

手機基礎程式設計

由於本書會用到 Android 手機 APP 應用軟體程式設計，本文使用 MIT 開發的 App Inventor 2，所以本章節主要是教各位讀者 App Inventor 2 基本操作(曹永忠, 許智誠, & 蔡英德, 2015a, 2015b, 2015c, 2015d)，希望讀者能仔細閱讀，因為在往後實作時，重覆的部份就不在重覆敘述之。

如何執行 AppInventor 程式

由於我們寫好 App Inventor 2 程式後，都必需先使用 Android 作業系統的手機或平板進行測試程式，所以本節專門介紹如何在手機、平板上測試 APPs 的程式。

首先，如下圖所示，我們在 App Inventor 2 程式模塊編輯畫面之中，在『Connect』的選單下，選取 AICompanion。

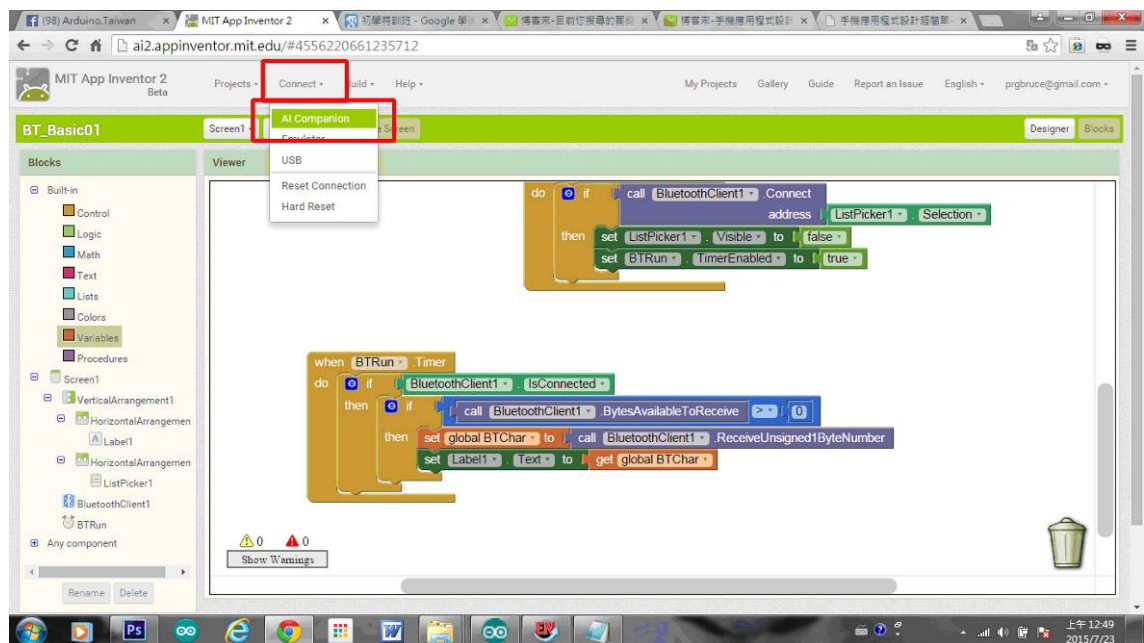


圖 1 啟動手機測試功能

如下圖所示，系統會出現一個 QR Code 的畫面。



圖 2 手機 QR CODE

如下圖所示，我們在使用 Android 的手機、平板，執行已安裝好的『MIT App Inventor 2 Companion』，點選之後進入如下圖。

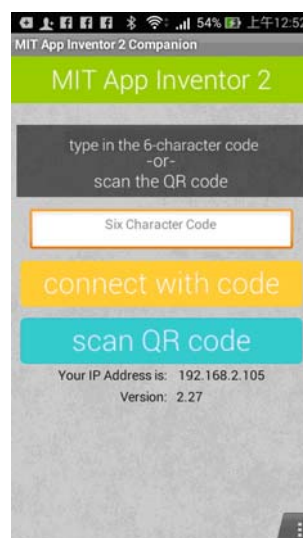


圖 3 啟動 MIT_AI2_Companion

如下圖所示，我們在選擇『scan QR code』，點選之後進入如下圖。

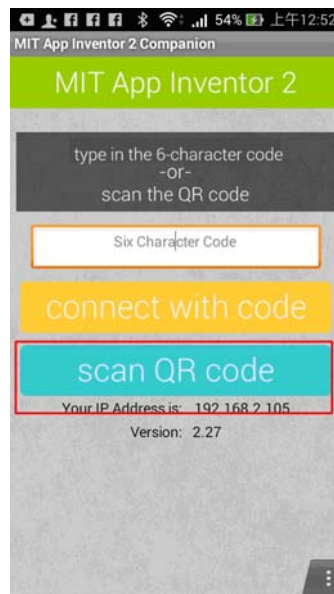


圖 4 掃描 QRCode

如下圖所示，手機會啟動掃描 QR code 的程式功能，這時後只要將手機、平板的 Camera 鏡頭描準畫面的 QR Code 就可以了。



圖 5 掃描 QRCodeing

如下圖所示，如果手機會啟動掃描 QR code 成功的話，系統會回傳 QR Code 碼到如下圖所示的紅框之中。



圖 6 取得 QR 程式碼

如下圖所示，我們點選如下圖所示的紅框之中的『connect with code』，就可以進入測試程式區。

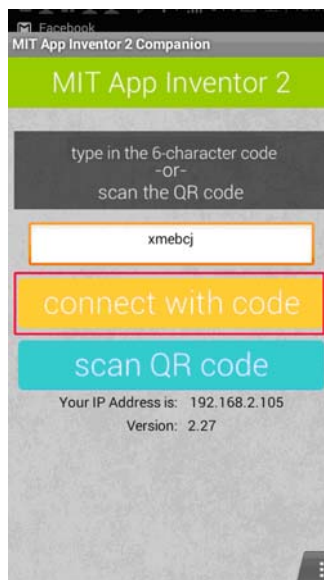


圖 7 執行程式

如下圖所示，如果程式沒有問題，我們就可以成功進入測試程式的主畫面。



圖 8 執行程式主畫面

上傳電腦原始碼

本書有許多 App Inventor 2 程式範例，我們如果不要一一重寫，可以取得範例網站的程式原始碼後，讀者可以參考本節內容，將這些程式原始碼上傳到我們個人帳號的 App Inventor 2 個人保管箱內，就可以編譯、發佈或進一步修改程式。

首先，如下圖所示，我們在 App Inventor 2 程式模塊編輯畫面之中，在『Projects』的選單下。

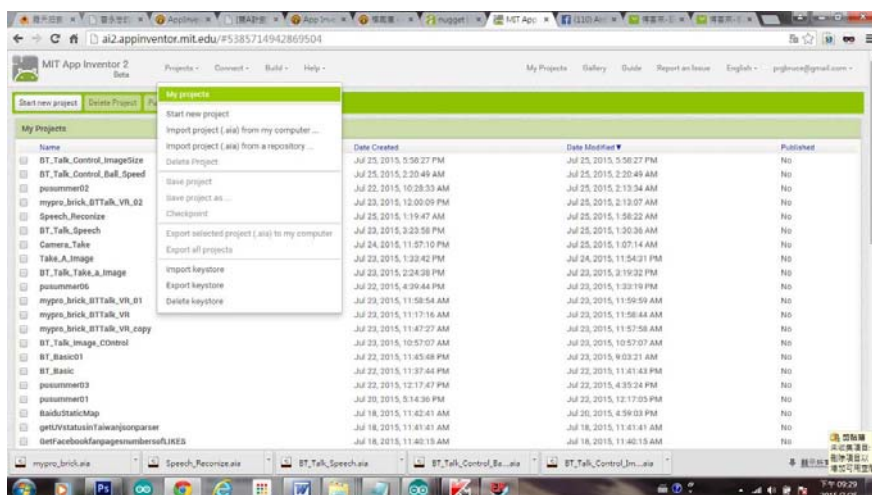


圖 9 切換到專案管理畫面

如下圖所示，我們在 App Inventor 2 程式模塊編輯畫面之中，點選在『Projects』的選單下『import project (.aia) from my computer』。

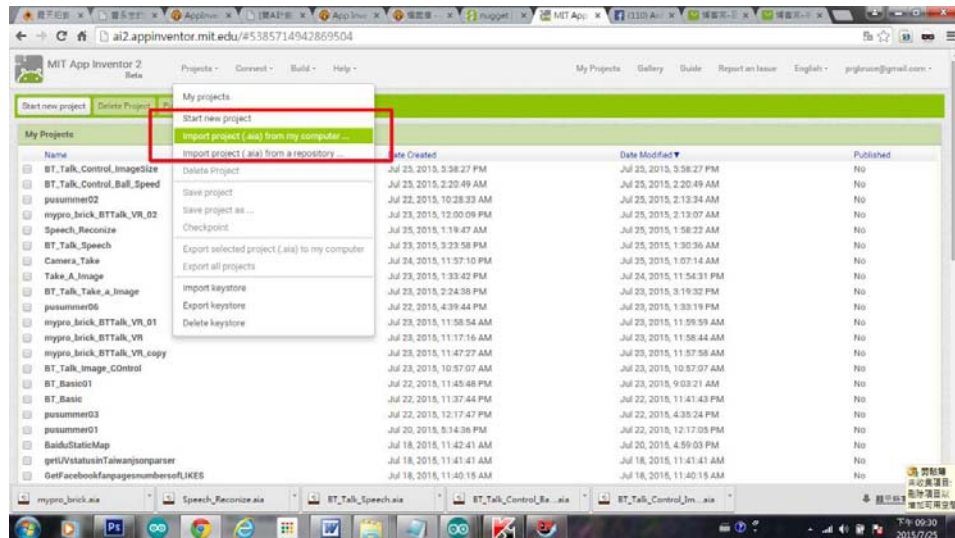


圖 10 上傳原始碼到我的專案箱

如下圖所示，出現『import project...』的對話窗，點選在『選擇檔案』的按鈕。

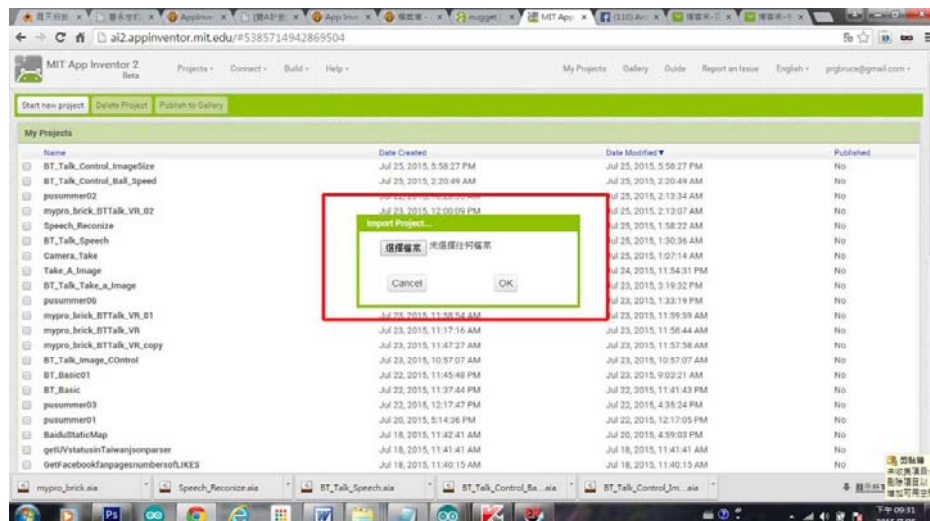


圖 11 選擇檔案對話窗

如下圖所示，出現『開啟舊檔』的對話窗，請切換到您存放程式碼路徑，並點選您要上傳的『程式碼』。

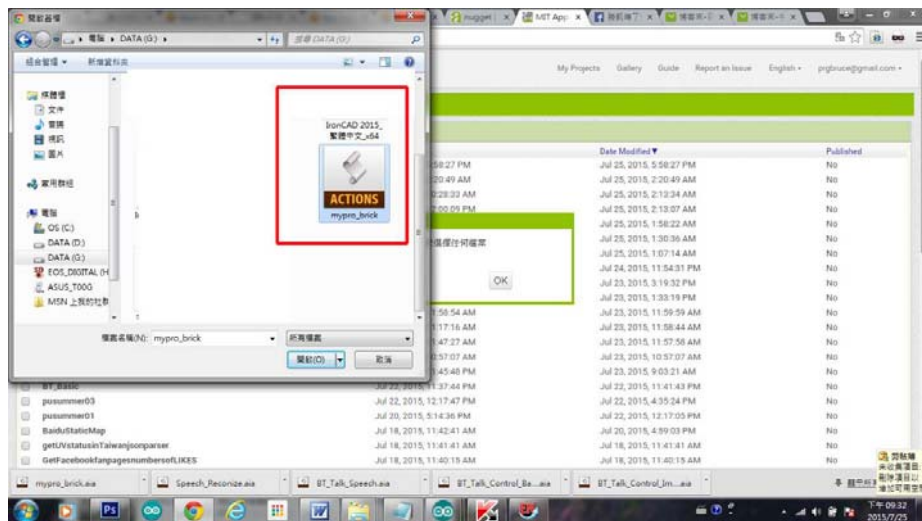


圖 12 選擇電腦原始檔

如下圖所示，出現『開啟舊檔』的對話窗，請切換到您存放程式碼路徑，並點選您要上傳的『程式碼』，並按下『開啟』的按鈕。

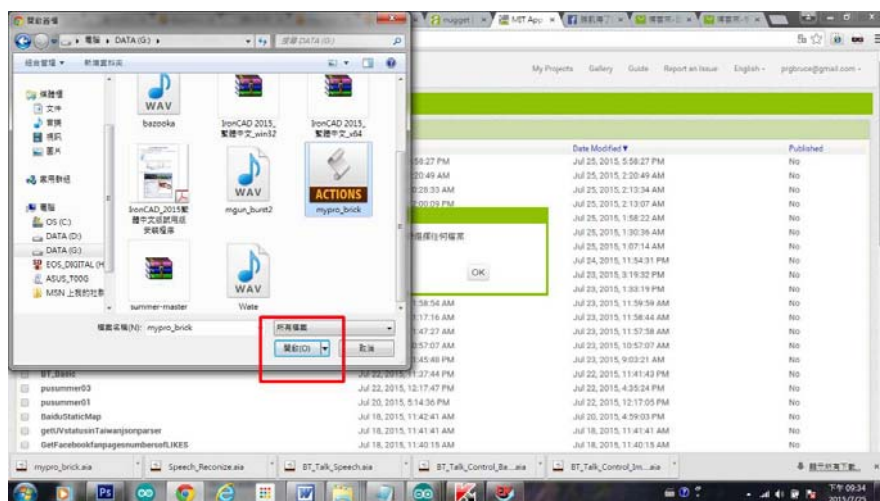


圖 13 開啟該範例

如下圖所示，出現『import project...』的對話窗，點選在『OK』的按鈕。

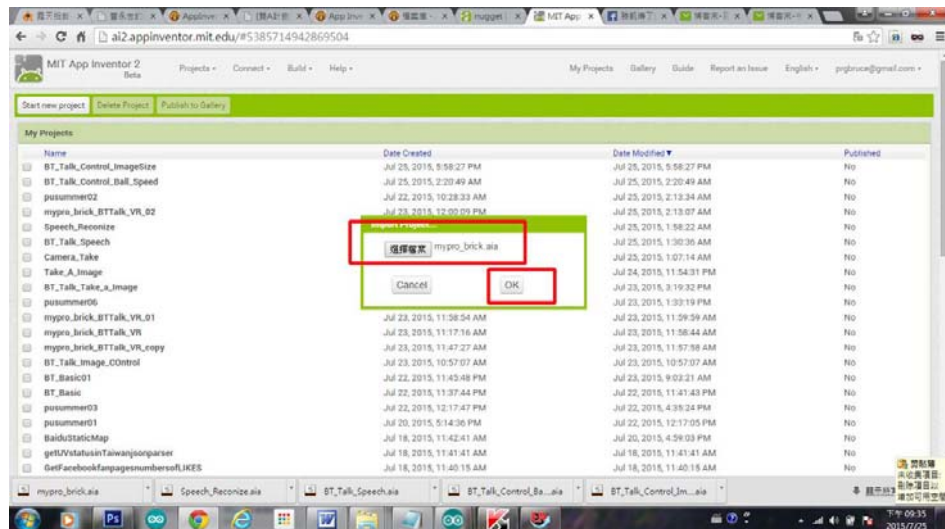


圖 14 開始上傳該範例

如下圖所示，如果上傳程式碼沒有問題，就會回到 App Inventor 2 的元件編輯畫面，代表您已經正確上傳該程式原始碼了。

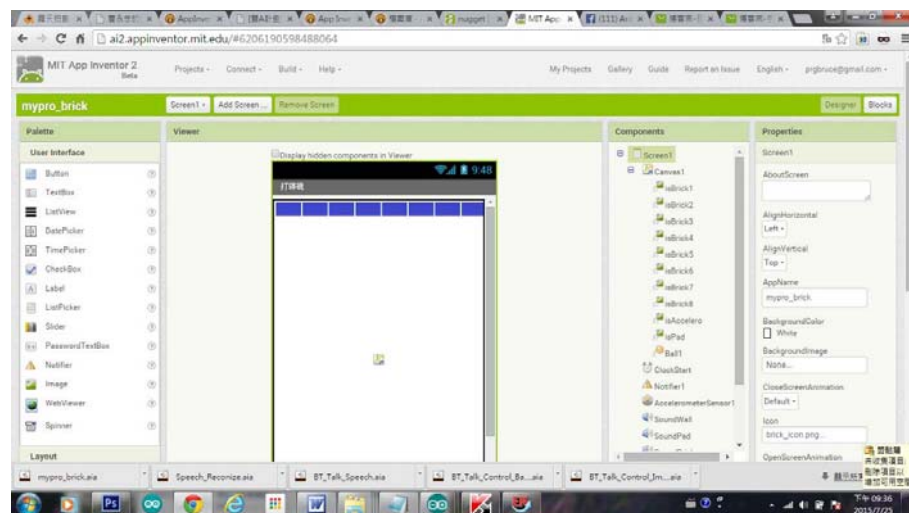


圖 15 上傳範例後開啟該範例

Arduino 藍芽通訊

Arduino 藍芽通訊是本書主要的重點，本節介紹 Arduino 開發板如何使用藍芽模組與與模組之間的電路組立。

如下圖所示，這個實驗我們需要用到的實驗硬體有下圖.(a)的 Arduino Mega

2560 與下圖.(b) USB 下載線、下圖.(c) 藍芽通訊模組(HC-05)：



圖 16 藍芽通訊模組(HC-05)所需零件表

如下圖所示，我們可以看到連接藍芽通訊模組(HC-05)，只要連接 VCC、GND、TXD、RXD 等四個腳位，讀者要仔細觀看，切勿弄混淆了。

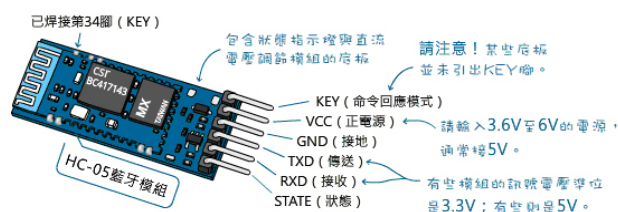


圖 17 附帶底板的 HC-05 藍牙模組接腳圖

資料來源：趙英傑老師網站(<http://swf.com.tw/?p=693>)(趙英傑, 2013, 2014)

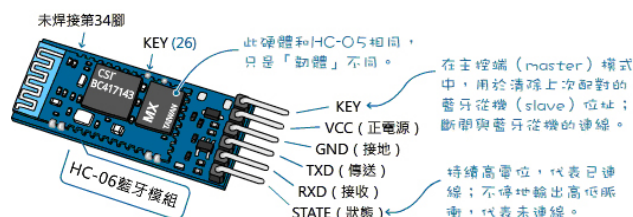


圖 18 附帶底板的 HC-06 藍牙模組接腳圖

資料來源：趙英傑老師網站(<http://swf.com.tw/?p=693>)(趙英傑, 2013, 2014)

如下圖所示，我們可以知道只要將藍芽通訊模組(HC-05)的 VCC 接在 Arduino 開發板 +5V 的腳位(有的要接 3.3V)，GND 接在 Arduino 開發板 GND 的腳位，剩下

的 TXD、RXD 兩個通訊接腳，如果要用實體通訊接腳連接，就可以接在 Arduino 開發板 Tx0、Rx0 的腳位，如果使用 Arduino Mega 2560 開發板又可以多三組通訊腳位可以使用，或者讀者可以使用軟體通訊埠，也一樣可以達到相同功能，只不過速度無法如同硬體的通訊埠那麼快。

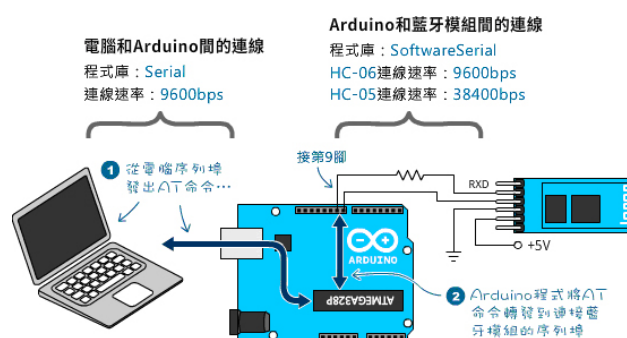



圖 19 連接藍芽模組之簡圖

資料來源：趙英傑老師網站(<http://swf.com.tw/?p=712>)(趙英傑, 2013, 2014)

由於本書使用 HC-05 藍牙模組，所以我們遵從下表來組立電路，來完成本節的實驗：

表 1 HC-05 藍牙模組接腳表

HC-05 藍牙模組	Arduino 開發板接腳
VCC	Arduino +5V Pin
GND	Arduino Gnd Pin
TX	Arduino Mega 2560 digital Pin 8
RX	Arduino Mega 2560 digital Pin 9
	

我們遵照前面所述，將 fayduino Uno 開發板的驅動程式安裝好之後，作者參考上表與上圖之後，完成電路的連接，完成後如下圖所示之藍牙模組 HC-05 接腳實際組裝圖。

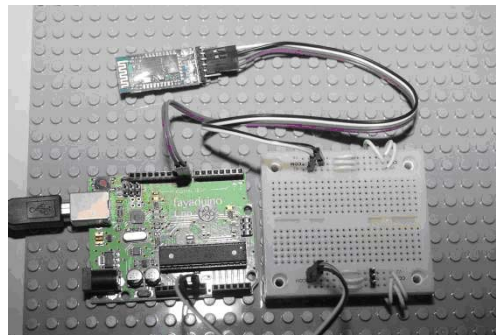


圖 20 藍牙模組 HC-05 接腳實際組裝圖

我們遵照前幾章所述，將 fayduino Uno 開發板的驅動程式安裝好之後，我們打開 Arduino 開發板的開發工具：Sketch IDE 整合開發軟體，撰寫一段程式，如下表所示之藍牙模組 HC-05 測試程式一，來進行藍牙模組 HC-05 的通訊測試。

表 2 藍牙模組 HC-05 測試程式一

藍牙模組 HC-05 測試程式一(BT_Talk)
<pre>// ref HC-05 與 HC-06 藍牙模組補充說明（三）：使用 Arduino 設定 AT 命令 // ref http://swf.com.tw/?p=712 #include <SoftwareSerial.h> // 引用程式庫 // 定義連接藍牙模組的序列埠 SoftwareSerial BT(8, 9); // 接收腳, 傳送腳 char val; // 儲存接收資料的變數 void setup() { Serial.begin(9600); // 與電腦序列埠連線</pre>

```
Serial.println("BT is ready!");

// 設定藍牙模組的連線速率
// 如果是 HC-05，請改成 38400
BT.begin(9600);
}

void loop() {

// 若收到藍牙模組的資料，則送到「序列埠監控視窗」
if (BT.available()) {
    val = BT.read();
    Serial.print(val);
}

// 若收到「序列埠監控視窗」的資料，則送到藍牙模組
if (Serial.available()) {
    val = Serial.read();
    BT.write(val);
}
}
```

讀者可以看到本次實驗-藍牙模組 HC-05 測試程式一結果畫面，如下圖所示，以看到輸入的字元可以轉送到藍芽另一端接收端。

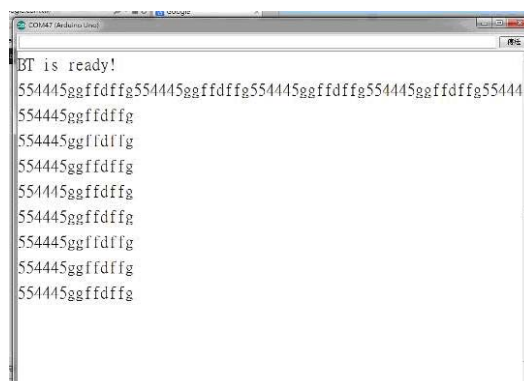


圖 21 藍牙模組 HC-05 測試程式一結果畫面

手機安裝藍芽裝置

如下圖所示，一般手機、平板的主畫面或程式集中可以選到『設定：Setup』。



圖 22 手機主畫面

如下圖所示，點入『設定：Setup』之後，可以到『設定：Setup』的主畫面，，
如您的手機、平板的藍芽裝置未打開，請將藍芽裝置開啟。



圖 23 設定主畫面

如下圖所示，開啟藍芽裝置之後，可以看到目前可以使用的藍芽裝置。



圖 24 目前已連接藍芽畫面

如下圖所示，我們要將我們要新增的藍芽裝置加入手機、平板之中，請點選下圖紅框處：搜尋裝置，方能增加新的藍芽裝置。



圖 25 搜尋藍芽裝置

如下圖所示，當我們要找到新的藍芽裝置，點選它之後，會出現下圖畫面，要

求使用者輸入配對的 Pin 碼，一般為『0000』或『1234』。



圖 26 第一次配對-要求輸入配對碼

如下圖所示，我們可以輸入配對的 Pin 碼，一般為『0000』或『1234』，來完成配對的要求。



圖 27 藍芽要求配對

如下圖所示，我們可以輸入配對的 Pin 碼，一般為『0000』或『1234』，來完成配對的要求，本書例子為『1234』。



圖 28 輸入配對密碼(1234)

如下圖所示，如果輸入配對的 Pin 碼正確無誤，則會完成配對，該藍芽裝置會加入手機、平板的藍芽裝置清單之中。



圖 29 完成配對後-出現在已配對區

如下圖所示，完成後，手機、平板會顯示已完成配對的藍芽裝置清單。



圖 30 目前已連接藍芽畫面

如下圖所示，完成配對的藍芽裝置後，我們可以用回上頁回到設定主畫面，完成新增藍芽裝置的配對。



圖 31 完成藍芽配對等完成畫面

安裝 Bluetooth RC APPs 應用程式

本書再測試 Arduino 開發板連接藍芽裝置，為了測試這些程式是否傳輸、接收命令是否正確，我們會先行安裝市面穩定的藍芽通訊 APPs 應用程式。

本書使用 Fadjar Hamidi F 公司攥寫的『Bluetooth RC』，其網址：
https://play.google.com/store/apps/details?id=appinventor.ai_test.BluetoothRC&hl=zh_TW
 ，讀者可以到該網址下載之。

本章節主要是介紹讀者如何安裝 Fadjar Hamidi F 公司攥寫的『Bluetooth RC』。

如下圖所示，在手機主畫面進入 play 商店。

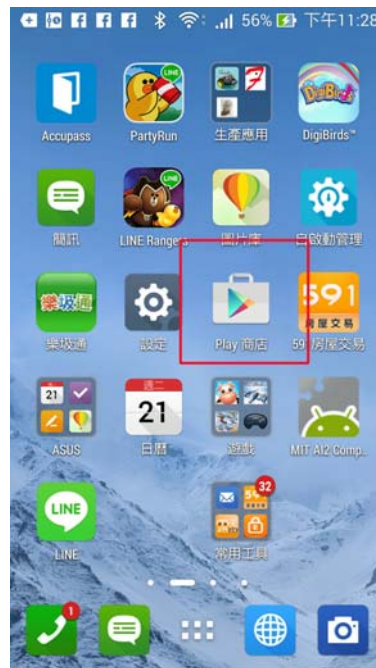


圖 32 手機主畫面進入 play 商店

如下圖所示，下圖為 play 商店主畫面。



圖 33 Play 商店主畫面

如下圖紅框處所示，我們在 Google Play 商店主畫面 - 按下查詢鈕。

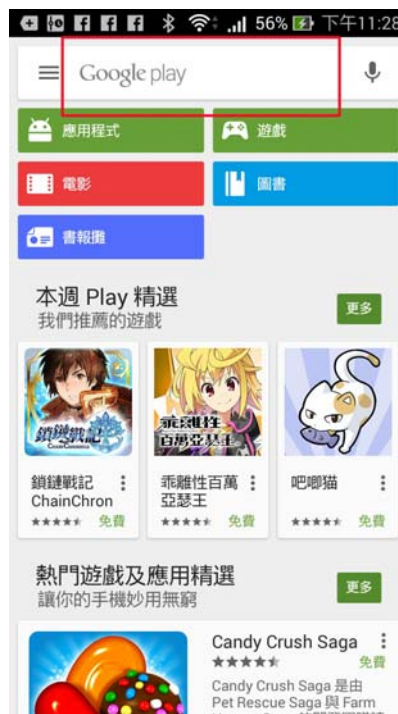


圖 34 Play 商店主畫面 - 按下查詢鈕

如下圖紅框處所示，我們在輸入『Bluetooth RC』查詢該 APPs 應用程式。

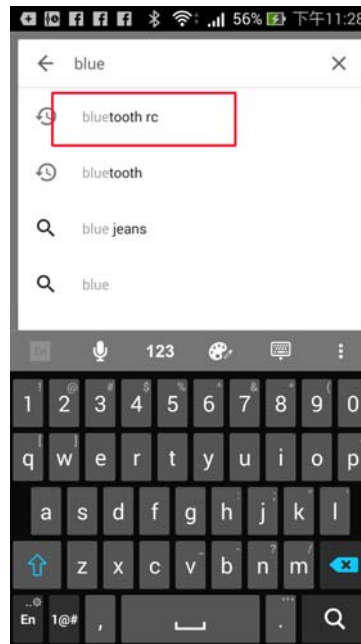


圖 35 Play 商店主畫面 - 輸入查詢文字

如下圖紅框處所示，我們在輸入『Bluetooth RC』查詢，找到 BluetoothRC 應用程式。

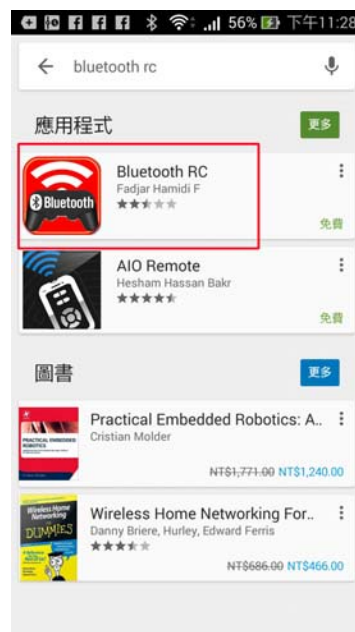


圖 36 找到 BluetoothRC 應用程式

如下圖紅框處所示，找到 BluetoothRC 應用程式 - 點下安裝。

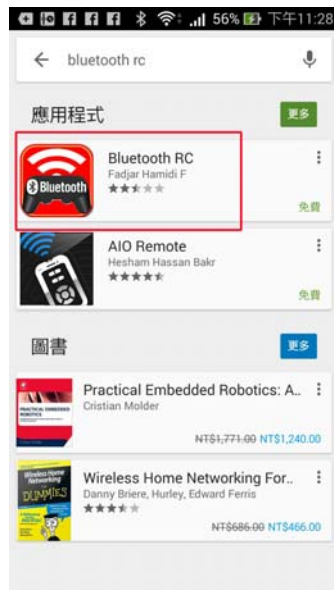


圖 37 找到 BluetoothRC 應用程式 -點下安裝

如下圖紅框處所示，點下『接受』，進行安裝。



圖 38 BluetoothRC 應用程式安裝主畫面要求授權

如下圖所示，BluetoothRC 應用程式安裝中。



圖 39 BluetoothRC 應用程式安裝中

如下圖所示，BluetoothRC 應用程式安裝中。



圖 40 BluetoothRC 應用程式安裝中二

如下圖所示，BluetoothRC 應用程式安裝完成。

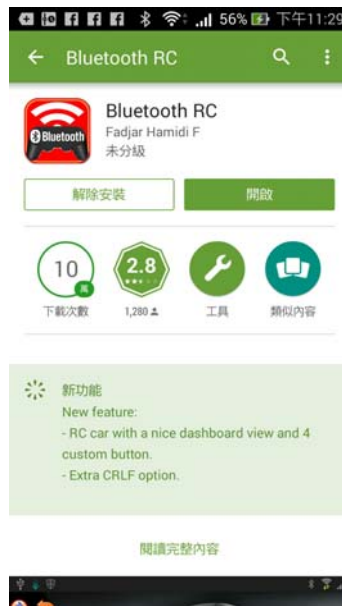


圖 41 BluetoothRC 應用程式安裝完成

如下圖紅框處所示，我們可以點選『開啟』來執行 BluetoothRC 應用程式。



圖 42 BluetoothRC 應用程式安裝完成後執行

如下圖所示，安裝好 BluetoothRC 應用程式之後，一般說來手機、平板的桌面或程式集中會出現『BluetoothRC』的圖示。



圖 43 BluetoothRC 應用程式安裝完成後的桌面

BluetoothRC 應用程式通訊測試

一般而言，如下圖所示，我們安裝好 BluetoothRC 應用程式之後，手機、平板的桌面或程式集中會出現『BluetoothRC』的圖示。



圖 44 桌面的 BluetoothRC 應用程式

如下圖所示，我們點選手機、平板的桌面或程式集中『BluetoothRC』的圖示，進入 BluetoothRC 應用程式。



圖 45 執行 BluetoothRC 應用程式

如下圖所示，為 BluetoothRC 應用程式進入系統的抬頭畫面。



圖 46 BluetoothRC init 應用程式執行中

如下圖所示，為 BluetoothRC 應用程式主畫面。



圖 47 BluetoothRC 應用程式執行主畫面

如下圖紅框處所示，首先，我們要為 BluetoothRC 應用程式選定工作使用的藍芽裝置，讀者要注意，系統必須要開啟藍芽裝置，且已將要連線的藍芽裝置配對完成後，並已經在手機、平板的藍芽已配對清單中，方能被選到。



圖 48 BluetoothRC 應用程式執行主畫面 - 選取藍芽裝置

如下圖所示，我們要可以選擇已經在手機、平板已配對清單中的藍芽，選定為

BluetoothRC 應用程式工作使用的藍芽裝置。

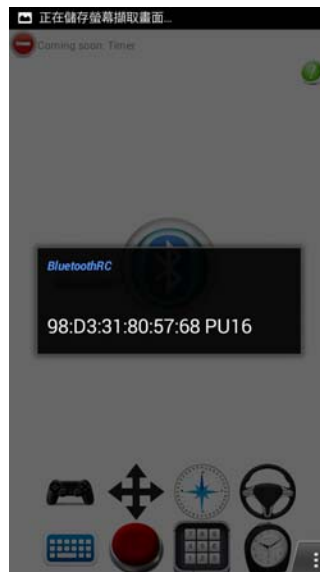


圖 49 BluetoothRC 應用程式執行主畫面 - 已配對藍芽裝置列表

如下圖紅框處所示，我們要可以選擇已經在手機、平板已配對清單中的藍芽，進行 BluetoothRC 應用程式工作使用。

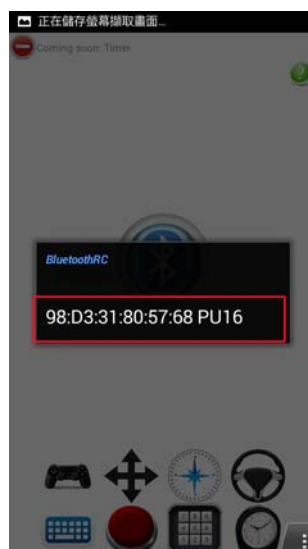


圖 50 BluetoothRC 應用程式執行主畫面 - 選取配對藍芽裝置

如下圖紅框處所示，系統會出現目前 BluetoothRC 應用程式工作使用藍芽裝置之 MAC。



圖 51 BluetoothRC 應用程式執行主畫面 - 完成選取藍芽裝置

如下圖紅框處所示，點選 BluetoothRC 應用程式執行主畫面紅框處 - 啟動文字通訊功能。

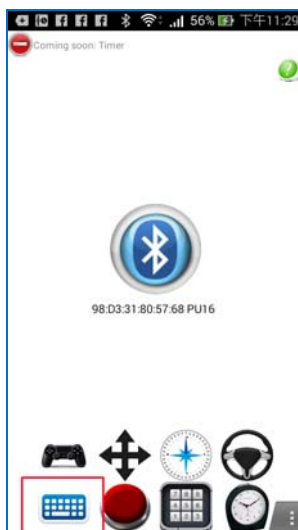


圖 52 BluetoothRC 應用程式執行主畫面 - 啟動文字通訊功能

如下圖所示，為 BluetoothRC 文字通訊功能主畫面。

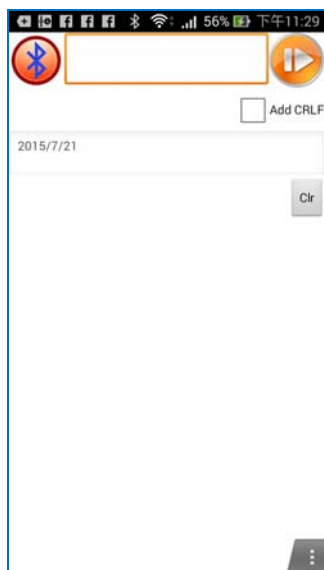


圖 53 BluetoothRC 文字通訊功能主畫面

如下圖紅框處所示，啟動藍芽通訊。

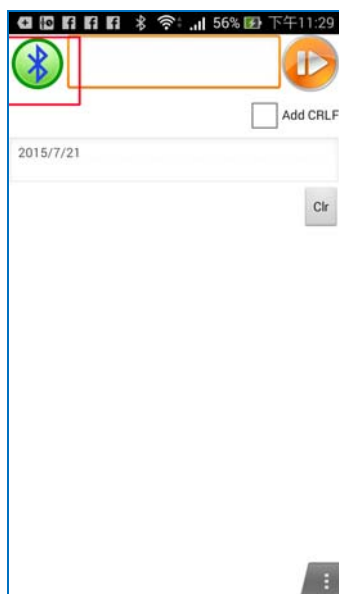


圖 54 BluetoothRC 文字通訊功能主畫面 -完成 開啟藍芽通訊

如下圖紅框處所示，我們可以輸入任何文字，進行藍芽傳輸。



圖 55 BluetoothRC 文字通訊功能主畫面 - 輸入送出文字

如下圖紅框處所示，按下向右三角形，將上方輸入的文字，透過選定的藍芽裝置傳輸到連接的另一方藍芽裝置。

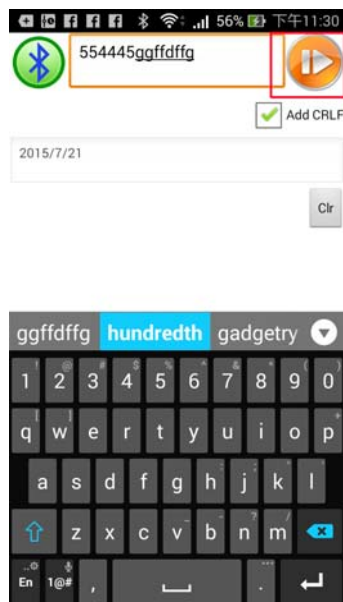
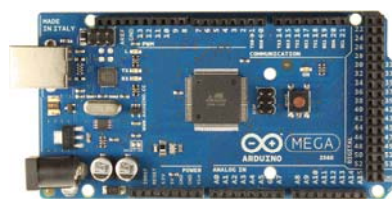


圖 56 BluetoothRC 文字通訊功能主畫面 - 傳送輸入文字

Arduino 藍芽模組控制

由於本章節只要使用藍芽模組(HC-05/HC-06)，所以本實驗仍只需要一塊 Arduino Uno 開發板、USB 下載線、8 藍芽模組(HC-05/HC-06)。

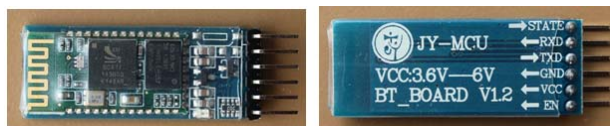
如下圖所示，這個實驗我們需要用到的實驗硬體有下圖.(a)的 Arduino Mega2560 與下圖.(b) USB 下載線、下圖.(c) 藍芽模組(HC-05/HC-06)：



(a). Arduino Mega2560



(b). USB 下載線

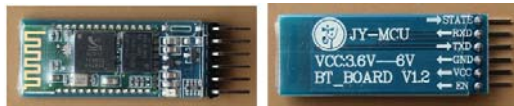


(c). 藍芽模組(HC-05/HC-06)

圖 57 藍芽模組(HC-05/HC-06)所需零件表

由於本書使用藍芽模組，所以我們遵從下表來組立電路，來完成本節的實驗：

表 3 使用手機控制風扇接腳表

藍芽模組(HC-05)	Arduino 開發板
VCC	Arduino Mega 2560 +5V
GND	Arduino Mega 2560 GND
TX	Arduino Mega 2560 digitalPin 11
RX	Arduino Mega 2560 digitalPin 12
 <p>藍芽模組(HC-05/HC-06)</p>	

我們遵照前幾章所述，將 **fayduino Uno** 開發板的驅動程式安裝好之後，我們打開 **Arduino** 開發板的開發工具：**Sketch IDE** 整合開發軟體，撰寫一段程式，如下表所示之藍芽模組(HC-05/HC-06)測試程式一，並將之編譯後上傳到 **Arduino** 開發板。

表 4 藍芽模組(HC-05/HC-06)

藍芽模組(HC-05/HC-06) (BT_Talk)
<pre> #include <SoftwareSerial.h> // 引用程式庫 // 定義連接藍牙模組的序列埠 SoftwareSerial BT(11, 12); // 接收腳, 傳送腳 char val; // 儲存接收資料的變數 void setup() { Serial.begin(9600); // 與電腦序列埠連線 Serial.println("BT is ready!"); // 設定藍牙模組的連線速率 // 如果是 HC-05，請改成 38400 BT.begin(9600); } void loop() { // 若收到藍牙模組的資料，則送到「序列埠監控視窗」 if (BT.available()) { val = BT.read(); Serial.print(val); } // 若收到「序列埠監控視窗」的資料，則送到藍牙模組 if (Serial.available()) { val = Serial.read(); BT.write(val); </pre>

```
}  
}
```

如下圖所示，我們執行後，會出現『BT is ready!』後，在畫面中可以接收到藍芽模組收到的資料，並顯示再監控畫面之中。



圖 58 Arduino 通訊監控畫面-監控藍芽通訊內容

如下圖所示，我們執行後，會出現『BT is ready!』後，我們再上方文字輸入區中，輸入文字。

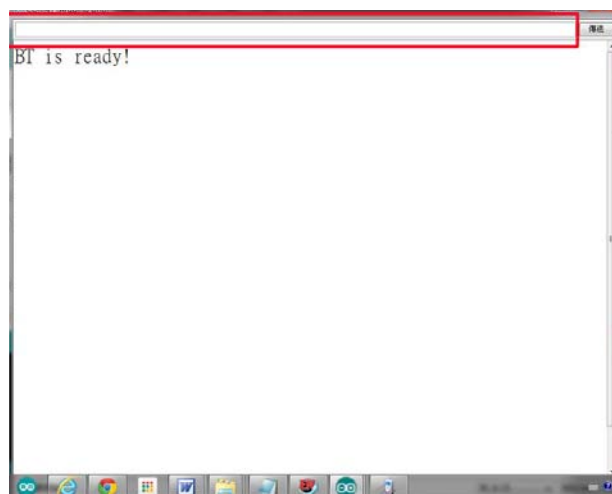


圖 59 Arduino 通訊監控畫面-輸入送出通訊內容字元輸入區

如下圖所示，我們執行後，會出現『BT is ready!』後，我們再上方文字輸入區中，輸入文字，按下右上方的『傳送』鈕，也會把上方文字輸入區中所有文字傳送到藍芽模組配對連接的另一端。



圖 60 Arduino 通訊監控畫面-送出輸入區內容

如下圖所示，我們執行後，藍芽模組配對連接的另一端上圖上方文字輸入區中輸入的文字。

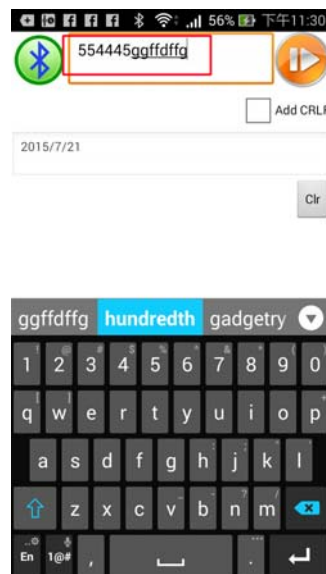


圖 61 BluetoothRC 文字通訊功能主畫面 - 輸入送出文字

如下圖所示，同樣的，我們執行手機、平板上的 Bluetooth RC 應用程式後，再下圖上方文字輸入區中輸入的文字。

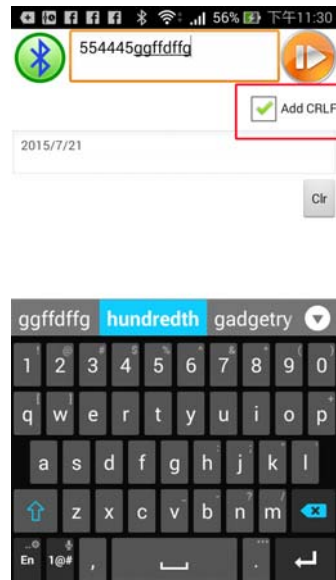


圖 62 BluetoothRC 文字通訊功能主畫面 - 傳送輸入文字(含回行鍵)

如下圖所示，同樣的，Arduino 通訊監控畫面會收到我們執行手機、平板上的 Bluetooth RC 應用程式其中文字輸入區中輸入的文字。



圖 63 Arduino 通訊監控畫面-送出輸入區內容

手機藍芽基本通訊功能開發

由於我們使用 Android 作業系統的手機或平板與 Arduino 開發板的裝置進行控制，由於手機或平板的設計限制，通常無法使用硬體方式連接與通訊，所以本節專門介紹如何在手機、平板上如何使用常見的藍芽通訊來通訊，本節主要介紹 App Inventor 2 如何建立一個藍芽通訊模組。

首先，如下圖所示，我們在 App Inventor 2 程式模塊編輯畫面之中，開立一個新專案。

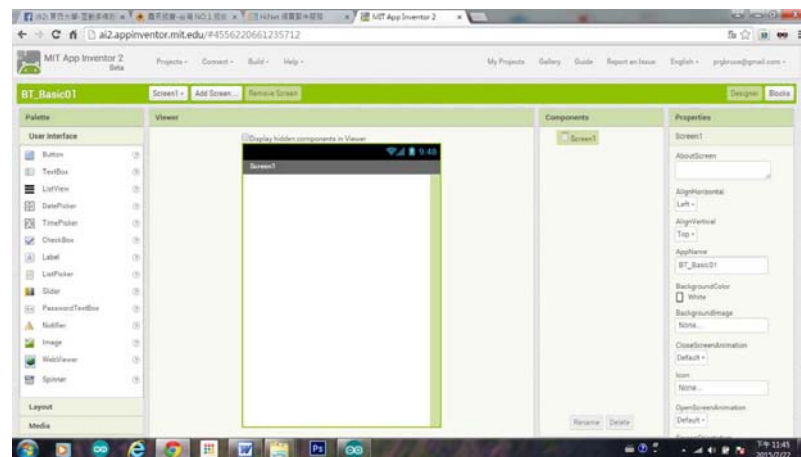


圖 64 建立新專案

首先，如下圖所示，我們在先拉出 VerticalArrangement1。

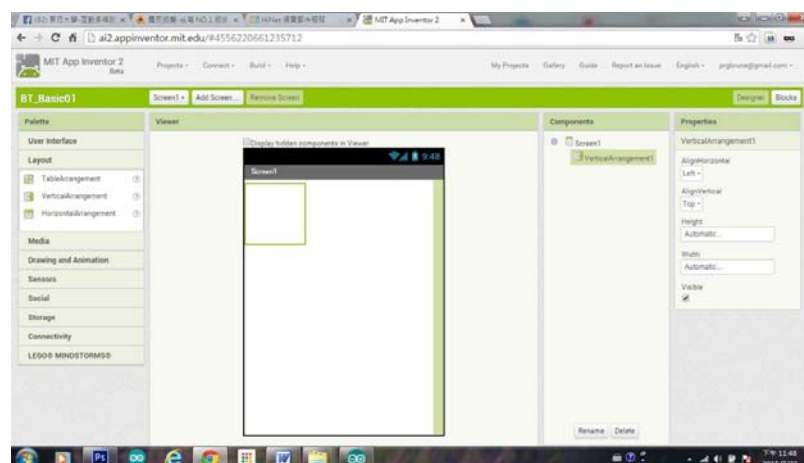


圖 65 拉出 VerticalArrangement1

如下圖所示，我們在拉出第一個 HorizontalArrangement1。

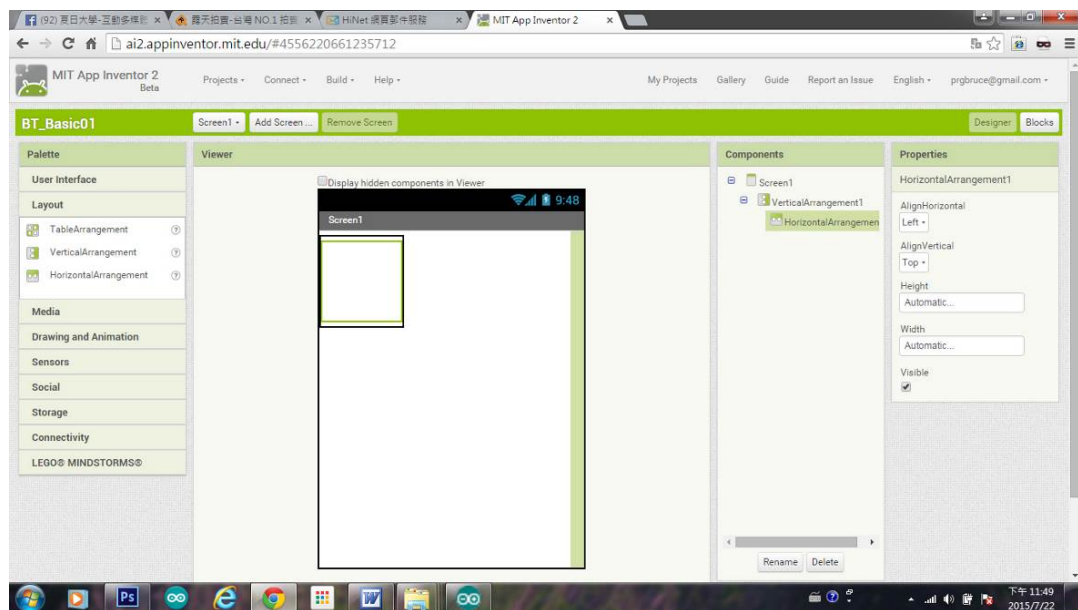


圖 66 拉出第一個 HorizontalArrangement1

如下圖所示，我們在拉出第二個 HorizontalArrangement2。

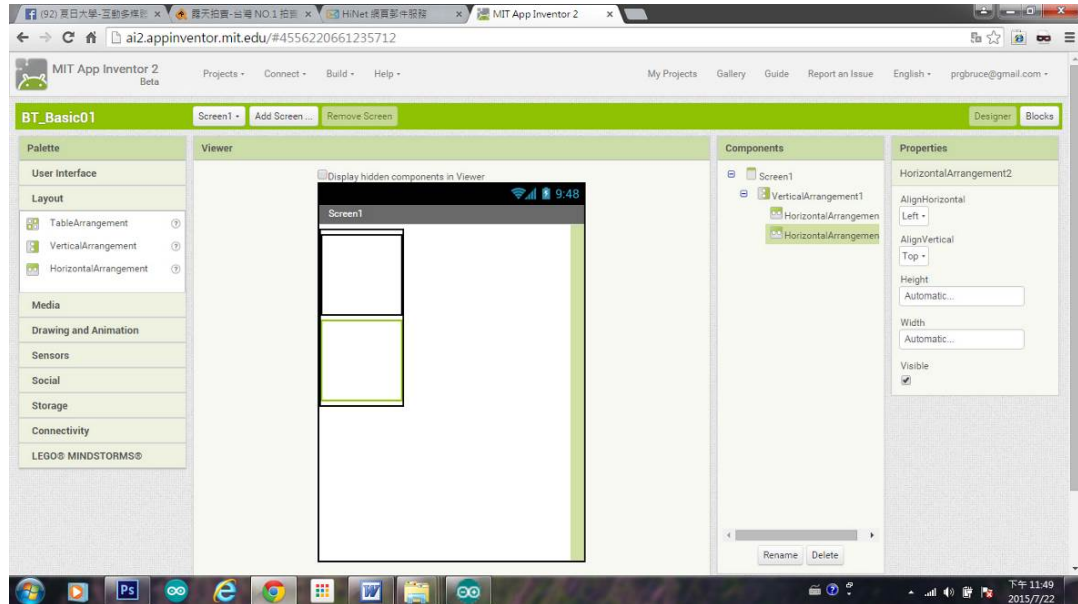


圖 67 拉出第二個 HorizontalArrangement2

如下圖所示，我們在第一個 HorizontalArrangement 內拉出顯示傳輸內容之 Label。

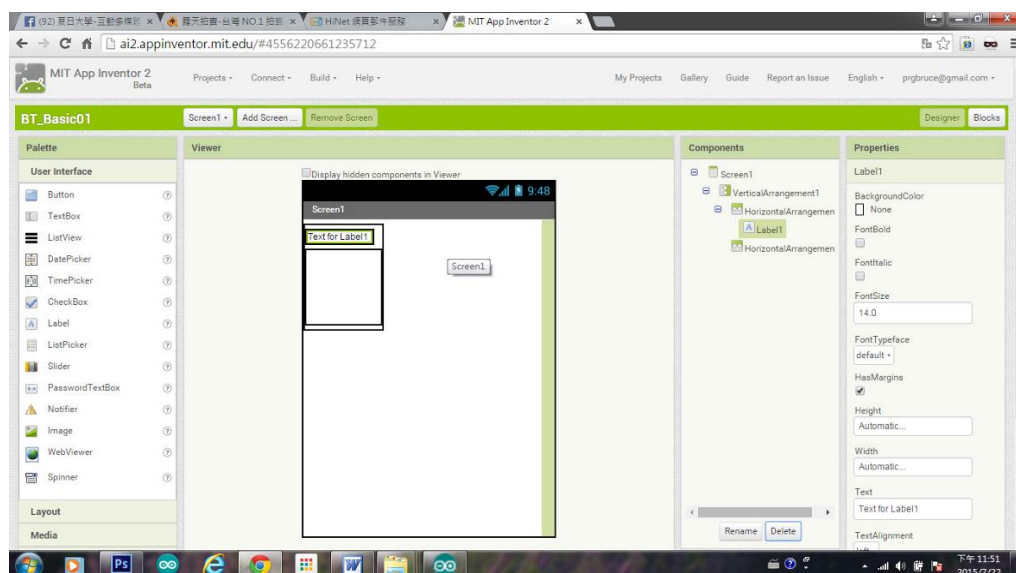


圖 68 拉出顯示傳輸內容之 Label

如下圖所示，我們修改在第一個 HorizontalArrangement 內拉出顯示傳輸內容之 Label 的顯示文字。

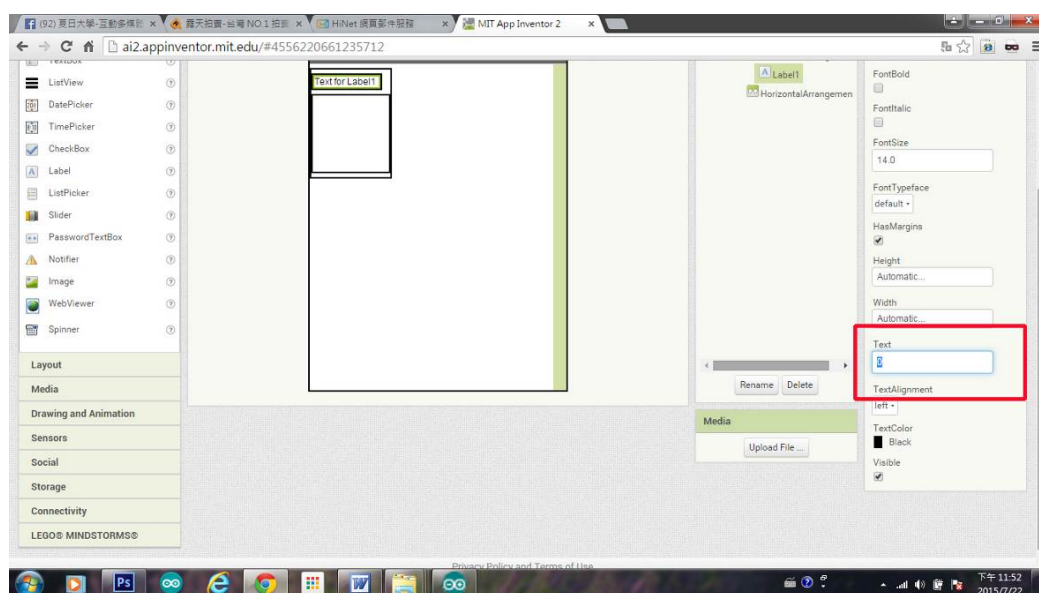


圖 69 修改顯示傳輸內容之 Label 內容值

如下圖所示，我們修改在第二個拉出的 HorizontalArrangement2 內拉出拉出 ListPictcker(選藍芽裝置用)。

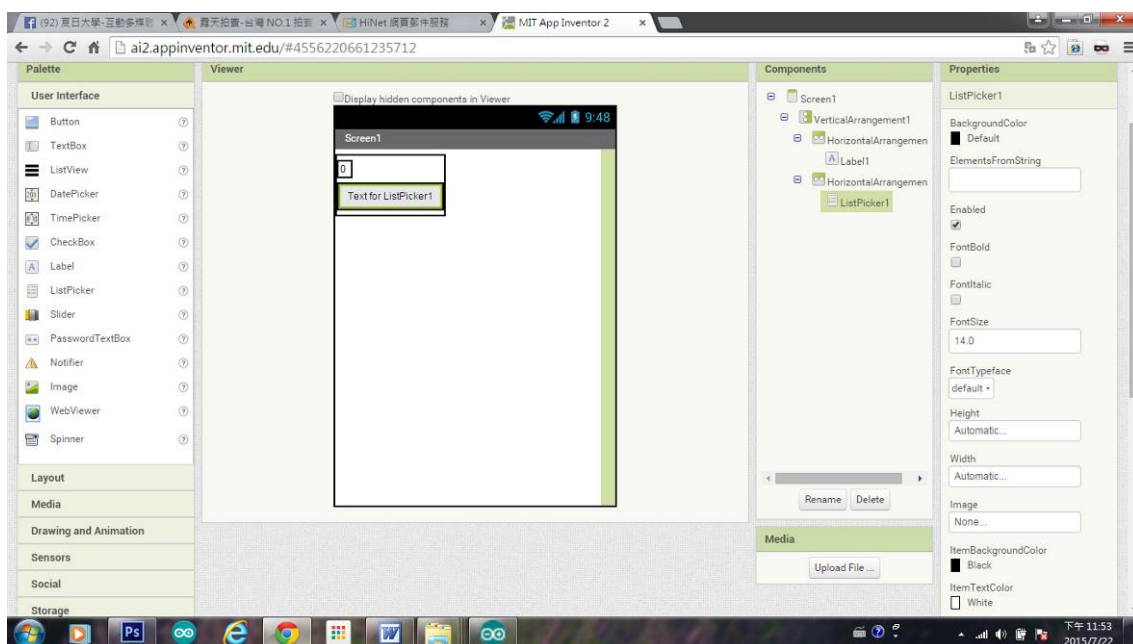


圖 70 拉出 ListPictker(選藍芽裝置用)

如下圖所示，我們修改在第二個拉出的 HorizontalArrangement2 內拉出拉出 ListPictker(選藍芽裝置用)改變其顯示的文字為『Select BT』。

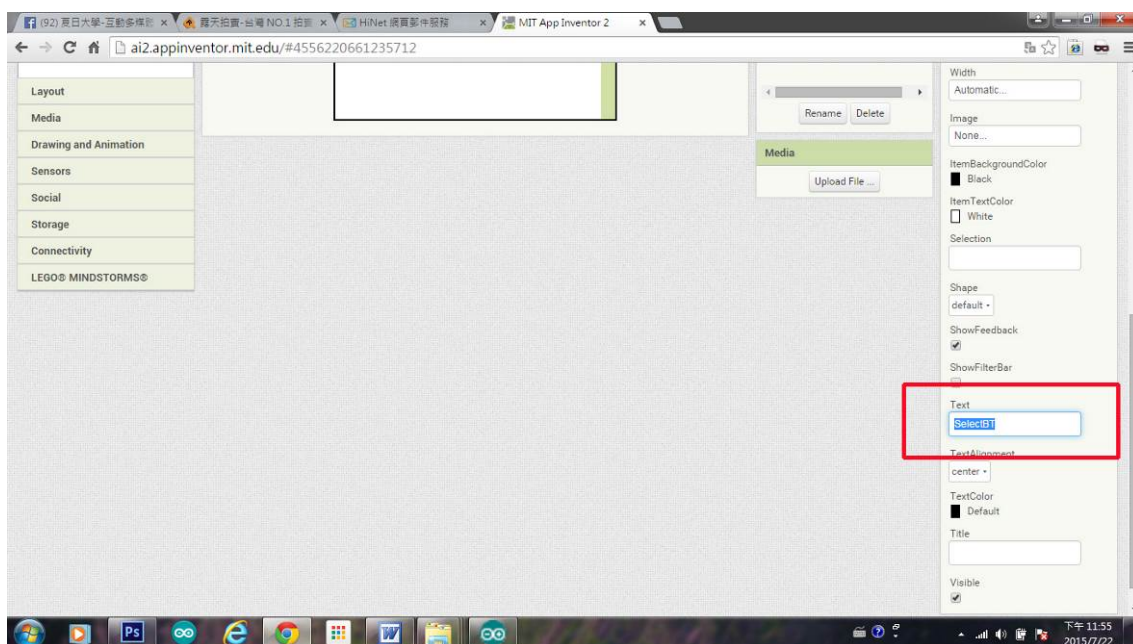


圖 71 修改 ListPictker 顯示名稱

如下圖所示，拉出藍芽 Client 物件。

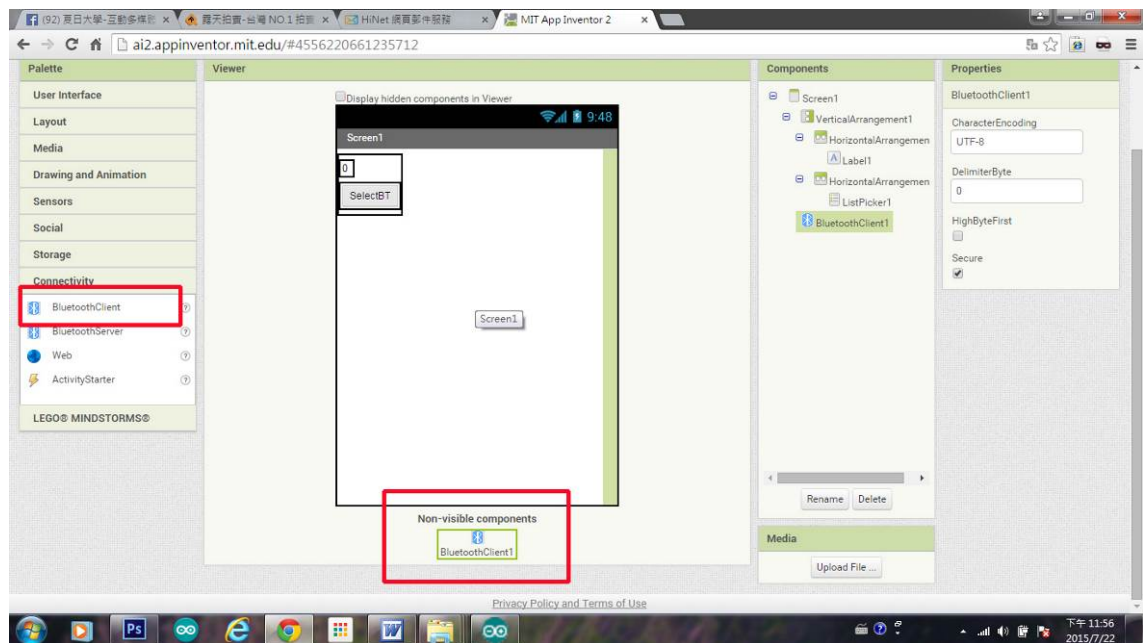


圖 72 拉出藍芽 Client 物件

如下圖所示，拉出驅動藍芽的時間物件。

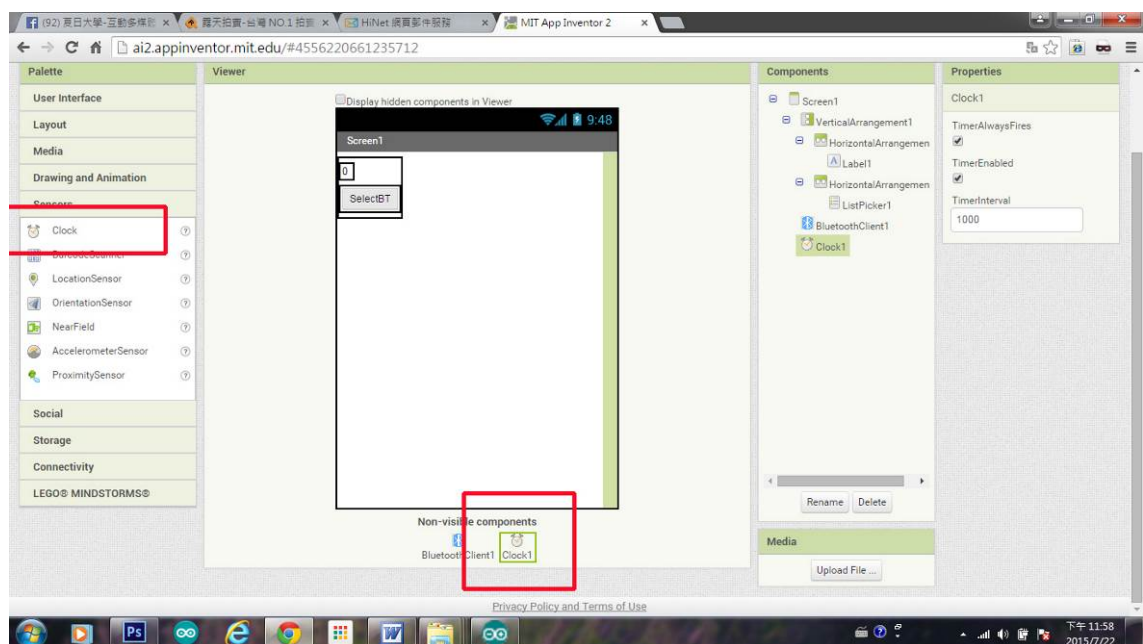


圖 73 拉出驅動藍芽的時間物件

如下圖所示，我們修改拉出驅動藍芽的時間物件的名稱為『BTRun』。

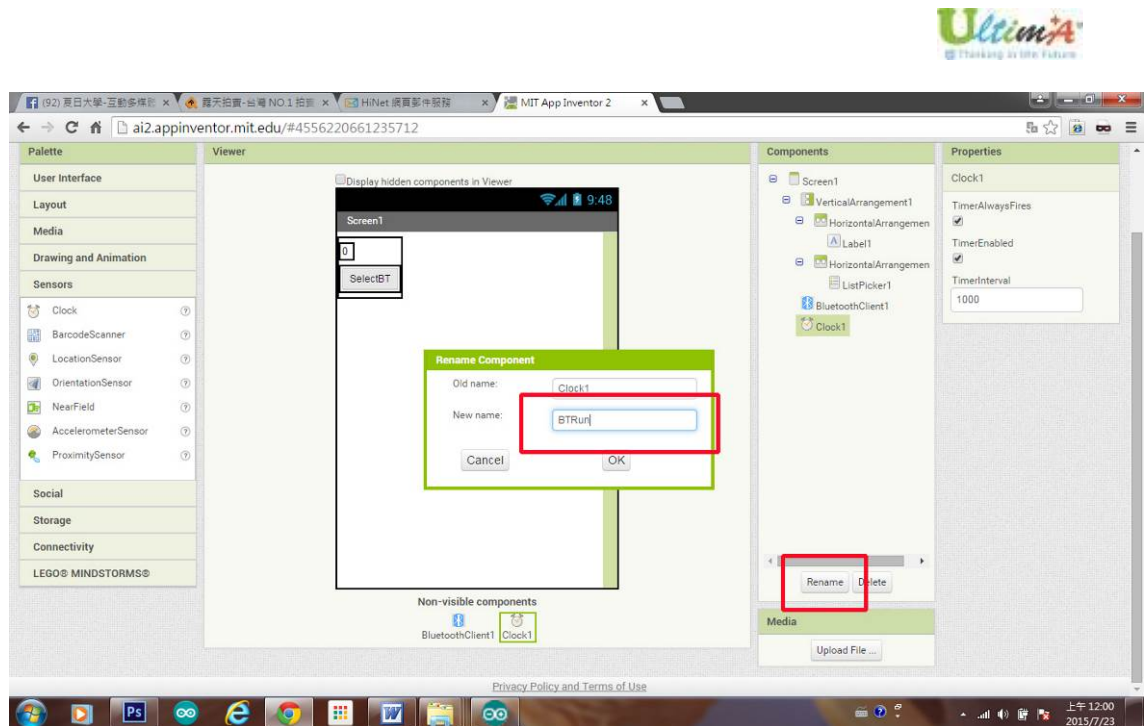


圖 74 修改驅動藍芽的時間物件的名字

如下圖所示，我們為了編修程式，請點選如下圖所示之紅框區『Blocks』按鈕。

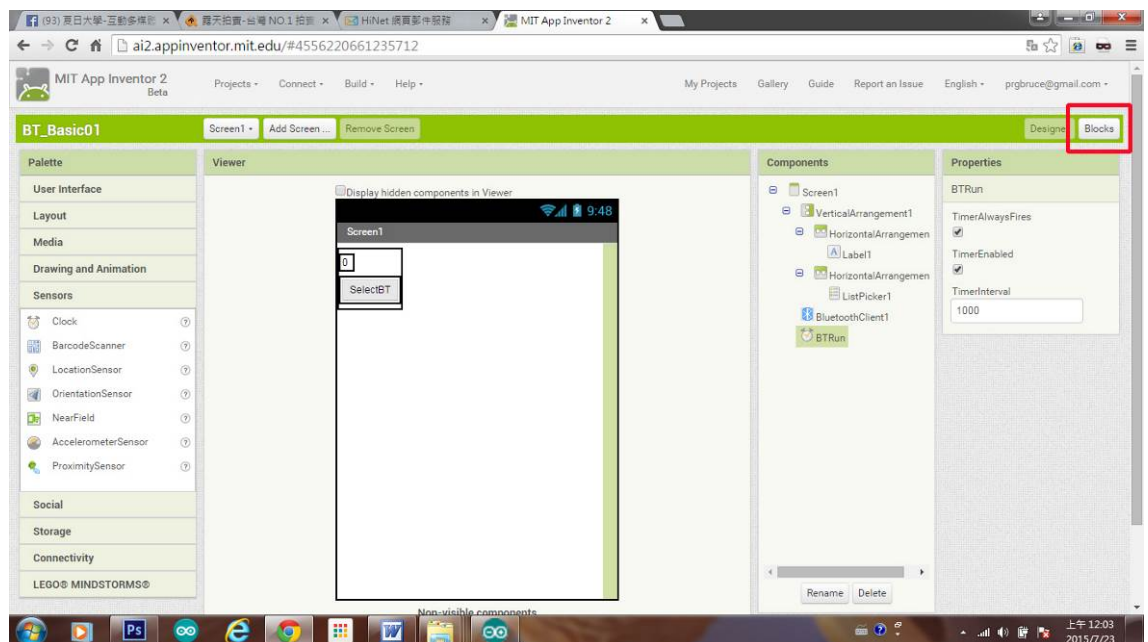


圖 75 切換程式設計模式

如下圖所示，，下圖所示之紅框區為 App Inventor 2 的程式編輯區。

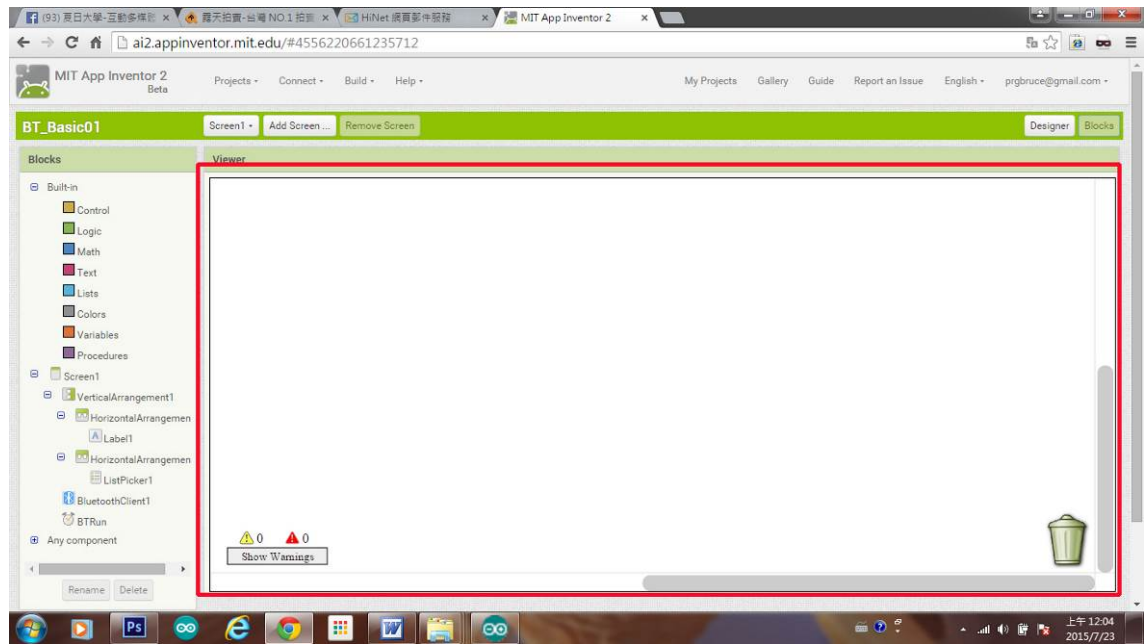


圖 76 程式設計模式主畫面

如下圖所示，我們在 App Inventor 2 的程式編輯區，建立 BTChar 變數。

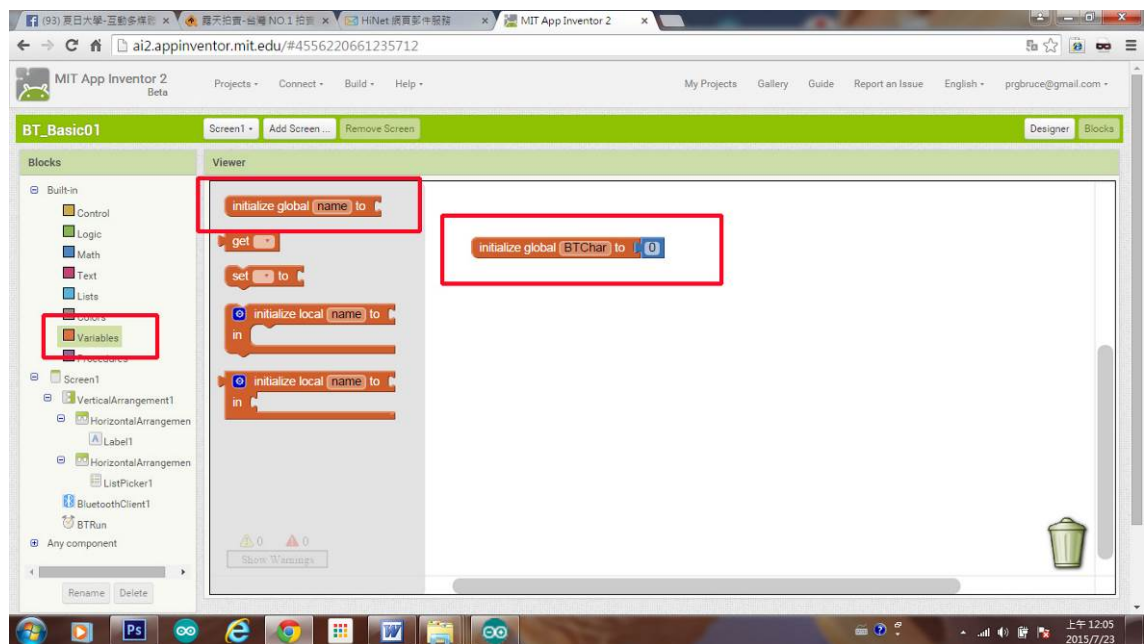


圖 77 建立 BTChar 變數

為了建全的系統，如下圖所示，我們進行系統初始化，在 Screen1.initialize 建立下列敘述。

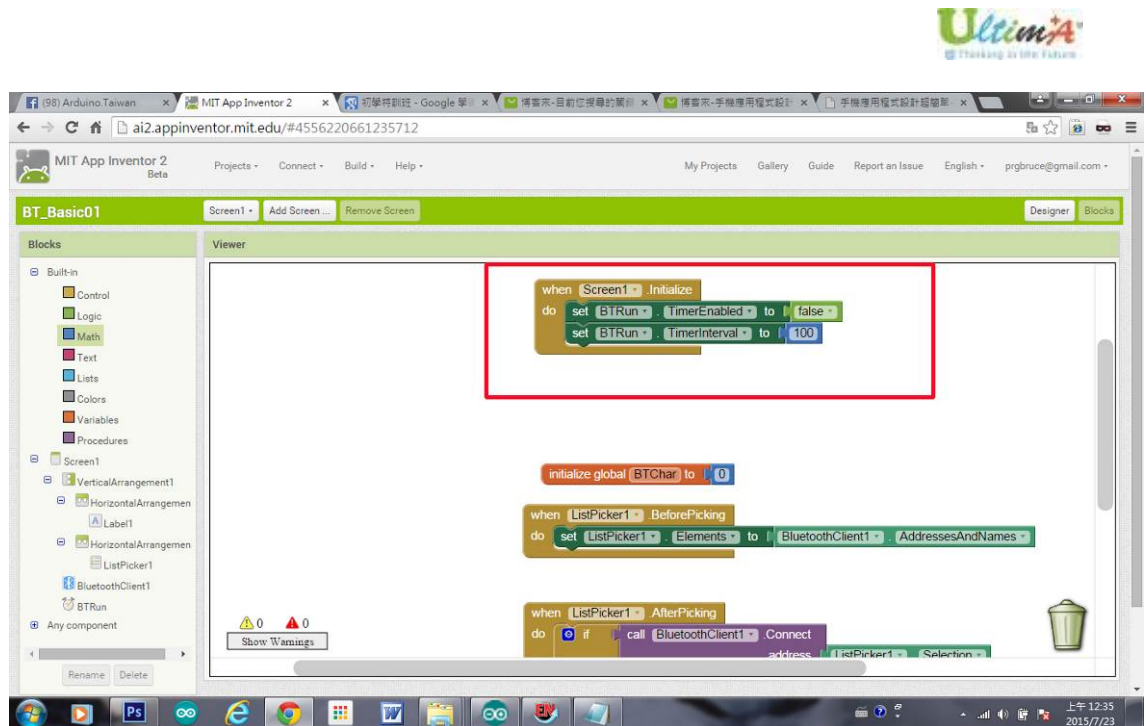


圖 78 系統初始化

首先，在點選藍芽裝置『ListPicker1』下，如下圖所示，我們在 ListPicker1.BeforePicking 建立下列敘述。



圖 79 將已配對的藍芽資料填入 ListPicker

首先，在點選藍芽裝置『ListPicker1』下，撰寫『判斷選到藍芽裝置後連接選取藍芽裝置』，如下圖所示，我們在 ListPicker1.AfterPicking 建立下列敘述。

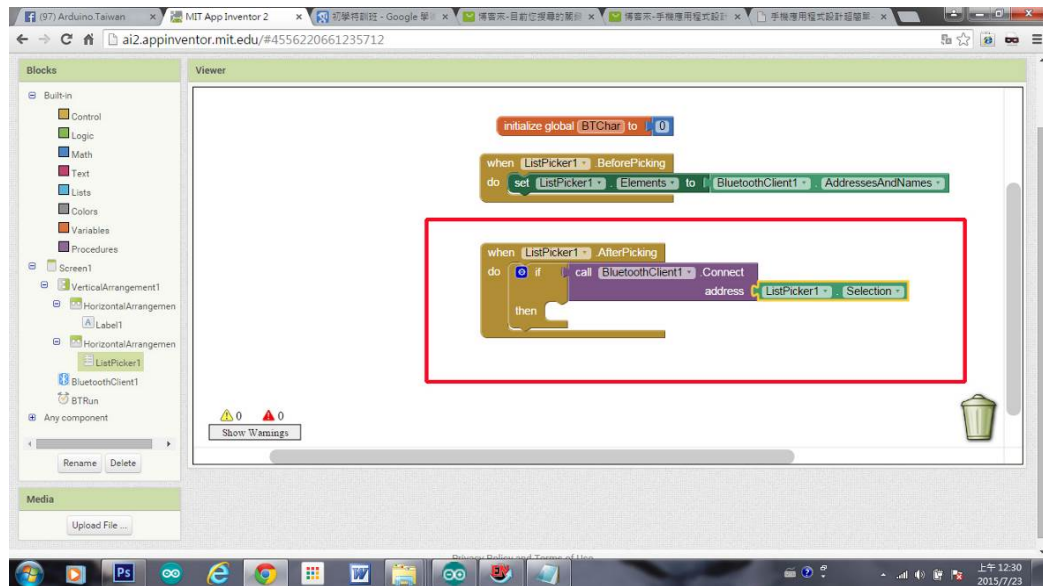


圖 80 判斷選到藍芽裝置後連接選取藍芽裝置

如下圖所示，在點選藍芽裝置『ListPicker1』下，我們在 ListPicker1.AfterPicking 建立下列敘述，因為已經選好藍芽裝置，所以不需要選藍芽裝置『ListPicker1』，所以將它關掉，並開啟藍芽通訊程式所需要的『BTRun』時間物件。

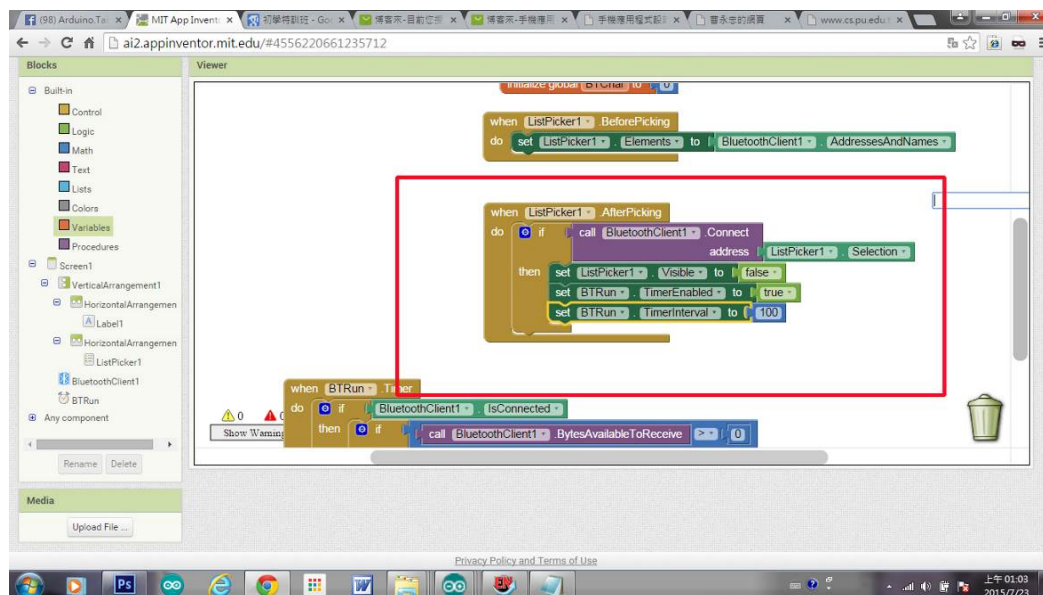


圖 81 連接藍芽後將 ListPickere 關掉

如下圖所示，在藍芽通訊程式所需要的『BTRun』時間物件下，我們為了確定藍芽已完整建立通訊，先行判斷是否藍芽已完整建立通訊。

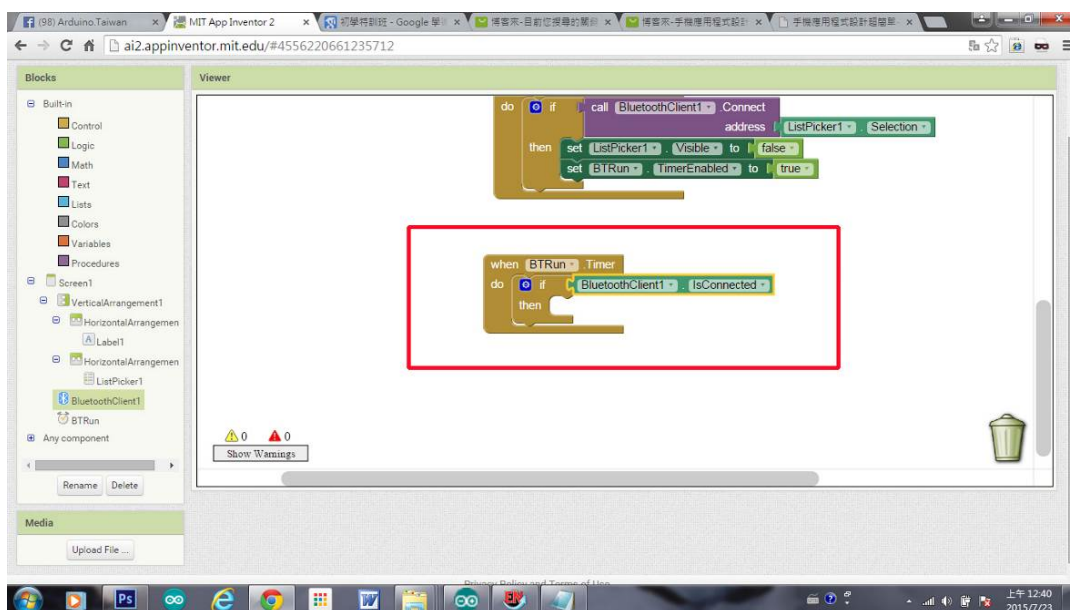


圖 82 定時驅動藍芽-判斷是否藍芽連線中

如下圖所示，在藍芽通訊程式所需要的『BTRun』時間物件下，如果藍芽已完整建立通訊，再判斷判斷是否藍芽有資料傳入。

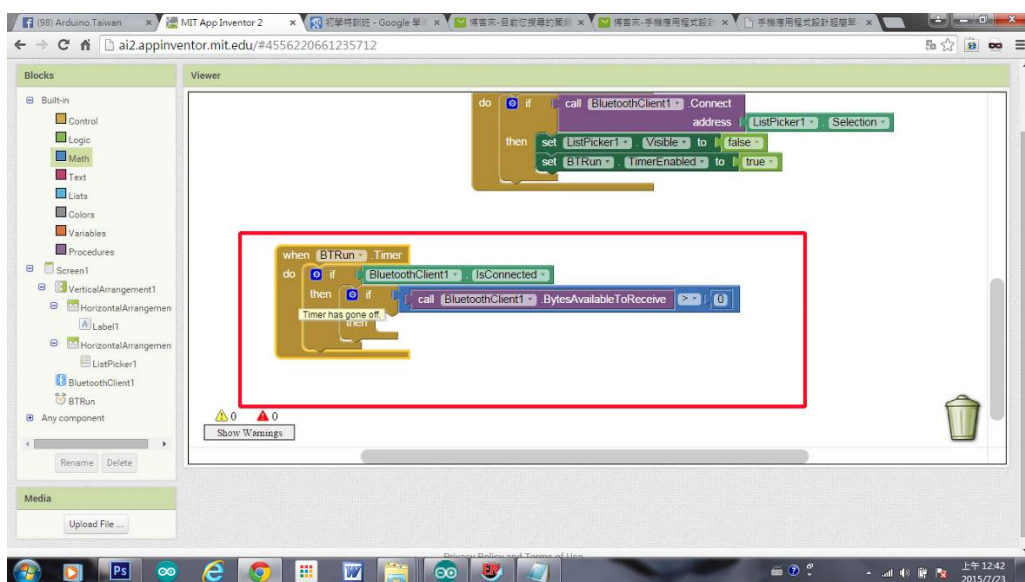


圖 83 定時驅動藍芽-判斷是否藍芽有資料傳入

如下圖所示，在藍芽通訊程式所需要的『BTRun』時間物件下，如果藍芽已完整建立通訊，再判斷判斷是否藍芽有資料傳入，再將此資料存入『BTChar』變數

裡面。

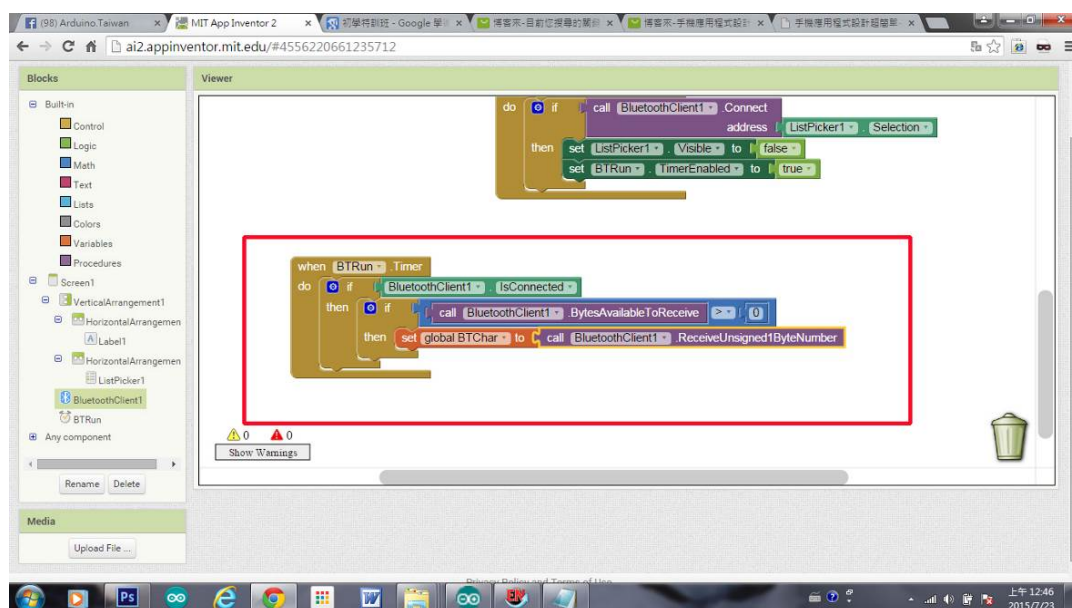


圖 84 定時驅動藍芽-讀出藍芽資料送入變數

如下圖所示，再將『BTChar』變數顯示在畫面的 Label1 的 Text 上。

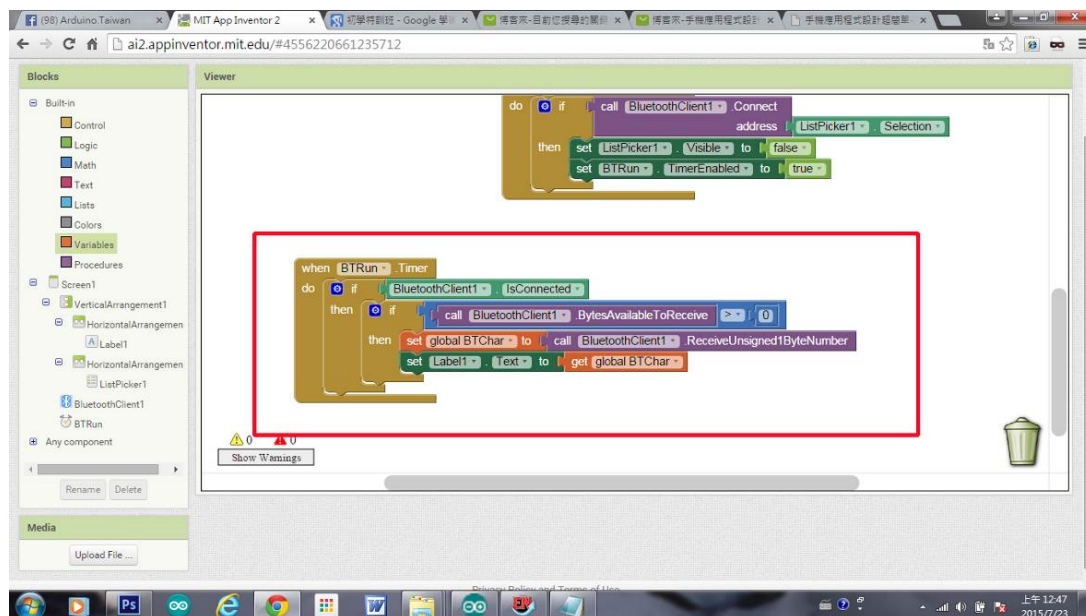


圖 85 定時驅動藍芽-顯示藍芽資料到 Label 物件

首先，如下圖所示，我們在 App Inventor 2 程式模塊編輯畫面之中，在『Connect』

的選單下，選取 AICompanion。

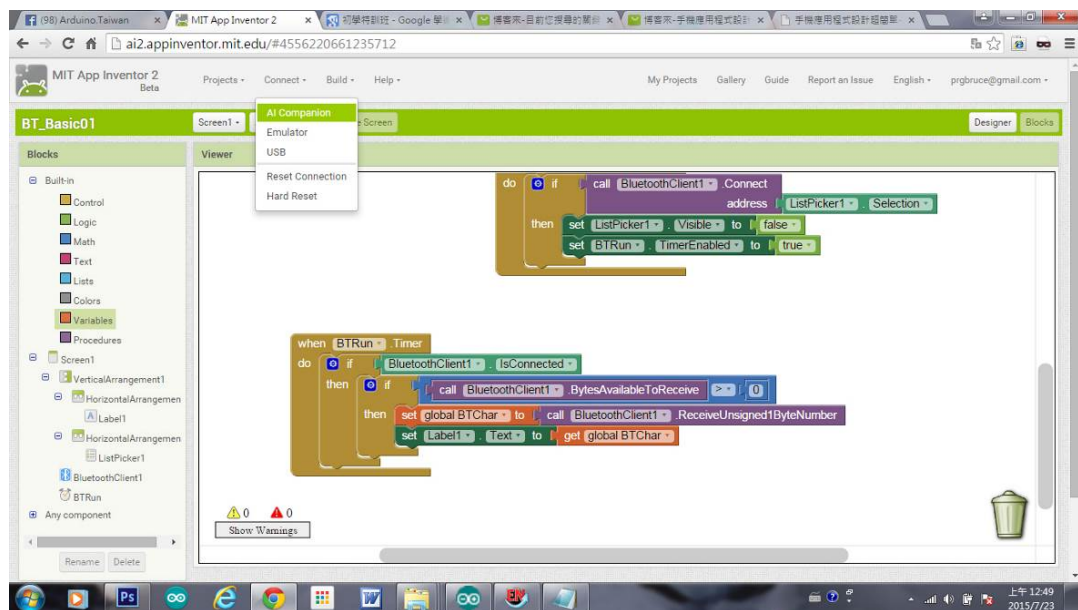


圖 86 啟動手機測試功能

如下圖所示，系統會出現一個 QR Code 的畫面。



圖 87 手機 QR CODE

如下圖所示，我們在使用 Android 的手機、平板，執行已安裝好的『MIT App

Inventor 2 Companion』，點選之後進入如下圖。



圖 88 啟動 MIT_AI2_Companion

如下圖所示，我們在選擇『scan QR code』，點選之後進入如下圖。

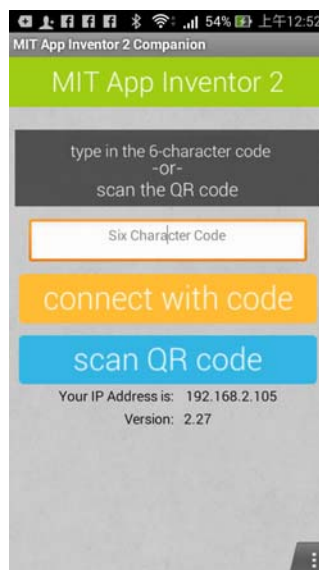


圖 89 掃描 QRCode

如下圖所示，手機會啟動掃描 QR code 的程式功能，這時後只要將手機、平板的 Camera 鏡頭描準畫面的 QR Code 就可以了。

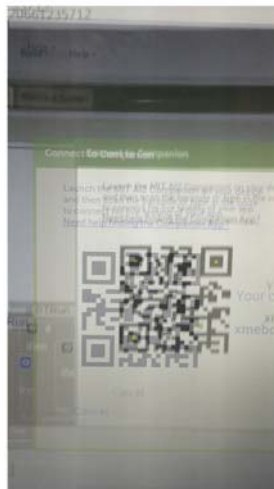


圖 90 掃描 QRCodeing

如下圖所示，如果手機會啟動掃描 QR code 成功的話，系統會回傳 QR Code 碼到如下圖所示的紅框之中。

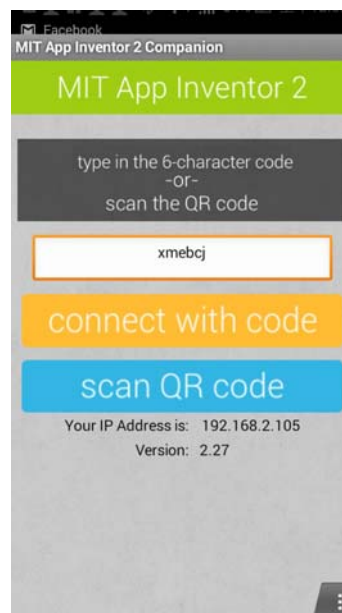


圖 91 取得 QR 程式碼

如下圖所示，我們點選如下圖所示的紅框之中的『connect with code』，就可以進入測試程式區。



圖 92 執行程式

如下圖所示，如果程式沒有問題，我們就可以成功進入測試程式的主畫面。

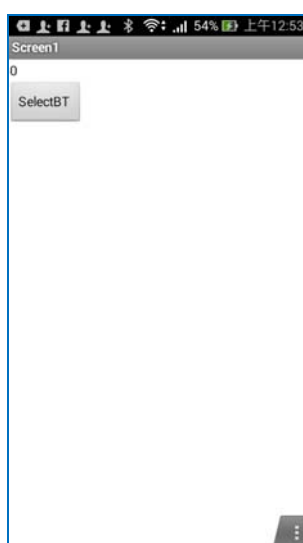


圖 93 執行程式主畫面

如下圖所示，我們先選擇『SelectBT』來選擇藍芽裝置。

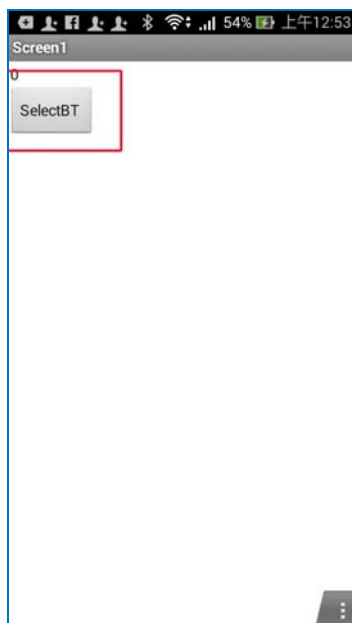


圖 94 選藍芽裝置

如下圖所示，會出現手機、平板中已經配對好的藍芽裝置。

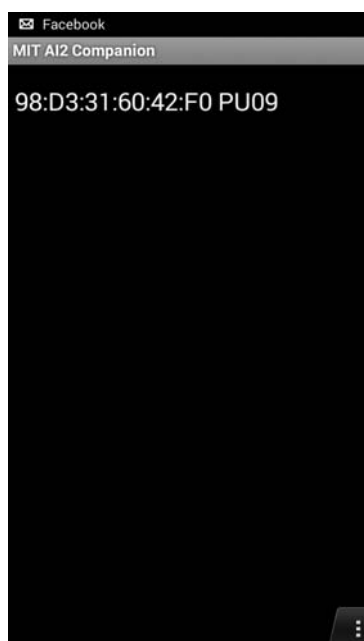


圖 95 顯示藍芽裝置

如下圖所示，我們可以選擇手機、平板中已經配對好的藍芽裝置。

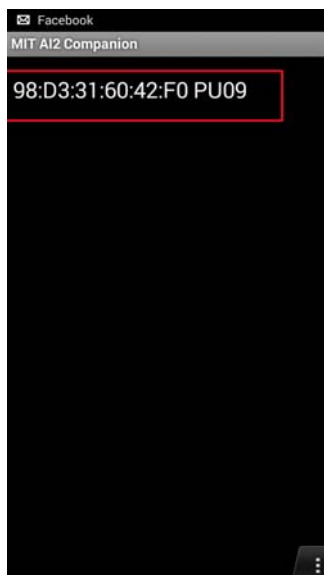


圖 96 選取藍芽裝置

如下圖所示，如果藍芽配對成功，可以正確連接您選擇的藍芽裝置，則會進入通訊模式的主畫面，可以接收配對藍芽裝置傳輸的資料，並顯示在上面。



圖 97 接收藍芽資料顯示中

章節小結

本章主要將完整的攜帶式空氣粒子感測裝置，加入手機 APP 應用程式功能，可以在沒有網路環境中，也可以更加輕鬆使用，並透過本章節的解說，相信讀者會上述模組之應用、手機 APP 應用程式開發、空氣粒子感測裝置開發應用，有更深入的了解與體認。

參考文獻

曹永忠, 許智誠, & 蔡英德. (2015a). *Arduino 手机互动编程设计基础篇:Using Arduino to Develop the Interactive Games with Mobile Phone via the Bluetooth* (初版 ed.). 台湾、彰化: 渥瑪數位有限公司.

曹永忠, 許智誠, & 蔡英德. (2015b). *Arduino 手機互動程式設計基礎篇:Using Arduino to Develop the Interactive Games with Mobile Phone via the Bluetooth* (初版 ed.). 台湾、彰化: 渥瑪數位有限公司.

曹永忠, 許智誠, & 蔡英德. (2015c). *Arduino 乐高自走车:Using Arduino to Develop an Autonomous Car with LEGO-Blocks Assembled* (初版 ed.). 台湾、彰化: 渥瑪數位有限公司.

曹永忠, 許智誠, & 蔡英德. (2015d). *Arduino 樂高自走車:Using Arduino to Develop an Autonomous Car with LEGO-Blocks Assembled* (初版 ed.). 台湾、彰化: 渥瑪數位有限公司.

趙英傑. (2013). *超圖解Arduino 互動設計入門*. 台灣: 旗標.

趙英傑. (2014). *超圖解Arduino 互動設計入門(第二版)*. 台灣: 旗標.