
大氣壓力實驗 APP 的開發實作

李冠杰^{1,3} 洪耀正^{1,2*}

¹逢甲大學 電機學系

²逢甲大學 通識教育中心

³程堡兒童程式教育中心

壹、背景與動機

隨著智慧型手機普及、感測器元件的發展，現在的手機可以測量的物理量越來越多，包含加速度、光照度、磁場等等。很多以往需要依靠昂貴設備才能進行的科學實驗，現在已經可以透過智慧型手機完成(曾耀寰，2017；李欣、羅道正、江俊明，2017；謝怡靜、江俊明，2017)。這不但解決了部分實驗囿於儀器昂貴而難以推廣的問題，更可以設計出貼近生活的科學實驗，強化生活與科學的連結。

以智慧型手機進行科學實驗除了上述的好處之外，若能進一步教授學生撰寫實驗量測 APP，則可把程式邏輯教育與自然科學教育結合，起到相輔相成的作用。以程式學習的角度來看，這樣可以賦予程式學習的明確動機，增加學生學習動力。以物理教育的觀點來看，其能把學生從儀器使用者的角色轉為設計者的角色，得以幫助學生拓展不同的思維模式，深耕科學素養教育。

基於以上概念，我們已經開發了能夠測量加速度、磁場、光照度的 APP 教學模

組(洪耀正、陳淑慧，2019)。在本文中，我們將這個概念擴充到大氣壓力的量測與實驗設計上，亦即開發大氣壓力 APP 的實作模組。由於大氣是看不見的，因此學生在學習初期不容易直觀地理解大氣壓力的各種現象，最後往往依賴老師解釋而進行被動學習。因此開發大氣壓力 APP 並應用到相關壓力實驗，學生將可直接觀察大氣壓力變化而有助於理解現象成因。本文便以最簡單的抽氣罐實驗為例，檢驗我們 APP 的設計結果。

貳、大氣壓力實驗設計

一、實驗目的

此為大氣壓力變化的觀察實驗：將在智慧型手機放入抽氣罐中，自行撰寫 APP 將手機內建的氣壓感測器的量測值讀取並顯示出來，取代傳統在抽氣罐中放入氣球作為氣壓指示物。

二、實驗原理

在一密封且體積 V 固定的容器中，持續抽出容器中的空氣，造成氣體的莫耳數 n 下降。假設環境溫度 T 不變，根據理想氣

*為本文通訊作者

體方程式 $PV = nRT$ ，故 $P \propto n$ ，容器內的大氣壓力 P 也會跟著下降。

三、實驗流程

準備可抽氣的密封罐(大小需可放入一隻智慧型手機)、具有氣壓感測器的智慧型手機(相關型號請見附錄 1)。開啟撰寫好的氣壓量測 APP 後，將智慧型手機放入罐子，並開始抽去罐子中的空氣，如圖 1 所示。在過程中持續觀察 APP 顯示的數值變化。



載有氣壓量測 APP 的手機

圖 1、大氣壓力實驗示意圖

參、大氣壓力實驗 APP 實作

App Inventor 2(AI2) 的最大特點為利用積木式的「程式方塊」來取代傳統的程式碼，就算是非資工科系的學生，也可以在妥善的引導下完成程式撰寫。因此本文一樣利用 AI2 實作 APP。AI2 的註冊、登入、基本操作與程式基礎可參考先前的一些文章與書籍(洪耀正, 2018; 洪耀正、陳

淑慧, 2019)。

以下是大氣壓力實驗 APP 的實作引導。

一、讀取氣壓感測器的量測值並顯示在 APP

1. 為了顯示大氣壓力數值，因此先由左側「元件面板」的「使用者介面」分類拖曳一個「標籤」元件加入工作面板的手機螢幕。其中「標籤元件」的用途為在 APP 顯示文字與數字。

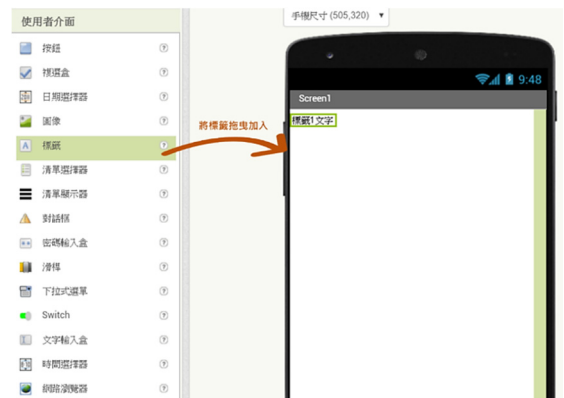


圖 2、拖曳加入一個「標籤」元件

同時可依個人喜好修改標籤的屬性，讓 APP 看起來更美觀。



圖 3、修改「標籤」的屬性

2. 在「元件面板」的「感測器」分類找到「Barometer」(氣壓感測器)並加入。由於感測器是不可視元件(意即使用者在螢幕上看不到該元件),所以加入後會顯示在手機畫面下方。



圖 4、加入氣壓感測器

其中 Barometer(氣壓感測器)的屬性維持預設即可。



圖 5、氣壓感測器的屬性不需更改

3. 點選右上角「程式設計」,就可以切換到程式撰寫頁面撰寫程式。



圖 6、切換至程式設計頁面

4. 加入「當氣壓改變」事件:在 AI2 中,所有的指令動作都需要有一個事件來觸發。在此我們希望只要氣壓值一有變化就馬上顯示最新數值,因此我們使用「當氣壓改變」事件。點選「Barometer1」分類,拖曳並加入「當氣壓改變」事件方塊。



圖 7、加入「當氣壓改變」事件方塊

5. 當氣壓改變時,將標籤 1 的文字設為最新的氣壓測量值:如圖加入「設標籤 1 的文字為」的指令,此時會顯示紅色叉叉的 Error,這是因為後方還沒有給值,給值之後 Error 就會消失。



圖 8、加入「設標籤 1 的文字為」指令

- 取出並呈現即時氣壓值：每當氣壓值改變時，「當氣壓改變」事件方塊內建的「pressure」變數就會呈現最新的氣壓值，因此我們只要取用這個方塊就可以取得即時氣壓值。



圖 9、取用「pressure」變數方塊

「pressure」變數只有氣壓數值，因此我們利用合併文字元件來加上單位。

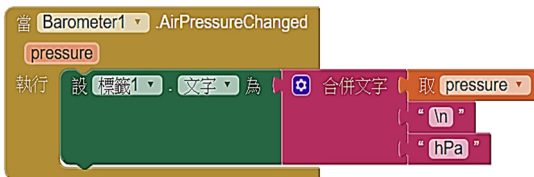


圖 10、使用「合併文字」方塊來加上單位

注意：「合併文字」在「文字」分類內，點擊方塊左上角的藍色齒輪可以增減要合併的文字個數。



圖 11、可以透過藍色齒輪選項修改要合併的文字個數

這樣就初步完成 APP 了，接下來可以使用智慧型手機來測試 APP 執行結果。目前版本沒有大氣壓力校正機制，下一小節將增添此功能。

- 利用手機在 Google Play 商店搜尋「MIT AI2 Companion」(APP 模擬套件)，並進行安裝。



圖 12、安裝 APP 模擬套件「MIT AI2 Companion」

安裝完成後開啟「MIT AI2 Companion」，點選 scan QR code

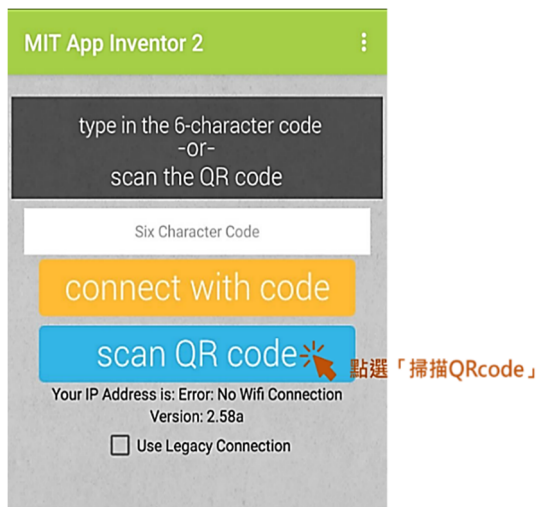


圖 13、開啟「MIT AI2 Companion」並點選「掃描 QR code」

- 在 AI2 程式頁面，點選上方選單的「連線」，並選擇 AI Companion 連線方式，便可生成該 APP 的 QR code。利用「MIT AI2 Companion」的「掃描 QR code」進行掃描，就可以看到剛剛撰寫 APP 的執行結果了。

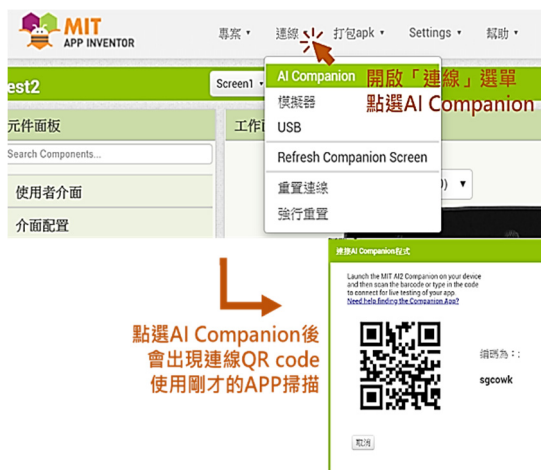


圖 14、生成 QR code，並使用 MIT AI2 Companion 掃描

二、校正大氣壓力

由於不同手機內的氣壓感測器存在測量上的誤差，因此在進行實驗前，我們必需使用標準大氣壓力計測量當下正確的氣壓值，並以此為基準進行修正。為了簡化問題，我們假設感測器所量到的數值與氣壓實際值的差值(offset)為一常數，且關係為：

$$P_{\text{實際值}} - P_{\text{感測器值}} = \text{offset} \quad (1)$$

其中 $P_{\text{實際值}}$ 為標準大氣壓力計的數值， $P_{\text{感測器值}}$ 為手機量得數值， offset 為校正值。如欲精準地求得 offset 值，可以親自到附近的氣象觀測站，該處的儀器都有定期進行校正，可取得相當準確的氣壓值，再由此計算出校正值。或者也可以使用物理實驗室內有定期校正的氣壓計。求得校正值後，我們只要在 APP 程式內將感測器值加上 offset ，即可逼近實際值。

以下我們逐步實作如何加入校正功能。

(一) 加入按鈕、文字輸入盒和標籤等元件：

由於校正需要使用者輸入數字、進行確認，以及回復到原本數值，因此我們需要回到 AI2「畫面編排」頁面，加入一個「文字輸入盒」讓使用者輸入標準大氣壓力計量到的 $P_{\text{實際值}}$ ，以及兩個「按鈕」元件(確認按鈕與復原按鈕)，同時再多加一個「標籤」元件來提醒使用者是否已校正。

考慮美觀，我們加入「水平配置」，並且將兩個「按鈕」元件放入其中。



圖 15、加入「水平配置」元件

同時將「水平配置」、「按鈕」寬度都設為填滿，這樣按鈕配置起來就相對整齊美觀。



圖 16、增加兩個「按鈕」元件到「水平配置」元件內

最後加入文字輸入盒、標籤，將元件寬度設為填滿，並修改相關屬性。



圖 17、加入「文字輸入盒」、「標籤」元件，並修改屬性

擺完後，元件清單會列出目前所有元件：標籤 1、標籤 2、按鈕 1、按鈕 2。然而系統預定的命名方式容易造成辨識不易，因此可以使用下方的「重新命名」鈕將元件重新命名，這樣在後續的程式撰寫上可以降低出錯的機會。



圖 18、點選「重新命名」將元件改名，方便程式撰寫時的辨識

(二) 切換到「程式撰寫」頁面，加入一個新的變數並命名為 offset：

在程式語言中，變數是一個非常重要的工具，程式中幾乎所有資料都是以變數的形式儲存。要使用變數存放某個值的話，必須先「宣告變數」，在 AI2 中就是「初始化全域變數」這個指令，宣告完變數後，就能透過程式取用或是修改這個值。

由於我們需要程式記住 offset，所以需要定義一個變數來存放這個值。因此我們新增一個變數並命名為 offset，同時將初始值設為 0，如圖所示。

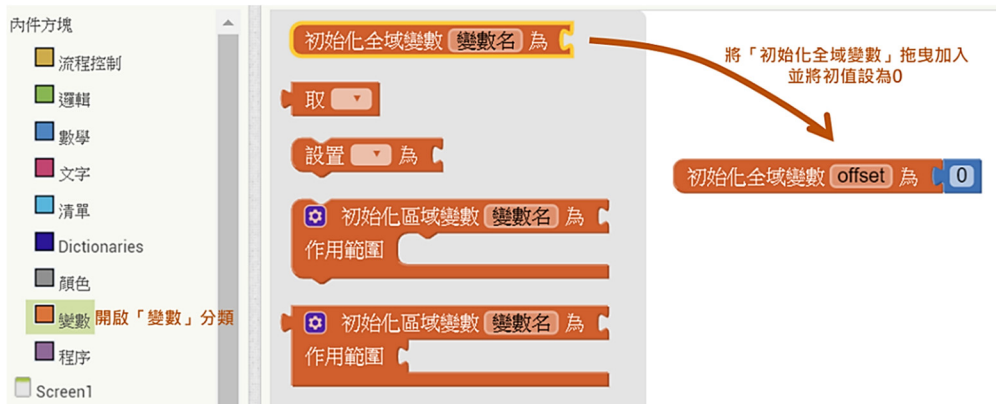


圖 19、新增一個「初始化全域變數」指令，並將變數命名為「offset」。

(三) 設置 offset 數值：

APP 使用者可在「文字輸入盒」讓使用者輸入標準大氣壓力計量到的 $P_{實際值}$ ，並利用「確定」鈕完成校正。如前所述，在 AI2 所有的指令動作都需要有一個事件來觸發，因此首先需要加入「當確定按鈕被點選」事件。

在點選確定按鈕後，我們要將校正提示標籤的文字改為已校正，並計算出要用來校正的值並存入變數 offset，其計算公式如方程式(1)所示。在此， $P_{感測器值}$ 即為「pressure」的值，而 $P_{實際值}$ 是使用者輸入到「文字輸入盒」的值，所以我們可以用「文字輸入盒的文字」這個方塊來取得這個值。

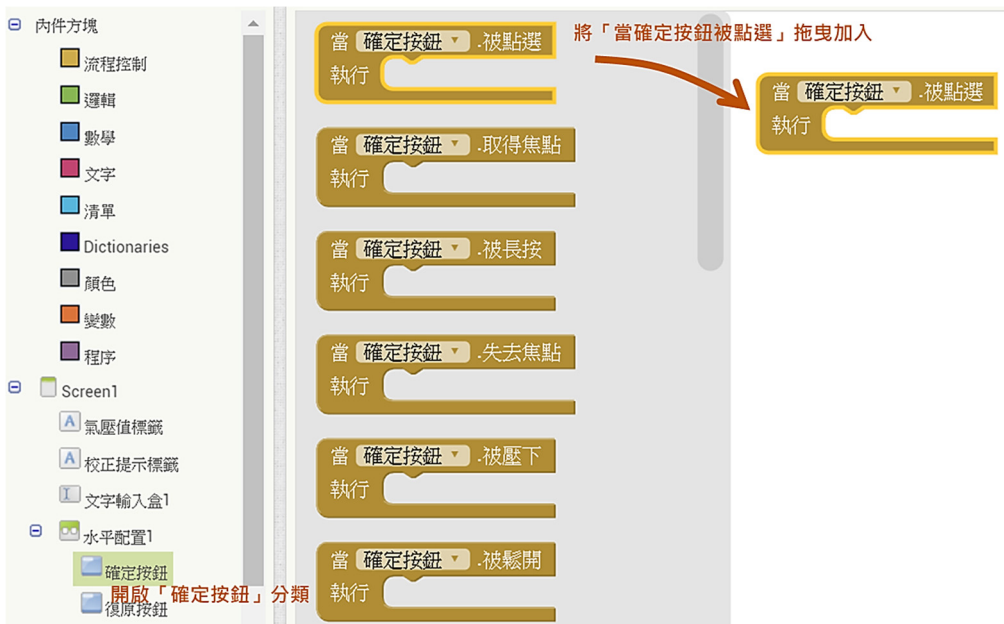


圖 20、加入「當確定按鈕被點選」事件

這裡有個程式上的小細節，就是「pressure」這個方塊只能在「當氣壓改變」事件方塊中取用，在「確定按鈕被點選」的事件方塊中不能直接拿來用。因此需要先将「pressure」這個值先用另一個變數(在此我們命名為「氣壓值」)存起來後，再取用該變數。

因此程式方塊可建立如圖 21 所示。

(四) 顯示校正後的大氣壓力值：

將即時量到的 pressure 加上 offset，即

可得到修正後的大氣壓力值。同時我們可以在「數學」分類找到「將數字設為小數形式」，藉以將數值取到小數點後兩位。結果如圖 22。

(五) 設定還原按鈕：

點選復原鈕後，要將校正提示標籤的文字改回尚未校正、將文字輸入盒的文字清空，並將 offset 變數回復為 0。因此我們在程式加入「當復原按鈕被點選」事件方塊，如圖 23。



圖 21、設置 offset 數值

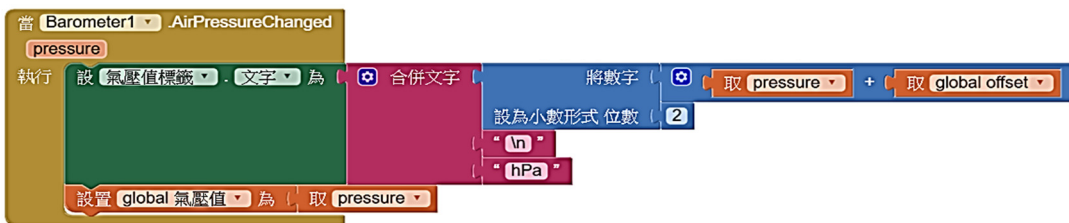


圖 22、修改結果的公式，並將數值取到小數點後兩位



圖 23、設定還原按鈕

肆、成果展示

我們以抽氣罐實驗為例，檢驗 APP 的量測結果，影片如下方連結：
<https://youtu.be/M0lHsWNhfdw>。操作流程如下圖：

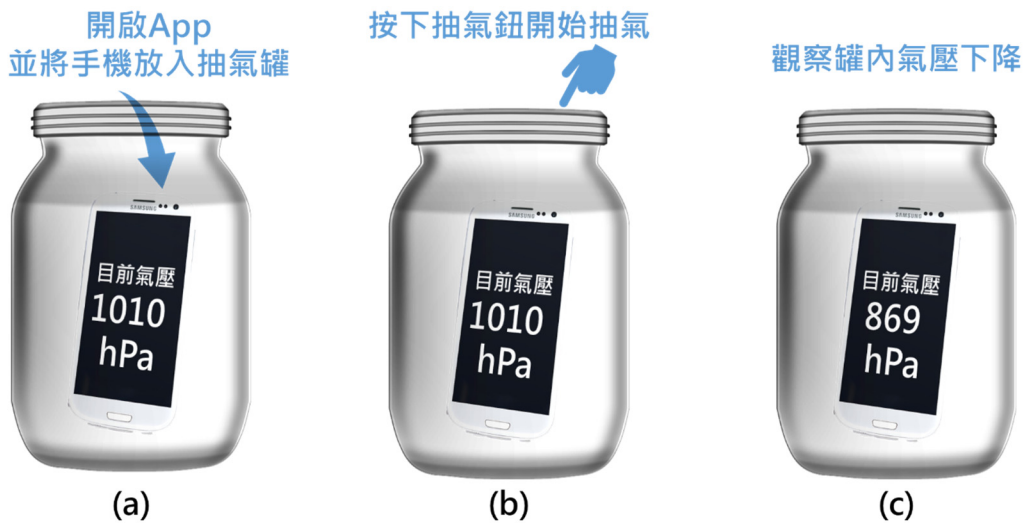


圖 24、(a)開啟 App 後將手機放入抽氣罐並蓋上蓋子 (b)按下抽氣罐的抽氣按鈕後，開始將罐中空氣抽出 (c)透過 App 觀察罐內氣壓下降情形

伍、結語

在九年一貫課綱中僅列為「重大議題」的資訊教育，在十二年國教課綱中升等，隸屬於「科技領域」中的「資訊科技」學科，以學生學習怎麼用電腦解決問題為宗旨。為了達成這樣子的理想，學生必

須具備各科整合的能力，因此在擬定此領域的能力指標時，最好也能參照其他領域的目標，讓資訊科技和科學、自然領域互相搭配(葉俊巖、羅希哲，2015)。

本文所提供的實驗模組即為自然領域與資訊科技的結合。雖然這是個簡單的物

理實驗，但卻補足了以往資訊科技課程缺乏情境動機的問題，學生在實作程式中能有明確的目標，而在操作物理實驗時，也能根據實驗狀況或需求去調整程式碼，此套模式可將學習方式拉升到不同的層級，培養學生的綜合能力。

針對有興趣的學生，本文所介紹的 APP 還可以再改進，作法如附錄 2 所示。教師可視學生的學習狀況決定是否介紹。

陸、參考資料

Cell Phones with Barometer : <https://www.phonegg.com/list/303-Cell-Phones-with-Barometer>, PHONEGG
李欣、羅道正、江俊明(2017)。普通物理實

驗的創新－利用手機 APP 量測地球磁場。**物理教育學刊**，**18(2)**，83-98。

洪耀正(2018)。**App Inventor 2 零基礎創意程式設計**。新北市：全華書局。

洪耀正、陳淑慧(2019)。物理實驗量測 App 的開發與實作。**科學教育月刊**，**419**，48-59。

葉俊巖、羅希哲(2015)，以 Maker 的角度來看臺灣小學的資訊教育，**臺灣教育評論月刊**，**4 (12)**，頁 110-114

曾耀寰(2017)。利用智慧手機測量單擺周期。**物理教育學刊**，**18(1)**，65-76。

謝怡靜、江俊明(2017)。普通物理實驗創新－利用手機 APP 驗證馬呂士定律。**物理教育學刊**，**18(2)**，99-106。

顧炳宏(2008)。大氣壓力之引導探究式教學活動設計。**物理教育學刊**，**9(1)**，123-136。

附錄 1、具備氣壓感測器的智慧型手機

表一、具備氣壓感測器的手機一覽表

具備氣壓感測器(barometer)的手機列表			
Apple iPhone SE (2020) Global A2296	Xiaomi Poco F2 Pro	Apple iPhone SE (2020) NA A2275	Apple iPhone SE (2020) CN A2298
Samsung Galaxy S20 5G US SM-G981U1 128GB 12GB RAM	Samsung Galaxy S20 5G Sprint US SM-G981U 128GB 12GB RAM	Samsung Galaxy S20 5G T-Mobile US SM- G981U 128GB 12GB RAM	Samsung Galaxy S20 5G AT&T US SM-G981U 128GB 12GB RAM
Samsung Galaxy S20 5G Canada SM-G981W 128GB 12GB RAM	Huawei Mate Xs	Samsung Galaxy S20+ 5G Global 128GB	Samsung Galaxy S20+ 5G Global 256GB
Samsung Galaxy S20+ 5G Global 512GB	Samsung Galaxy S20+ 5G Global 512GB	Samsung Galaxy S20+ 5G Global 512GB Dual SIM	Samsung Galaxy S20+ 5G Global 256GB Dual SIM
Samsung Galaxy S20+ 5G Global 128GB Dual SIM	Samsung Galaxy S20+ 5G Global 512GB eSIM	Samsung Galaxy S20+ 5G Global 256GB eSIM	Samsung Galaxy S20+ 5G Global 128GB eSIM
Samsung Galaxy S20+ US	Samsung Galaxy S20+ US Dual SIM	Samsung Galaxy S20 5G USCC US SM-G981U 128GB 12GB RAM	Samsung Galaxy S20 5G VZW US SM-G981U 128GB 8GB RAM
Samsung Galaxy S20+ 5G USA	Samsung Galaxy S20 Ultra 5G USA 512GB Dual SIM	Samsung Galaxy S20 Ultra 5G USA 512GB	Samsung Galaxy S20 Ultra 5G USA 256GB Dual SIM
Samsung Galaxy Z Flip	Samsung Galaxy S20 5G Global SM-G981B/DS Dual SIM	Samsung Galaxy S20+ Global Dual SIM	Samsung Galaxy S20+ Global
Samsung Galaxy S20 Ultra 5G USA 128GB	Samsung Galaxy S20 Ultra 5G USA 128GB Dual SIM	Samsung Galaxy S20 Ultra 5G USA 256GB	Ulefone Armor 2
Samsung Galaxy S10+ Canada	Samsung Galaxy S10 SM-G9730 China	Huawei Mate X	Google Pixel 4 US
Google Pixel 4 XL USA	Samsung Galaxy Note10+ SM-N975F/DS Global	Samsung Galaxy Note10+ SM-N9750 CN/TW	Samsung Galaxy Note10+ SM-N9750/DS
Samsung Galaxy Note10+ SM-N975W	Samsung Galaxy Note10+ SM-N975U	Samsung Galaxy Note10	Huawei Mate 30 RS Porsche
LG V50S ThinQ 5G	Huawei Mate 30	Huawei Mate 30 5G	Huawei Mate 30 pro
Huawei Mate 30 pro 5g	Google Pixel 4 Global		

參考資料：<https://www.phonegg.com/list/303-Cell-Phones-with-Barometer>

附錄 2、使用微型資料庫(TinyDB)改良程式

一、用途

程式中的變數只要在 APP 關閉後就會消失，因此每次開啟 APP 後都必須再輸入標準氣壓重新校正一次，但如果今天手邊沒有標準氣壓計就無法進行校正。因此，若我們可以將校正後計算出的 offset 永久保存於手機的話，之後進行實驗時就不需要再校正一次，在使用上會方便許多。

微型資料庫(TinyDB)是 Android 平台上一個輕量級的儲存類別，可以將資料儲存到手機之中，不同的資料給定不同的標籤(tag)名稱來儲存，因此 APP 應用程式可利用標籤來呼叫出不同的資料。

二、程式撰寫

1. 加入微型資料庫元件



圖 25、加入「微型資料庫」元件

2. 切換到「程式設計」頁面，並加入「當 Screen1 初始化」事件方塊

這個事件方塊會在剛開啟 APP 時執行一次，所以我們通常用來設定這個程式的初始狀態，在這個例子中，一開啟 APP 時要把先前存在微型資料庫中的 offset 值給讀取出來並交由變數儲存。



圖 26、將「當 Screen1 初始化」事件方塊加入

3. 呼叫 tinyDB 並取得值

標籤是微型資料庫中用來辨識不同筆資料的依據，取用時指定的標籤名稱要與儲存時使用的標籤名稱相同，標籤名稱可以自訂，這裡設定名稱是「offsetTAG」。

若微型資料庫中找不到該標籤名稱，則表示還沒有儲存過，也就是該手機未曾進行過校正，此時我們就應該將 offset 先設為 0，故無標籤時之回傳值要填 0。

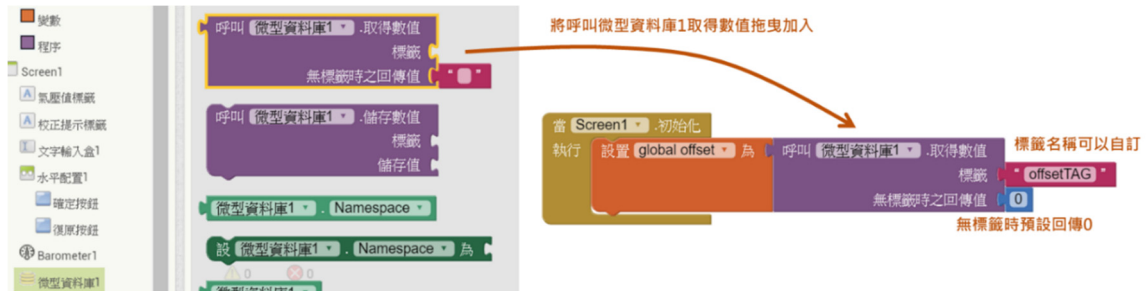


圖 27、將「呼叫微型資料庫 1 取得數值」加入，並指定標籤名稱和回傳值

4. 判斷是否已校正過，若已校正過，則校正提示標籤要顯示「已校正」

判斷是否已校正的方法只需要判斷 offset 是否不為 0 即可，因為 offset 為小數，即使手機的感測器結果非常準，校正後 offset 的值也幾乎不可能完全為 0。



圖 28、加入「如果...則...」方塊

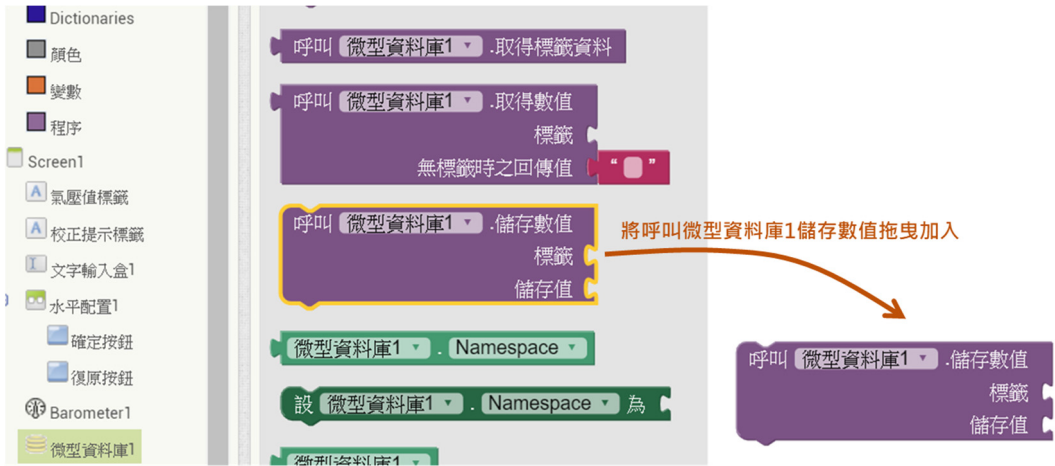
若 offset 不等於 0，則校正提示標籤會顯示「已校正」；若 offset 等於 0，則如果...則...程式方塊便不會執行，校正提示標籤的文字會維持預設的「尚未校正」。



圖 29、加入條件「offset 不等於 0」，若條件成立的話則更改校正提示標籤的文字

5. 在確定按鈕被點選後，將計算完的 offset 存入微型資料庫

千萬別忘記了要將資料存入後，日後才能讀出來使用，因此在每次確認按鈕被點選後，我們都必須將計算過後的 offset 存入微型資料庫。



開啟微型資料庫1分類

圖 30、將「呼叫微型資料庫 1 儲存數值」加入



圖 31、將 offset 的值以「offsetTAG」這個標籤存入微型資料庫

如此便可以將 offset 的數值儲存於手機中，關閉 APP 資料依舊能保存。