

App Inventor 期末專案

以藍牙操控之互動展示裝置

指導老師：曾吉弘

隊員：張群昇、陳彥竹

一、 專題目標

透過利用 App Inventor 所開發的手機，來控制電腦畫面中 Processing 所繪製的方格陣列色塊，使用者透過翻轉手機，來改變畫面中的陣列的色塊；轉動手機將改變畫面中方塊顏色的 RGB，期望使用者在操作的過程中，呈現出互動裝置藝術設計之目的。

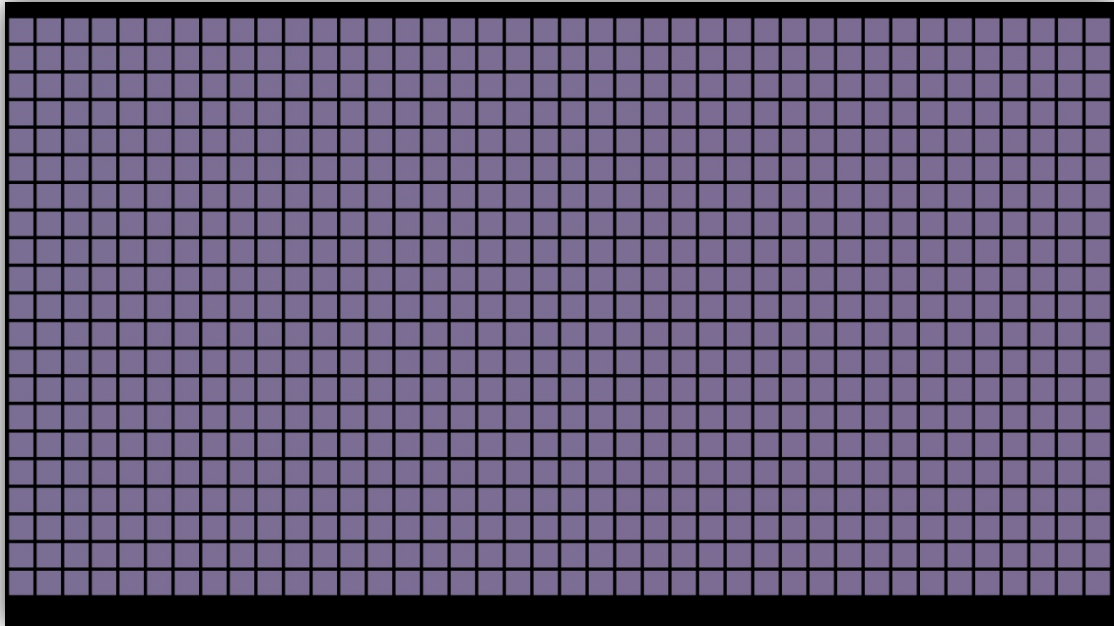


圖 1：Processing 畫面示意圖

二、 功能應用說明

1. **藍牙傳輸應用：**
透過手機的藍牙裝置與電腦相連接。
2. **Orientation Sensor 應用：**
透過手機的上下左右翻轉，運用偵測 Pitch、Roll 以及 Azimuth 三軸的數值，改變 Processing 所設計的畫面。
3. **Processing 應用：**
透過 Processing 撰寫顯示的互動畫面，並與手機做藍牙的連線的數值接收。

三、裝置設定

1. App Inventor 藍牙連結設定：

在 Inventor 裡插入藍牙，並設定以 450 毫秒的頻率傳送三軸資訊。



圖 2：App Inventor 畫面按鈕配置圖

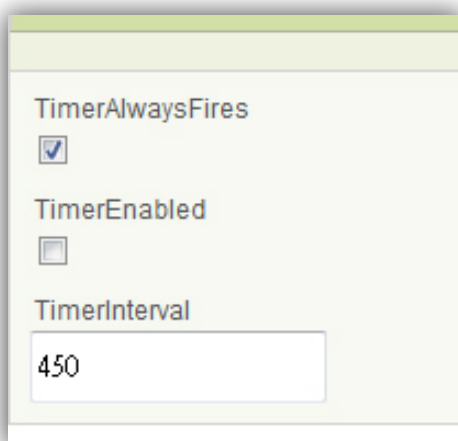


圖 3：App Inventor 時脈頻率設定

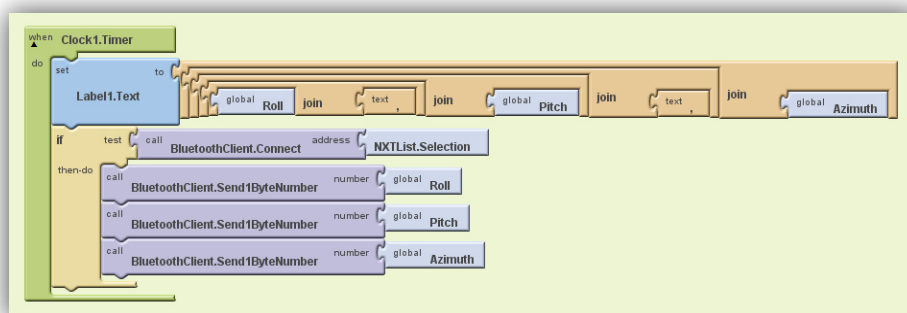


圖 4：App Inventor 時脈之程式架構

2. 電腦與 Processing 藍牙設定：

由於我們希望透過行動裝置的傳值來改變電腦端的 Processing 畫面，因此需先將電腦的藍牙"寫入"設定開啟，方法如下圖步驟。

(1) 右下角藍牙圖案點右鍵



圖 5：電腦之藍牙連線設定一

(2) 選擇 COM 連接埠，點下面"新增"



圖 6：電腦之藍牙連線設定二

(3) 接著會出現下面畫面，選擇"連入"，然後按確定



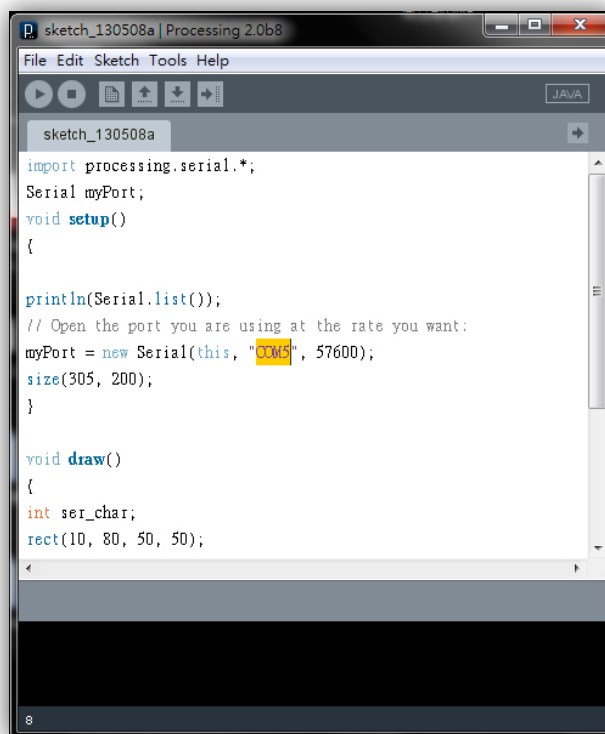
圖 7：電腦之藍牙連線設定三

(4) 即可看見電腦藍牙"連入"的 COM 值



圖 8：電腦之藍牙連線設定四

(5) Processing 設定同樣的 COM 值連入即可收到來自行動裝置的數值



```
sketch_130508a | Processing 2.0b8
File Edit Sketch Tools Help
sketch_130508a
import processing.serial.*;
Serial myPort;
void setup()
{
  println(Serial.list());
  // Open the port you are using at the rate you want:
  myPort = new Serial(this, "COM3", 57600);
  size(305, 200);
}

void draw()
{
  int ser_char;
  rect(10, 80, 50, 50);
}
```

圖 9：電腦之藍牙連線設定五

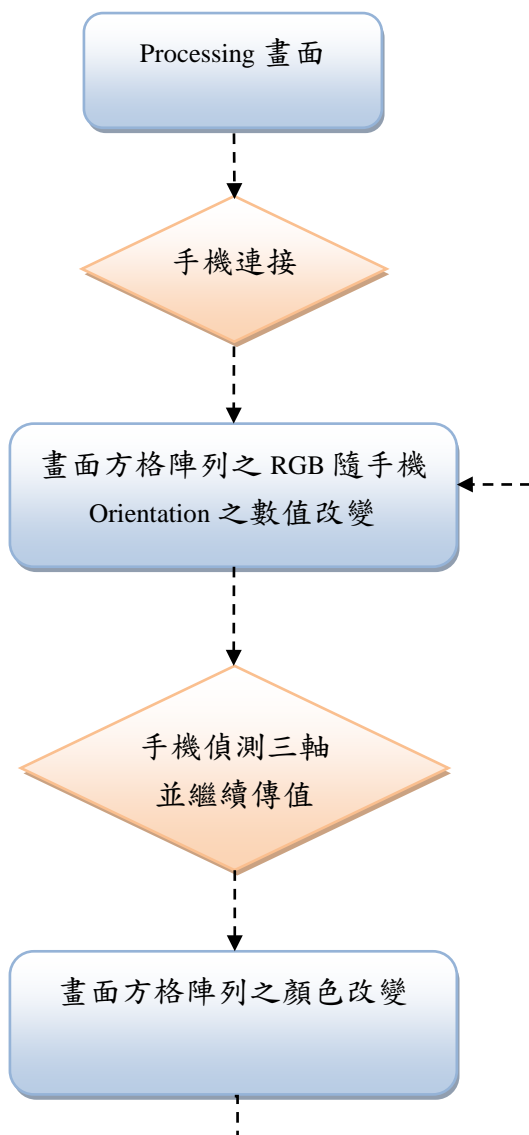
四、使用零件表

表 1：使用零件表

名稱	數量
Computer	1
Smart Device With Andriod	1

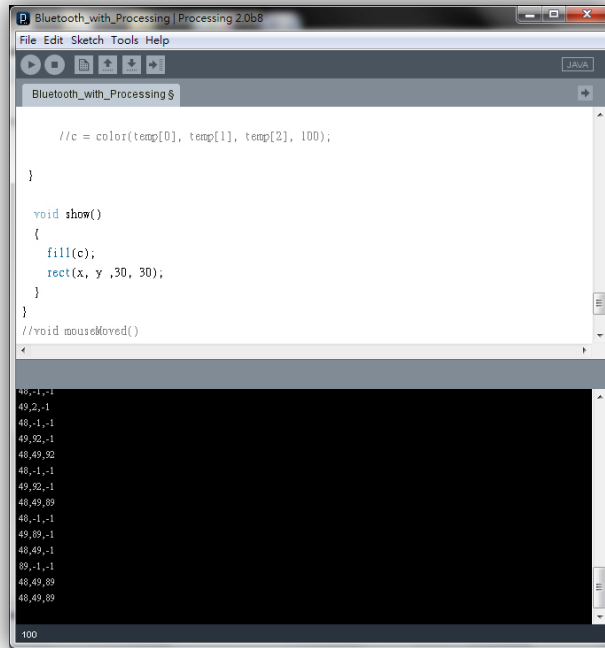
五、 程式流程圖

本專題開發之 App Inventor 與 Processing 控制流程圖如下，當電腦尚未被手機藍牙連線前，不會產生任何效果；而當接收到手機藍牙連線後，在手機的部分會以 450 毫秒的頻率，透過接收三軸的數值傳給 Processing 以改變畫面，進而重複之。



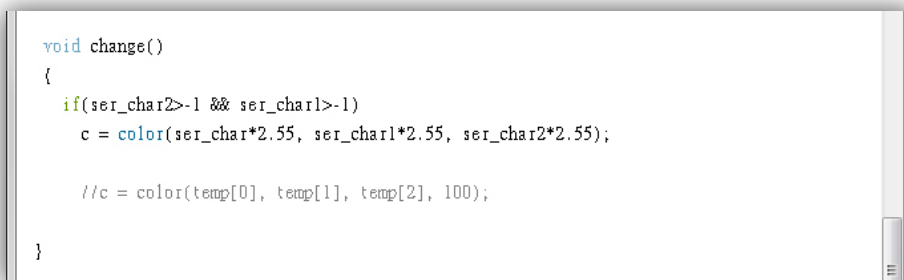
六、問題與解決

由於在藍牙傳輸上會有掉值的問題，因此在 Processing 接收時，未被接收到的值則會以"-1"呈現，但另一方面，在測試 Processing 之藍牙數值接收時，我們發現當 3 軸的數值都接收到時，其接收之數值順序，一定會是我們原先期望的數值順序，因此可以透過篩選"-1"的動作來設定。



```
Bluetooth_with_Processing | Processing 2.0b9
File Edit Sketch Tools Help
Bluetooth_with_Processing $
//c = color(temp[0], temp[1], temp[2], 100);
}
void show()
{
  fill(c);
  rect(x, y, 30, 30);
}
//void mouseMoved()
48,49,88
49,2,-1
48,-1,-1
49,92,-1
48,49,92
48,-1,-1
49,92,-1
48,49,88
48,-1,-1
49,89,-1
48,49,-1
39,-1,-1
48,49,88
48,49,88
100
```

圖 10：Processing 之藍牙數值接收狀態



```
void change()
{
  if(ser_char2>-1 && ser_char1>-1)
    c = color(ser_char*2.55, ser_char1*2.55, ser_char2*2.55);

  //c = color(temp[0], temp[1], temp[2], 100);
}
```

圖 11：Processing 之數值篩選架構

七、專案進度甘特圖表

表 2：專案進度甘特圖表

	工作名稱	開始	完成	期間	Mar	Apr					May					Jun		
					27	3	10	17	24	1	8	15	22	29	4			
1	陀螺儀設計	03/27	04/10	2 周														
2	藍牙傳輸設計	04/10	05/01	3 周														
3	Processing 畫面設計	05/01	05/22	3 周														
4	整合設計與優化	05/22	06/19	2 周														

八、專案工作分配表

表 3：專案進度甘特圖表

工作項目	負責人員 / 輔助人員
Andriod 程式設計	陳彥竹 / 張群昇
Processing 程式設計	張群昇 / 陳彥竹

九、結論與未來期望

由於當初在設定藍牙連接，以及 Processing 數值接收時，花了我們蠻多的時間在測試，因為學期期末發表時間的關係，我們在畫面效果的呈現上比較有限，如果還有機會往後繼續延伸，希望可以加入更多的畫面動畫元素，如方格陣列大小變形之波浪呈現，或是方格陣列色彩漸層的設計。

十、參考資料

1. [App Inventor 中文學習網](#)
2. [Processing](#)
3. [Processing on Android - orientation detection](#)