嵌入式系統設計

期末專題報告

平衡儀

指導教授：陳慶瀚

學生：馬英傑

學號983003110

E-mail：[finalsys@hotmail.com](mailto:finalsys@hotmail.com)

2014年1月

摘要

使一個平面保持水平，或者是維持在某個特定的角度，這項條件是許多不同的工作所需的基本條件，在進行物理實驗時，要確保設置實驗儀器之桌面與地面維持水平；飛行器在進行轉向時，要以特定角度傾斜機身，提供轉向所需之向心力；無人飛行器在停旋於空中時，要提供機身各方面之升力，使機身達成水平，諸如上述之事件，都需要維持某種平衡。

本專題製作之儀器，便是藉由Smart Sensor中的加速度計，來達成自動調整平衡的功能，並透過設計的展示平台，驗證此功能的實作成功與否。

1. 問題描述

姿態控制是飛行器的控制器中很重要的部分，在一架四軸飛行器(Quadcopter)的運動模式中，停旋是一種使飛行器定點於空中，維持一定的高度及方向，透過四個旋翼，來達成扭矩的平衡以及抵銷重力，而四軸飛行器在空中停旋時，若遭遇預期以外的外力，諸如風吹等自然現象，原本提供的平衡便會被打破，要回復平衡，便須藉由偵測飛行器當前的姿態改變，對四個旋翼的出力進行修正。

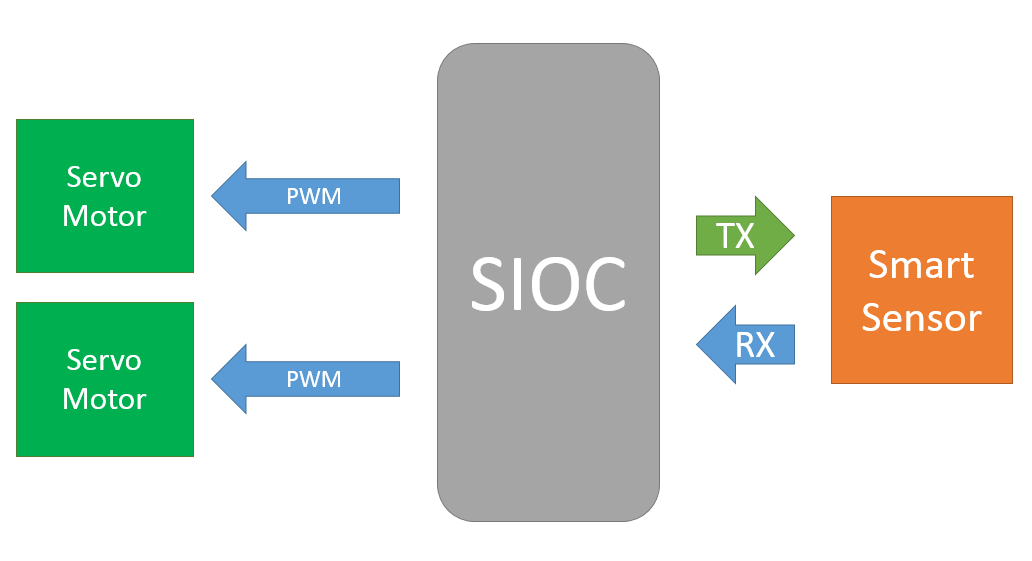
在研究四軸飛行器的運動時，停旋這個動作令我非常感興趣，透過控制器的自我校正，對輸出進行補償，進而維持某種所需的狀態，應該能應用於許多場合，輔助許多工作的進行，因此我希望在嵌入式系統的期末專題中，實作這樣功能，也由於目前我並無想法將此項功能運用於某個地方，因此便設計了一個測試平台，去證實此功能的實作。

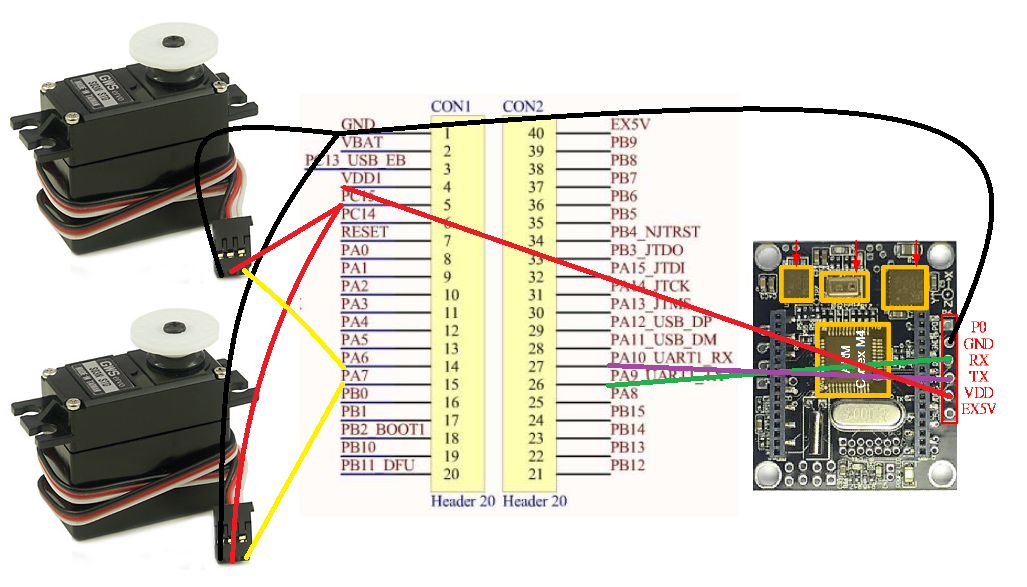
1. 原理和方法

本專題用以偵測平面水平度之感測器為加速度計，透過Smart Sensor所傳回之加速度計的三個向量分量，由重力加速度分配於六個平面(X+,X-,Y+,Y-,Z+,Z-)之情形，判斷平台之傾斜狀況，並以此做為依據，調整控制伺服馬達之PWM訊號。

在伺服馬達的控制部分，是採用累加的方式調整，由於硬體設計的部分讓兩個伺服馬達分別控制平台之X軸傾斜量與Y軸傾斜量，因此可大略視為兩個馬達之間互不影響(independent)，在PWM訊號的操作上，便可依X軸之重力加速度大小來決定1號碼達之PWM訊號應該漸增或漸減，2號碼達之PWM訊號則由Y軸上的重力加速度來判斷。

1. 硬體架構

3.1系統架構圖

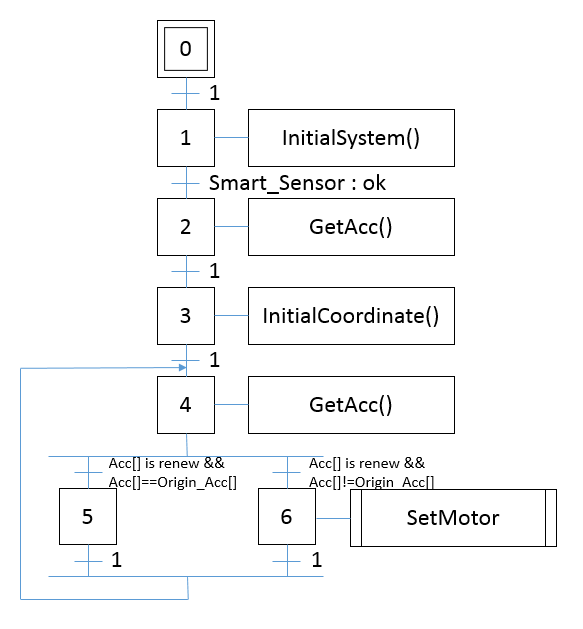
3.2系統線路

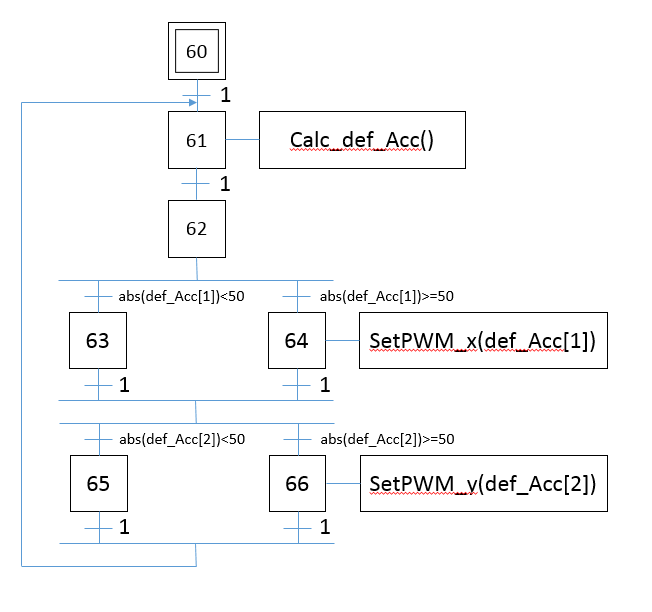
1. 系統設計

4.1 IDEF0系統架構

4.2 Grafcet

4.2.1 平衡儀系統主控制器



4.2.2 SetMotor

4.3 主要狀態與action

|  |  |  |
| --- | --- | --- |
| 狀態 | 動作 | 功能說明 |
| 0 | 0 | 初始狀態 |
| 1 | InitialSystem() | 加速度計致動、馬達位置歸零 |
| 2 | GetAcc() | 取得加速度值 |
| 3 | InitialCoordinate() | 紀錄初始位置(的加速度值) |
| 4 | GetAcc() | 取得加速度值 |
| 5 | Null | Do noting |
| 6 | SetMotor | 調整馬達位置 |
| 60 | 0 | SetMotor初始狀態 |
| 61 | Calc\_def\_Acc() | 計算目前位置與初始位置之加速度差量 |
| 62 | Null | Do nothing |
| 63 | Null | Do noting |
| 64 | SetPWM\_x(def\_Acc[1]) | 依照def\_Acc[1]之值調整伺服馬達1之PWM訊號 |
| 65 | Null | Do noting |
| 66 | SetPWM\_y(def\_Acc[2]) | 依照def\_Acc[1]之值調整伺服馬達2之PWM訊號 |

4.3 作業系統C code合成

**void main()**

**{**

**……**

**}**

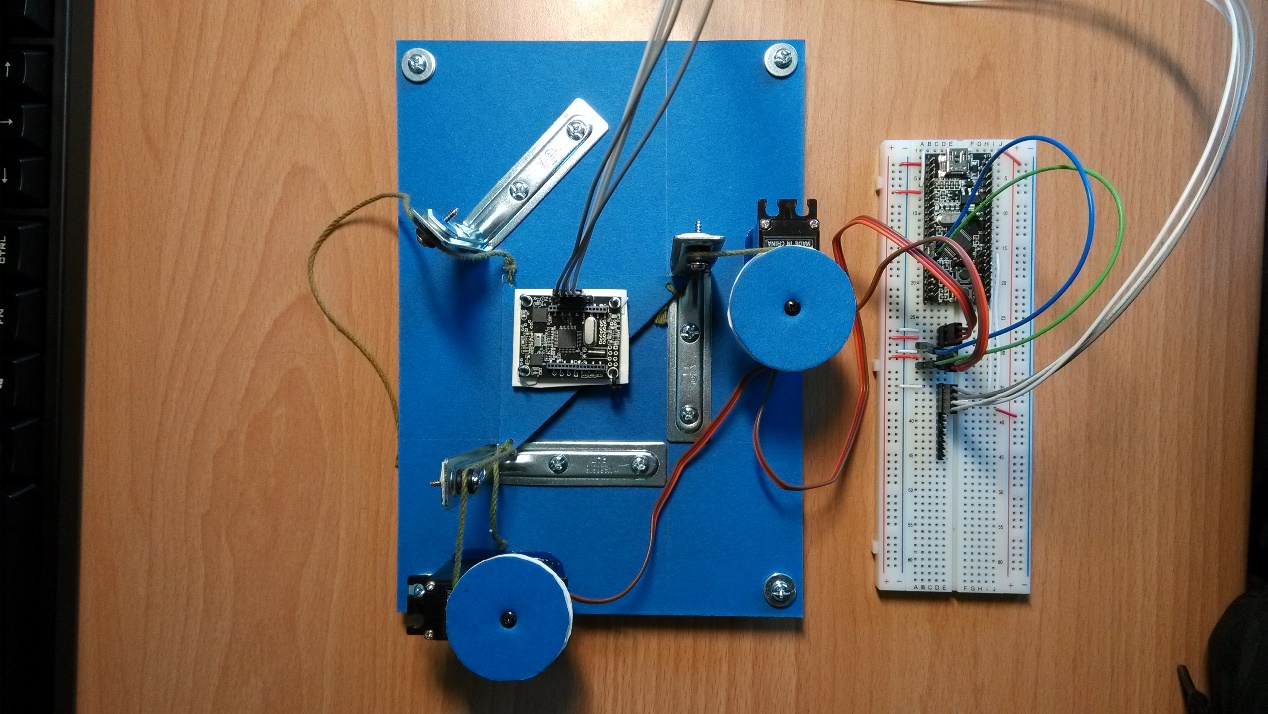
…

狀態轉移模擬驗證…

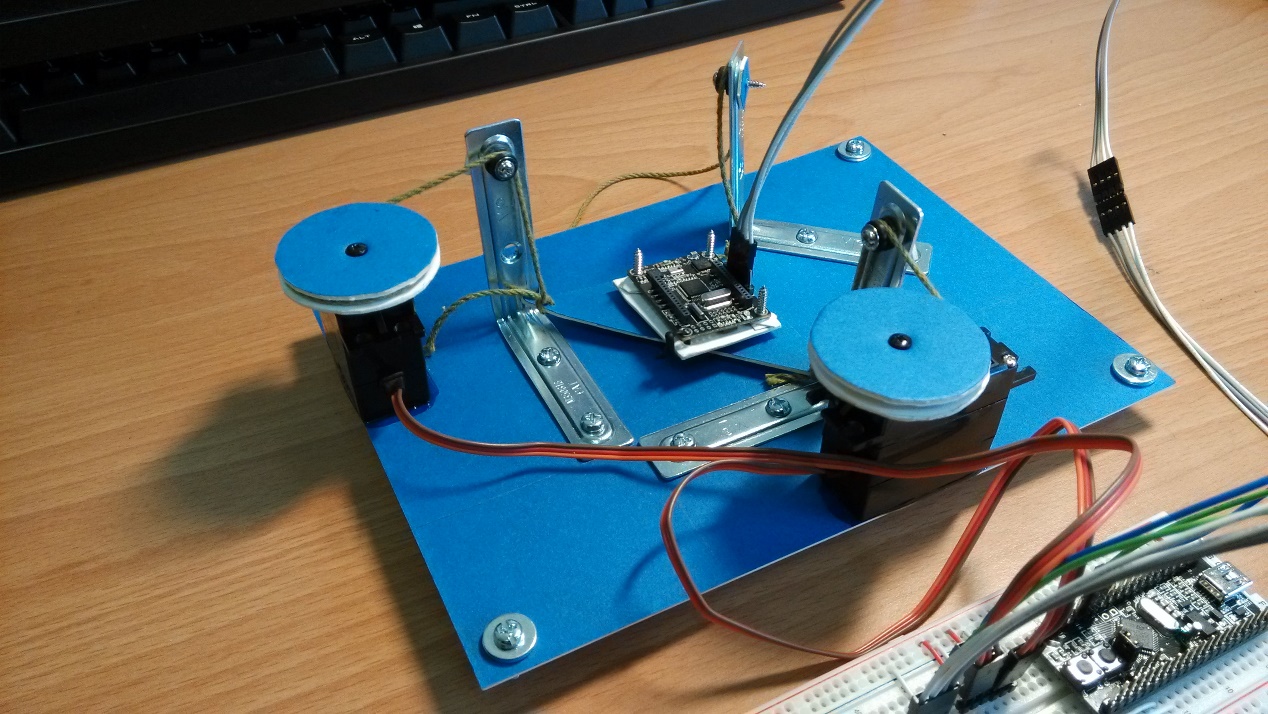
**4.4 底層函式設計與實作**

…….

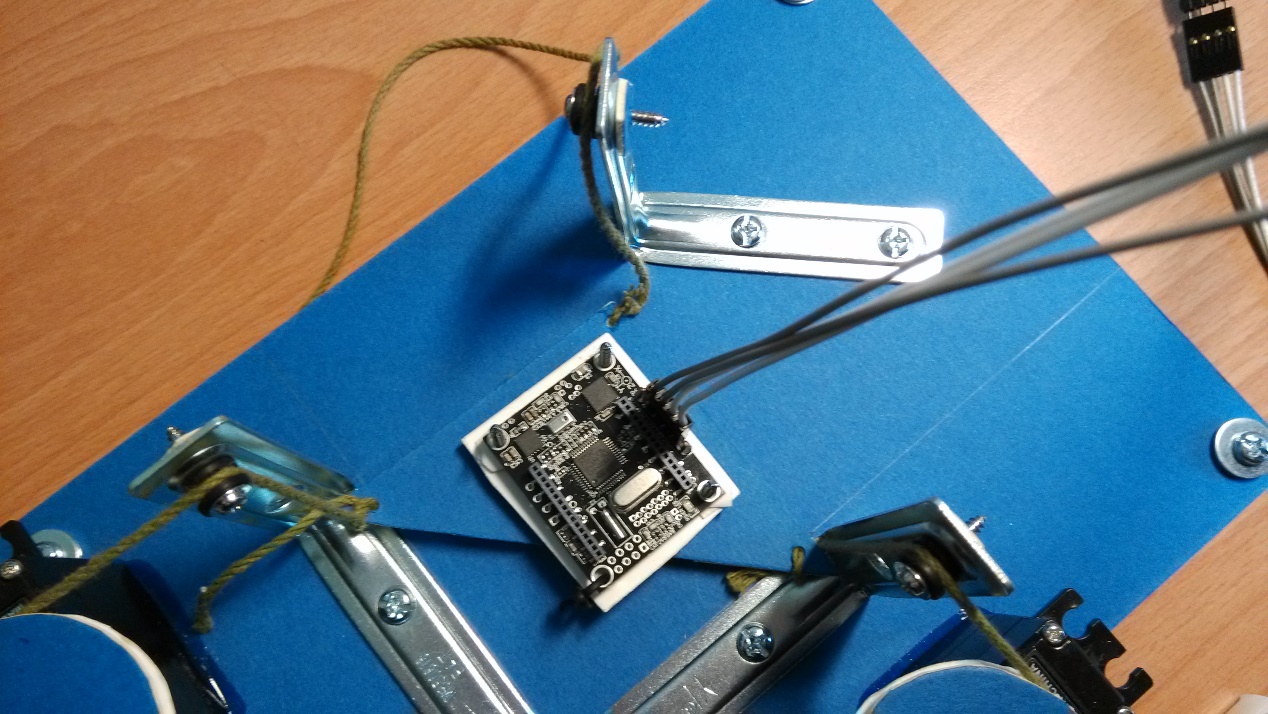
1. 系統整合驗證
   1. Demo平台及SIOC



* 1. Demo平台(側視圖)



* 1. 升降平台(Smart Sensor)



* 1. 伺服馬達



* 1. Demo影片

附件：Demo.mp4

1. 心得

製作平衡儀作為期末專題題目，是在期末結束前的一周才更改，要將Smart Sensor的使用方法摸透、建立伺服馬達以及放置Smart Sensor的平台之間的運動關係，以及製作出測試用的demo平台，在僅僅一周的時間內其實無法將每個細部都研究徹底，因此在程式編寫的許多地方都做了妥協，例如直接使用AT Command的方式去向Smart Sensor要加速度計的數值，以遞增遞減來處理pwm訊號的補償等等，花較多時間去構思的部分，反而是demo平台的架設，由於手邊擁有的伺服馬達最大旋轉角度為270度，而其中較能平順的工作範圍僅有180度，要將旋轉的路徑(半圓)轉化為上升及下降，並不是一件簡單的工作，最初的想法是在伺服馬達的前端裝設一十字型的框架，並將馬達設置於平台底下，透過十字形框架的轉動來推動平台，但實際測試後發現能改變的高度太小，不易於觀察，第二個想法為設計一連桿機構來達成推動平台的工作，但由於連桿機構實體不易製作而作罷，往後又構思了許多結構，但也因為諸多的實行困難而放棄，最終使用定滑輪的方式，改為拉升平台才解決伺服馬達運動以及平台升降間連動方式的問題，雖然在實際製作後，運行上還有許多問題，但已能藉由外力來解決，並測試程式的正確性。

本專題雖然在匆忙之下完成，但也讓我獲益良多，透過實作，讓我看見了嵌入式系統在程式以外會遇上的困難，並提共了我許多的想法，在往後應用這項技術時，應能更適切的去設計軟體及硬體。