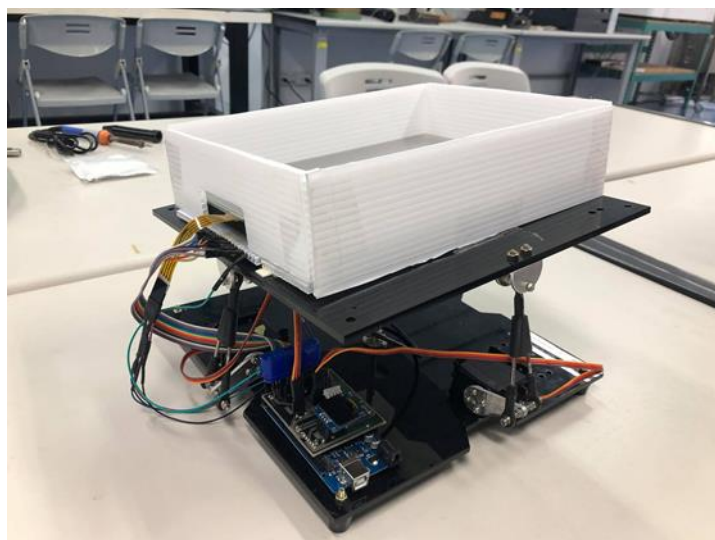


臺北市立大安高級工業職業學校專題製作競賽  
「專題組」作品說明書

類別:電機與電子群

作品名稱:平衡球

關鍵詞:Arduino、PID 控制、觸控原理



學生 組長:陳庠蓁

組員:田青艷

組員:洪珮倫

組員:毛世鑫

指導老師:林家德老師

中華民國 110 年 1 月

## 目錄

圖目錄 .....	ii
表目錄 .....	iii
壹、摘要 .....	1
貳、研究動機 .....	1
參、主題與課程之相關性與教學單元之說明 .....	2
一、外殼設計 .....	2
二、電路板製作 .....	2
三、程式撰寫 .....	3
肆、研究方法 .....	4
一、研究流程 .....	4
二、使用材料及工具 .....	5
(一)、硬體介紹 .....	5
(二)、軟體介紹 .....	8
(三)、機構原理 .....	9
伍、研究結果 .....	14
一、操作模式 .....	14
(一)、主流程圖 .....	14
(二)、Mode0 自動平衡模式 .....	15
(三)、Mode1 參數調整模式 .....	16
(四)、Mode2 規律繞方/繞圓模式 .....	17
(五)、Mode3 追點遊戲模式 .....	19
二、成果展示 .....	20
陸、討論 .....	21
柒、結論 .....	22
捌、參考資料及其它 .....	22

## 圖目錄

圖 1	漢堡機器人送餐.....	1
圖 2	機座底板設計圖.....	2
圖 3	電路板設計圖.....	2
圖 4	ARDUINO 程式.....	3
圖 5	專題流程圖.....	4
圖 6	電阻屏觸控介紹圖.....	6
圖 7	MEGA2560 重要接腳介紹.....	6
圖 8	PS2 接腳介紹.....	7
圖 9	伺服馬達零件外觀圖及配件.....	7
圖 10	OLED 顯示螢幕外觀.....	8
圖 11	ARDUINO 標誌.....	8
圖 12	ALTIUM DESIGNER 標誌.....	9
圖 13	ONSHAPE 標誌.....	9
圖 14	PID 控制器的方塊圖.....	10
圖 15	PID 演算法公式.....	10
圖 16	不同比例增益 KP 下，受控變數對時間的變化.....	11
圖 17	不同比例增益 KI 下，受控變數對時間的變化.....	11
圖 18	不同比例增益 KD 下，受控變數對時間的變化.....	12
圖 19	PID 人工調整方式.....	13
圖 20	齊格勒－尼科爾斯方法 PID 調整方式.....	13
圖 21	主程式流程圖.....	14
圖 22	球體平衡流程圖.....	15
圖 23	改變映射目標上下限值.....	15
圖 24	更改 KP 或 KD 參數.....	16
圖 25	更改 DX 或 DY 參數.....	16
圖 26	調整速度(SP)和次數(C)流程圖.....	17
圖 27	繞方流程圖.....	17
圖 28	繞圓流程圖.....	18
圖 29	設定可移動邊界.....	19
圖 30	追點遊戲.....	19
圖 31	遊戲分數.....	19
圖 32	MODE3 螢幕顯示畫面.....	20
圖 33	成品外觀.....	20
圖 34	MODE0 自動平衡模式.....	20
圖 35	MODE1 參數調整模式.....	20
圖 36	MODE3 追點遊戲模式.....	20
圖 37	MODE2 規律繞方模式.....	21
圖 38	MODE2 規律繞圓模式.....	21

## 表目錄

表 1 材料與設備.....	5
表 2 LED 電氣特性.....	7

## 壹、摘要

在科技快速發展的今日，我們發現送餐機器人在運送過程中因晃動而讓食物翻倒的案例不計其數。秉持著這種理念，我們擬定了本次的專題計劃。我們模擬送餐機器人在運送過程中可能會遇到的問題，利用電阻屏模擬機器人手上的盤子，而球體就是食物。在專題中我們使用 Mega2560 為我們的開發板核心，利用 Arduino 軟體編寫程式碼並設定四種模式，讓四種模式的畫面顯示在 OLED 上。藉由 PS2 搖桿控制球體位置、切換模式、三個參數、圖形等，再利用 Onshape 軟體繪製機座的底板與上板，將製作好的圖形送至雷射切割機，並用堅固的壓克力板作為材料。最後因成品整體美觀需要，我們運用 Altium Designer 軟體繪製並印刷電路板，使線路整潔且乾淨。

## 貳、研究動機

現今科技越來越發達，許多人力漸漸被機器所取代，市面上也出現許多機器人導覽、送餐、聊天等功能，這些科技演變使生活更加便利。

曾經在新聞上看到一家速食店正在測試送餐機器人，他們把餐點放在機器人的餐盤上，並設定送餐位置，客人便可從機器人的餐盤上拿取餐點享用，送餐的過程中很緩慢，深怕餐點不小心掉落以及打翻，我們為了避免此類的意外事故發生，利用球體模擬餐點在餐盤上的晃動程度，使之達到平衡，便可加快送餐速度。



圖 1 漢堡機器人送餐

## 參、主題與課程之相關性與教學單元之說明

### 一、外殼設計

在高三的專題製作實習課程中，我們學習使用 3D 列印繪圖工具、3D 列印機及雷射切割機的操作方法，利用這些技術可以製作出符合我們機構的需求。一開始製作外殼時，因考量到體積大、材質堅硬、成本等因素，我們選用雷射切割來解決以上種種問題，將繪製的 DXF 轉檔後，使用壓克力板切割成型。

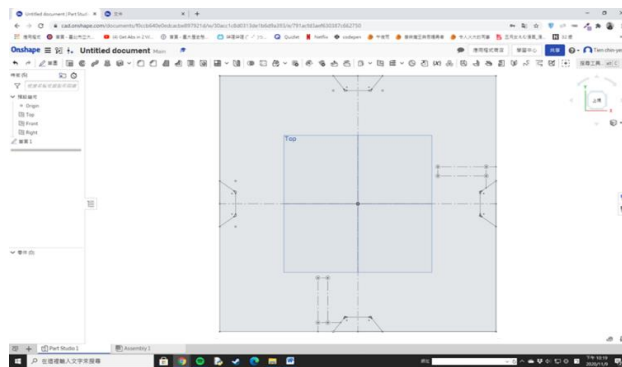


圖 2 機座底板設計圖

### 二、電路板製作

在高三微處理機實習課程中，我們使用 Altium Designer 繪製電路圖、佈線，進行曝光、顯影、蝕刻再到鑽孔焊接，將所有電路整合至電路板上，避免因線路纏繞打結而接觸不良並且可增加整體美觀。

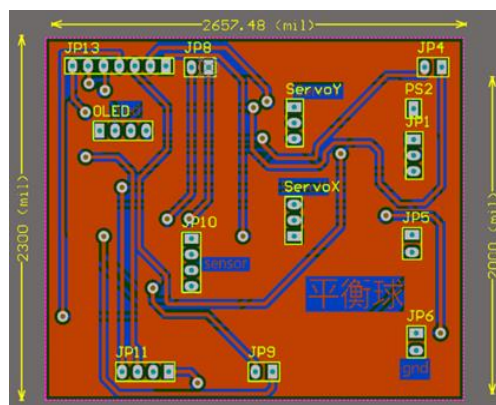
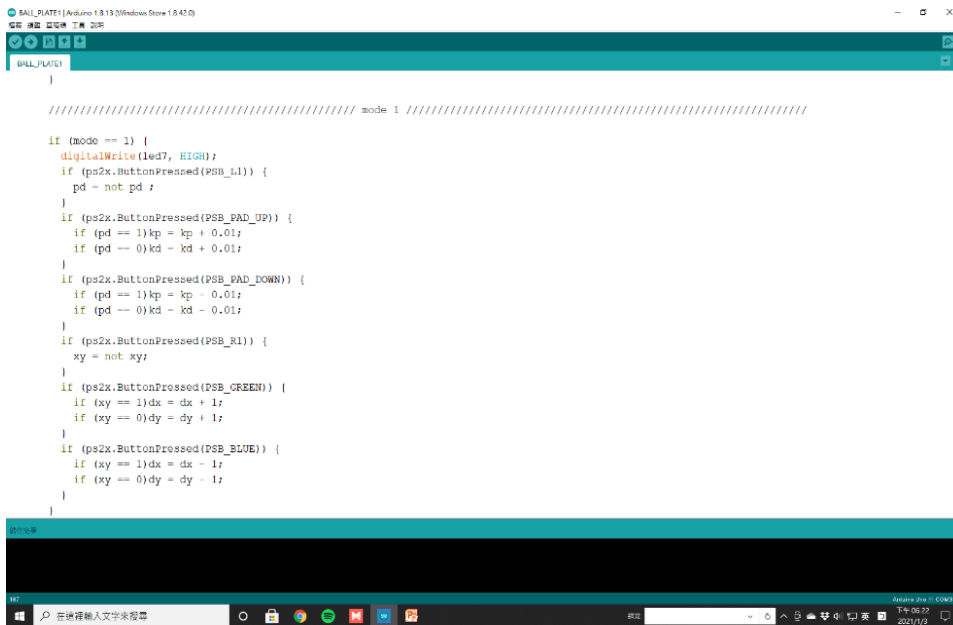


圖 3 電路板設計圖

### 三、程式撰寫

在高二的數位邏輯實習課程中，我們從 CPLD 程式撰寫中學到一些不同的語法表示，以及程式的邏輯推斷，藉此對程式有了初步認識。到了高二的微處理機實習，我們開始使用 Arduino 撰寫，應用簡單的演算法、邏輯判斷，自行設計電路運作方式。這次的專題也是使用 Arduino 來調整 PID 參數以及撰寫 PS2 搖桿不同的模式功能。



```
//////////////////////////////////////////////////////////////////////////////// mode 1 ////////////////////////////////////////////////////////////////////////////  
  
if (mode == 1) {  
  digitalWrite(LED7, HIGH);  
  if (ps2x.ButtonPressed(PSB_L1)) {  
    pd = not pd ;  
  }  
  if (ps2x.ButtonPressed(PSB_PAD_UP)) {  
    if (pd == 1)kp = kp + 0.01;  
    if (pd == 0)kd = kd + 0.01;  
  }  
  if (ps2x.ButtonPressed(PSB_PAD_DOWN)) {  
    if (pd == 1)kp = kp - 0.01;  
    if (pd == 0)kd = kd - 0.01;  
  }  
  if (ps2x.ButtonPressed(PSB_R1)) {  
    xy = not xy;  
  }  
  if (ps2x.ButtonPressed(PSB_GREEN)) {  
    if (xy == 1)dx = dx + 1;  
    if (xy == 0)dy = dy + 1;  
  }  
  if (ps2x.ButtonPressed(PSB_BLUE)) {  
    if (xy == 1)dx = dx - 1;  
    if (xy == 0)dy = dy - 1;  
  }  
}
```

圖 4 Arduino 程式

## 肆、研究方法

### 一、研究流程

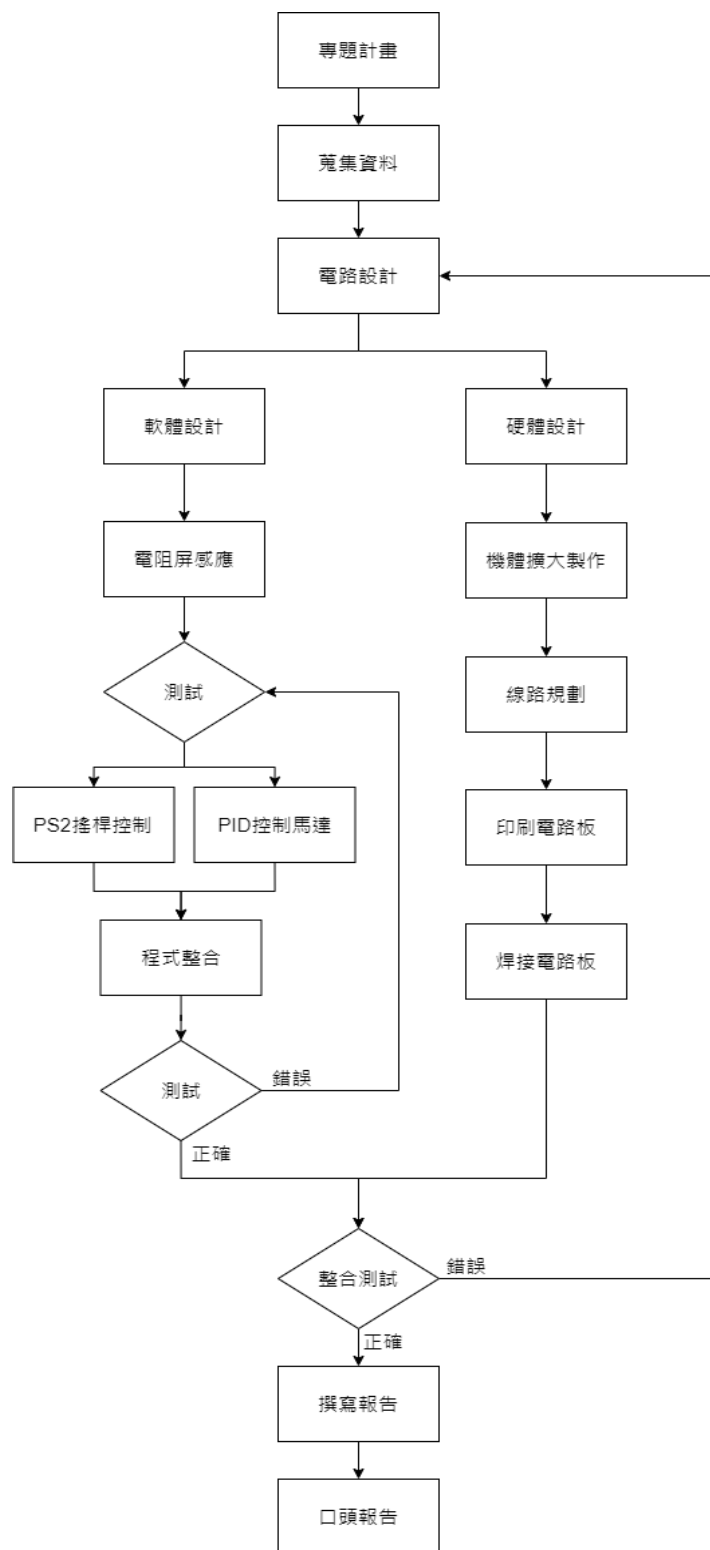


圖 5 專題流程圖



## 二、使用材料及工具

表 1 材料與設備

類別	名稱	應用說明
硬體	電阻屏	輸出球體位置
硬體	Arduino Mega2560	撰寫、驅動程式
硬體	PS2 搖桿	控制球體方向
硬體	伺服馬達	操縱平台角度
硬體	OLED 液晶顯示模組	顯示功能模式
軟體	Arduino	撰寫程式
軟體	Altium Designer	繪製電路圖
軟體	Onshape	繪製外殼

### (一)、硬體介紹

#### 1、電阻屏

##### (1) 四線式電阻屏

電阻式觸摸屏是一種傳感器，基本上是薄膜加上玻璃的結構，薄膜和玻璃相鄰的一面上均塗有 ITO（納米銦錫金屬氧化物）塗層，ITO 具有很好的導電性和透明性。當觸摸操作時，薄膜下層的 ITO 會接觸到上層的 ITO，經由感應器傳出相應的電信號，經過轉換電路送到處理器，通過運算轉化為屏幕上的 X、Y 電壓值，而完成點選的動作，並呈現在屏幕上。很多 LCD 模塊都採用了電阻式觸摸屏，這種屏幕可以用四線、五線、七線或八線來產生屏幕偏置電壓，同時讀回觸摸點的電壓。

##### (2) 觸控原理

電阻式觸控面板的主要部分是一塊與顯示器表面配合非常好的電阻薄膜屏。它是一種多層的複合薄膜，由一層玻璃或有機玻璃作為基層，表面鍍有一層透明的導電層，上面再蓋有一層外表硬化處理、光滑防刮的塑膠層。它的內表面也鍍有一層透明導電層，在兩層導電層之間有許多細小（小於千分之一英寸）的透明隔離點把它們隔開絕緣。當手指觸摸螢幕時，平常絕緣的兩層導電層在觸摸點位置就接觸在一起，控制器偵測到接通後，其中一面導電層接通 X 軸方向的 5 伏特均勻電壓場，另一導電層將接觸點的電壓引至控制卡進行 A/D 轉換，得到電壓值後與 5 伏特相比即可獲得觸摸點的 X 軸坐標，同理得出 Y 軸的坐標，這就是所有電阻技術觸控面板共同的基本原理。

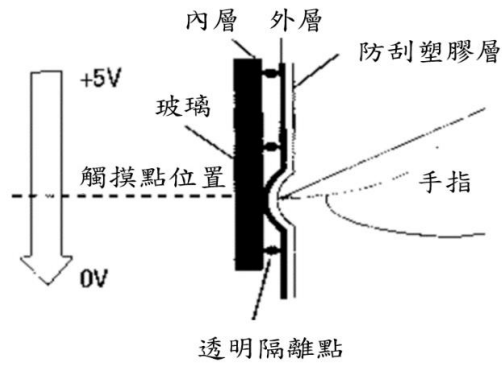


圖 6 電阻屏觸控介紹圖

## 2、 Arduino Mega2560

Arduino Mega2560 是採用 USB 介面的核心電路板。具有 54 個數字輸入輸出，適合需要大量 IO 介面的設計。處理器核心是 ATmega2560，具有 54 個數字輸入/輸出口、16 個模擬輸入(PWM)、4 個 UART 介面、一個 16MHz 晶體振盪器、一個 USB 口、一個電源插座、一個 ICSP header 和一個重置按鈕。板上有支援一個主控板的所有資源。Arduino Mega2560 也能相容為 Arduino Uno 設計的擴充套件板。可以自動選擇 3 種供電方式：外部直流電源通過電源插座供電；電池連線電源聯結器的 GND 和 VIN 引腳；USB 介面直流供電。

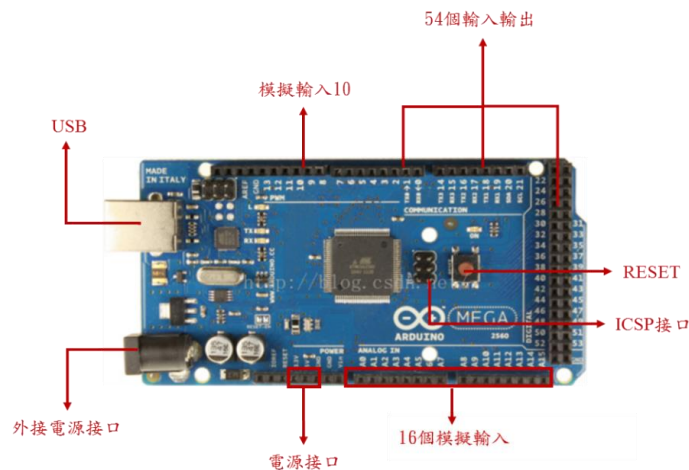


圖 7 Mega2560 重要接腳介紹

## 3、 PS2 搖桿

PS2 控制器採用特殊的介面接頭，沒辦法簡單地和 Arduino 控制板相連，需另外與 PS2 搖桿轉接頭做連接並且利用 Arduino 程式庫來做接腳判定。

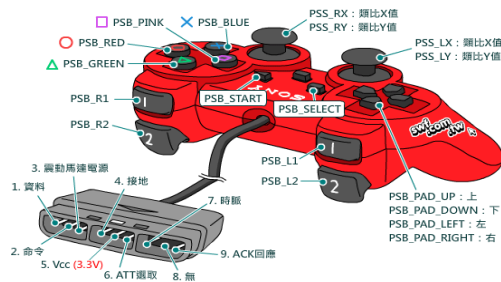


圖 8 PS2 接腳介紹

#### 4、 伺服馬達

應用於電機的伺服控制，將感測器裝在電機與控制對象機器上，偵測結果會返回伺服放大器與指令值做比較。由此可知，因為伺服電機是以回饋訊號控制，因此伺服電機的動作特性是進行位置定位控制和動作速度控制。它的主要特點是轉速可以精確控制，速度控制範圍廣，輸出功率大且效率也高。



圖 9 伺服馬達零件外觀圖及配件

#### 5、 OLED 液晶屏顯示模組

表 2 LED 電氣特性

解析度	128*64
可視角度	>160°
支援眾多控制晶片	Arduino、51 系列、MSP430 系列、STM32、CSR 晶片等
功耗程度	全屏點亮時 0.08W，正常全屏顯示漢字 0.06W (遠低於 TFT、LCD 等技術)
寬電壓支援	無需任何修改，直接支援 3V~5V 直流
工作溫度	-40°C~70°C
模組體積(長寬厚)	27.0MM*27.0MM*4.1MM
IO 口佔用少	採用 I2C 通信方式，最多只要 4 個 IO 口就能驅動
驅動晶片	SSD1306

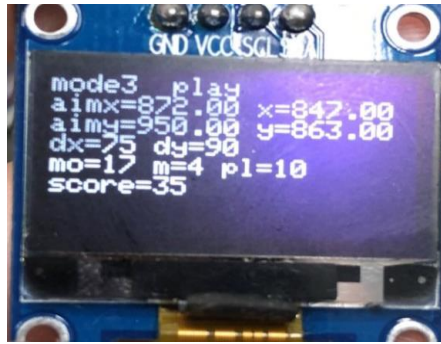


圖 10 OLED 顯示螢幕外觀

## (二)、軟體介紹

### 1、 Arduino

Arduino 包括一個硬體平台——Arduino Board，和一個開發工具——Arduino IDE。兩者都是開放的，既可以獲得 Arduino 開發板的電路圖，也可以獲得 Arduino IDE 的原始碼，甚至也提供了基本的接口和 USB 轉串列模組。使用者只需要用一個 USB 線就可以連接電腦和 Arduino Board，完成編程和燒錄，不需要專門的下載器。

Arduino 使用一種簡化的專用程式語言，為 C++ 的程式語言架構；IDE 可免費下載，並開放原始碼和跨平台，極為便利。在本次專題中，我們利用 Arduino 編寫程式碼，設定 PID 三個參數值，將電阻屏測得的值分別減去設定的目標值(目標值即為中心點)，就會得到誤差值，再丟回 Arduino 進行 PID 公式的誤差運算，再將算出的值轉換成馬達的角度，控制伺服馬達使球體能夠維持在電阻屏中心點，並控制 PS2 搖桿切換 4 種模式，使球體能夠在電阻屏上繞方和繞圓或追點遊戲。



圖 11 Arduino 標誌

### 2、 Altium Designer

Altium Designer 是原 Protel 軟件開發商 Altium 公司推出的一體化的電子產品開發系統，主要運行在 Windows 操作系統。這套軟件通過把原理圖設計、電路仿真、PCB 繪製編輯、拓撲邏輯自動佈線、信號完整性分析和設計輸出等技術的完美融合，提供了全新的設計解決方案。讓大家可以輕鬆進行設計，而熟練使用這一軟件將使電路設計的質量和效率

大大提高。因為本次專題電路較為複雜，如果只使用杜邦線及麵包板接線會使成品極不美觀，而使用此軟體能將我們設計的電路成品化，使專題成果更為具體。因此我們使用此軟體繪製電路圖再印刷電路板，繪製完成後再轉檔，然後再進行曝光、顯影、蝕刻出電路板，最後再鑽孔焊接。



圖 12 Altium Designer 標誌

### 3、 Onshape

Onshape 是唯一將 CAD、數據管理、協作工具、實時分析結合在一起的產品開發平台，也是全球第一個全雲端的 3D 整合型軟體。Onshape 與 TinkerCAD 同樣是 3D 繪圖軟體，但在 TinkerCAD 上，多數的物件設計是透過像積木拖拉的方式組合，並透過 Group 以及 Ungroup 兩個動作達到組合與分離的動作。而 Onshape 則與一般工程物件的製作較為相似，是在一個平面先繪製好外型，再將外型拉長或延伸至希望的大小，有需要也可將物件切割或去除。本次專題中，我們為了擴大底座而使用此軟體，將繪製好的圖送到雷射切割機，且使用堅固壓克力板作為材料製作底板與上板。



圖 13 Onshape 標誌

### (三)、機構原理

#### 1、控制原理

PID 控制器是由比例單元 (Proportional)、積分單元 (Integral) 和微分單元 (Derivative) 組成，透過調整這三個單元的增益來調整其特性。

PID 控制器主要適用於動態特性不隨時間變化的系統。PID 控制器是一個在工業控制應用中常見的反饋迴路部件，這個控制器把收集到的數據和一個參考值進行比較，然後把這個差別用於計算新的輸入值，這個新的輸入值的目的是可以讓系統的數據達到或者保持在參考值。

PID 控制器的比例單元(P)、積分單元(I)和微分單元(D)分別對應目前誤差、過去累計誤差及未來誤差。若是不知道受控系統的特性，一般認為 PID 控制器是最適用的控制器。藉由調整 PID 控制器的三個參數，可以調整控制系統，設法滿足設計需求。控制器的響應可以用控制器對誤差的反應快慢、控制器過衝的程度及系統震盪的程度來表示。

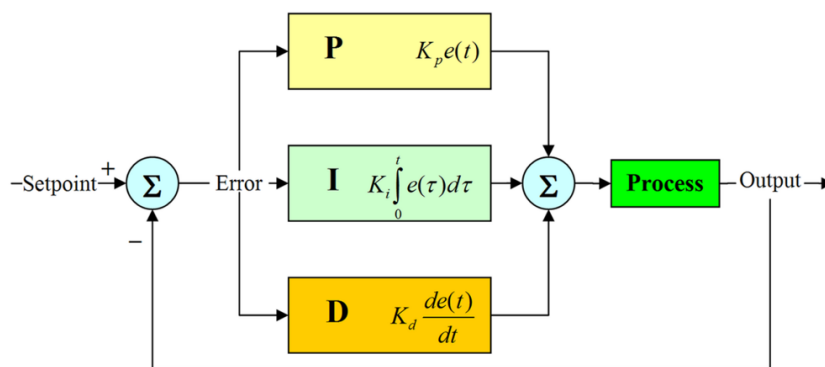


圖 14 PID 控制器的方塊圖

$$u(t) = MV(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t)$$

圖 15 PID 演算法公式

### (1) 比例控制 (proportion)

比例控制考慮當前誤差，將誤差值和一個常數  $K_p$  (表示比例) 相乘。比例控制只成立於誤差不為零時，當誤差越大，比例控制輸出值就越大，反之誤差小，輸出就變小。當誤差是 0 的時候，控制器的輸出也是 0。比例控制不具有消除干擾的能力。若要讓受控輸出非零的數值，就要需要有一個穩態誤差或偏移量。

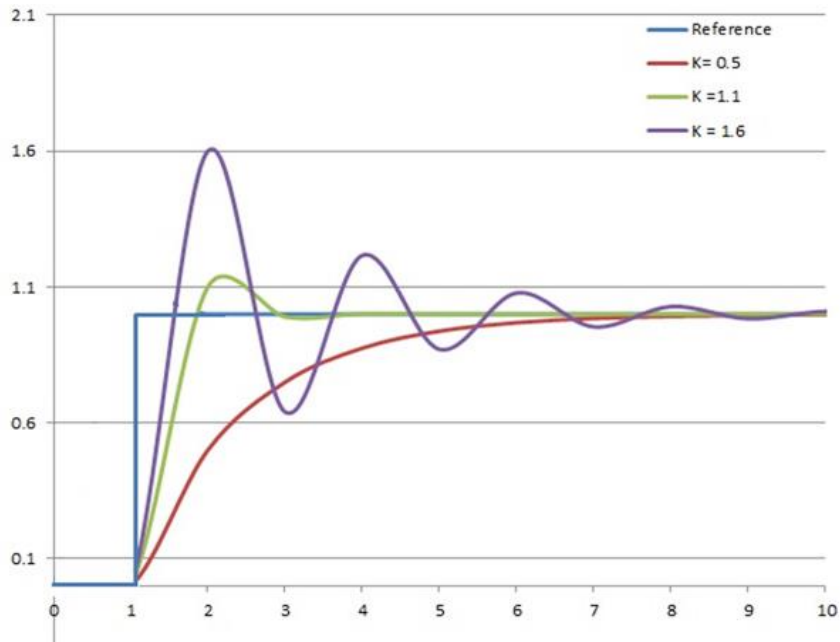


圖 16 不同比例增益  $K_p$  下，受控變數對時間的變化

## (2) 積分控制 (integral)

積分控制考慮過去的誤差，將過去一段時間內的誤差和乘上參數  $K_i$  所得出來的結果。如果只有單純的比例控制無法消除干擾及誤差，系統會來回震盪，無法達到平衡點。但在加入積分控制後，系統的誤差將會慢慢地減少，最後將會穩定下來。積分參數的值越大，系統將會更快達到平衡點，但參數過大時會導致過衝的現象。

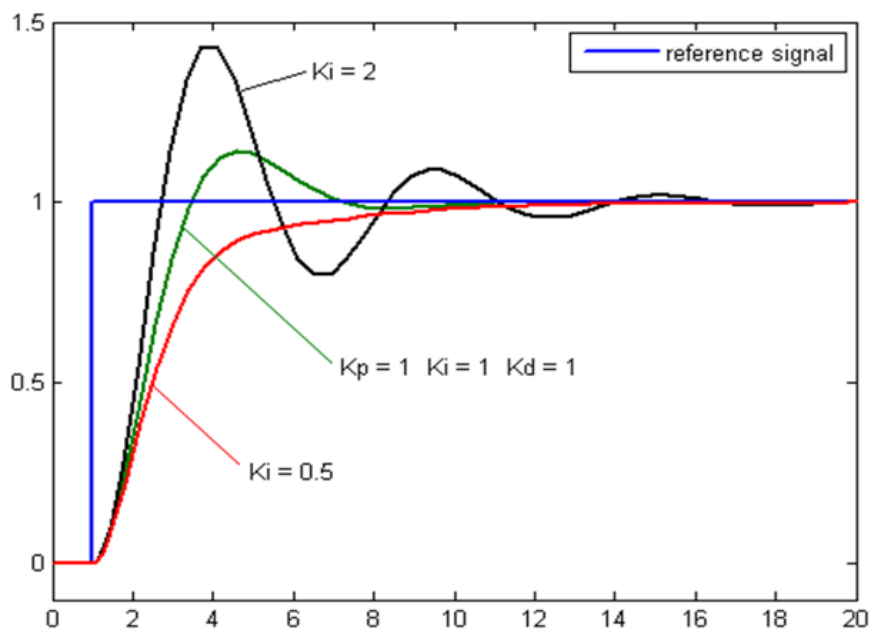


圖 17 不同比例增益  $K_i$  下，受控變數對時間的變化



### (3) 微分控制 (differential)

微分控制考慮未來的誤差，計算誤差的一階導函數，乘上  $K_d$  所得。此控制項會對系統的改變作出反應，結果越大，系統將可以做出更快的反應，讓整體更快達到平衡點，但在慢速的系統中無法明顯的看出差異。

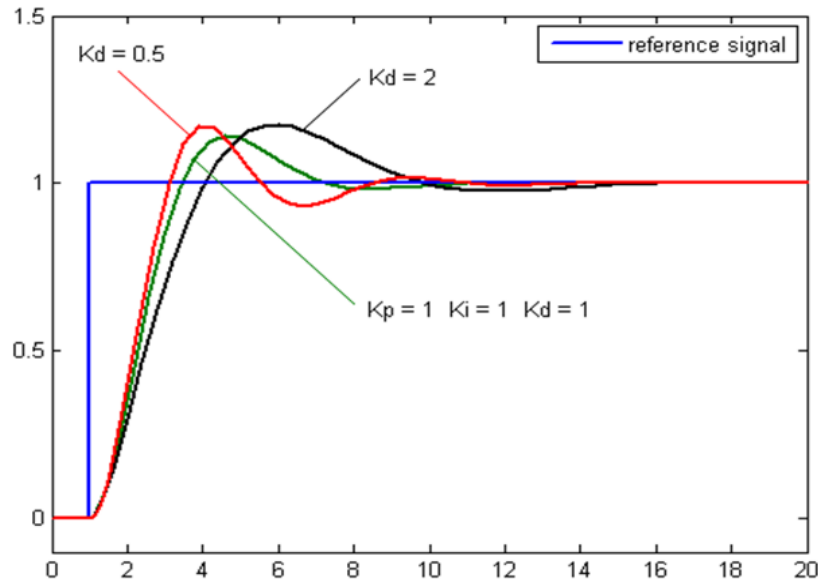


圖 18 不同比例增益  $K_d$  下，受控變數對時間的變化

## 2、參數調整

PID 的參數調試是指透過調整控制參數讓系統達到最佳的控制效果，而穩定性（不會有發散性的震盪）是首要條件。PID 控制器的參數若適當調試，會有很好的效果。一般初始設計常會需要不斷的電腦模擬，並且修改參數，一直達到理想的性能為止。因此 PID 控制器的參數未挑選妥當，其控制器輸出可能是不穩定的，也就是其輸出發散，過程中可能有震盪，也可能沒有震盪。下面我們介紹常見幾種調整參數的方法。

### (1) 人工調整

若需在系統仍有負載的情形進行調試，有一種作法是先將  $K_i$  及  $K_d$  設為零，增加  $K_p$  一直到迴路輸出震盪為止，之後再將  $K_p$  設定為「1/4 振幅衰減」（使系統第二次過衝量是第一次的 1/4）增益的一半，然後增加  $K_i$  直到一定時間後的穩態誤差可以被修正為止。不過  $K_i$  可能會造成不穩定，若有需要可以增加  $K_d$ ，並確認在負載變動後迴路可以夠快的回到其設定值，但是  $K_d$  太大會造成響應太快及過衝。一般而言反應快速的 PID 會有輕微的過衝，只是有些系統不允許過衝。因此需要將回授系統調整為過阻尼系統，而  $K_p$  比造成震盪  $K_p$  的一半還要小很多。



調整方式	(on) 上升時間	超調量	安定時間	穩態誤差	穩定性 <sup>[10]</sup>
↑ $K_p$	減少 ↓	增加 ↑	小幅增加 ↗	減少 ↓	變差 ↓
↑ $K_i$	小幅減少 ↘	增加 ↑	增加 ↑	大幅減少 ↓↓	變差 ↓
↑ $K_d$	小幅減少 ↘	減少 ↓	減少 ↓	變動不大 →	變好 ↑

圖 19 PID 人工調整方式

(2) 齊格勒－尼科爾斯方法

齊格勒－尼科爾斯方法是另一種啟發式的調試方式，這個方法由 John G. Ziegler 和 Nathaniel B. Nichols 在 1940 年代導入，一開始也是將  $K_i$  及  $K_d$  設定為零，增加比例增益直到系統開始等振幅振盪為止，當時的增益稱為  $K_u$ ，而振盪週期為  $P_u$ ，可用以下的方式計算增益：

控制器種類	$K_p$	$K_i$	$K_d$
<i>P</i>	$0.50K_u$	-	-
<i>PI</i>	$0.45K_u$	$1.2K_p/P_u$	-
<i>PID</i>	$0.60K_u$	$2K_p/P_u$	$K_p P_u/8$

圖 20 齊格勒－尼科爾斯方法 PID 調整方式

(3) PID 調試軟體

大部份現代的工業設備不再用上述人工計算的方式調試，而是用 PID 調試及最佳化軟體來達到一致的效果。軟體會收集資料，再建立模型，並提供最佳的調試結果，有些軟體甚至可以用參考命令的變化來進行調試。

數學的 PID 調試會將脈衝加入系統，再用受控系統的頻率響應來設計 PID 的參數。若是響應時間需要數分鐘的系統，就會建議用數學 PID 調試。因為用人工調整法可能要花好幾天才能找到可讓系統穩定的參數。最佳解不太容易找到，有些數位的迴路控制器有自我調試的程序，利用微小的參考命令來計算最佳的調試值。但也有其他調試的公式，依不同的性能準則所產生。許多有專利的公式已嵌入在 PID 調試軟體及硬體模組中。一些先進的 PID 調試軟體也可以在動態的情況下用演算法調整 PID 迴路，這類軟體會先將程序建模，給微擾量，再根據響應計算參數。

## 伍、研究結果

### 一、操作模式

#### (一)、主流程圖

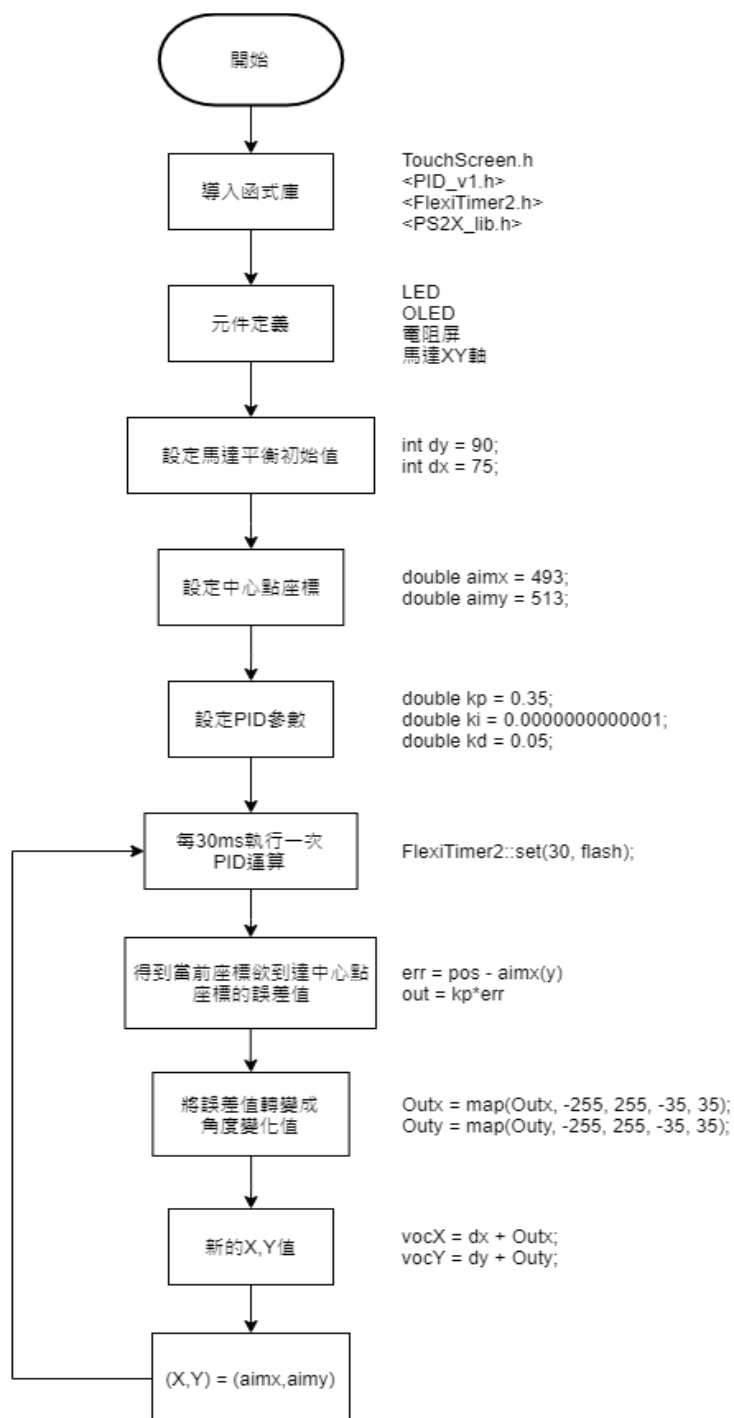


圖 21 主程式流程圖

## (二)、Mode0 自動平衡模式

- 1、 使用搖桿(PSS\_LX, PSS\_LY)改變目標值(aim)，不論如何移動球的位置，球都會回到目標值點上並平衡，若球移動到設定範圍，該範圍對應的 LED 燈會亮。(圖 22)
- 2、 若按下 PSB\_RED，目標值會回到原始設定值，按下 PSB\_GREEN 和 PSB\_BLUE 可改變映射目標的上下限值。(圖 23)

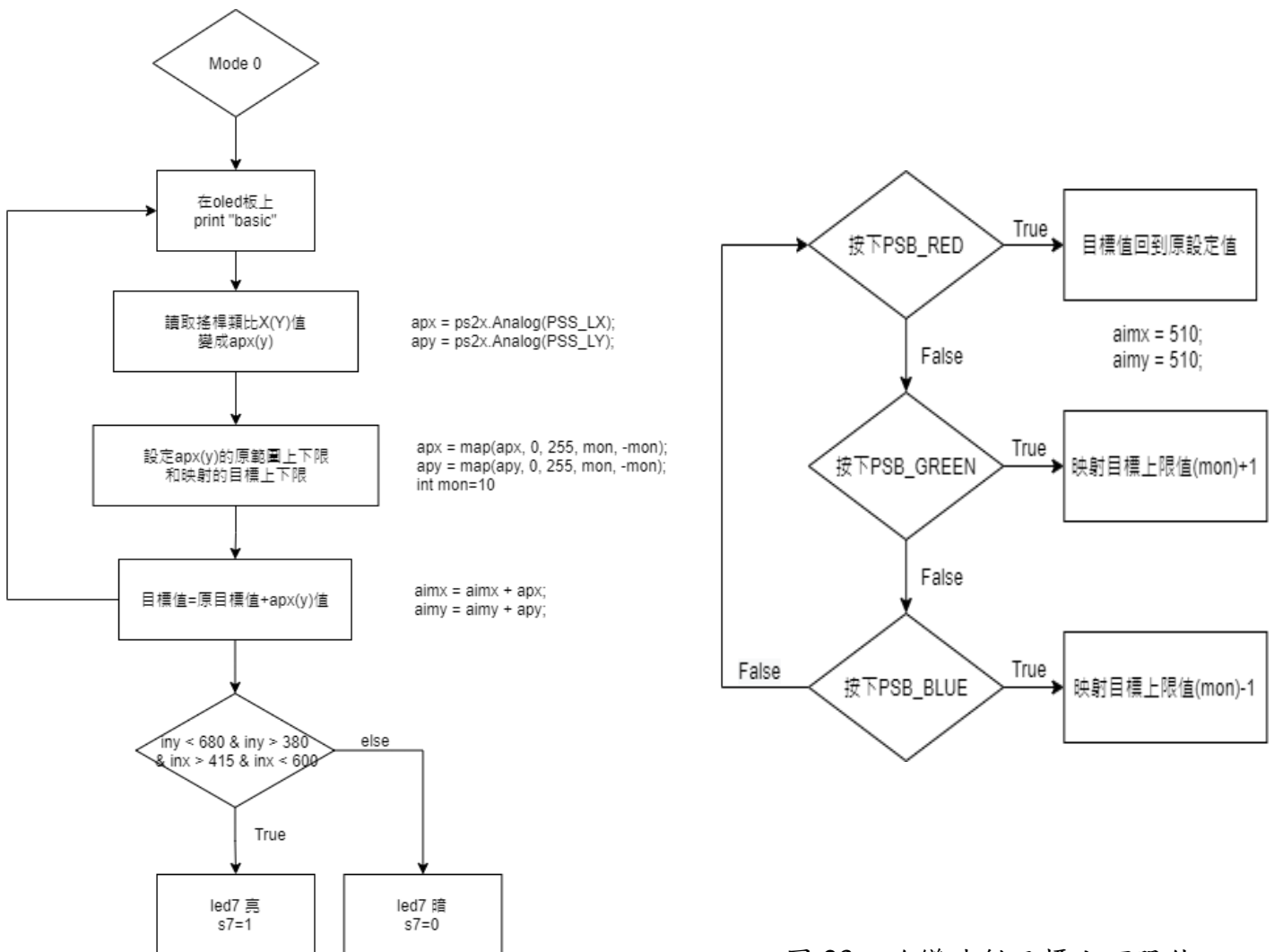


圖 23 改變映射目標上下限值

圖 22 球體平衡流程圖

### (三)、Mode1 參數調整模式

- 1、 使用 PSB\_PAD\_UP 和 PSB\_PAD\_DOWN 微調 PID 的  $kp$  和  $kd$  的參數(預設  $kp = 0.35$ 、 $kd = 0.05$ )，透過 PSB\_L1 決定要改  $kp$  或  $kd$ 。(圖 24)
- 2、 使用 PSB\_GREEN 和 PSB\_BLUE 調整馬達的  $dx$  和  $dy$  的參數(預設  $dx = 75$ 、 $dy = 90$ )，透過 PSB\_R1 決定要改  $dx$  或  $dy$ 。(圖 25)

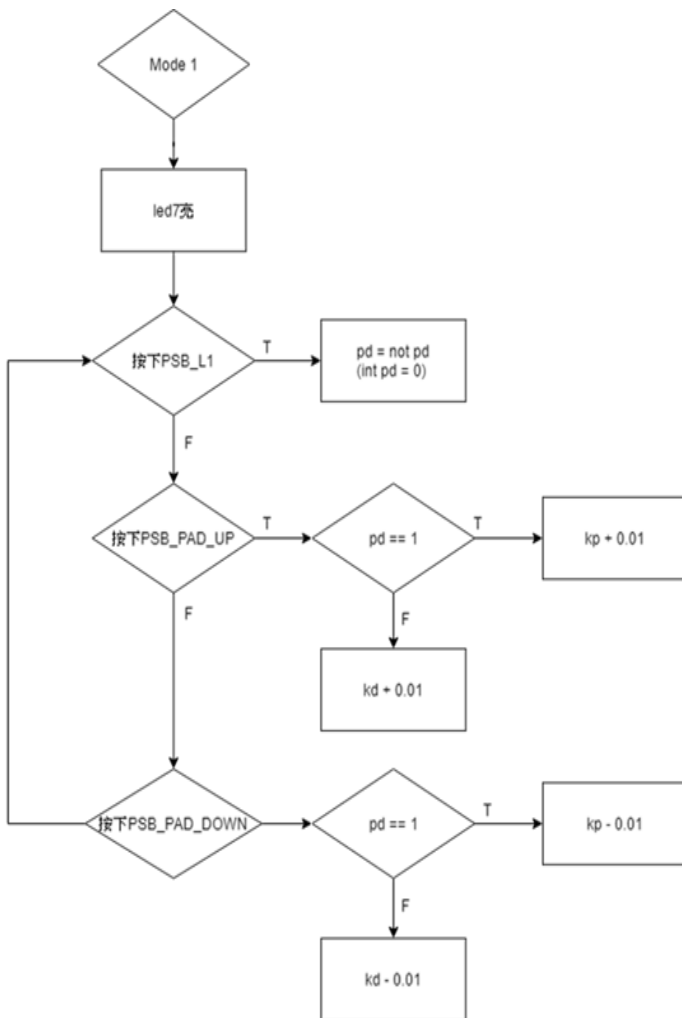


圖 24 更改  $kp$  或  $kd$  參數

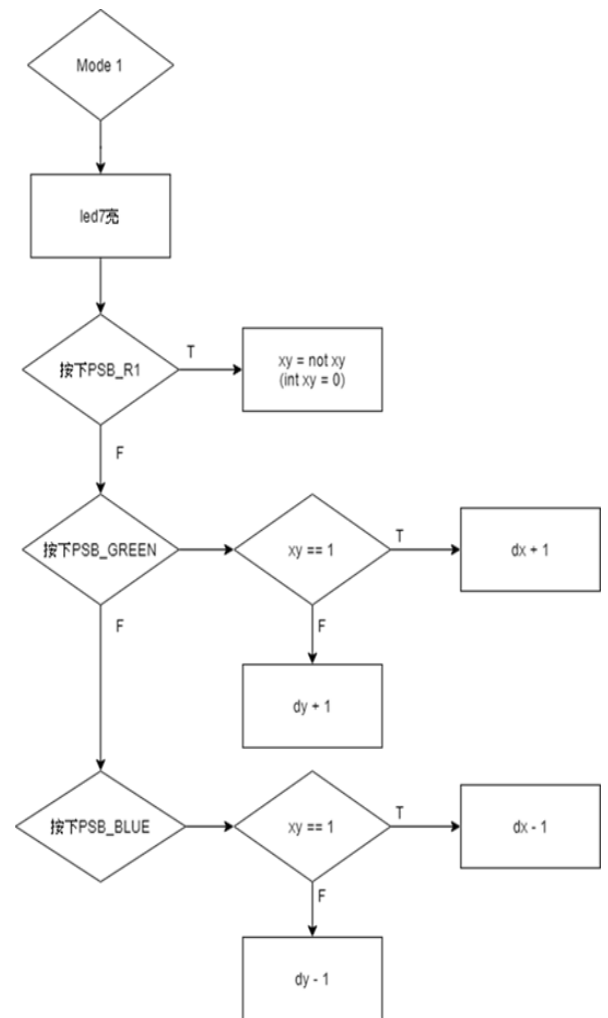


圖 25 更改  $dx$  或  $dy$  參數

(四)、Mode2 規律繞方/繞圓模式

- 1、 使用 PSB\_PAD\_UP 和 PSB\_PAD\_DOWN 來調整速度(sp)和次數(c)，我們預設  $sp = 5$ 、 $c = 6$ ，透過 PSB\_L1 可以決定要改 sp 或 c。(圖 26)
- 2、 按下 PSB\_PINK，開始規律繞方程式，運用在四個目標點間的移動，達到規律繞方的成果，在執行次數/2 之後，球回到原始目標值。(圖 27)
- 3、 按下 PSB\_RED，開始規律繞圓程式，運用圓的公式計算座標，達到規律繞圓的成果，若按下 PSB\_BLUE，結束循環，球回到原始目標值。(圖 28)

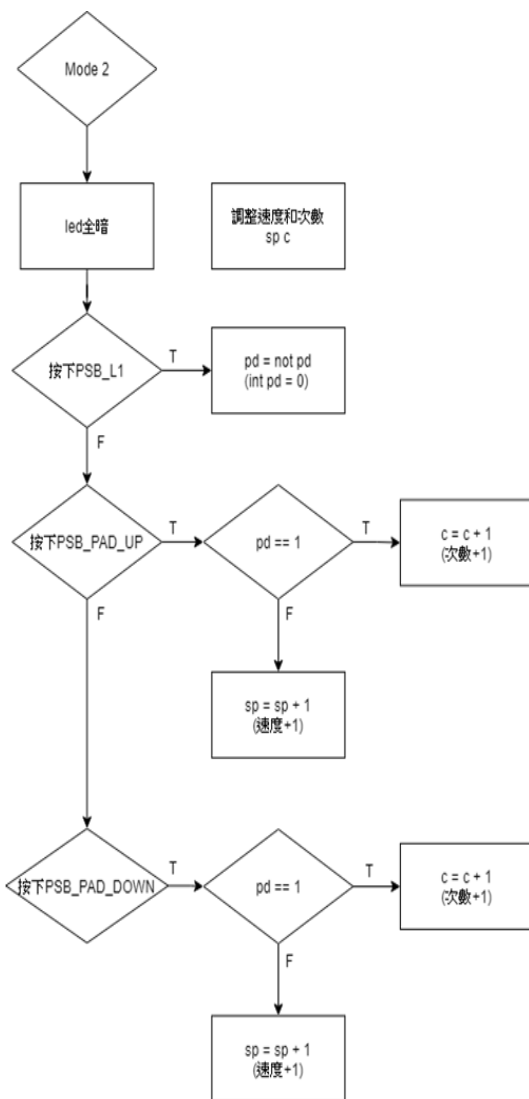


圖 26 調整速度(sp)和次數(c)流程圖

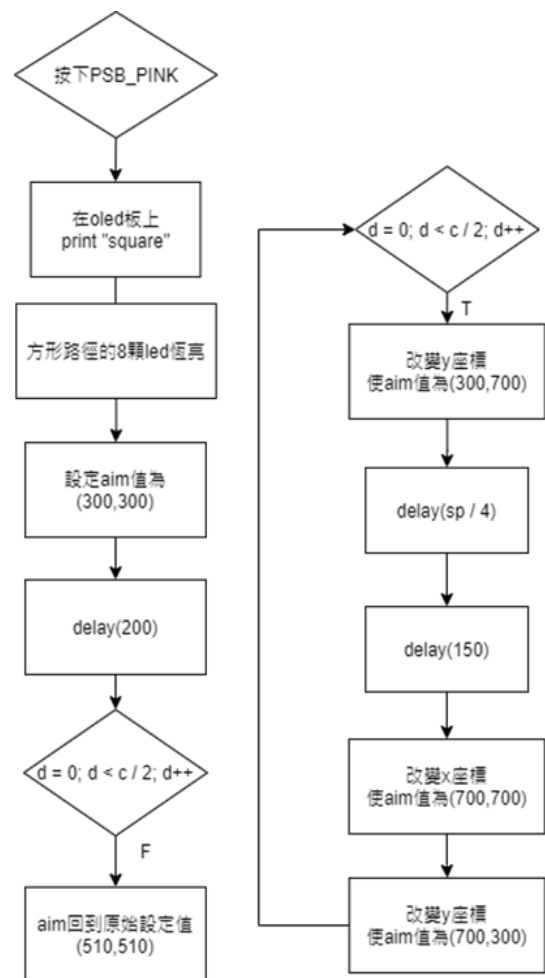


圖 27 繞方流程圖

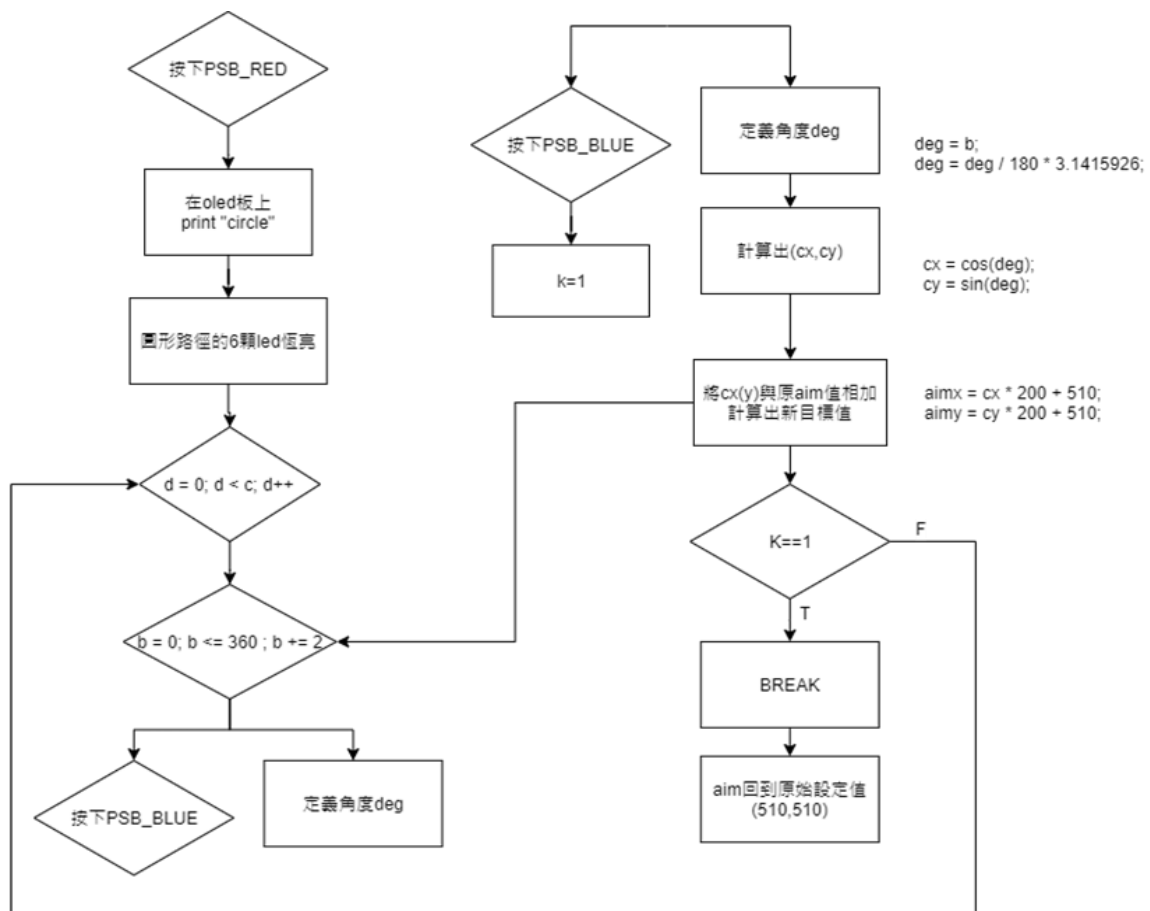


圖 28 繞圓流程圖

(五)、Mode3 追點遊戲模式

- 1、 透過用搖桿改變目標值，球跟著目標值移動的模式，製造出用搖桿來控制球位置的成果。(圖 30)
- 2、 按下 PSB\_GREEN，w 變數在 1~14 中隨機取一數；按下 PSB\_RED，目標值會回到原始設定值。(圖 31)
- 3、 假設 w 變數為 13，若球移動到該參數的設定範圍，則遊戲成功，分數會顯示在 oled，w 再從 1~14 中選一數。(圖 32)

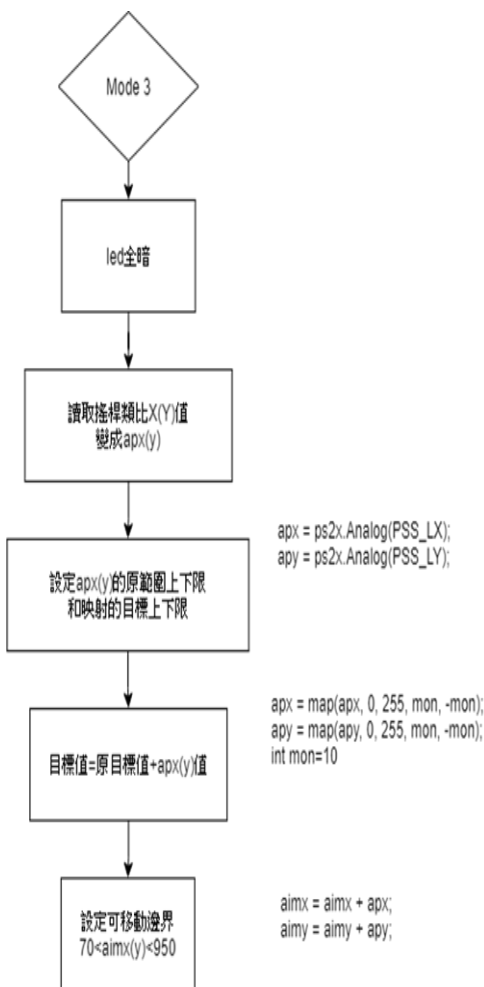


圖 29 設定可移動邊界

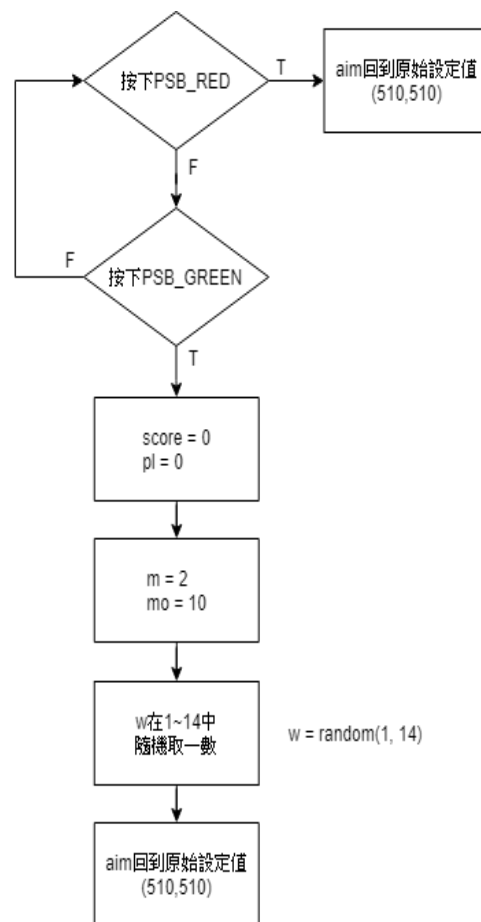


圖 30 追點遊戲

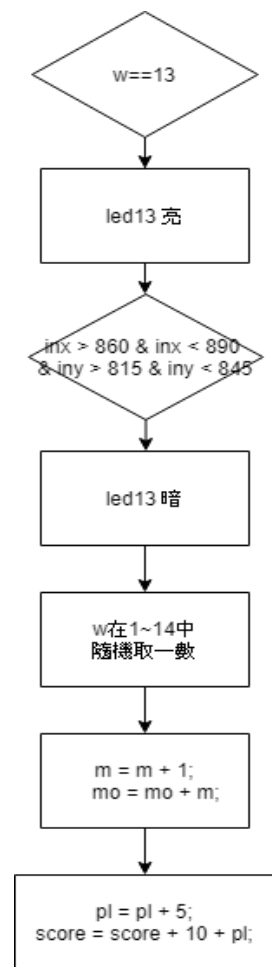


圖 31 遊戲分數



圖 32 Mode3 螢幕顯示畫面

## 二、成果展示



圖 33 成品外觀

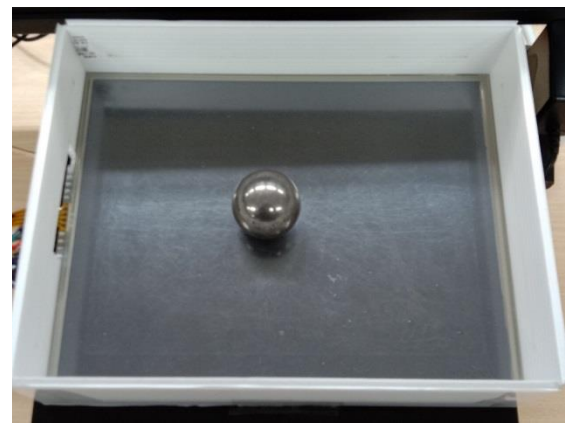


圖 34 Mode0 自動平衡模式

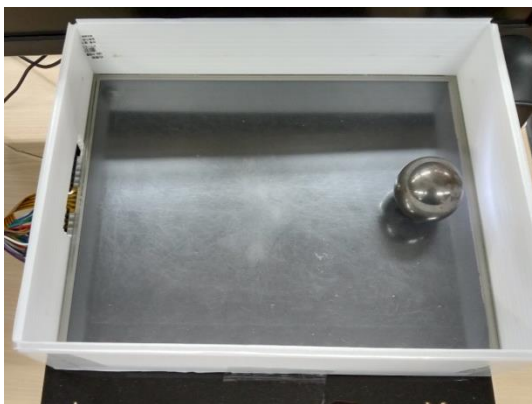


圖 35 Mode1 參數調整模式

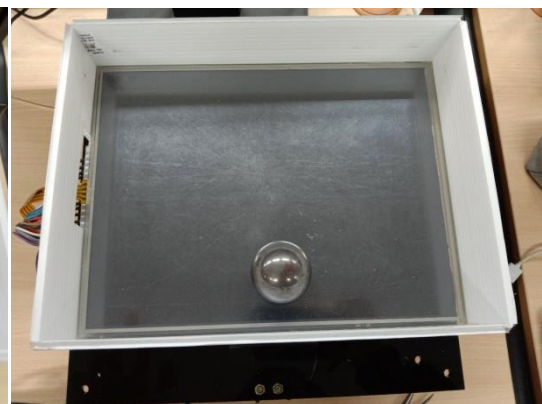


圖 36 Mode3 追點遊戲模式





圖 37 Mode2 規律繞方模式

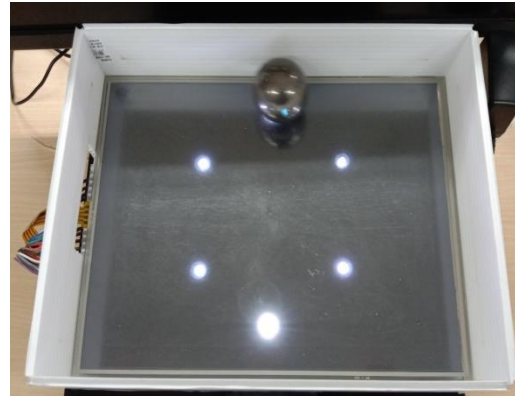


圖 38 Mode2 規律繞圓模式

## 陸、討論

### 一、機構繪圖軟體選用

原先想以 3D 列印製作機構底板，但後來考量到機構的體積較大、堅固性問題以及列印時間，所以我們改用雷射切割，不但可以承受機構重量，也能符合機構需求。

### 二、機構設計

在設計底板時，起初伺服馬達位置在底板對角，但是從中發現運轉角度無法達到最大角，經過多次修改測試，將馬達移至底板邊界中間位置，並在馬達放置處切出凹槽，避免向下轉動時與底板產生碰撞。

### 三、程式參數調整

由於我們對於程式撰寫的經驗不足加上製作硬體設備花費大量時間，我們利用上屆學長的程式進行修改，當我們進行馬達測試時，發現感測值與運動方向相反，此時我們更改程式中的 DIRECT 及 REVERSE，來改變 PID 控制的追蹤方向。當我們把程式燒錄進去時，OLED 顯示器上的 x、y 目標數值持續減少，經過多次測試，我們調整程式平衡點，讓 PID 去追此座標點。

## 四、電路板製作

在這次專題中，為了簡潔電路，我們利用 AD 製作電路板，在第一次洗電路板的過程中，由於對於操作步驟不熟悉，我們多次失敗，從中也學習到許多不可忽視的細節，最終完成理想中的電路板。

## 柒、結論

專題製作所需要具備的能力絕不是一人奮戰，需要組員間的互相配合與認真負責的態度。在遇到困難時，視需求上網搜尋資料、請教老師，讓我們能夠順利度過難關。過程中，我們學習到了許多專業技能，像是洗電路板、雷射切割、3D 列印、文書操作，且能夠把 C 語言融會貫通，相信這些技能在未來對我們大有幫助。

在製作電路板的時候，我們要先設計電路圖，我們利用多個排針的排列，組成電路圖，並完美配線。然而，在洗電路板的時候，出現一些人為疏失，導致線路沒有完整顯影在板子上，經過多次努力後，我們終於大功告成。

雖然我們完成了本次專題實作，但功能略少，仍有許多可以更加精進的部分，期許未來有機會能夠利用 Python 語法製作影像辨識迷宮，也希望平衡球的概念能被運用在送餐機器人上。

## 捌、參考資料及其它

- 1、2020 新版 0.96 吋白色 OLED 顯示屏 12864 液晶屏模塊 IIC 介面。2021 年 10 月 15 日。取自

<https://www.playrobot.com/display/2463-2020096oled-12864-iic.html>

- 2、ALTIUM DESIGNER。2020 年 12 月 10 日。取自

<https://baike.baidu.com/item/Altium%20Designer>

- 3、Onshape-3D 繪圖軟體。2021 年 11 月 05 日。取自

<https://livingtech.education/2017/03/06/onshape-3d%E7%B9%AA%E5%9C%96%E8%BB%9F%E9%AB%94/>

- 4、Arduino/簡介。2021 年 10 月 15 日。取自  
<https://zh.m.wikibooks.org/zh-tw/Arduino/%E7%AE%80%E4%BB%8B>
- 5、PID 控制器。2021 年 12 月 24 日。取自  
<https://zh.wikipedia.org/wiki/PID%E6%8E%A7%E5%88%B6%E5%99%A8>
- 6、梅克 2 工作室(2014)。Arduino 微電腦控制實習。台科大圖書。
- 7、旗立研究室(2020)。計算機概論。台北市:旗立資訊股份有限公司。