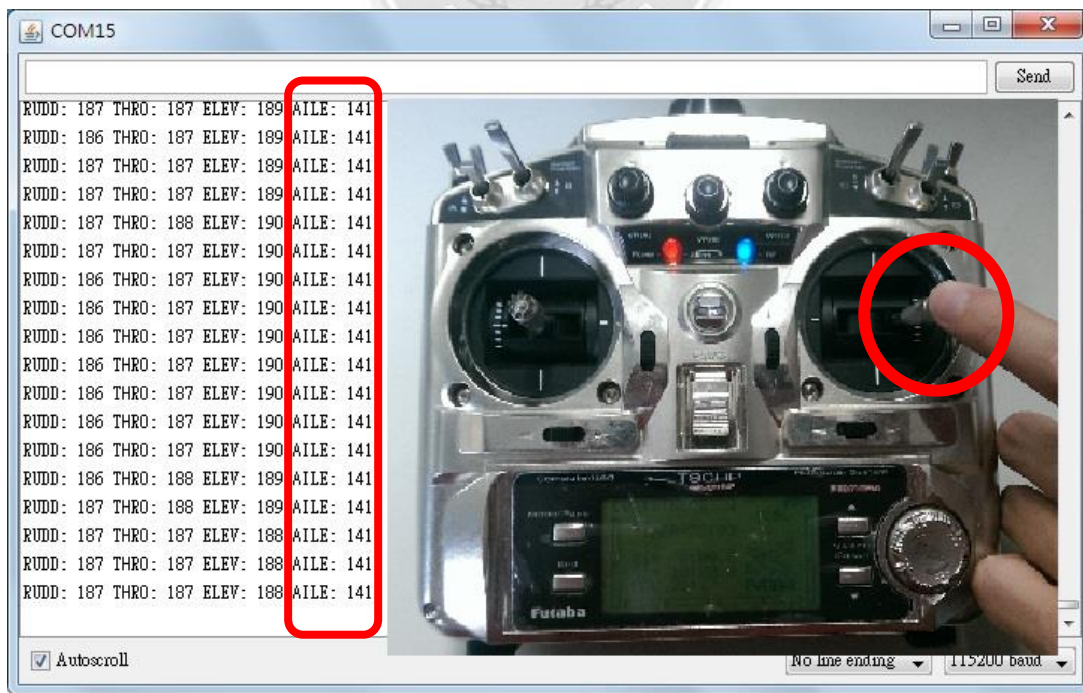


圖 4.23 搖控器於右滾轉位置

此時搖桿位於左滾轉位置，故 AILE 為最大值約 263。



圖 4.24 遙控器於左滾轉位置

#### 4.2.2 自動導航模式測試：

本文所撰寫之自動導航模式，由於在室內無法接收到 GPS 訊號，亦即目前點及參考點都無法更新，所以吾人採用給定座標點的方式進行，測試其所作的轉彎判斷以及偏航與俯仰的變化是否如預期般動作；載具判斷左右轉的測試無須使用地球的經緯度座標，可直接使用簡單的笛卡爾座標系統即可測試。本文直接設定參考點座標為(40,0)、目前點座標為(0,0)、目標點座標為(0,30)，用以測試是否做出右轉的判斷以及機載電腦(Arduino)輸出的模擬訊號偏航及俯仰部分的變化，當虛擬座標設定好後，便如下列步驟進行測試。流程圖如下。



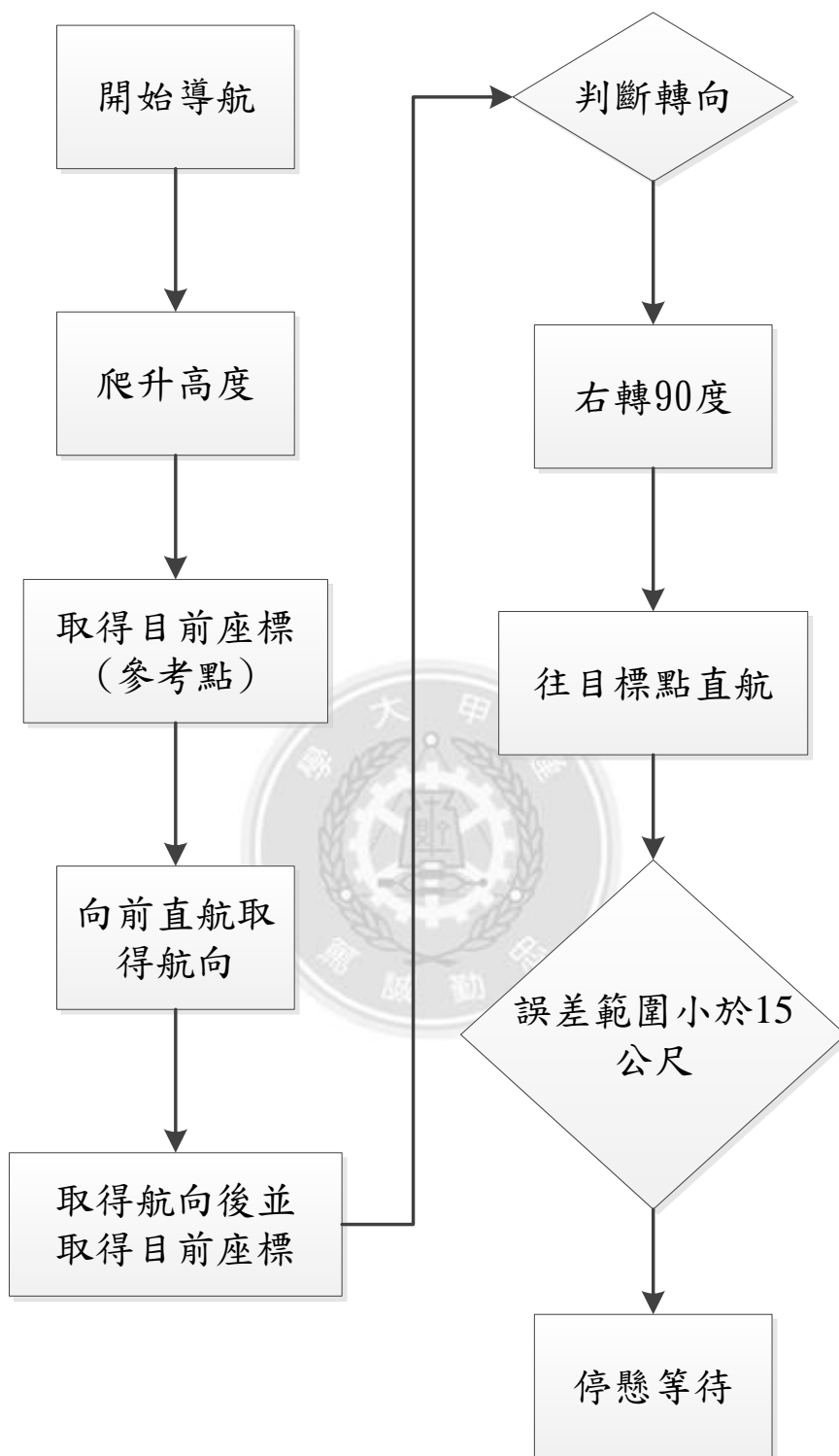


圖 4.25 導航運作流程圖

於參考點位置(40,0)，切換導航功能後，機體先爬升至一定高度後，在進行方向之導航，下圖為爬升時數據，機頭方向朝左。下列圖

示之箭頭方向 為表示升力方向，箭頭之長短則表示轉速之快慢。

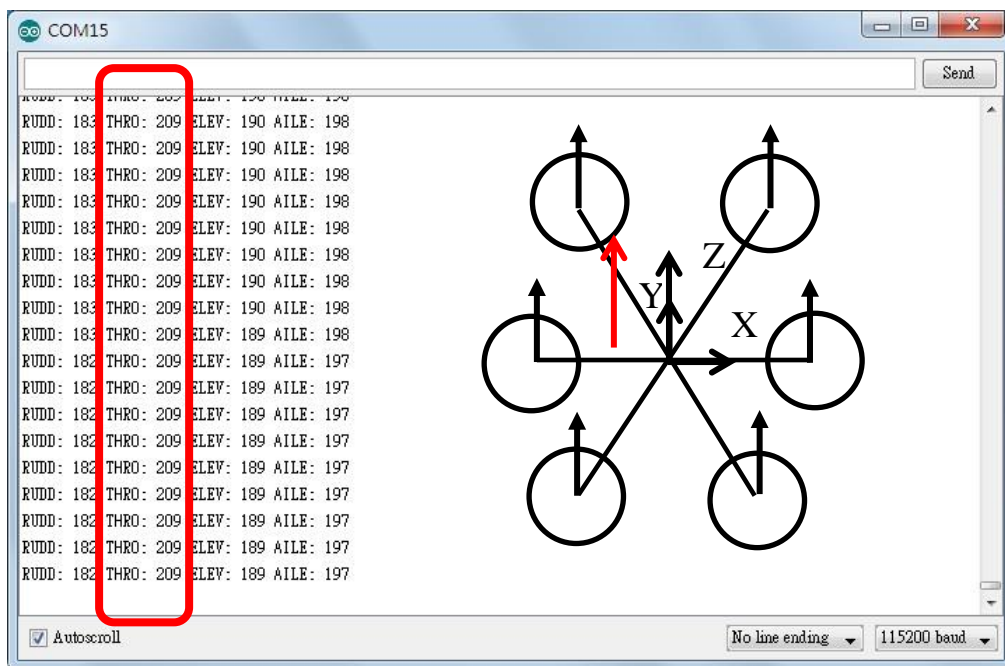


圖 4.26 多軸爬升數據圖

爬升至特定高度後，機體往前直行一段距離取得當時航向，如下

圖。

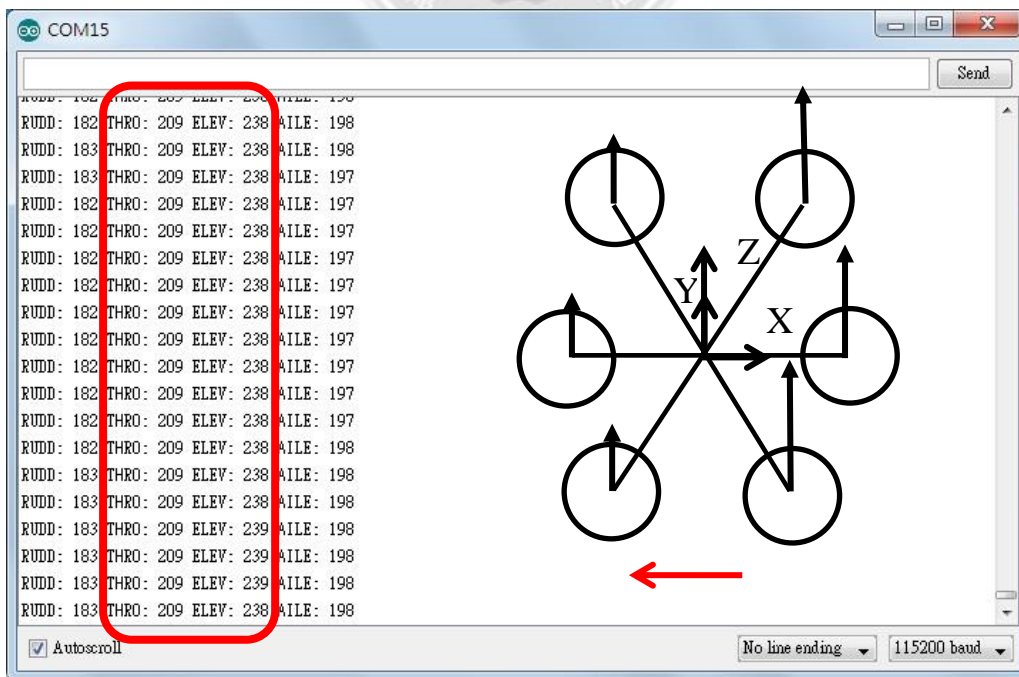


圖 4.27 多軸直航數據圖

此時到達目前點位置(0,0)，接著進行轉向，如下圖。

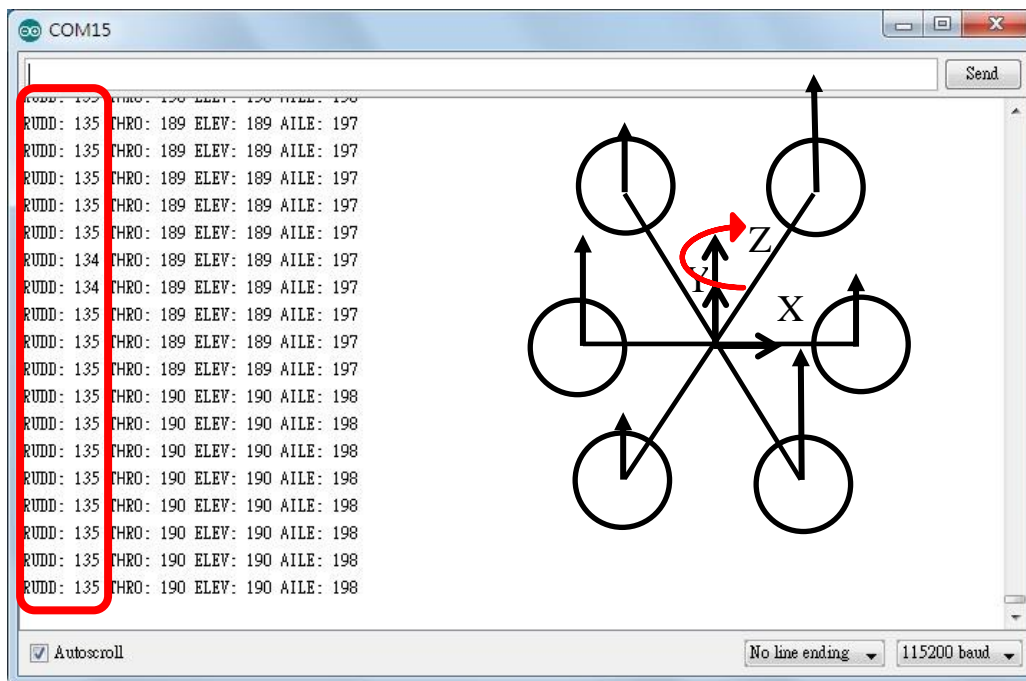


圖 4.28 多軸轉向數據圖

轉向後往目標點直航，前進直達目標點，如下圖。

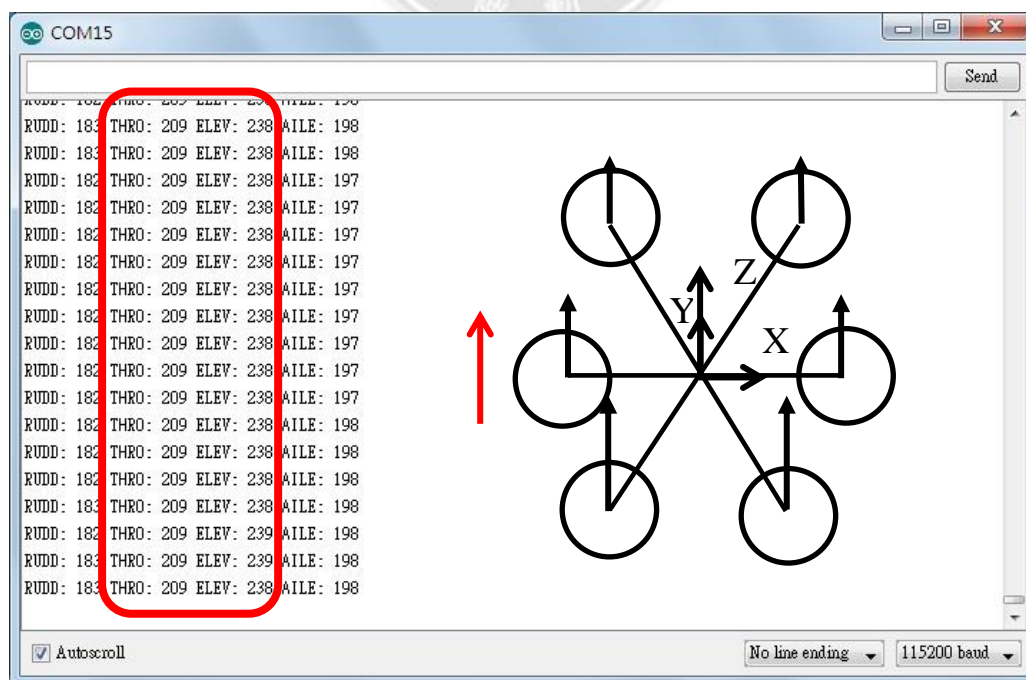


圖 4.29 多軸往目標點直航數據圖

直航至誤差範圍小於 5 公尺，判定抵達目標點位置(0, 30)後，做停懸的動作，等待地面操控人員下一步指令。

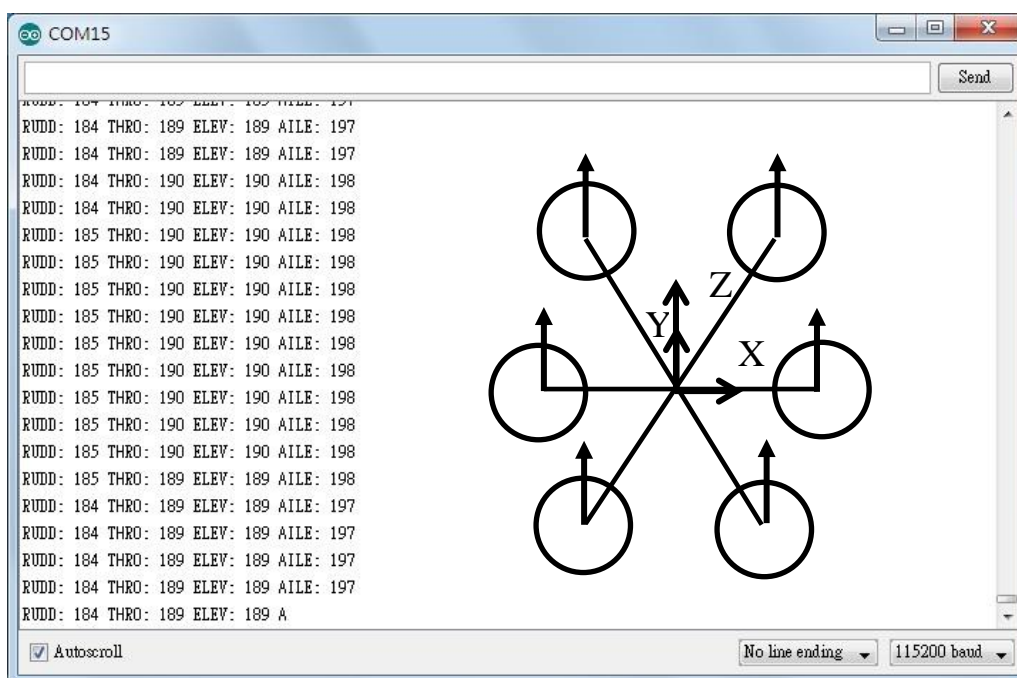


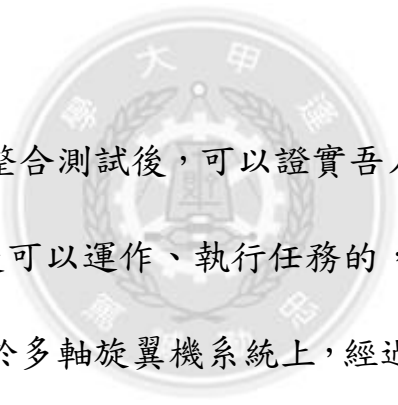
圖 4.30 多軸停懸數據圖

### 4.2.3 測試結果

本次的測試結果，在於各系統整合後可以順利讀取各項資料，且接收器所接收到的遙控器各項訊號動作，也有符合預先的程式設計執行，而多軸載具判斷之轉向結果也與吾人預先所規劃的如出一轍。至此，機載電腦(Arduino)能透過無線傳輸模組與地面監控站互相傳送資料，故證實其整合後，軟硬體皆可以順利執行轉向的判斷及控制，進而完成導航任務。

## 第五章

### 結論與未來展望



經過最後的模擬整合測試後，可以證實吾人所撰寫的 Arduino 程式及 LabVIEW 程式是可以運作、執行任務的，本文主旨為導航系統及設備可以直接裝載於多軸旋翼機系統上，經過簡單的地面測試後證實可以如預期般運作、使用。至此本論文的階段性目標已經達成。

由於吾人尚未做實際飛行測試，故待未來做實機飛行測試後才能有更多佐證，且吾人是第一次接觸導航系統的設計，所以採用的方式較為簡易，往往會出現一些延遲或是數據傳輸有些不穩定，所以比較好的方式為更加了解 Arduino 程式邏輯後，再重新整理並編輯程式，以求能達到更加完美的多軸載具導航控制。

在撰寫論文時，吾人發現到一些小問題，是關於通用於各家飛行

控制板的部分，目前吾人所撰寫之地面站僅能接收 TMF 單一廠商飛行控制板之資料，期望往後能對應多家廠商所出產之飛行控制板皆能使用吾人所開發之導航系統，且若能取得各家飛行控制板本身之資料，亦可以利用各家廠商本身所提供之 GPS 資料，可以取代吾人系統的 GPS 部分，如此不僅可以降低這套系統建構硬體之成本，亦可以降低兩組不同 GPS 所產生的誤差，提升此套系統的精準度。

最後本論文多軸旋翼機之導航系統架構雖已完成，未來仍可加以改良，可經由實際飛行測試加以驗證本文所開發之自主導航系統，並以最佳效率路徑為考量，提高效率及效能，使整體系統更加完整。



## 參考文獻

- 【1】 邱永賢 ；「應用 LabVIEW PDA 模組於導航系統之建構」；逢甲大學航太與系統工程研究所碩士論文；2009
- 【2】 張進欽 ；「應用 LabVIEW 虛擬儀控於機載控制系統之建構」；逢甲大學航太與系統工程研究所碩士論文；2009
- 【3】 翁子傑 ；「嵌入式系統於無人飛行載具自主導航之實現」；逢甲大學航太與系統工程研究所碩士論文；2010
- 【4】 劉祐瑜 ；「無人飛行載具地面監控站平台之建構」；逢甲大學航太與系統工程研究所論文；2006
- 【5】 Arduino ；<http://arduino.cc/en/Main/ArduinoBoardUno>
- 【6】 NOVATECH ESC AIR B4515 ；  
[http://www.novatech-tw.com/lib/dl\\_file.php?filename=BL-AIR110714.pdf&dir=products\\_dl](http://www.novatech-tw.com/lib/dl_file.php?filename=BL-AIR110714.pdf&dir=products_dl)
- 【7】 Arduino NEO-6M GPS 接收器 ；  
<http://www.u-blox.com/en/gps-modules/pvt-modules/neo-6-family.html>
- 【8】 Futaba 2.4Ghz 無線遙控發射器使用手冊 ；  
<http://manuals.hobbico.com/fut/9c-super-series-manual.pdf>
- 【9】 王閔正 ；「應用 GPS 於無人飛行載具導航系統之研究」；逢甲大學航太與系統工程研究所碩士論文；2006

【10】 3DR Radio Telemetry kit-433Mhz ；

<https://store.3drobotics.com/products/3dr-radio>





## 附錄 A

表 A.1 無線傳輸模組規格表【10】

Specification	3DR Radio Telemetry kit-433Mhz
Performance	
Duplex	Supports point-to-point half duplex
Receiver sensitivity	-121 dBm
Data rates	up to 250kbps
Power Requirements	
Supply Voltage	3.7-6VDC
Transmit current	100ma at 20dBm
Receive current	25mA