



平衡球

Ball On Plate

指導老師:林家德 老師

組員: 01陳沁

02李禹彤

03許巧穎

20鐘子婷

簡報大綱

➤ 前言

- 一、製作背景和目的
- 二、成員介紹
- 三、貢獻圖
- 四、甘特圖
- 五、系統架構圖

➤ 元件硬體介紹

一、硬體介紹

- 1) Pixy CMUcam5
 - 2) 伺服馬達
 - 3) OLED
 - 4) 3D列印
- ### 二、電路板設計

➤ 程式介紹

➤ 問題與解決方法

➤ 心得建議



前言

一、製作背景和目的

機器視覺常伴隨著影像伺服控制機構，
尤其在高精密的工業檢測系統上更是被廣泛的應用。
透過pixy，可進行球體瞬間影像之即時擷取與處理，
藉此辨別球體之位置、速度、加速度等重要特性，
將此資訊轉換為驅動機電伺服系統使用。

金屬平衡板產生傾斜，並盡可能地讓色球又快又準的到達所設定的目標。



二. 成員介紹(1)

陳沁

主要負責工作：
機構設計、3D繪圖



許巧穎

主要負責工作：
機構設計、3D繪圖



二. 成員介紹(2)

李禹彤

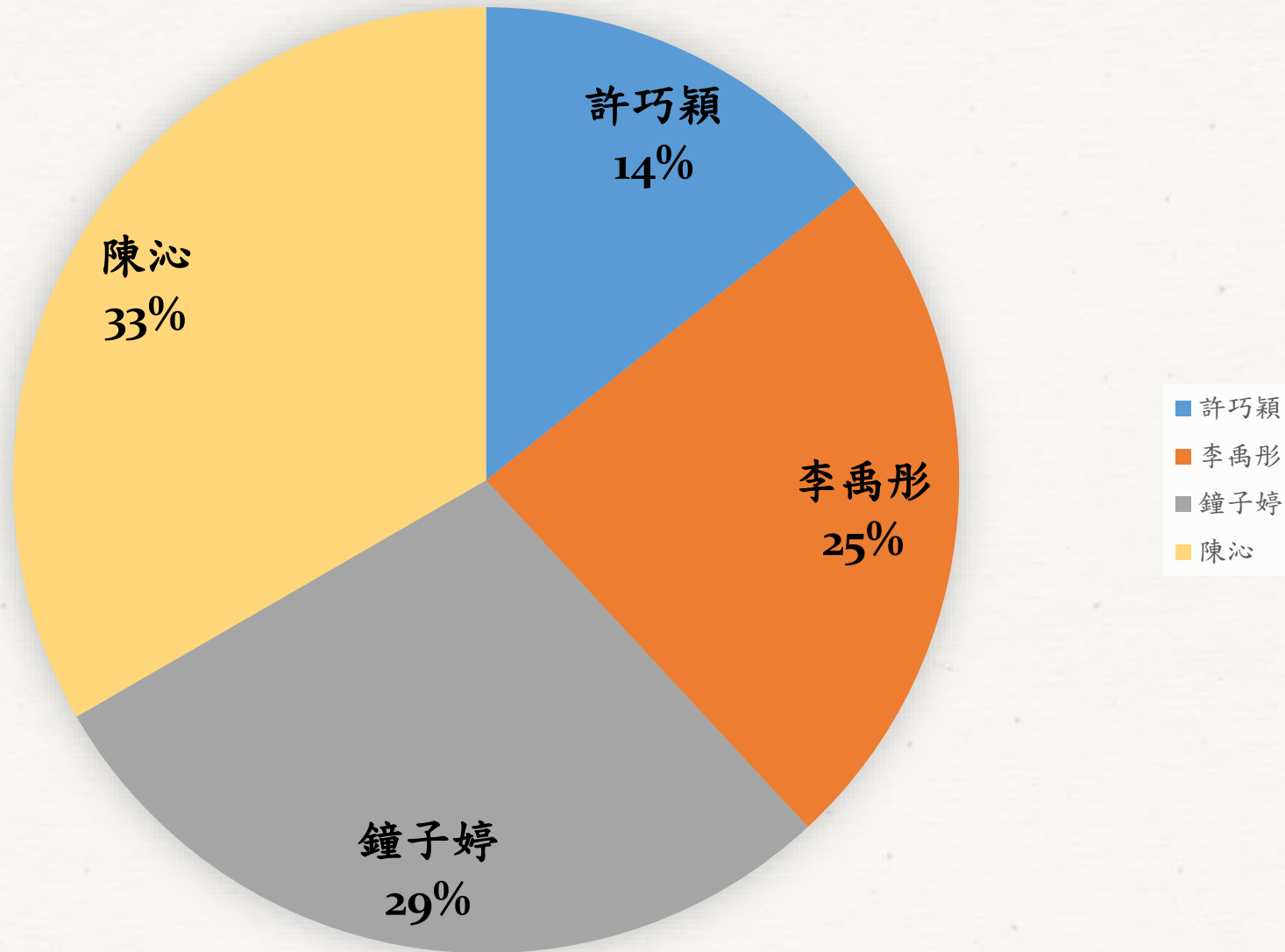
主要負責工作：
程式設計與撰寫



鐘子婷

主要負責工作：
程式設計與撰寫
材料準備

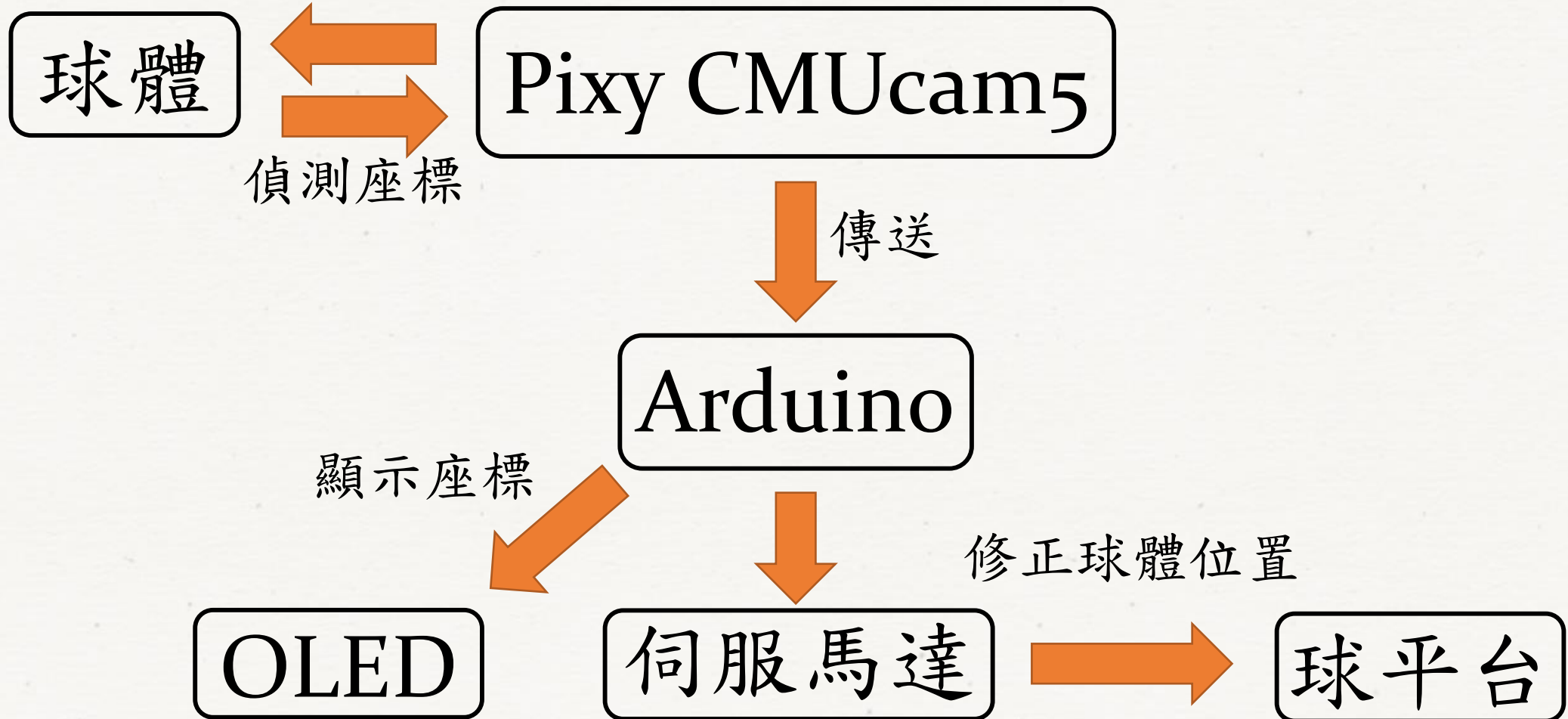
三、貢獻度圓餅圖



四.甘特圖

週次 \ 工作項目	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	負責成員	
收集資料	■	■																	全	
理論探討	■	■																	全	
機構規劃	■	■	■																全	
硬體規劃			■	■															全	
軟體規劃				■	■	■													全	
3D 繪圖	■	■	■	■	■	■													1	
3D 列印 +機構組裝					■	■													1	
電路板設計							■	■	■	■									2	
電路板焊接										■	■								1	
OLED	■	■	■	■	■	■													2	
Pixy 控制	■	■	■																2	
伺服馬達										■	■	■	■	■					2	
PID控制														■	■				2	
Arduino 程式撰寫					■	■	■	■	■	■	■	■	■	■	■				2	
製作印刷電路板													■	■	■	■	■	■	全	
旋轉平台 +固定架設計															■	■	■	■	1	
撰寫報告	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	依據當週負責人
預定進度	3	9	12	20	25	39	45	48	50	60	65	70	75	80	85	90	95	100	累積百分比%	

五、系統架構圖



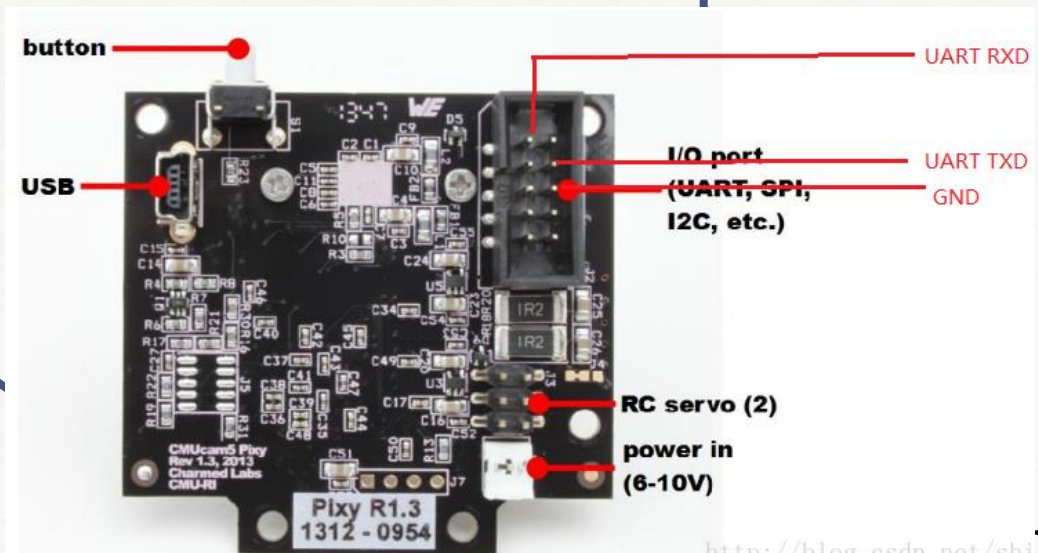


元件設備介紹

一、硬體介紹(1)

◆ Pixy CMUcam5

- 小巧，快速，易用，低成本，易於使用的視覺系統
- 學會檢測你教的物體
- 輸出每秒檢測50幀
- 提供所有Arduino，Raspberry Pi等函式庫
- 支持C / C ++和Python
- 通過幾種接口之一進行通信
UART串列，SPI，I²C，
USB，數位/類比

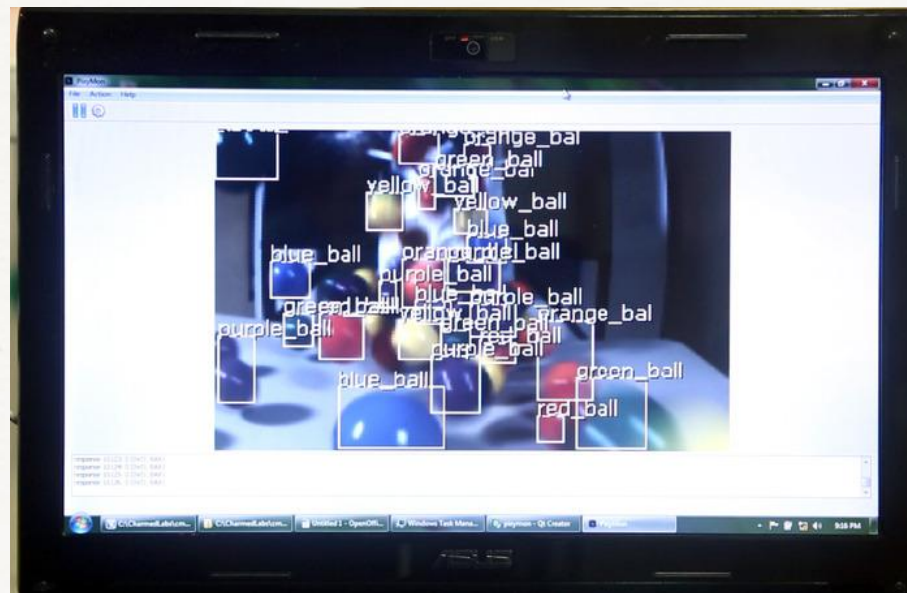


◆ Pixy CMUcam5

特色:

可以幾乎同時辨識出幾百個物件

每秒可以處理50張640*400大小的圖片
可精準抓取移動物體



硬體介紹(2)

◆ 伺服馬達 spektrum A6270

馬達工作電壓4.8~8.4V。

馬達扭力為3.13 kgf-cm。

馬達速度達0.18sec/60° (6V工作電壓)。

馬達材質為金屬齒輪及本體。

馬達重量76.5g。

數位式伺服馬達，僅須傳送一次PWM訊號即可控制馬達轉動。



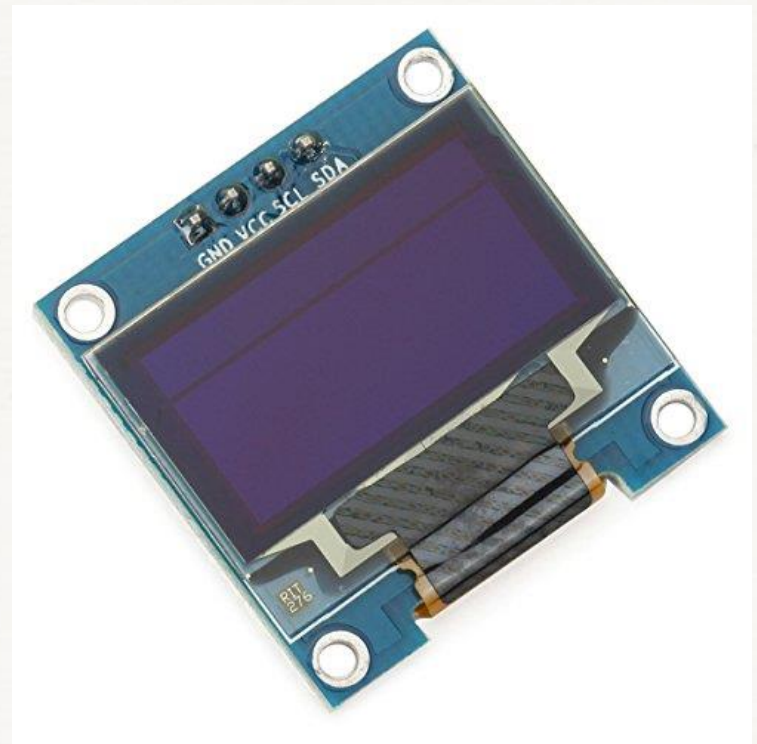
硬體介紹(3)

◆ 0.96吋OLED顯示器

屬於PMOLED(Passive matrix OLED)

特色：

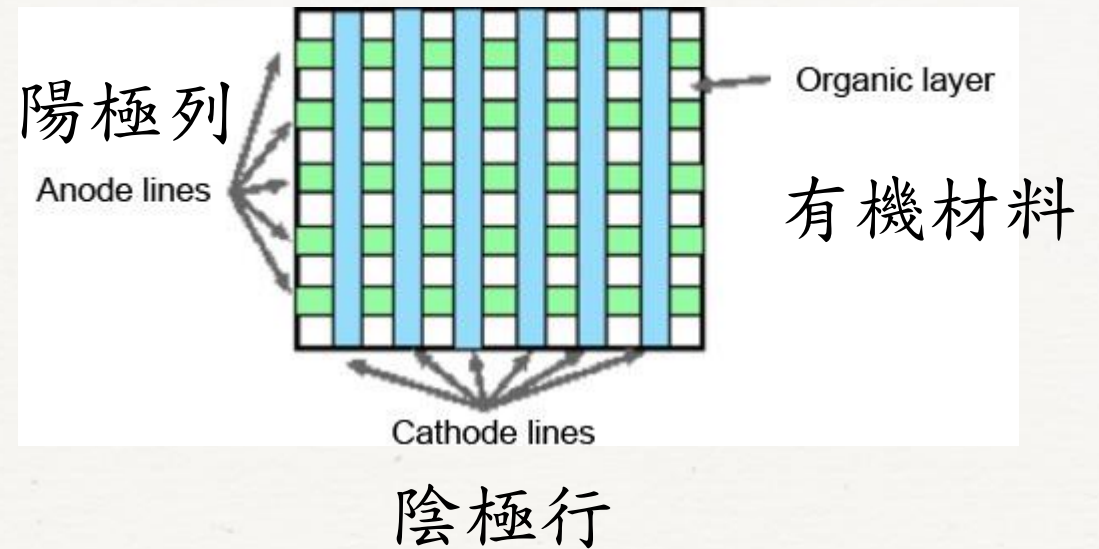
1. 結構簡單
2. 成本低
3. 自發光
4. 高亮度
5. 視角廣
6. 功耗低



◆ 0.96吋OLED顯示器

- 發光原理

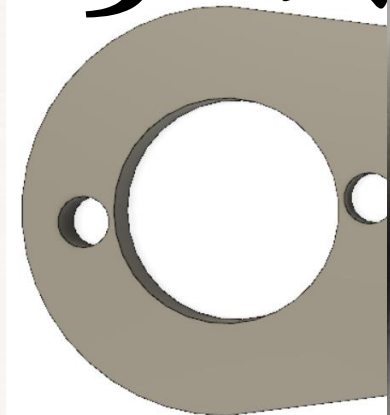
採用循序掃描的驅動方式
當電流通過的時候
行列交叉點被激活
進而發光



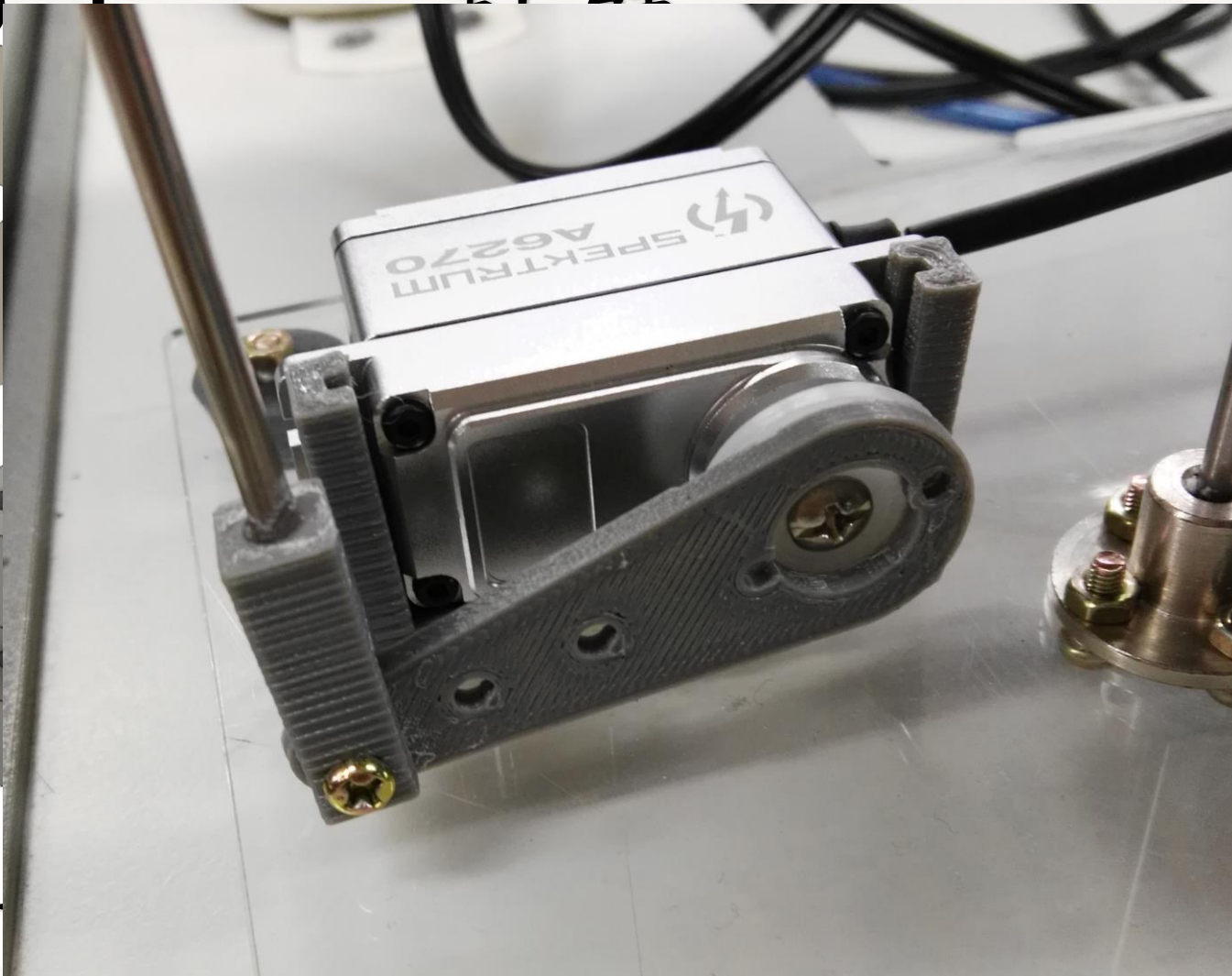
硬體介紹(4)

◆3D列印

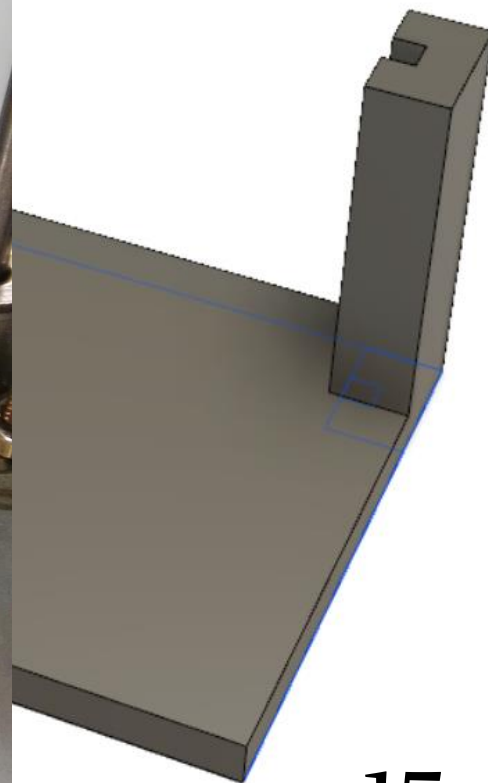
封壁



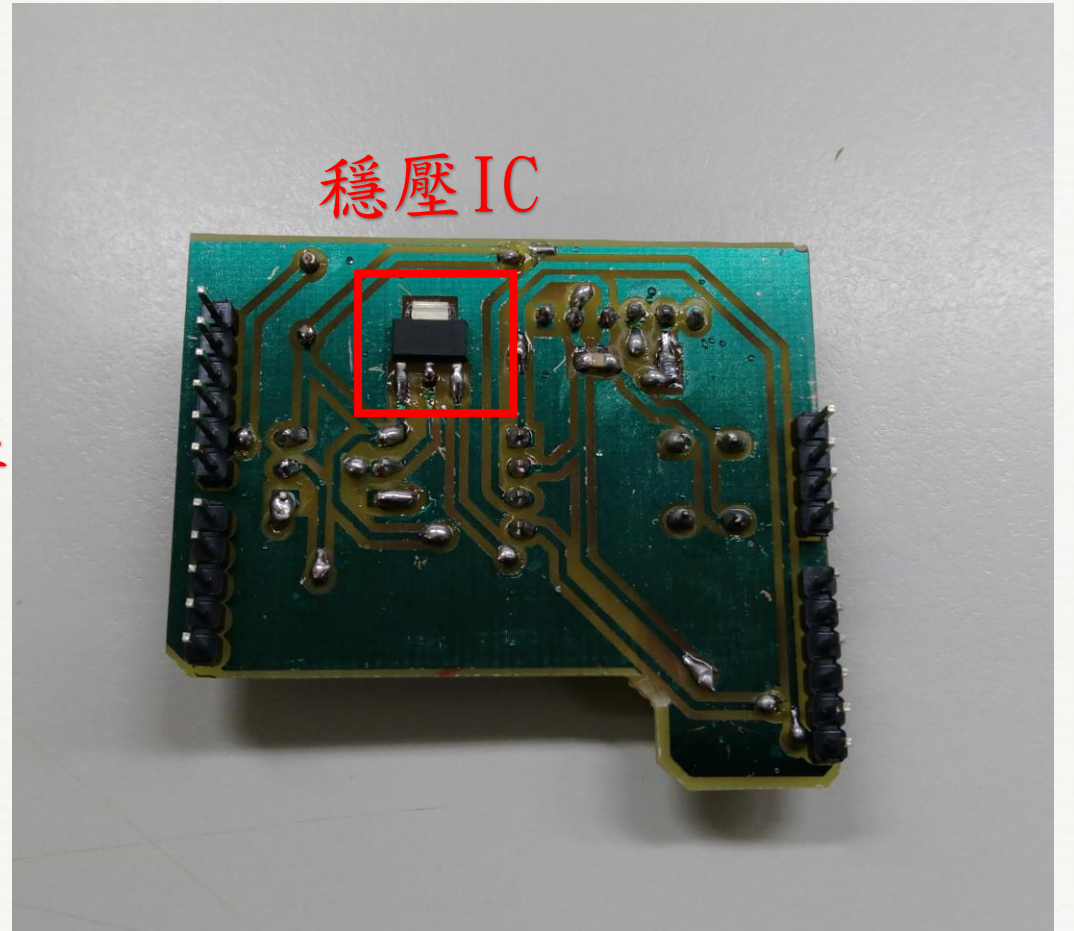
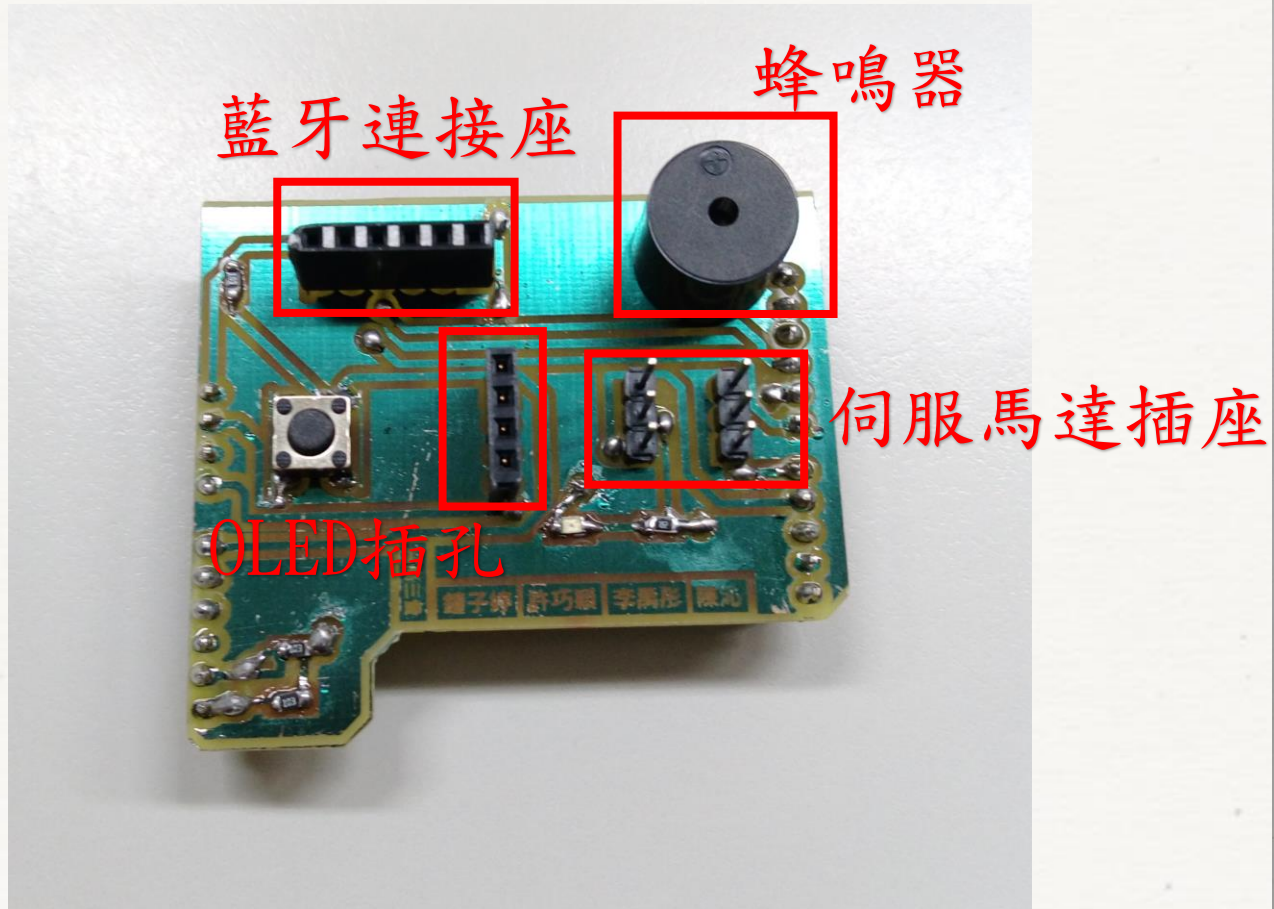
轉接



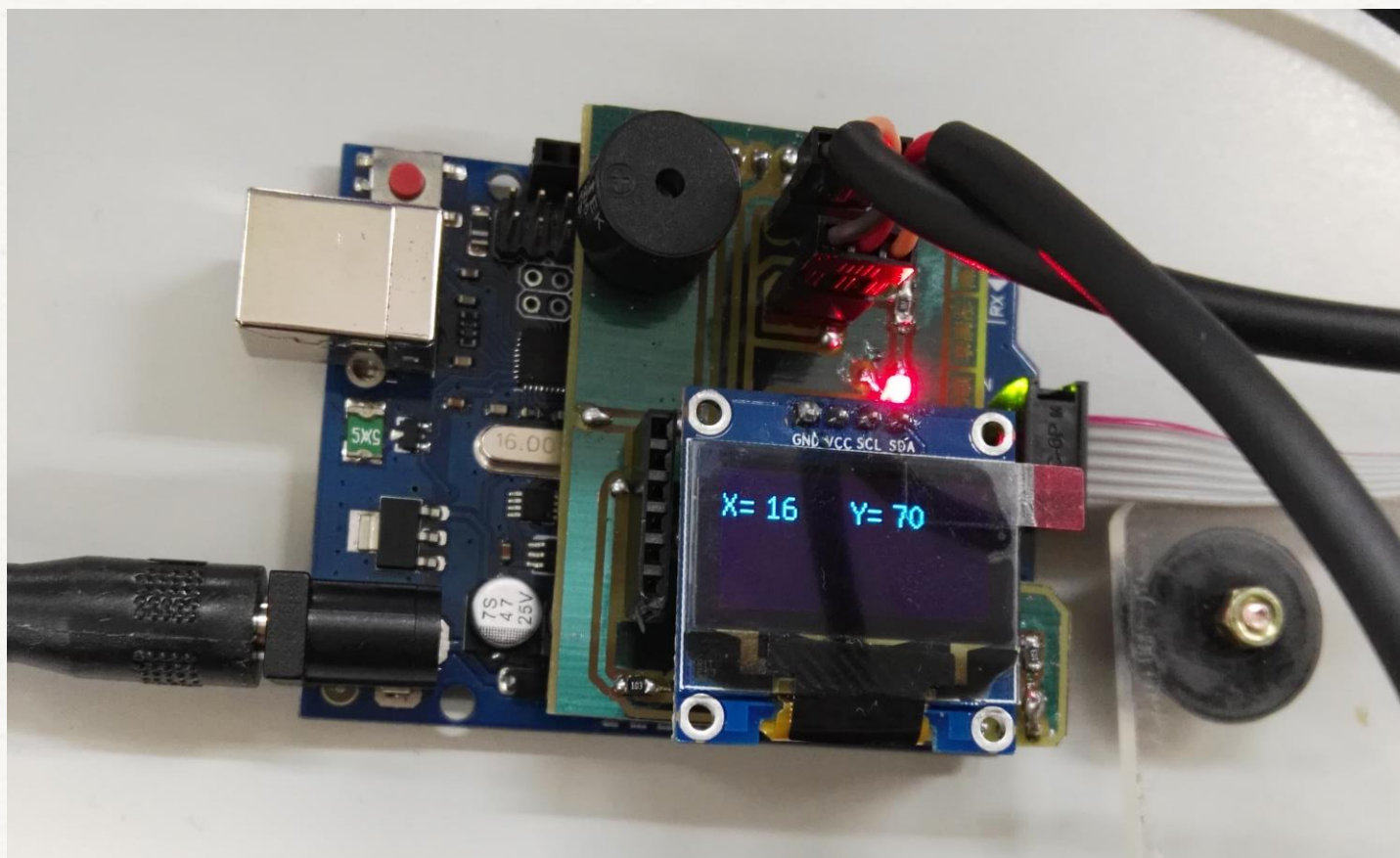
達底座



二、電路板設計



與Arduino Uno結合





程式介紹

```
#include <Adafruit_SSD1306.h>
#include<Adafruit_GFX.h>
#include <U8glib.h>
U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE | U8G_I2C_OPT_DEV_0); // I2C / TWI
#include <SPI.h>
#include <Pixy.h>
#include <Servo.h>
Servo myservoX, myservoY;
#define Buzzer 9
```

```
float Kp = 6.7, Ki = 0.0005, Kd = 0.1; // PID 參數值
double values;
Pixy pixy;
int x, y;
unsigned long lastT = millis();
unsigned int last_error = 0;
unsigned int last_error1 = 0;
```

```

void setup()
{
  u8g.begin();

  myservoX.attach(7);    // 角度限制 67, 108
  myservoY.attach(6);    // 角度限制 60, 110
  Serial.begin(57600);
  Serial.print("Starting...\n");
  pixy.init();
  u8g.setFont(u8g_font_unifont);
  u8g.setColorIndex(1);
}

void loop()
{
  uint16_t blocks;
  char buf[32];
  u8g.firstPage();
  do
  {
    draw();
  } while ( u8g.nextPage() );
  delay(1);
  blocks = pixy.getBlocks();
  if (blocks)
  {
    //if (i++%10==0)
    {
      x = pixy.blocks[0].x;
      y = pixy.blocks[0].y;
      sprintf(buf, "X=%d, Y=%d\t", x, y);/// 注意 %d 是給int用(2 bytes); long 要用 %ld;\n是換行
      Serial.print(buf);
      draw();
    }
  }
}

```

```

int error = 184 - x;          // 計算 x 與設定點的誤差量
int derivative = error - last_error; // 計算位置的比率誤差變化量(derivative)
last_error = error;          // 記住上次的比率誤差量，並計算輸出控制量 output
// 計算 PID 輸出控制量，error(P值)、derivative(D值)、last_error用來取D值的
int output = error * Kp + derivative * Kd;
Serial.println(output);
output = constrain(output+1500,1116,1642); // 限制 output 的最大值範圍
myservoX.writeMicroseconds(output);        // 輸出 X 馬達角度控制量

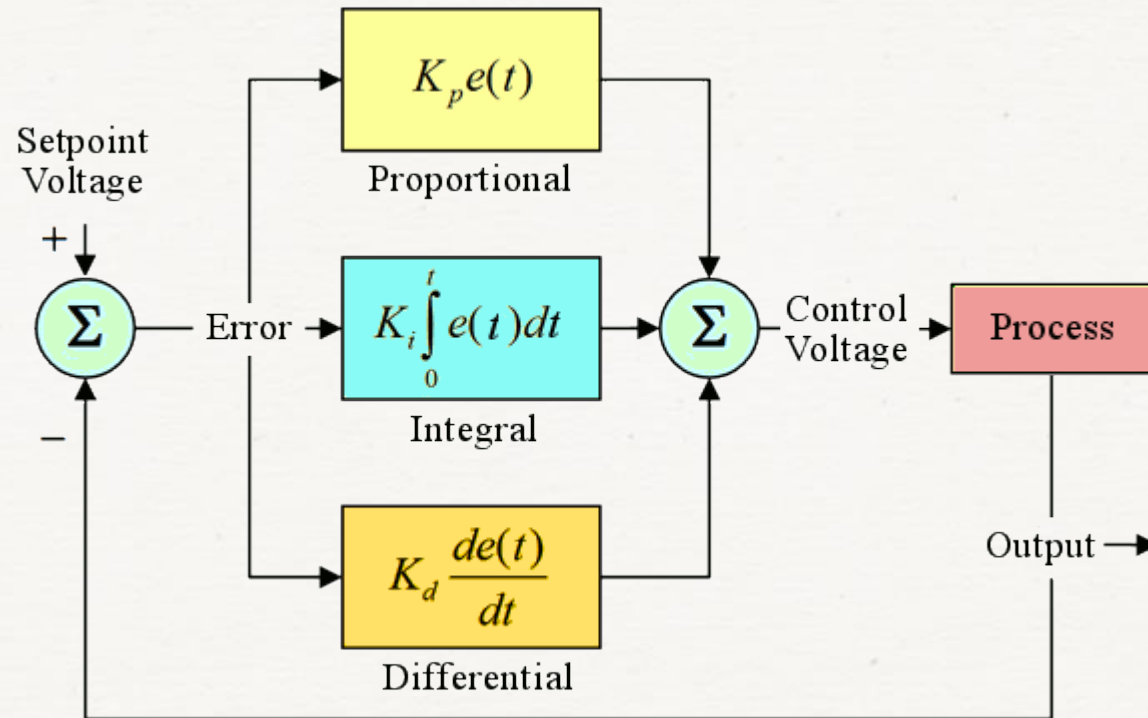
int error1 = 103 - y;         // 計算 y 與設定點的誤差量
int derivative1 = error1 - last_error1;
last_error1 = error1;
int output1 = error1 * Kp + derivative1 * Kd; // 計算 PID 輸出控制量
Serial.println(output1);
output1 = constrain(output1+1500,1295,1704); // 限制 output 的最大值範圍
myservoY.writeMicroseconds(output1);        // 輸出 Y 馬達角度控制量

delay(1);
}
}
}

```

```
void draw() {  
    // graphic commands to redraw the complete screen should be placed here  
    u8g.drawStr( 0, 22, "X=");  
    u8g.setPrintPos(20, 22);  
    u8g.print(x);  
    u8g.drawStr( 60, 22, "Y=");  
    u8g.setPrintPos(80, 22);  
    u8g.print(y);  
}
```

PID控制器的方塊圖



K_p ：比例增益

K_i ：積分增益

K_d ：微分增益

備註: K_p, K_i, K_d 皆為調適參數

e ：誤差=設定值 (SP) - 回授值 (PV)

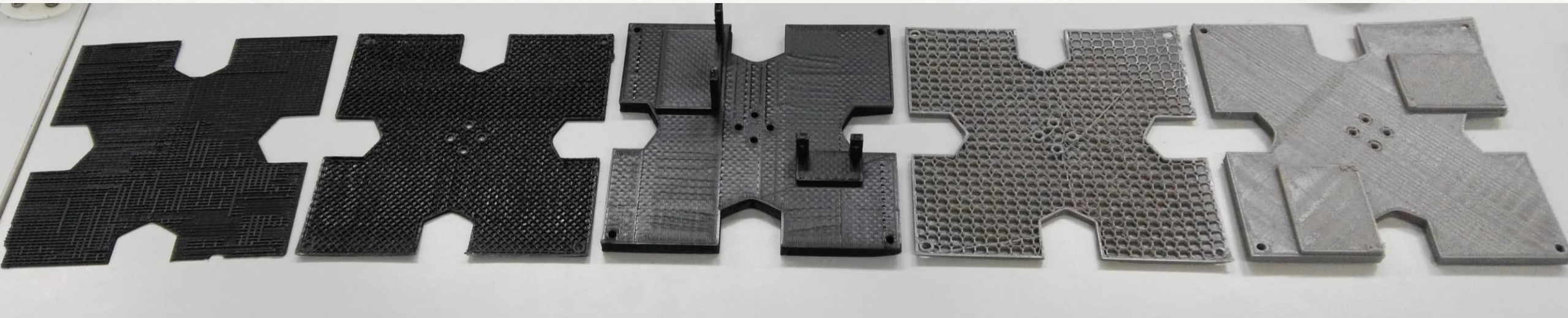
t ：目前時間

τ ：積分變數，數值從0到目前時間



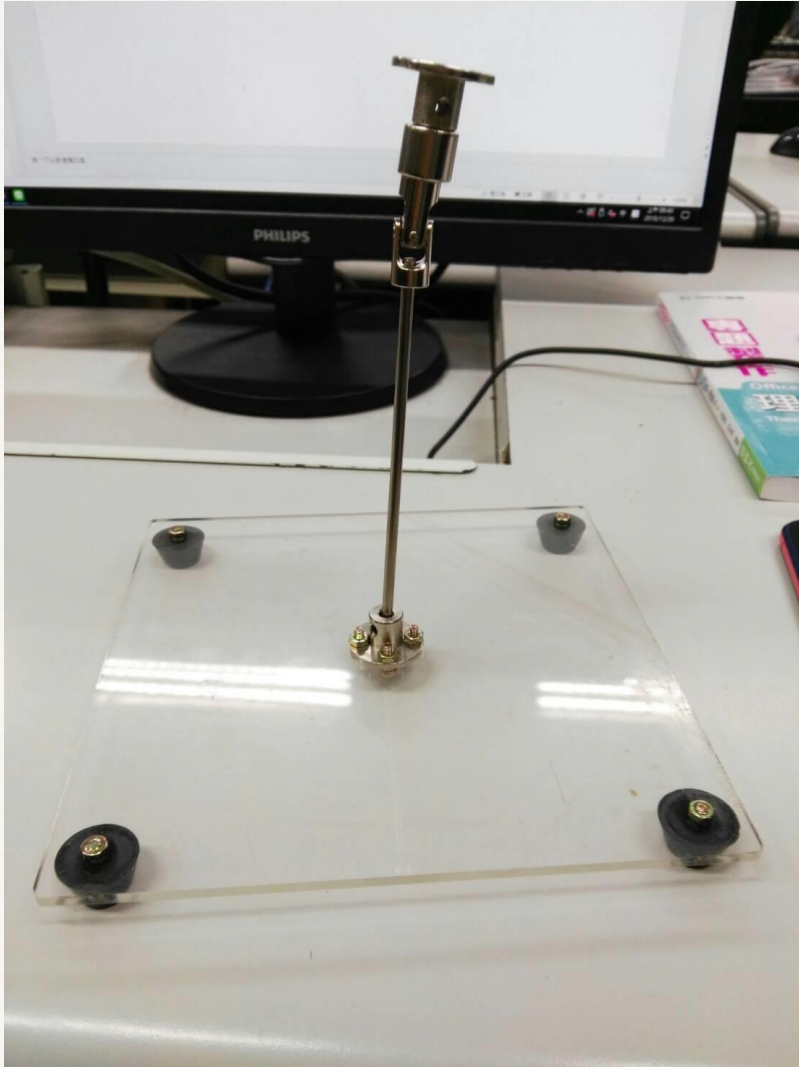
問題與解決方法

底板製作過程



多次3D印製底板失敗

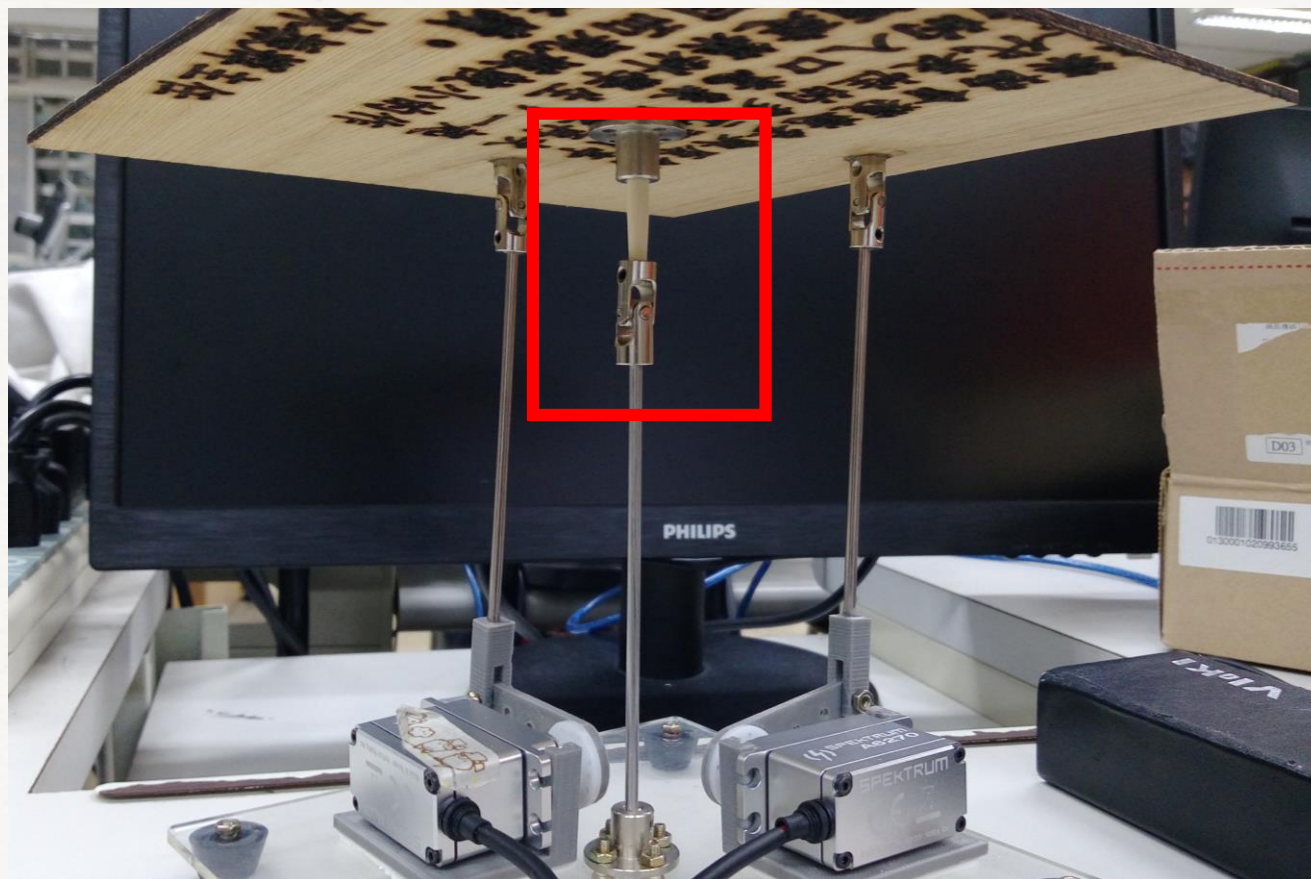
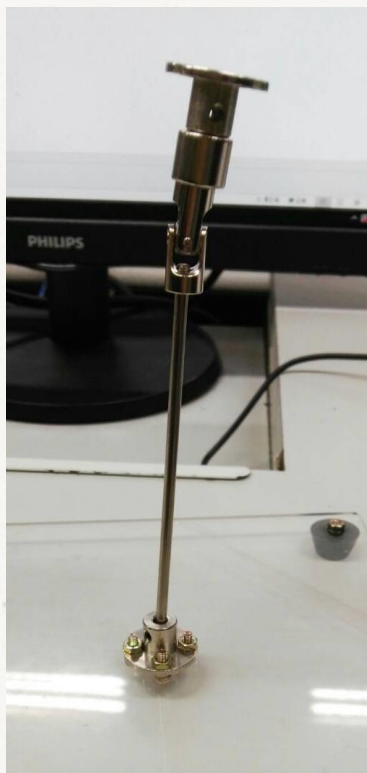
解決方法



改用壓克力板

中心軸長度問題

竹筴加長

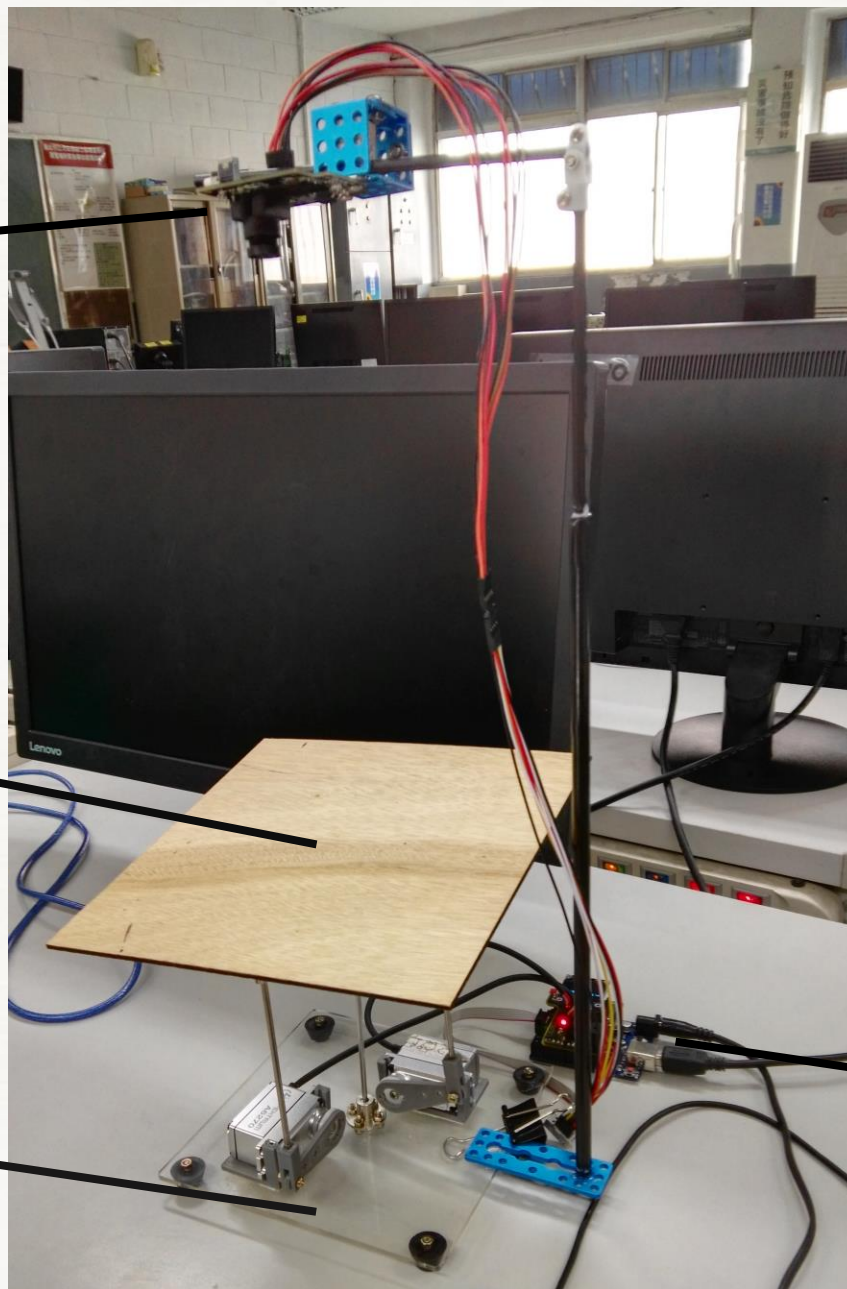


成品圖

PIXY

平台

底座



控制板



心得、建議



3

參考資料

CMUcam5 Pixy: 開源機器視覺感測器 - MakerLab 創客實驗室

<http://www.makerlab.me/guides/6/articles> 20140228

Cheng Zhang - Ball on Big Plate 20170812

<https://sites.google.com/view/pidan1231239/projects/ball-on-big-plate>

09-OLED顯示器 - 阿玉maker研究區 20170316

<https://sites.google.com/site/wenyumaker/09-oled-xian-shi-qi>

Thank you

感謝聆聽

