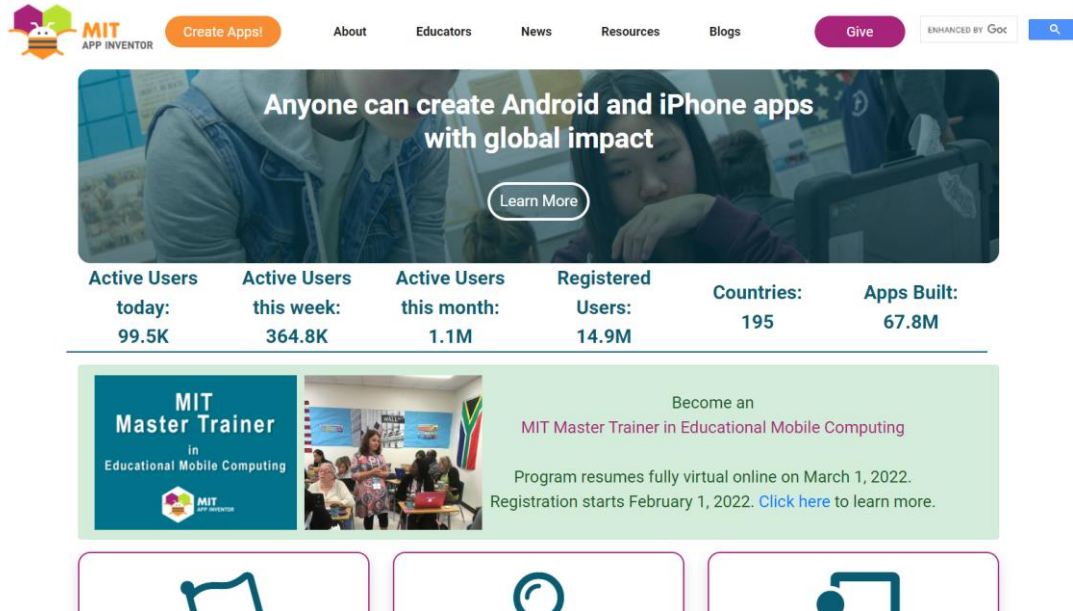


App Inventor 2 - CPR 教材

App Inventor 為一個透過視覺化圖形介面來設計行動應用服務的工具。

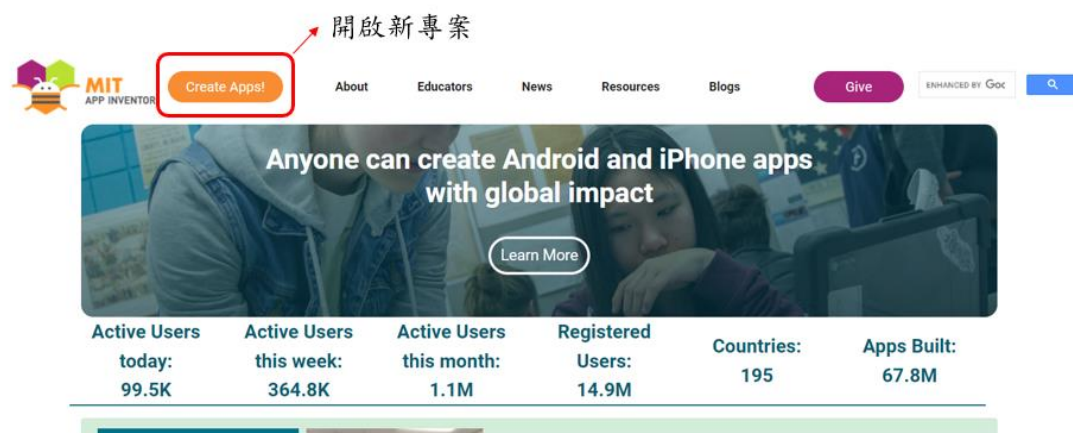
在 App Inventor 使用者可以透過瀏覽器編輯器中拖拉積木和 GUI 圖形化介面的方式來開發 Android App 而不用撰寫 Java 或是 Kotlin 程式碼（跟 Scratch Blocks Editor 使用上頗為類似，只是 App Inventor 比較聚焦在 Android 行動應用程式的開發上）。對於一些比較不熟程式語言的使用者來說，不失為訓練邏輯思考和熟悉行動應用開發的一方便工具。



The screenshot shows the MIT App Inventor homepage. At the top is the MIT App Inventor logo and a navigation bar with links: About, Educators, News, Resources, Blogs, and a Give button. Below the navigation bar is a large banner with the text "Anyone can create Android and iPhone apps with global impact" and a "Learn More" button. Under the banner is a statistics table. Below the table is a section for the MIT Master Trainer program, which includes a photo of a group of people and text about the program resuming fully virtual online on March 1, 2022. At the bottom are three icons representing different app categories.

| Active Users today: | Active Users this week: | Active Users this month: | Registered Users: | Countries: | Apps Built: |
|---------------------|-------------------------|--------------------------|-------------------|------------|-------------|
| 99.5K | 364.8K | 1.1M | 14.9M | 195 | 67.8M |

首先，我們進入 App Inventor 網頁後點選 Create App 按鈕，就會導到 Google 登入畫面，使用 Google 帳號登入後就會到專案編輯頁面（第一次登入會需要點選 Terms of Service），這是主要編輯應用程式的地方。



This screenshot is similar to the previous one, but with a red box highlighting the "Create Apps!" button in the top navigation bar. A red arrow points from the text "開啟新專案" (Open new project) to the button. The rest of the page content is identical to the previous screenshot.

再來，輸入自己的 Google 信箱帳號，此 App Inventor 2 是跟 Google 帳號連結的，所以之後換其他電腦開發軍可以開啟專案。

繁體中文 ▼ 說明 隱私權 條款

To use App Inventor for Android, you must accept the following terms of service.

User Postings

By submitting or distributing your User Postings, you affirm, represent, and warrant that you are the creator and owner of or have the necessary licenses, rights, consents, and permissions to reproduce and publish the User Postings, and to authorize MIT and the Site's users to reproduce, modify, publish, and otherwise distribute your User Postings as necessary to exercise the license granted by you below. You, and not MIT, are solely responsible for your User Postings and the consequences of posting or publishing them.

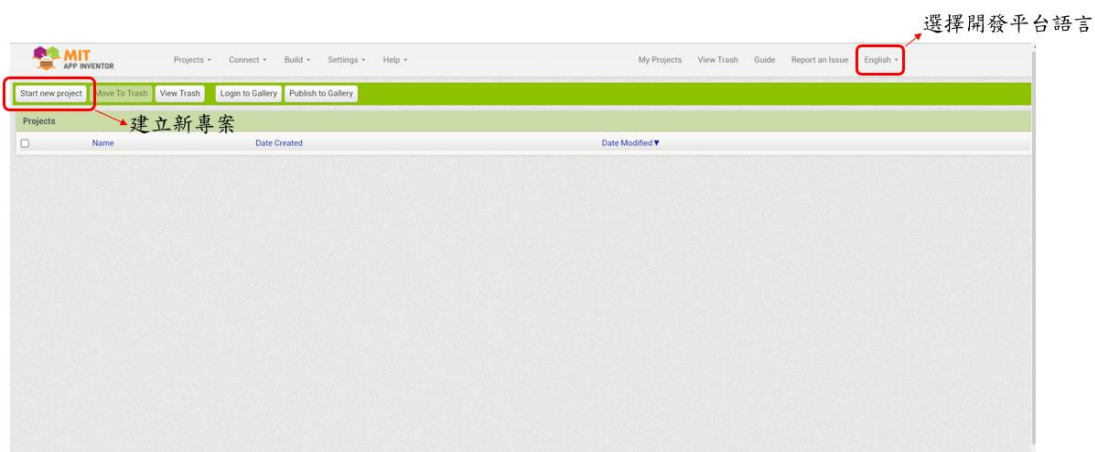
MIT has no proprietary rights in your User Postings. You have the right to download or delete them from the server at any time.

You are required to follow these community guidelines.

partake in any activity listed.

接受此項服務

完成以上設定，可直接開啟 App Inventor 2 開發平台頁面，可先選擇右上角的語言選擇自己熟悉的語言(此教學使用中文)，完成後可點選左邊的新增專案按鈕。



此時會跳出輸入專案名稱視窗，這邊我們輸入 CPR app



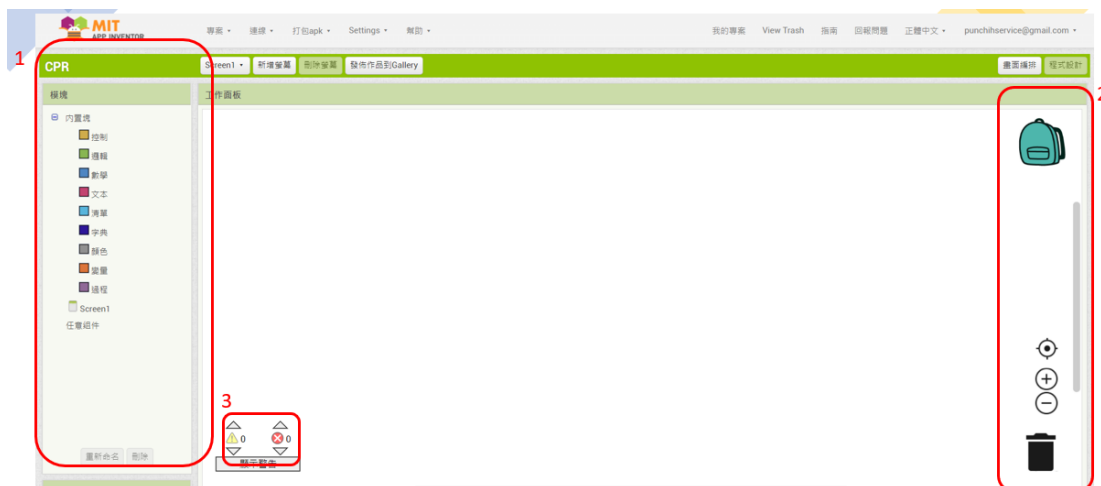
完成專案名稱輸入後，跳轉至編輯畫面頁面。



編輯畫面頁面基礎介紹，此頁面為設定版面的地方，可根據自行的需求在左側選取要的內容，以及設定元件的名稱及 ID 代碼(方便後續開發)，以及可先在此頁面設定元件大小、顏色等版面資訊，完成後可跳轉至程式頁面進行開發。

1. 此為放置元件的地方，選取要使用的元件點選並拖移至手機畫面中。
2. 顯示目前有多少元件，以及若需要調整元件 ID 代碼，可點選下方重新命名來位元件建立元件名稱 ID 代碼。
3. 若需要先為元件設計外觀，包含大小、顏色等資訊，可在此地方調整。
4. 若元件在版面上開發完畢，可點選這邊的程式程式設計，跳轉至程式開發頁面。

切換至程式開發頁面介紹



1. 內置塊，程式基礎邏輯方塊，可把需要的邏輯點選後拖移至白色地方進行編輯。
2. 可把方塊存取起來方便下次使用以及點選拖移至垃圾桶刪除。
3. 開發程式邏輯，若有警告(代表有部分功能未完整，但可以使用)會顯示在上面以及錯誤(代表有功能失敗，無法使用，須修正完畢才能測試 APP)。

我們先設計一個按鈕及一個標籤，點選按鈕後，標籤名字切換成指定文字

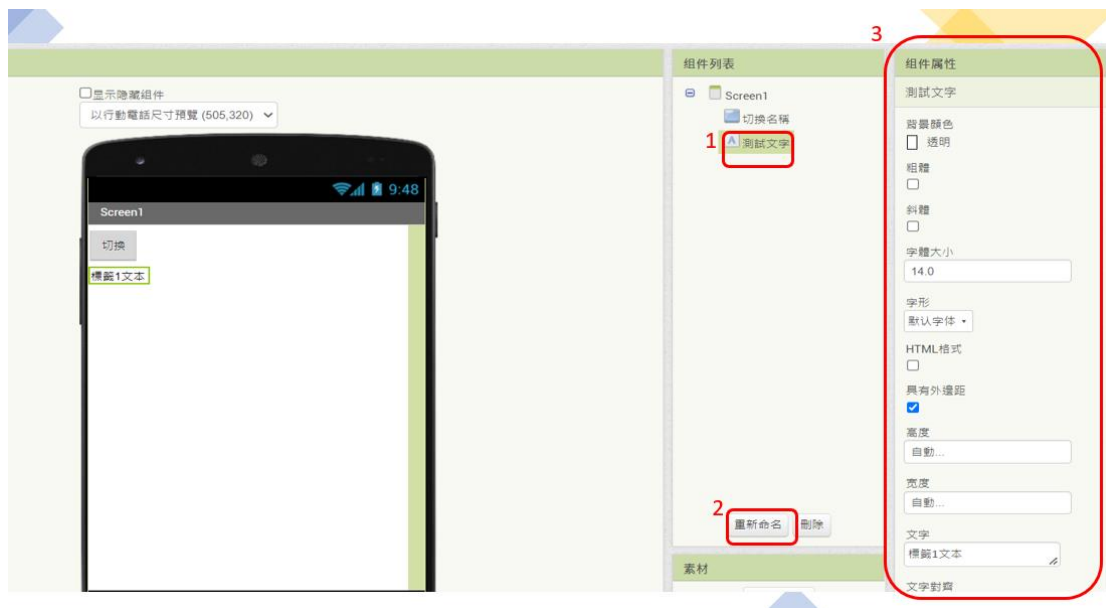


先進行按鈕設定操作

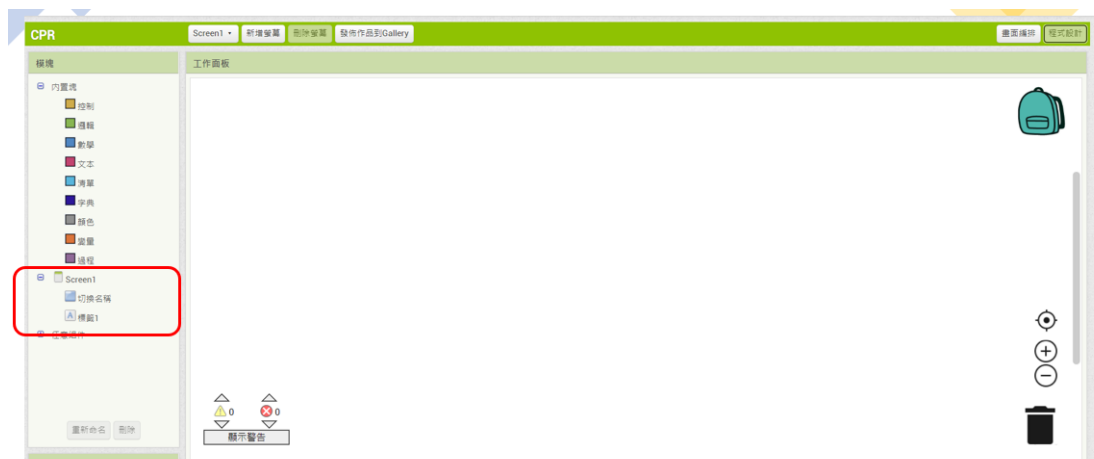


1. 選擇元件 – 按鈕
2. 點選重新命名 (為後面方便開發及教學)
3. 輸入新的名稱，我們這輸入 – 切换名稱
4. 可以依照自己的喜好進行元件版面設計

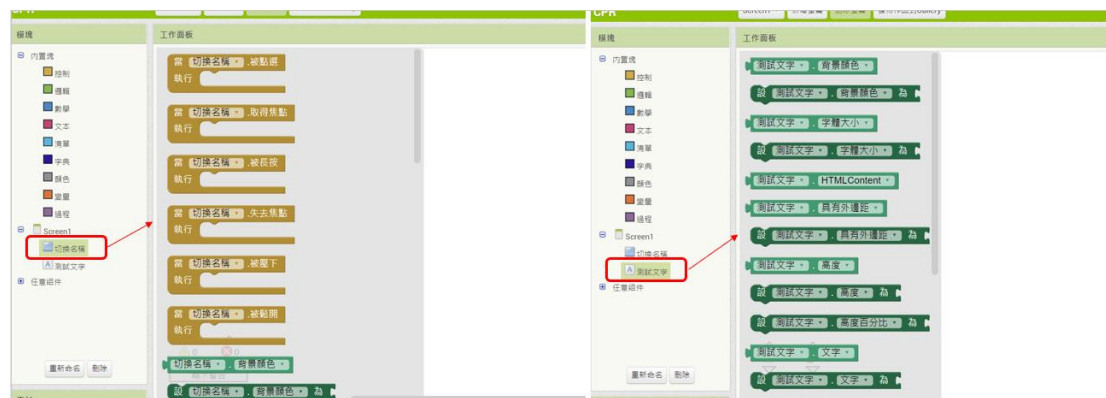
進行標籤設定操作



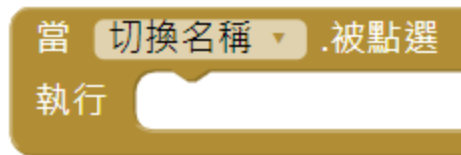
1. 選擇元件 – 標籤
 2. 重新命名，並設定新的名稱 – 測試文字
 3. 把文字的地方從 標籤1 文本 改成空的
- 完成版面設定之後，切換到程式設計頁面



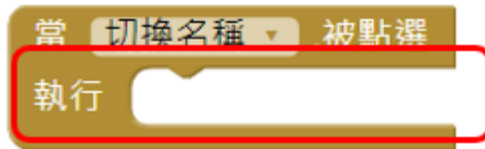
此時我們就可以看到在左側邏輯功能的 Screen1 多了兩個內容，直接各自點開來看看裡面有很多邏輯方塊可以選擇。



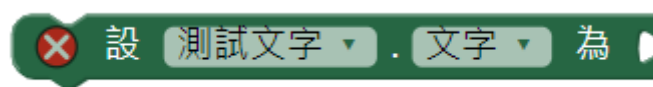
此時我們先選擇切換名稱 (按鈕)，選擇方塊第一個[當切換名稱.被點選]這項目並拖移置白色區域。



當切換名稱 (按鈕)被點選的時候，程式就會執行這個地方，所以我們要把這方塊的白色地方填滿。



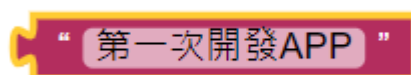
再來，我們來拉取測試文字 (標籤)，選擇方塊[設 測試文字 文字為]的深綠色方塊，其實因為每個深綠色的方塊下拉選單都能夠設定不一樣的内容，所以可隨機拉取一個深綠色的方塊，拉到白色介面後下拉選單選擇[文字]



我們看到這深綠的方塊左側有一個紅叉的錯誤圖案，代表此項目有問題，所以代表在旁邊我們必須給他一個文字



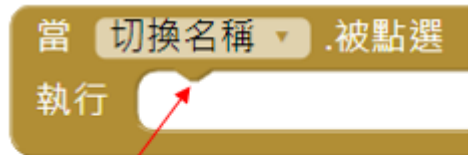
選擇內置塊的文本，在選擇第一個内容並把粉紅方塊拖移至白色區域，完成後，在粉紅方塊的中間輸入文字 - 第一次開發 APP。



完成後我們把粉紅色方塊移動深綠色的文字方塊，並把拼圖拼起來

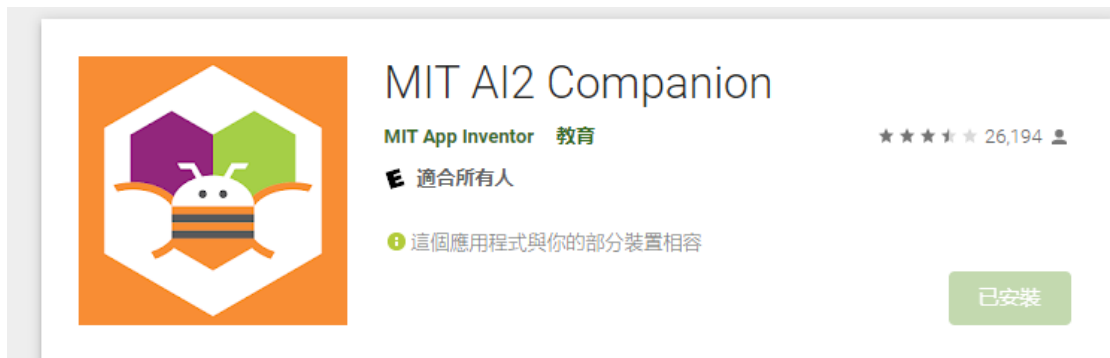


我們就會發現，在深綠的文字方塊，原本旁邊有紅叉，若粉紅色方塊接起來後，紅叉就會不見變成一個正常的流程。完成後我們把整塊方塊也連接到咖啡色的按鈕方塊中。



完成了以上的基礎教學，我們需要執行在安卓手機上

1. 請先去 Play 商店輸入 MIT AI2 Companion 並下載



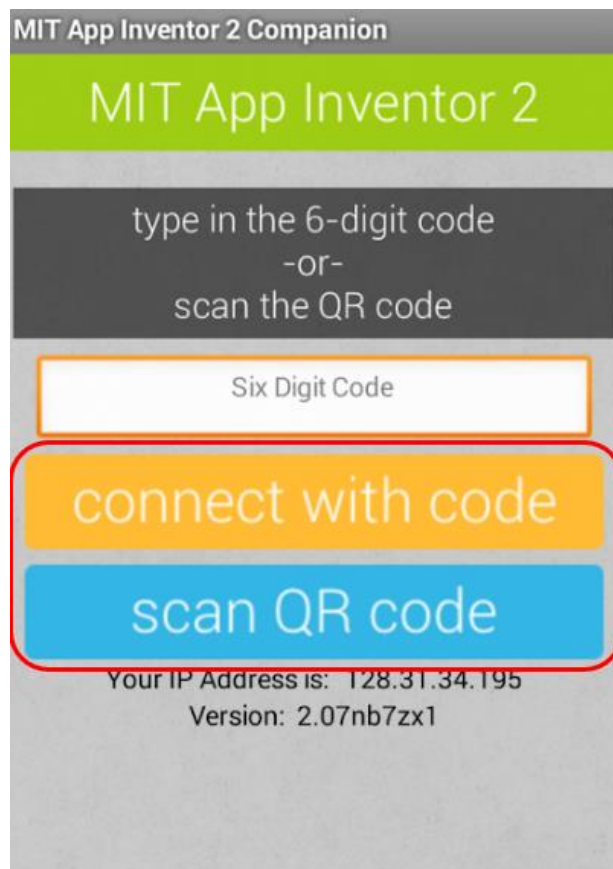
2. 在 app inventor2 開發網頁上，點選連線-AI Companion 程式



3. 點選完成後畫面會彈出一個小視窗並呈現 QR code 狀態



4. 開啟手機 App MIT AI2 Companion APP，可選擇 connect with code 或者是選擇 scan QR code。



5. 輸入完畢或者是掃描完畢後，畫面會開始運轉，完成後手機就會出現 app 畫面。
6. 點選 APP 上的切換按鈕，會看到原先的文字從 標籤 1 文本 切換成 第一次開發 APP 文字。

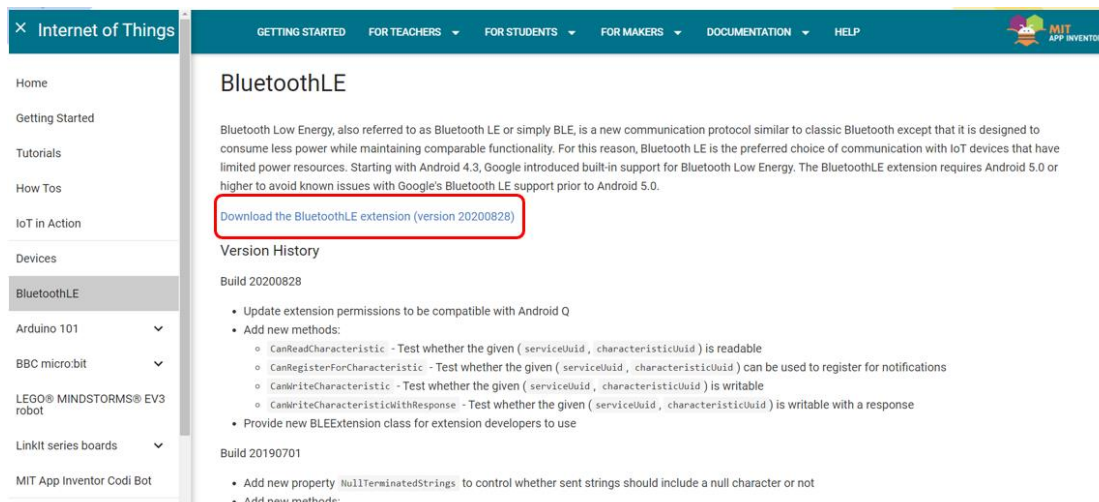


完成了基礎教學，以下開始進行透過藍牙連線 Robboni 來執行 CPR 訓練的 APP 操作流程。

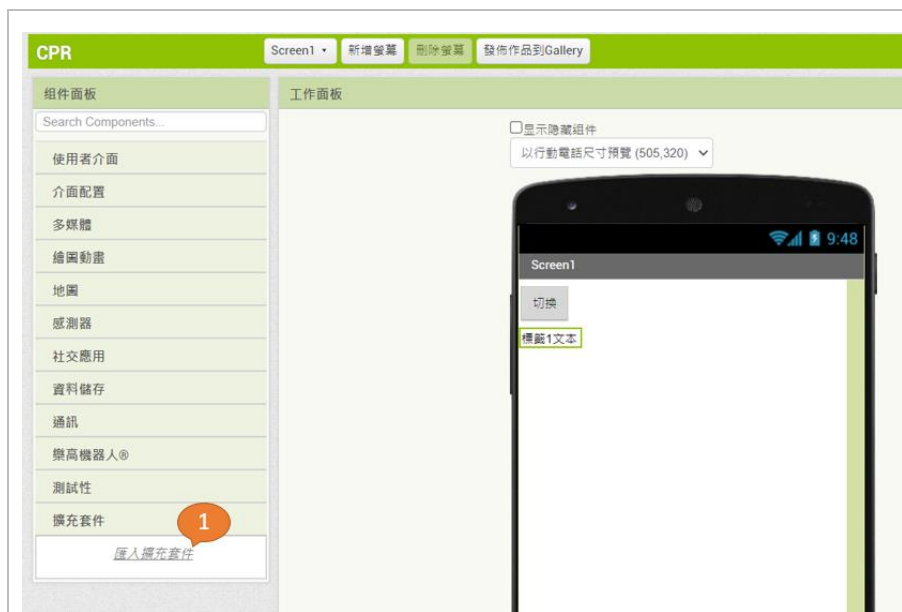
1. 藍牙套件下載並安裝

因 app inventor 原始平台並未支援藍牙 BLE 技術，所以我們必須先下載 BLE 套件來安裝至平台請開啟此頁面

<https://iot.appinventor.mit.edu/#/bluetoothle/bluetoothleintro>，並進行 BLE 套件下載



完成下載後，開啟 app inventor 畫面編排頁面



1. 點選左側擴充套件並選擇匯入擴充套件



1. 選擇檔案，選擇剛剛下載的 BLE 套件。
2. 完成後，點選 Import 進行匯入動作。

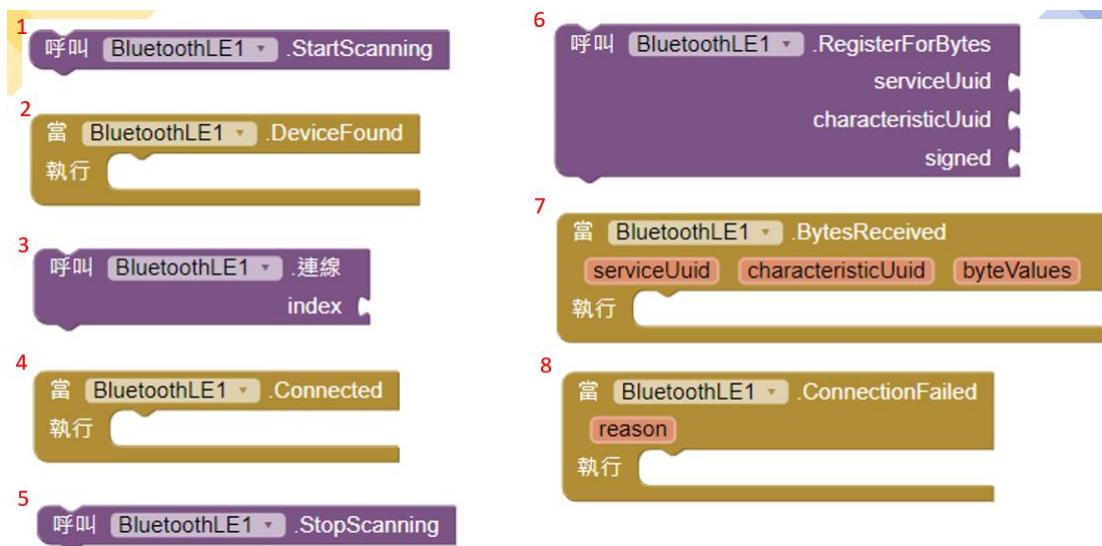
完成後在畫面上看到 BluetoothLE 的元件，完成後把該元件拉到畫面中，我們會發現，在右邊的元件列表中出現了 BluetoothLE 的元件，但手機模擬畫面並未出現元件，這是因為此套件屬於功能面導向的套件，所以並不會出現任何畫面內容在畫面中。



切換是程式設計，我們會看到 BluetoothLE 能選擇的方塊功能很多



介紹會使用到的方塊模組以及按照實際連線流程

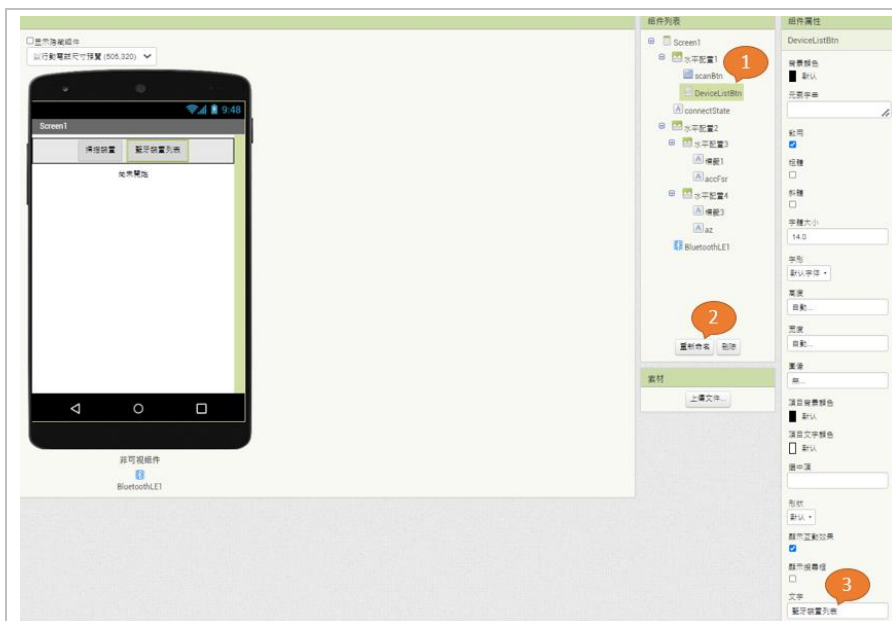


1. 開啟藍牙掃描模式
2. 抓取到目前環境中可連線的藍牙裝置列表
3. 連線所目標的藍牙裝置
4. 當藍牙連線成功會跳轉到此模組執行
5. 關閉藍牙廣播，減少資源浪費
6. 與藍牙裝置溝通，需要註冊特定的資料欄位，告知藍牙裝置需要傳送哪些資訊給 APP
7. 當註冊完畢後，會開始收取到資料並呈現至執行欄位
8. 如果連線異常失敗，會跳轉至此模組執行

2. 版面配置

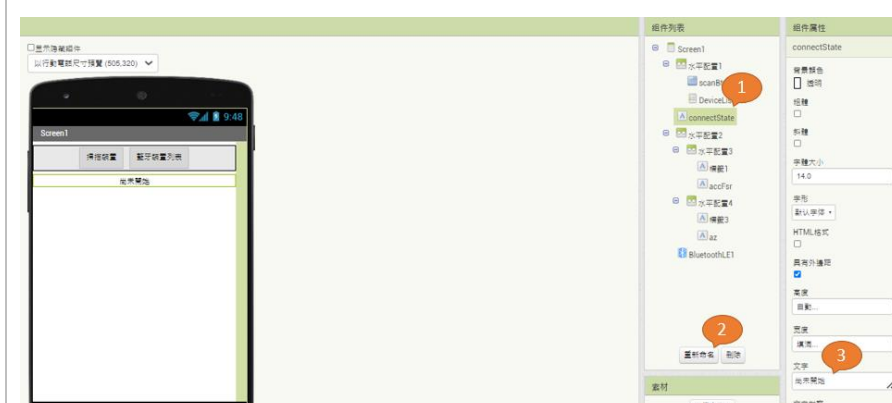
把基礎教學的元素刪除，並點選左側介面配置中的水平配置並移至畫面中。

| | |
|--|--|
| | <ol style="list-style-type: none">1. 點選介面配置 – 水平配置，拖拉置畫面中 |
| | <ol style="list-style-type: none">1. 拉一個按鈕至水平配置裡。2. 拉一個清單選擇器至水平配置裡。3. 拉一個標籤至水平配置下方。 |
| | <p>設定第一個按鈕，此按鈕要做的事情是點選後會啟動藍牙掃描功能。</p> <ol style="list-style-type: none">1. 在組件列表中，點選按鈕。2. 選擇重新命名，修改成 - scanBtn。3. 修改文字預設值，修改成 - 掃描裝置 |



設定清單選擇器，此按鈕要做的事情是點選後會抓取到手機附近的藍芽設備。

1. 在組件列表中，點選清單選擇器按鈕。
2. 選擇重新命名，修改成 - DeviceListBtn。
3. 修改文字預設值，修改成 - 藍牙裝置列表



設定標籤，此標籤要顯示目前連線狀態。

1. 在組件列表中，點選標籤。
2. 選擇重新命名，修改成 - connectState。
3. 修改文字預設值，修改成 - 尚未開始



設定水平配置版面調整。

1. 在組件列表中，點選水平配置。
2. 在組件屬性點選水平對齊，選擇 - 居中3。
3. 在寬度選擇填滿。

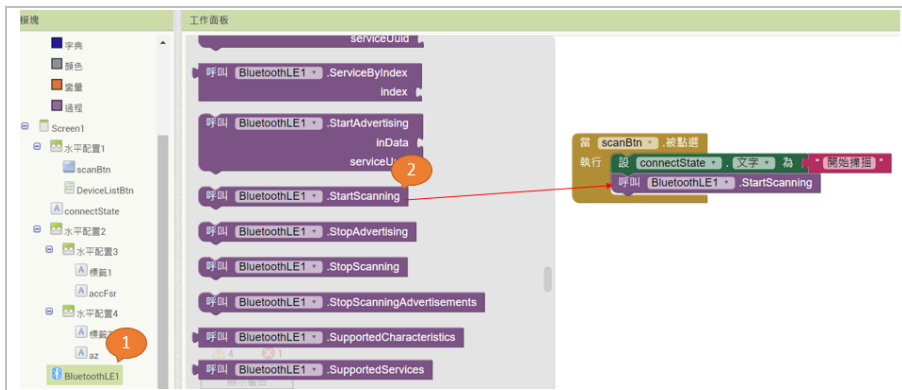
清單選擇器介紹，此元件點選後，會根據帶入的資料表，呈現一個滿版的列表畫面，可呈現資料也可點選需要的內容，用在這可把抓取到的藍牙裝置呈現至此列表中。

調整了元件的 ID 名稱以及文字呈現內容



切換至程式設計頁面，並把方塊拉出，此項目是點選掃描裝置按鈕後，會啟動藍牙掃描功能，同時把 connectState 標籤拉出來並串接在此，當點選掃描按鈕時，同時把 connectState 標籤文字改成開始連線

| | |
|--|--|
| | <p>設定按鈕觸發功能。</p> <ol style="list-style-type: none">點選 scanBtn。選擇被點選的功能，並拖移至版面中。 |
| | <p>設定標籤顯示</p> <ol style="list-style-type: none">點選 connectState。選擇顯示文字方塊，並拖移至被點選的執行中。 |
| | <p>設定本文內容</p> <ol style="list-style-type: none">點選文本方塊。選擇第一個文字方塊。並修改內部文字 - 開始掃描。 |

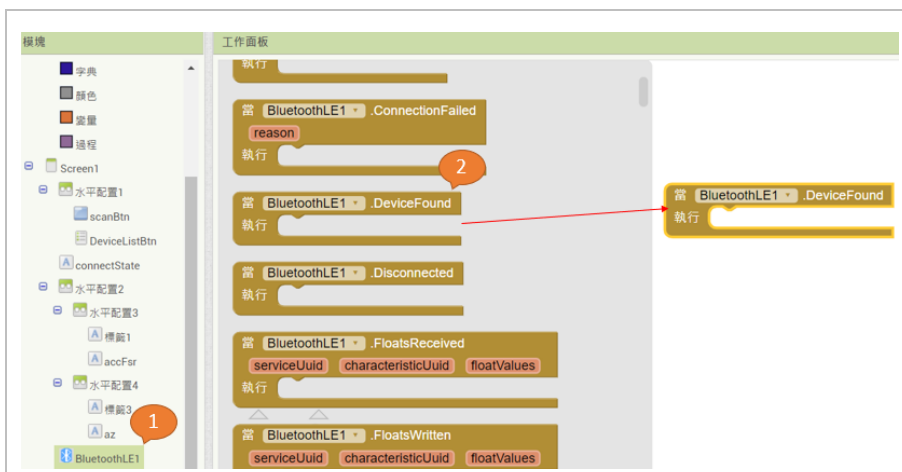


點選按鈕，也一起啟動藍牙掃描功能。

1. 點選 BluetoothLE1。
2. 選擇 StartScanning 按鈕，並拖移至被點選執行內。



完成後，接續的是我們把掃描抓取到的設備顯示在 DeblceListBtn 裡面。



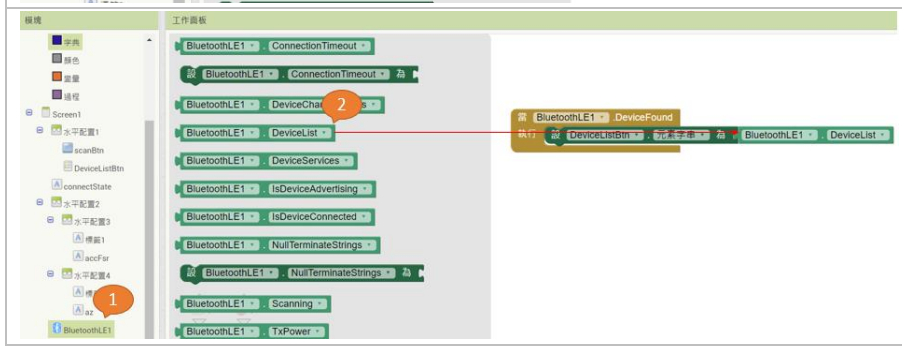
BluetoothLE1，設定一個當抓取到設備後的方塊。

1. 點選 BluetoothLE1。
2. 選擇 DeviceFound 方塊，並拖移至版面中。



設定清單列表

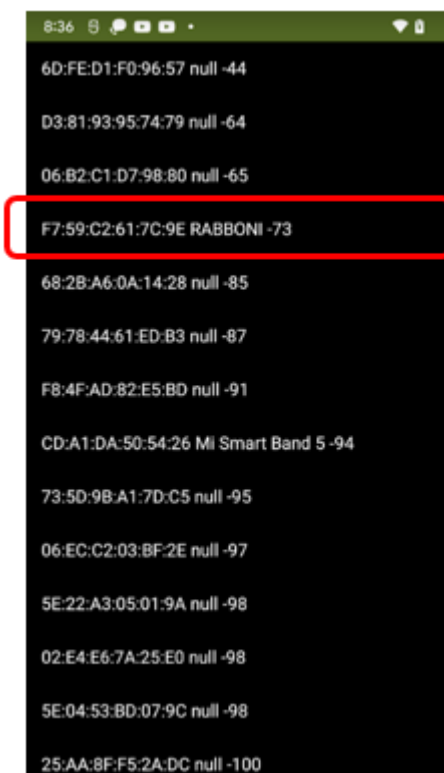
1. 點選 DeviceListBtn。
2. 選擇元素字串並拖移至 DeviceFound 執行中。



把藍牙抓取到的裝置寫入至 DeviceListBtn


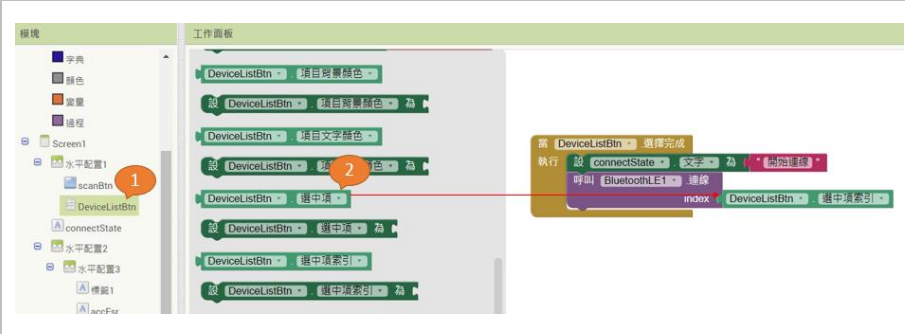
1. 點選 BluetoothLE1。
2. 選擇 DeviceList 方塊，拖移至 DeviceListBtn 後並串接上。

完成後請執行在手機中，請先點選藍牙裝置 Rabboni 的廣播按鈕，點選 APP 的掃描裝置按鈕，完成後請再點選藍牙裝置列表頁面

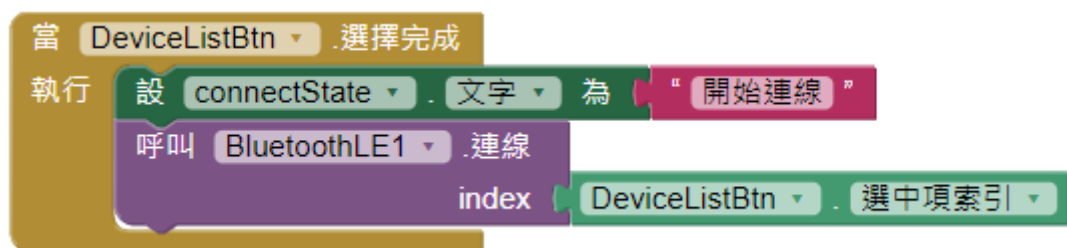


再來要完成點選要的裝置 [RABBONI] 並連線此藍牙裝置

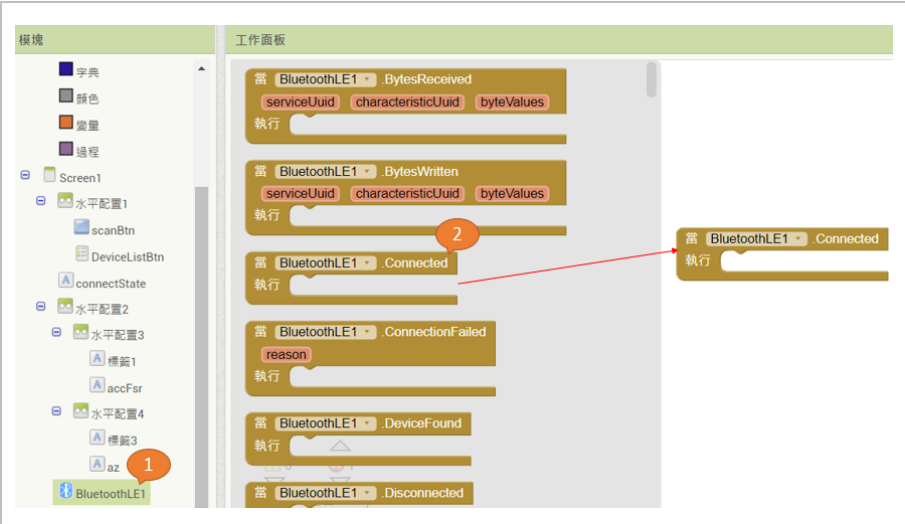
| | |
|---|---|
| <p>This screenshot shows the development environment. In the '物件' (Objects) panel on the left, the 'DeviceListBtn' button is selected, indicated by a red circle with the number 1. In the '工作面板' (Workspace) on the right, the '當 DeviceListBtn 選擇完成' (When DeviceListBtn is selected) event block is highlighted with a red circle with the number 2.</p> | <p>在清單選擇器點選被選擇的物件</p> <ol style="list-style-type: none"> 1. 點選 DeviceListBtn。 2. 選擇 選擇完成方塊，拖移至版面中。 |
| <p>This screenshot shows the '工作面板' (Workspace) where the 'connectState' variable is being configured. The '當 DeviceListBtn 選擇完成' event block is selected, and the '設定 connectState 文字' (Set connectState text) block is being added to the execution chain, indicated by a red circle with the number 2.</p> | <p>當被選擇後，connectState 修改文字</p> <ol style="list-style-type: none"> 1. 點選 connectState。 2. 選擇 文字方塊並拖移至 DeviceListBtn 選擇完成內的執行並串接。 |
| <p>This screenshot shows the '物件' (Objects) panel where the 'scanBtn' button is selected, indicated by a red circle with the number 1. In the '工作面板' (Workspace), the '當 scanBtn 被點選' (When scanBtn is clicked) event block is selected, and the '設定 connectState 文字' (Set connectState text) block is being added to the execution chain, indicated by a red circle with the number 2. The '開始掃描' (Start scanning) block is also visible, indicated by a red circle with the number 3.</p> | <p>當成功選擇後，提供文字給 connectState 顯示</p> <ol style="list-style-type: none"> 1. 點選本文 2. 選擇第一個文字方塊。 3. 並修改內部文字 - 開始連線。 |

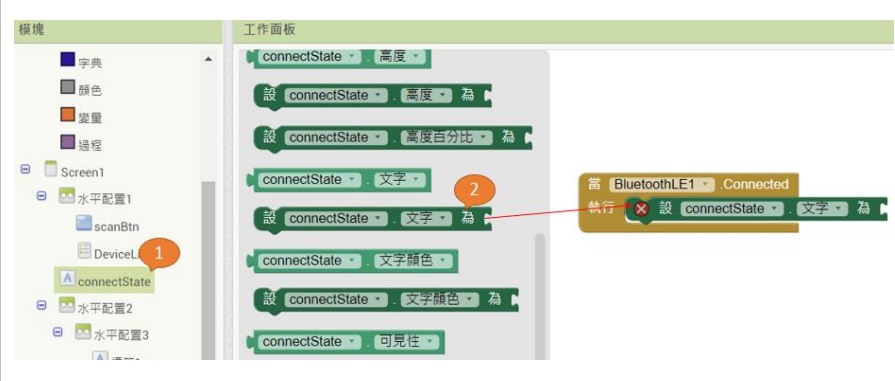

| | |
|---|--|
|  | <p>啟動與藍芽設備連線-1</p> <ol style="list-style-type: none"> 1. 點選 BluetoothLE1。 2. 選擇 連線方塊並拖移至 DeviceListBtn 執行內，並串接到 connectState 下方。 |
|  | <p>DeviceListBtn 選擇到的物件</p> <ol style="list-style-type: none"> 1. 點選 DeviceListBtn。 2. 選擇 選中項方塊，拖移至連線方塊的 index 並串接起來。 |

當 DeviceListBtn 清單選擇器點選要的項目時，會把 connectState 標籤 文字改為 開始連線，以及把 BluetoothLE1 的連線流程，根據選擇的項目進行連線



此時要監測是否有連線成功，如果連線成功後，把 connectState 標籤文字改成 連線成功。

| | |
|---|---|
|  | <p>藍牙裝置連線成功功能</p> <ol style="list-style-type: none"> 1. 點選 BluetoothLE1。 2. 選擇 Connected 方塊，拖移至版面中。 |
|---|---|

| | |
|--|--|
|  | <p>當藍牙裝置連線成功後後，connectState 修改文字</p> <ol style="list-style-type: none"> 1. 點選 connectState。 2. 選擇 文字方塊並拖移至 BluetoothLE1 Connected 內的執行並串接。 |
|  | <p>當連線成功後，提供文字給 connectState 顯示</p> <ol style="list-style-type: none"> 1. 點選本文 2. 選擇第一個文字方塊。 3. 修改內部文字 – 連線成功。 |



同時我們也要註冊三組 CPR 所需要抓取的內容以及流程

1. 讀取資料流程

服務 ID : 0000fff0-0000-1000-8000-00805f9b34fb

特徵 ID : 0000fff7-0000-1000-8000-00805f9b34fb

2. 寫入資料流程

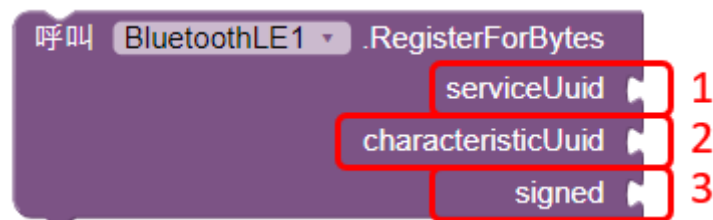
服務 ID : 0000fff0-0000-1000-8000-00805f9b34fb

特徵 ID : 0000fff6-0000-1000-8000-00805f9b34fb

3. 讀取藍牙裝置 Rabboni 感測值資訊

服務 ID : 00001600-0000-1000-8000-00805f9b34fb

特徵 ID : 00001601-0000-1000-8000-00805f9b34fb

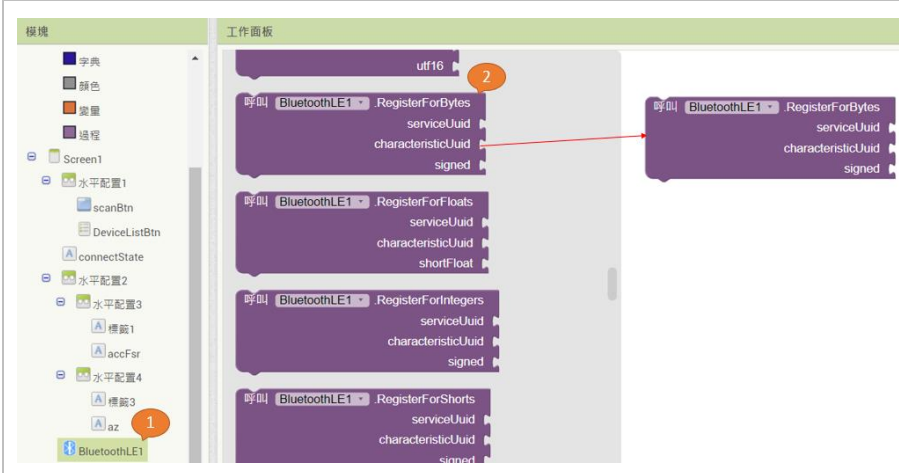


如何把資料註冊至 BluetoothLE1

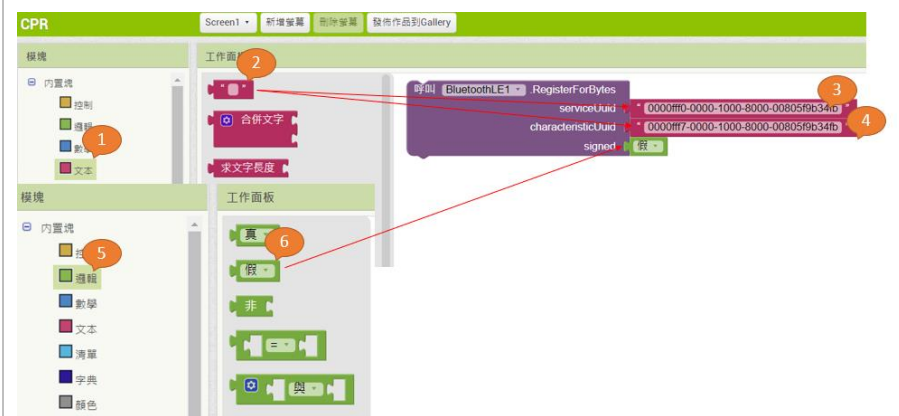
1. serviceUuid : 服務 ID

2. characteristicUuid : 特徵 ID

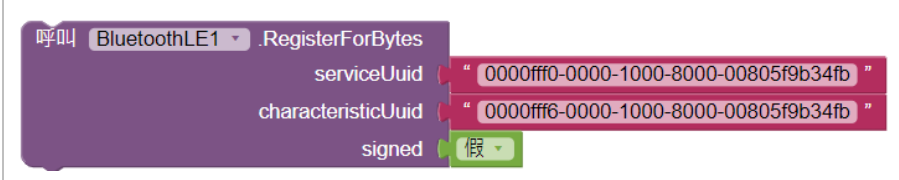
3. signed : 填入邏輯中的 假，告知讀取的資料內容，無須進行正負號判斷



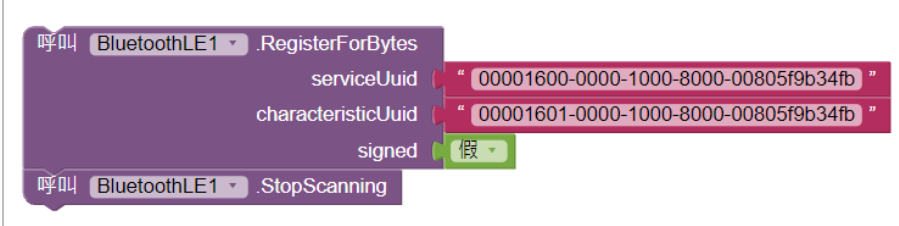
- 註冊藍牙服務功能
1. 點選 BluetoothLE1
 2. 選擇 RegisteredForBytes 方塊並拖移至版面中。



- 前面提供的 UID 寫入至 RegisteredForBytes 方框內
1. 點選本文。
 2. 選擇第一個文字方塊，拖移至 ServiceUuid、characteristicUuid 並串接起來。
 3. 填入讀取資料流程服務 ID。
 4. 填入讀取資料流程特徵 ID。
 5. 點選邏輯。
 6. 選擇 假方塊，拖移至 RegisteredForBytes 內的 signed 並串接起來。

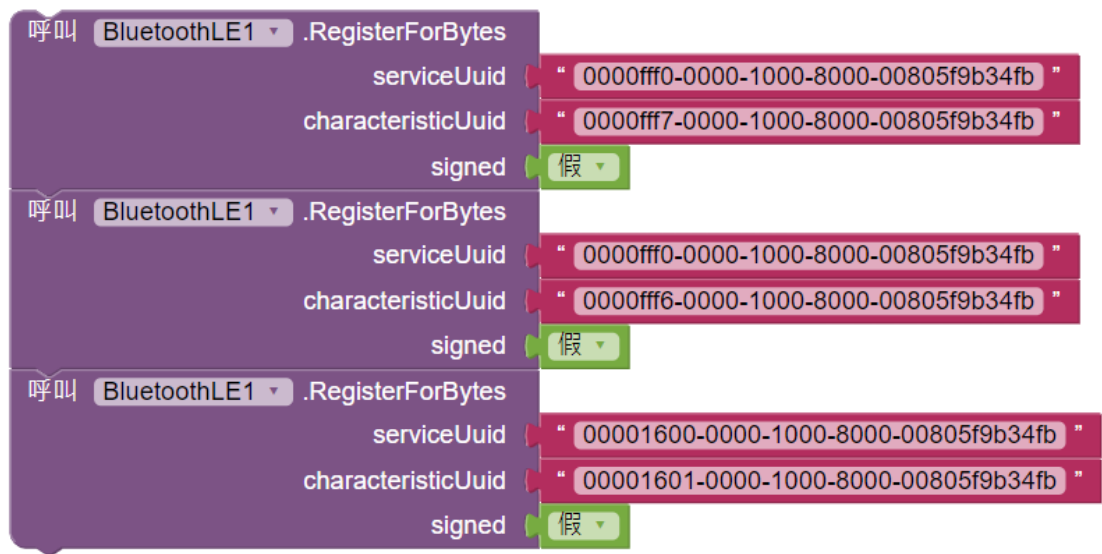


一樣的流程，再做一個寫入資料流程

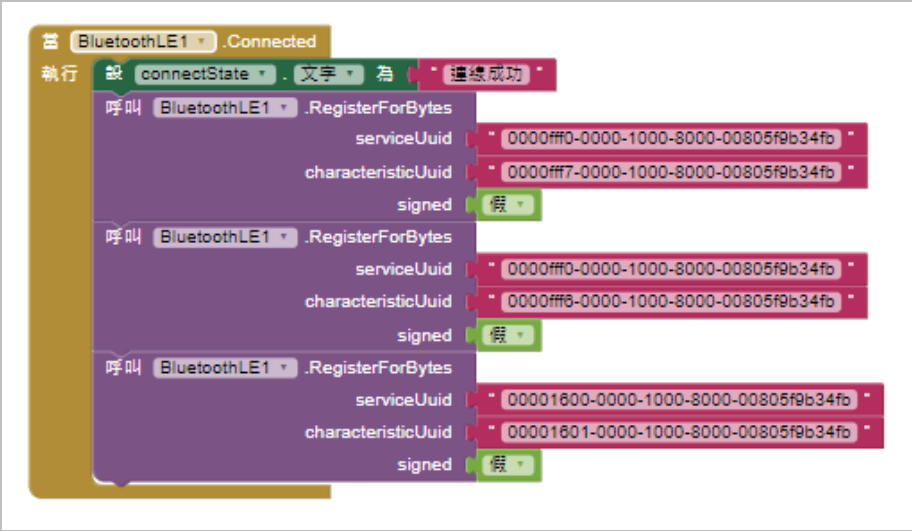


讀取藍牙裝置 Rabboni 感測值資訊

完成後把三個 RegisterForBytes 串接再一起

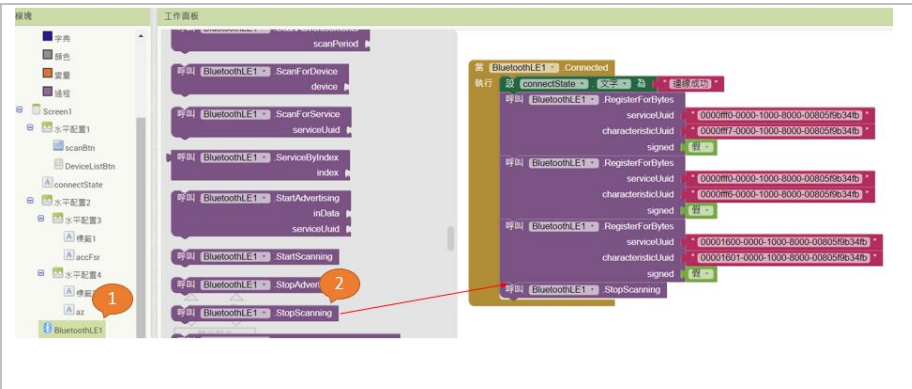


當連線成功時，把以上註冊流程一起寫入



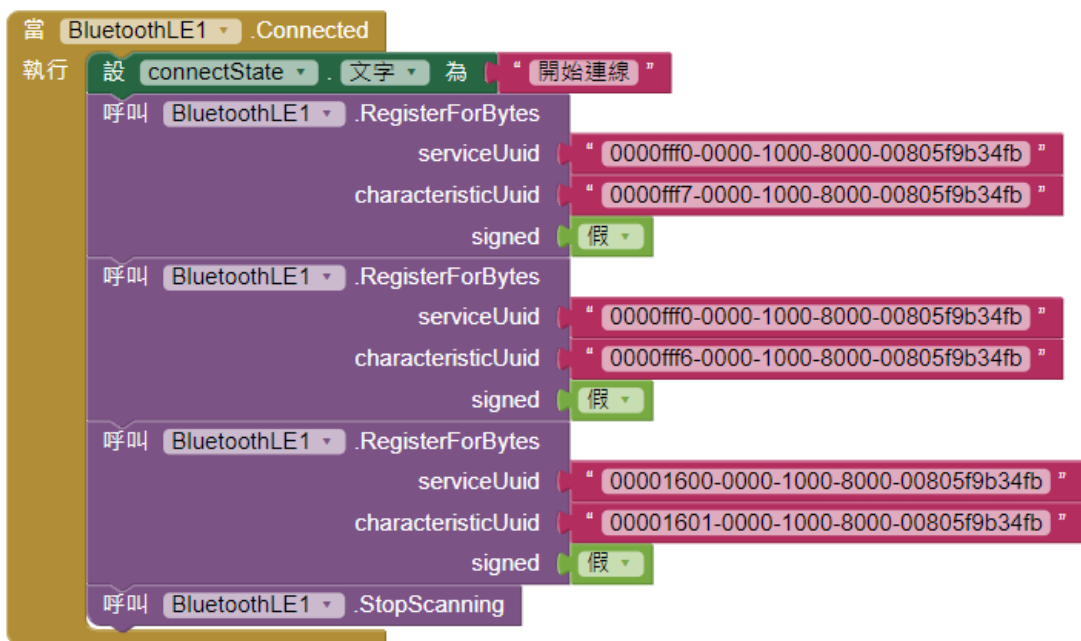
當連線成功時，把 RegisterForBytes 三個方塊一起放入至 BluetoothLE1 Connected 方塊內

同時因為連線成功，無須再進行掃描，把 BluetoothLE1 中的停止掃描方塊串接至此連線成功的方塊中



當連線成功後，把掃描功能關閉

1. 點選 BluetoothLE1。
2. 選擇 StopScanning 方塊，拖移至 BluetoothLE1 Coonected 方塊內並接續在 RegisterForBytes 方塊下方。



當有連線成功，就需要做連線失敗的流程，告知使用者目前狀況



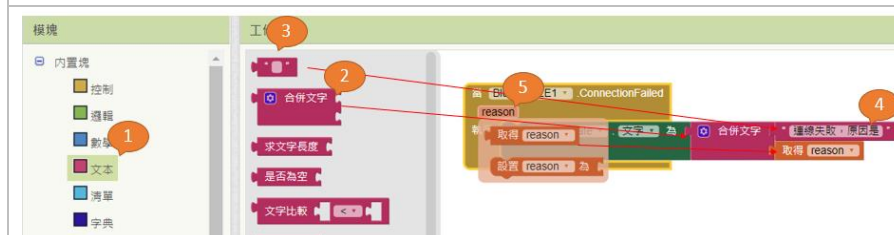
當連線失敗後，會進入此功能

1. 點選 BluetoothLE1。
2. 選擇 ConnectionFailed 中並拖移至版面中。
3. 可點選 ConnectionFailed 方塊中，此方塊有提供如果有 failed 會告知 reason 文字。



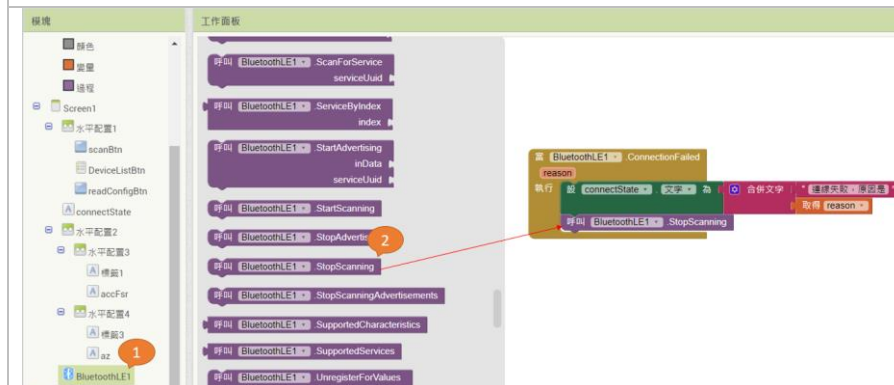
當連線失敗後，connectState 修改文字

1. 點選 connectState。
2. 選擇 文字方塊並拖移至 ConnectionFailed 方塊內的執行並串接。



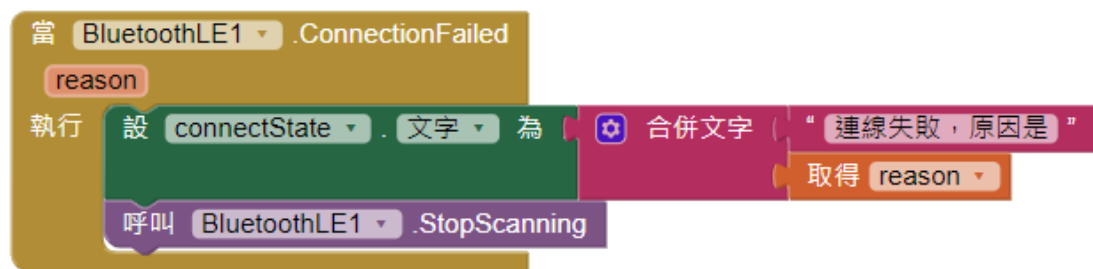
顯示文字

1. 點選本文。
2. 選擇 合併文字方塊，拖移至 connectState 方塊後方並串接。
3. 選擇第一個文字方塊，拖移至合併文字後方並串接。
4. 修改內部文字 — 連線失敗，原因是
5. 點選 ConnectionFailed 的 reason 方塊，選擇取得 reason 方塊並拖移至合併文字方塊後面串接。



如果已連線失敗，把藍牙掃描功能關閉

1. 點選 BluetoothLE1。
2. 選擇 StopScanning 方塊並移動至執行裡面並連接。



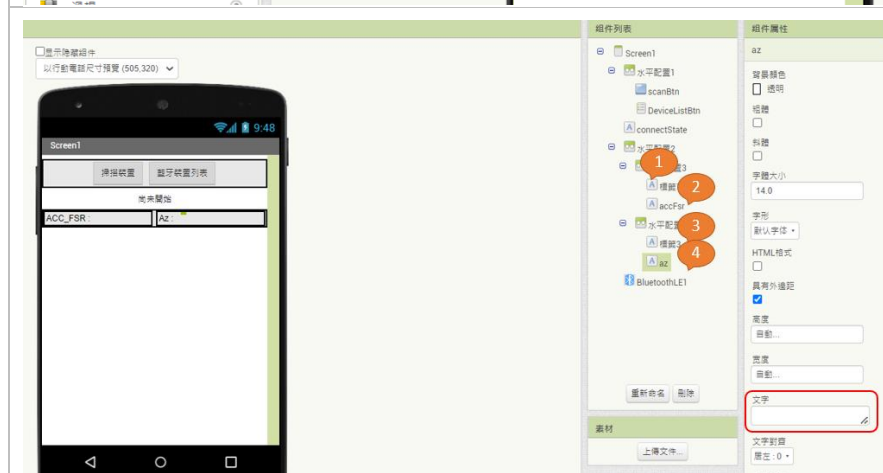
開始製作收取資料頁面並根據不同裝置設定內容進行調整
回到畫面編排的頁面



1. 點選介面配置 – 水平配置，先拖拉一個至 connectState 標籤下方。完成後在拉兩個水平配置至這個水平配置中，所以說有兩個水平配置放置在一個水平配置中。



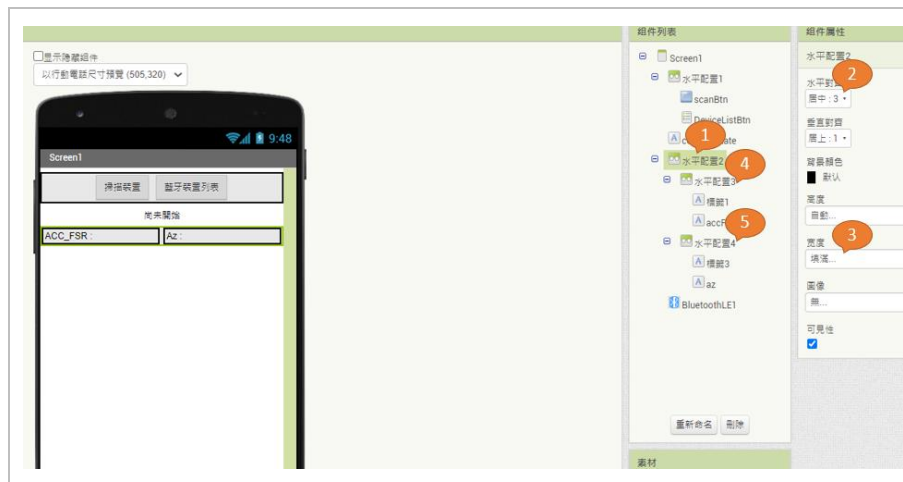
1. 點選介面配置 – 標籤，在上方內部的兩個水平配置，也在各拉取兩個標籤至水平配置內。



修改標籤 ID 碼

1. 選擇左側的水平配置中，第一個標籤，只需在右側的預設文字框內輸入 ACC_FSR :。
2. 選擇左側的水平配置中，第二個標籤，並重新命名 ID 為 accFsr。
3. 選擇右側的水平配置中，第一個標籤，只需在右側的預設文字框內輸入 Az :。
4. 選擇右側的水平配置中，第二個標籤，並重新命名 ID 為 az。

再進行水平配置調整



修改三個水平配置版面，以利於展示。

1. 點選水平配置。
2. 水平對齊中，選擇居中 3。
3. 寬度選擇填滿。
4. 按照以上步驟，但在水平對齊中，選擇居左 1。
5. 按造以上步驟，但在水平對齊中，選擇居左 1。

此設置方式是為了讓呈現上能夠更完整，各自的第一個標籤文字是抬頭文字，後面才是根據藍牙資料顯示之內容

完成以上配置後，選擇程式設計進行後續開發

在開發前先介紹控制方塊，此方塊是進行判斷是否是開發者要的资料，是的话再进行后续



1. 判断成立的逻辑
2. 如果逻辑成立的话就会进入 则 这边的流程

也可以點選左上角設定的藍色 icon，往下增加判断流程



在前面有介绍我们注册了三个流程，实际上读取是两个，我们需要判断各自回来的资料是谁的，所以我们必须运用控制方块来进行判断

再來介紹邏輯方塊介紹



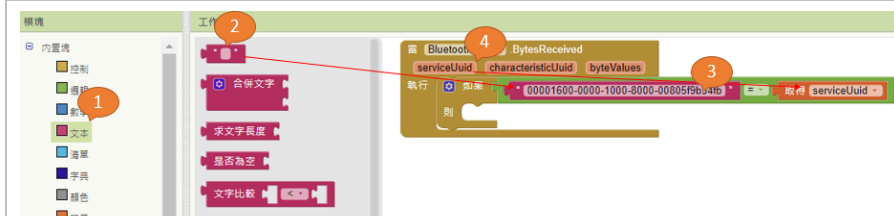
在綠色方塊中可以看到有 等於 的綠色方塊，這意思是說如果左邊等於右邊，代表這個邏輯就是成立的。

因為在前方我們註冊了有兩個會回傳資訊的服務，讀取資料流程、讀取藍牙裝置 Rabboni 狀態列表，我們再回傳資料的時候必須分配好。

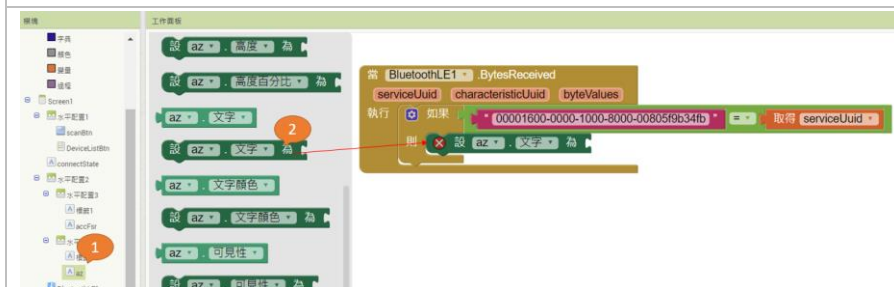
| | |
|--|--|
| | <p>讀取藍牙裝置回傳資料</p> <ol style="list-style-type: none">1. 點選 BluetoothLE1。2. 選擇 BytesReceived 方塊並拖移至版面中。3. 回傳的資料來源自哪個服務 ID。4. 回傳的資料來源自哪個特徵 ID。5. 回傳的資料。 |
| | <p>加入控制，來判斷回傳的來源。</p> <ol style="list-style-type: none">1. 點選控制。2. 選擇如果/則方塊，拖移至 BluetoothLE1 的 BytesReceiverd 中的執行。 |



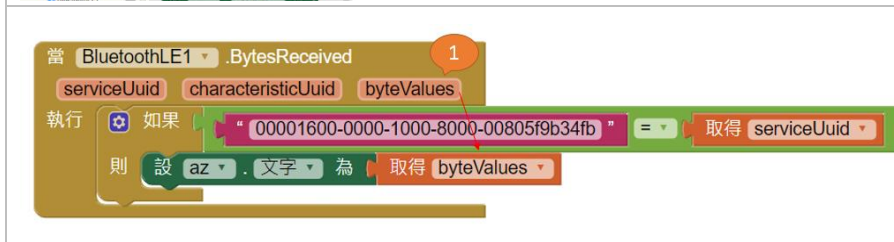
- 加入邏輯判斷，協助控制方塊
1. 點選邏輯。
 2. 選擇等於方塊，拖移至控制方塊如果旁並串接上。



- 加入邏輯內容
1. 點選文本。
 2. 選擇第一個文字方塊，拖移至邏輯方塊第一個欄位內。
 3. 內容請輸入讀取藍牙裝置 Rabboni 感測值資訊的服務 ID。
 4. 點選 BluetoothLE1 的 ByteReceived 中 serviceUuid 並拖移至邏輯方塊第二個欄位內。



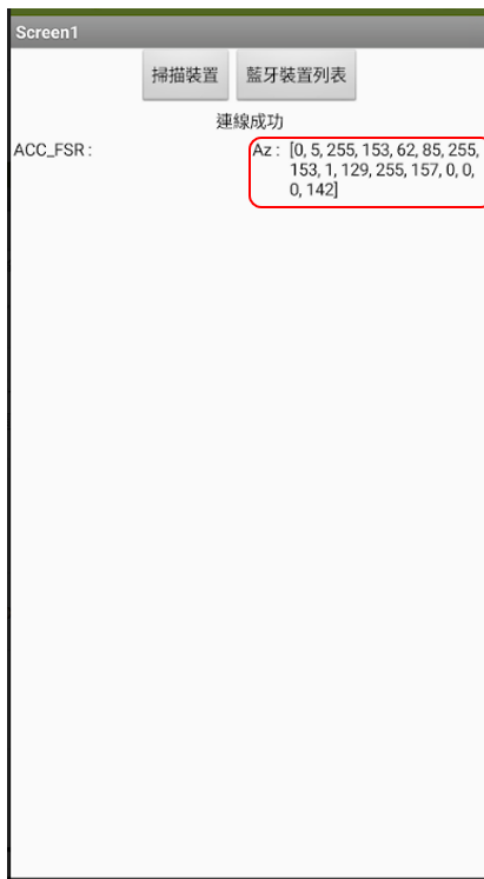
- az 接收藍牙裝置 Rabboni 感測值資訊
1. 點選 az。
 2. 選擇 文字方塊並拖移至控至方塊內的執行並串接。



1. BluetoothLE1 的 BytesReceived 中 byteValues 方塊拖移至 az 文字的後方並串接。

根據以上步驟我們實際測試。

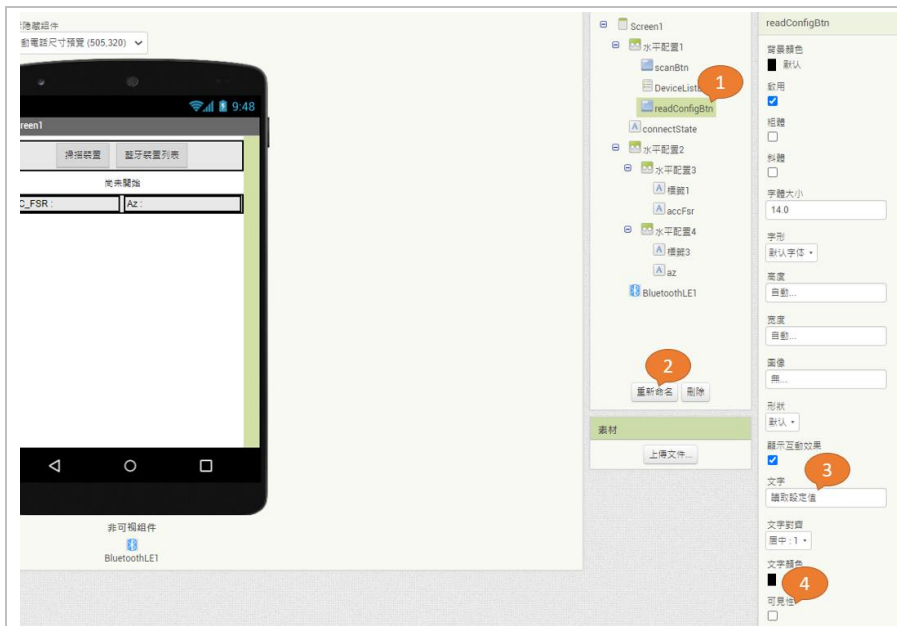
1. 裝置廣播鍵啟動。
2. 點選 APP 中的掃描裝置按鈕。
3. 點選 APP 中的藍牙裝置列表按鈕。
4. 彈出滿版裝置列表並點選 Rabboni 裝置。
5. 如果有成功看到成功連線。



如果有看到以上回傳資訊，代表資料已經有成功串接。

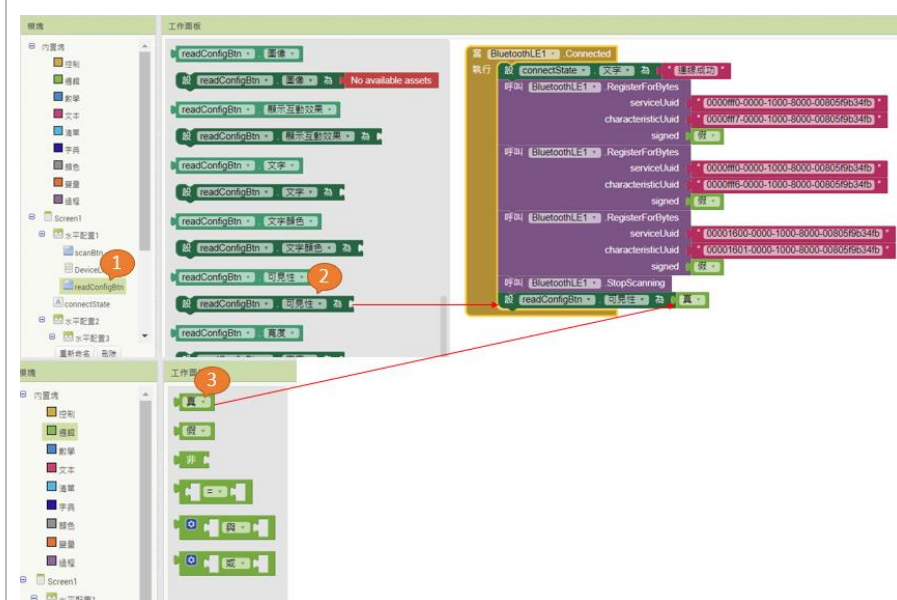
我們也需要讀取 Rabboni 藍牙裝置目前的設定參數，利用參數再進行後續換算，以及收到的資訊列表只會用到一個數字，所以我們要把資料裁切並取得第二個

| | |
|--|--|
| | <p>設定初始全域變數</p> <ol style="list-style-type: none"> 1. 點選變量。 2. 選擇初始化全域變數方塊並拖移至版面中。 3. 內容參數設定為 configBytes。 4. 選擇取得方塊並拖移至初始化全域變數方塊後方並串接，下拉選單選擇全域 configBytes |
| | <p>需寫入參數到 Rabboni 藍牙裝置來讀取特定資料回來</p> <ol style="list-style-type: none"> 1. 點選 BluetoothLE1。 2. 選擇 WriteBytes 方塊並拖移至版面中。 |
| | <p>BluetoothLE1 的 WriteBytes 方塊所需參數</p> <ol style="list-style-type: none"> 1. 點選文本。 2. 選擇第一個文字方塊並拖移至三個至 serviceUuid、characteristicUuid 和 values 並串接。 3. 點選邏輯 假方塊，移至 BluetoothLe1 的 WriteBytes 方塊的 signed 後面並串接。 |
| | <p>回到畫面編排，新增一個按鈕</p> <ol style="list-style-type: none"> 1. 點選使用者介面 按鈕，並拖移至最上面的水平配置中最右側 |



設定元件介面配置

1. 點選組件列表至該按鈕。
2. 重新命名，修改 ID 碼為 readConfigBtn。
3. 修改預設文字為 讀取設定值。
4. 可見性把打勾刪除，此時在版面上的按鈕就會不見。

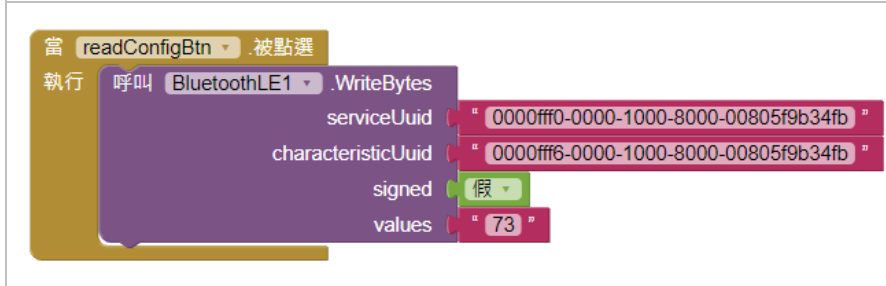


如果連線成功，把 readConfigBtn 按鈕顯示出來。

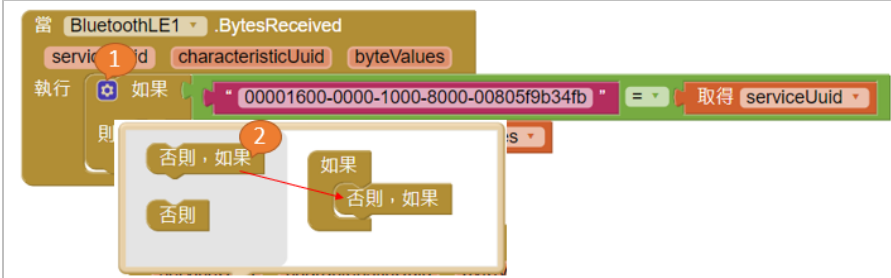
1. 點選 readConfigBtn。
2. 選擇可見性方塊，拖移至 BluetoothLE1 Connected 方塊下方，並接續 StopScanning 方塊下面。
3. 點選邏輯 真方塊並拖移至可見性後面並串接。



1. 點選 readConfigBtn。
2. 選擇被點選方塊並拖移至版面中。

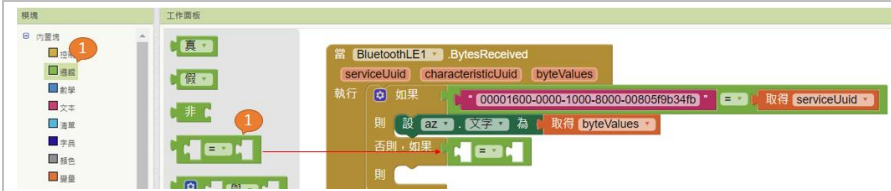


1. 把前面 BluetoothLE1 的 WriteBytes 方塊拖移至 readConfigBtn 被點選的執行內。



增加 Received 判斷流程

1. 點選設定圖案 icon，並跳出小視窗。
2. 點選否則，如果並拖移至旁的如果下方並串接。



加入邏輯判斷，協助控制方塊

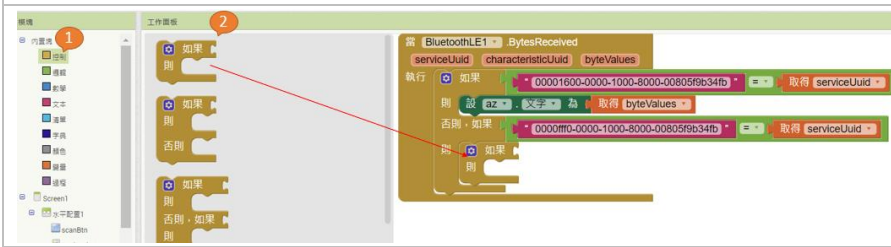
1. 點選邏輯。

選擇等於方塊，拖移至控制方塊如果旁並串接上。



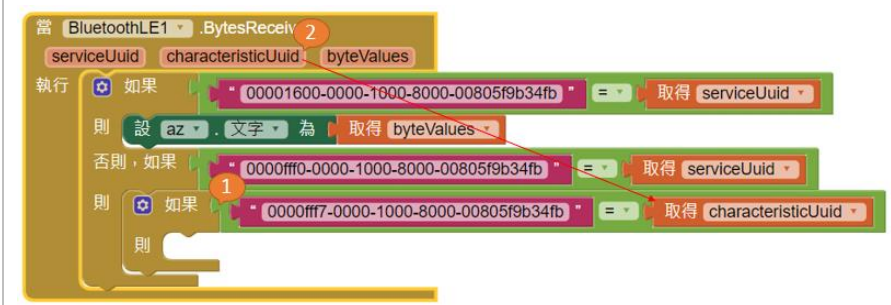
加入邏輯內容

1. 點選文本。
2. 選擇第一個文字方塊，拖移至邏輯方塊第一個欄位內。
3. 內容請輸入讀取資料流程的服務 ID
4. 點選 BluetoothLE1 的 ByteReceived 中 serviceUuid 並拖移至邏輯方塊第二個欄位內。



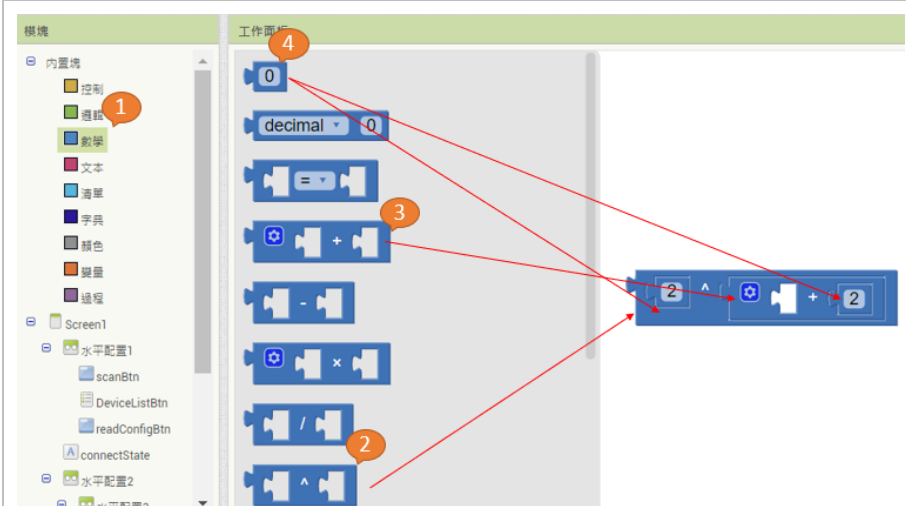
再增加一個控制方塊至如果方塊裡面

1. 點選控制。
2. 選擇如果方塊拖移至否則，如果下的則後方並串接。

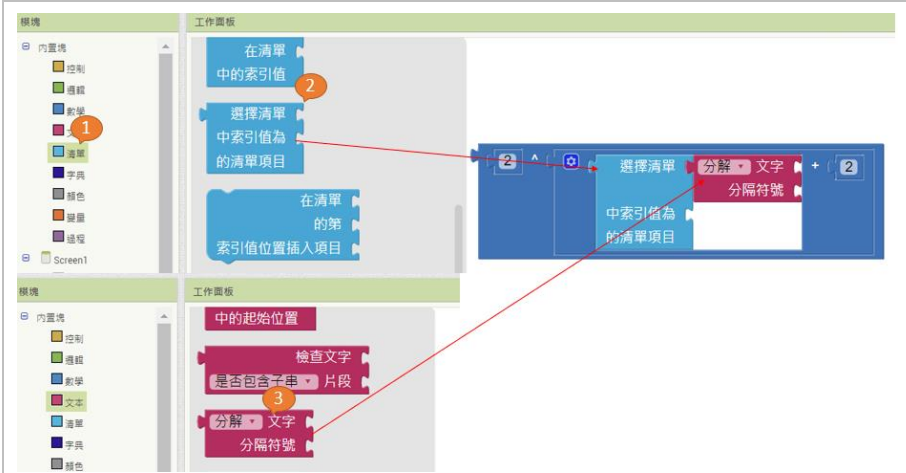


增加 Received 判斷流程

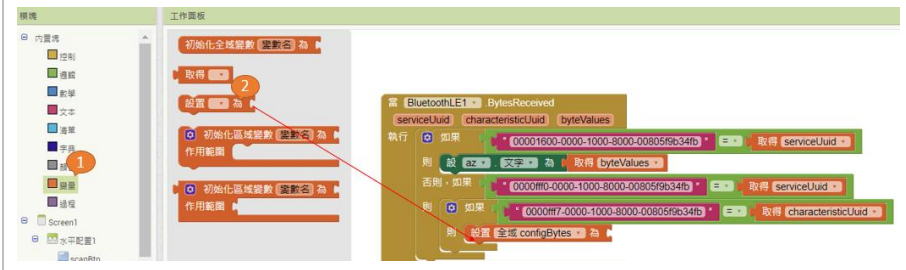
1. 按照前面步驟，一樣新增一個邏輯進來。
2. 邏輯 等於方塊的後方，請選擇 characteristicUuid 方塊拖移至後方並串接。



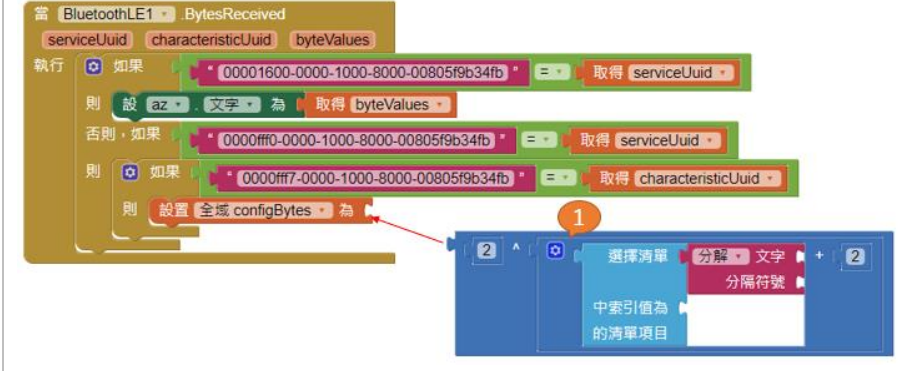
- 加入數學運算
1. 點選數學。
 2. 選擇次方[^]方塊並拖移至版面中。
 3. 選擇相加方塊，並移至次方塊的第二個欄位。
 4. 選擇常數方塊，移至次方塊中的第一個欄位以及相加方塊的第二個欄位，並修改數字為 2。



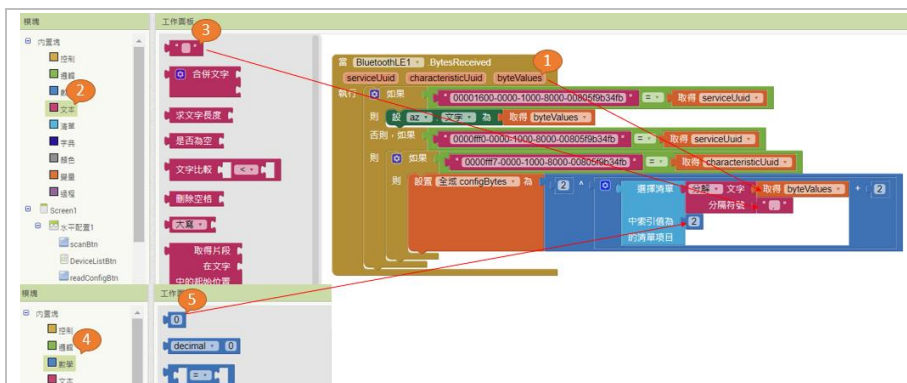
- 加入拆解資料序列方法
1. 點選清單。
 2. 選擇 選擇清單中索引值方塊並拖移至相加方塊的第一個欄位。
 3. 點選文本並選擇分解文字方塊並拖移至清單方塊的第一個選擇清單旁並串接。



1. 點選變數。
2. 選擇設置方塊，拖移至 BluetoothLE1 的 ByteReceived 內的控制方塊則之後，並選擇 configBytes

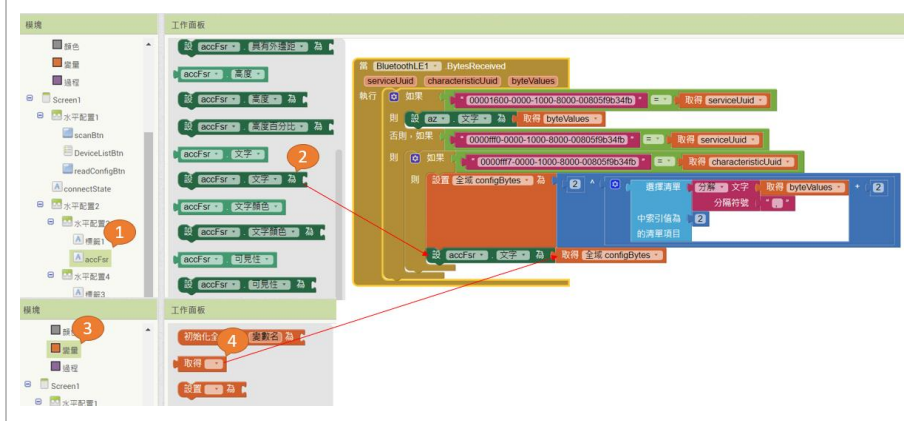


1. 把數學方塊整個加入至變數後方便串接。



添加資料來源以及拆解項目

1. BluetoothLE1 的 BytesReceived 中的 byteValues 拖移至分解文字旁並串接。
2. 點選文本。
3. 選擇第一個文字方塊並拖移至分解方塊的分解符號旁，並輸入[,]，利用這個來拆解內容。
4. 點選數學。
5. 選擇常數方塊並拖移至清單方塊中的索引值，並輸入數字[2]。

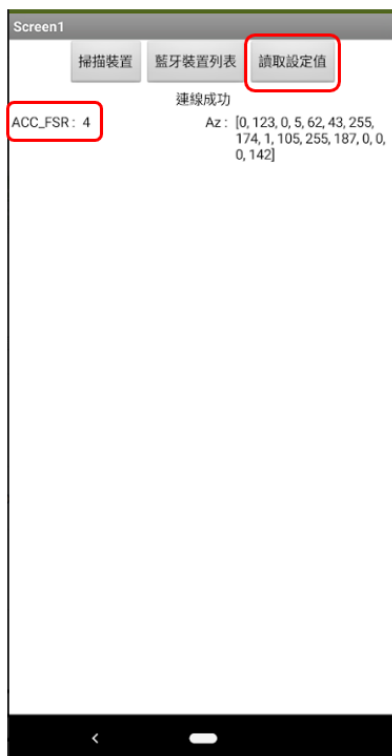


顯示 accFsr 資料

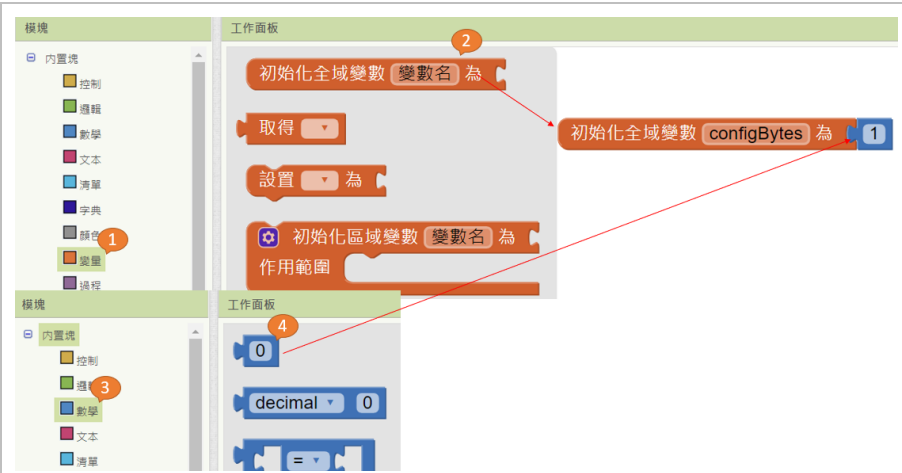
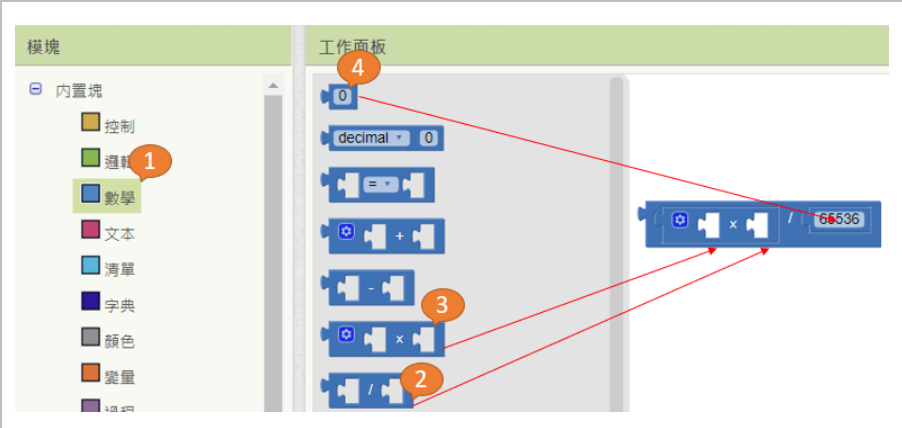
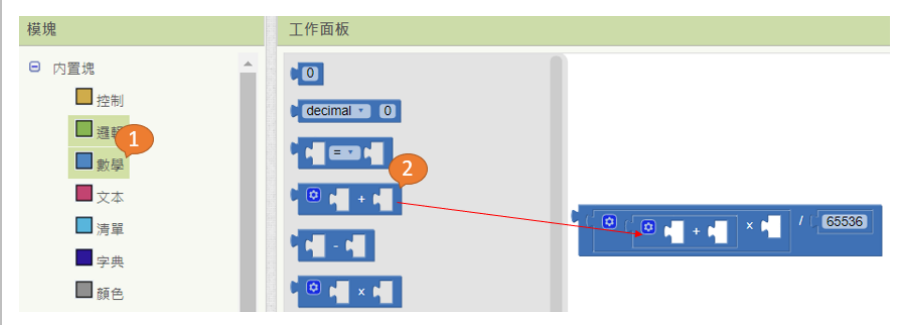
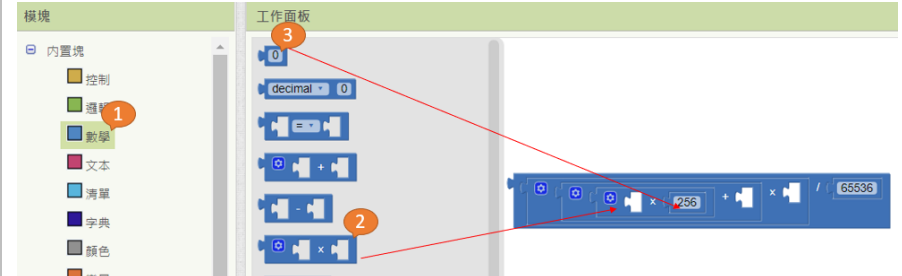
1. 點選 accFSR。
2. 選擇 文字方塊並拖移至控至方塊內的執行並串接。
3. 點選變量。
4. 選擇 取得方塊並拖移至文字方塊後方並串接，且選擇下拉選單內容 configBytes。

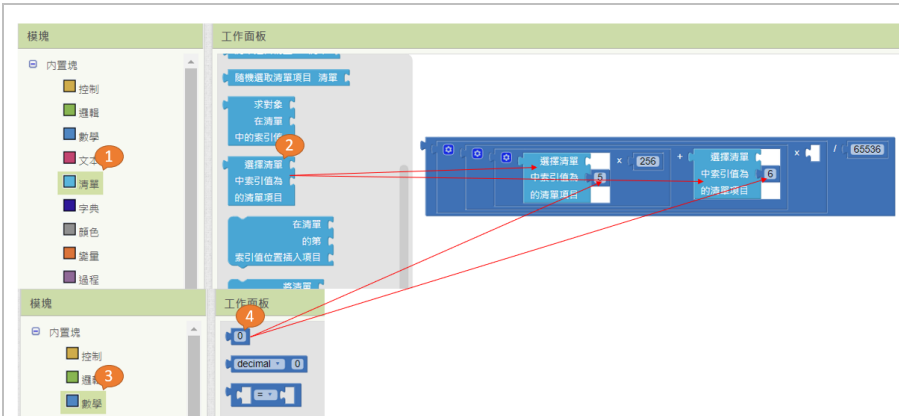
根據以上步驟我們實際測試。

1. 裝置廣播鍵啟動。
2. 點選 APP 中的掃描裝置按鈕。
3. 點選 APP 中的藍牙裝置列表按鈕。
4. 彈出滿版裝置列表並點選 Rabboni 裝置。
5. 如果有成功看到成功連線。
6. 若連線成功，讀取設定值的按鈕就會顯示出來。



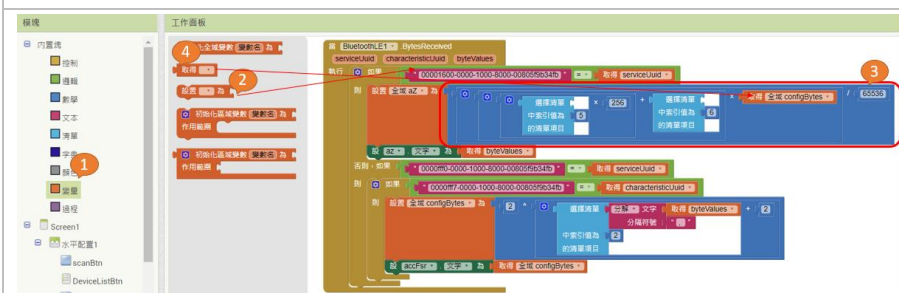
在取得 az 上，讀取藍牙裝置 Rabboni 感測值資訊，只需要讀取第五、第六的參數並換算得到實際 az 的值

| | |
|---|---|
|  | <p>設定 az 全域變數</p> <ol style="list-style-type: none"> 1. 點選變量。 2. 選擇初始化全域變數方塊並拖移至版面中。 3. 點選數學。 4. 選擇 常數方塊，並拖移至初始化全域變數後方便串接，常數修改成 1。 |
|  | <p>新增 aZ 數學公式</p> <ol style="list-style-type: none"> 1. 點選數學。 2. 選擇 相除方塊，並拖移至版面中。 3. 選擇 相乘方塊，並拖移至相除方塊的第一格欄位。 4. 選擇 常數方塊，並拖移至相除方塊的第二格欄位，調整參數至 65536。 |
|  | <ol style="list-style-type: none"> 1. 點選數學。 2. 選擇 相加方塊，並拖移至相乘方塊的第一格欄位。 |
|  | <ol style="list-style-type: none"> 1. 點選數學。 2. 選擇 相乘方塊，並拖移至相加方塊的第一格欄位。 3. 選擇 常數方塊，並拖移至相乘方塊的第二格欄位。 |



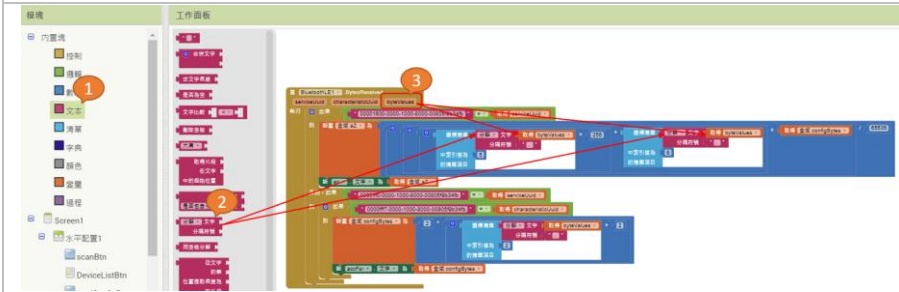
加入拆解資料序列方法

1. 點選清單。
2. 選擇 選擇清單中索引值方塊，並分別拖移至相乘的第一格欄位以及相加的第二個欄位並串接起來。
3. 點選數學。
4. 選擇常數，分別拖移至選擇清單中的各自索引值旁，分別填入常數 5 以及常數 6。



在 BluetoothLE1 的 ByteReceived

1. 點選變量。
2. 選擇 設置方塊，並拖移至 1600，第一個控制方塊之下並串接，下拉選單選擇全域 aZ。
3. 把上面完成的數學整包方塊拖移至設置方塊後方並串接。
4. 選擇 取得方塊，拖移至相乘方塊的第二格欄位，下拉選單選擇 configBytes。



修改註冊服務流程

1. 點選文本。
2. 選擇 分解方塊並拖移至各自清單選擇器的第一格欄位並串接。
3. 點選 byteValues 並拖移至各自分解方塊的第一格欄位。

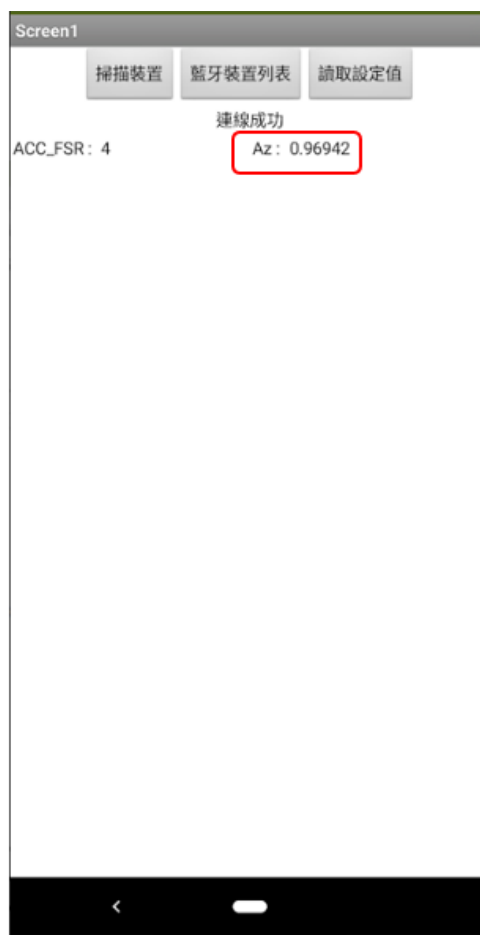


修改註冊服務流程

1. 點選原本 BluetoothLE1 Connected，裡面的最下方註冊方塊，並把此方塊轉移至 readConfigBtn 被點選方塊下方並串接。

根據以上步驟我們實際測試。

1. 裝置廣播鍵啟動。
2. 點選 APP 中的掃描裝置按鈕。
3. 點選 APP 中的藍牙裝置列表按鈕。
4. 彈出滿版裝置列表並點選 Rabboni 裝置。
5. 如果有成功看到成功連線。
6. 若連線成功，讀取設定值的按鈕就會顯示出來。
7. 點選讀取設定值後，會發現 ACC_FSR 跟 AZ 的資訊都會跑出來。



開始製作收取資料頁面並根據不同裝置設定內容進行調整
回到畫面編排的頁面



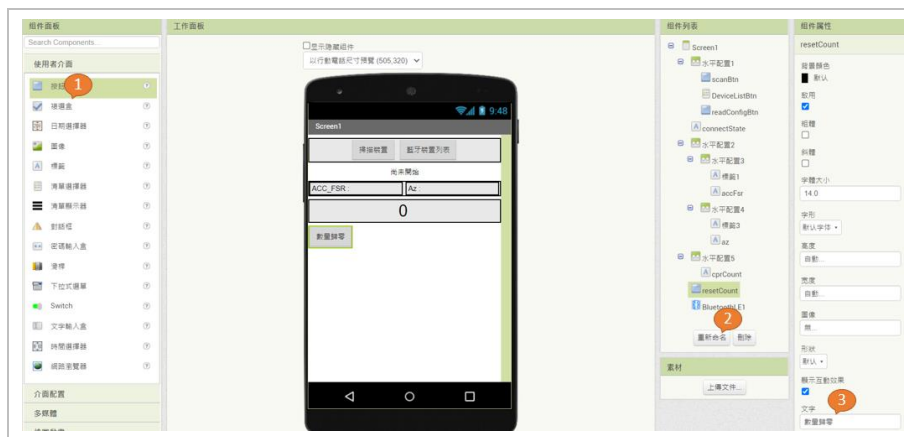
1. 點選介面配置 – 水平配置，先拖移至版面中。



1. 寬度，選擇填滿。
2. 水平對齊，下拉選單選擇居中 3。



1. 點選標籤並拖移至水平配置中。
2. 該標籤重新命名 ID 為 cprCount。
3. 字體大小調整至 30。
4. 文字預設至 0。



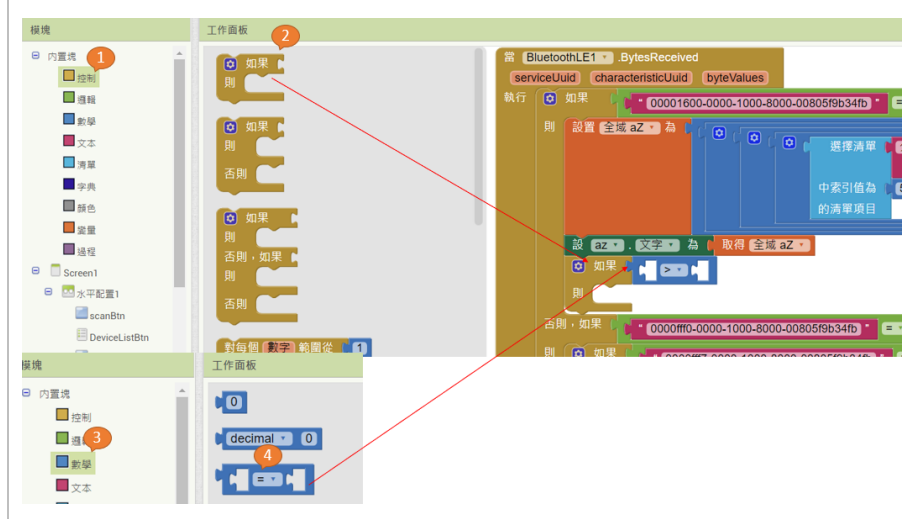
1. 點選按鈕，並拖移至版面中。
2. 該按鈕重新命名 ID 為 resetCount。
3. 按鈕預設文字修改成數量歸零。

切換至程式設計頁面進行編輯



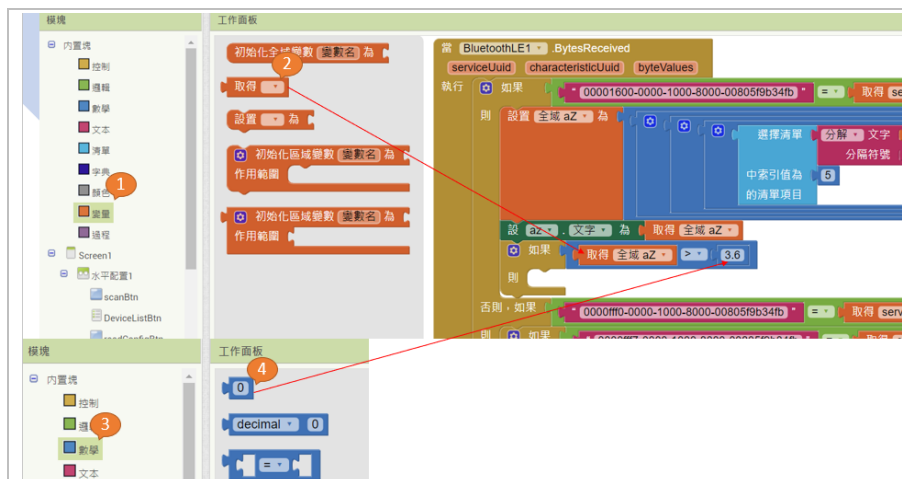
初始化全域變數 cprCount

1. 點選變量。
2. 選擇 初始化全域變數方塊並拖移至版面中，修改變數為 cprCount。
3. 點選數學。
4. 選擇 常數方塊並拖移至初始化全域變數方塊後方並串接。

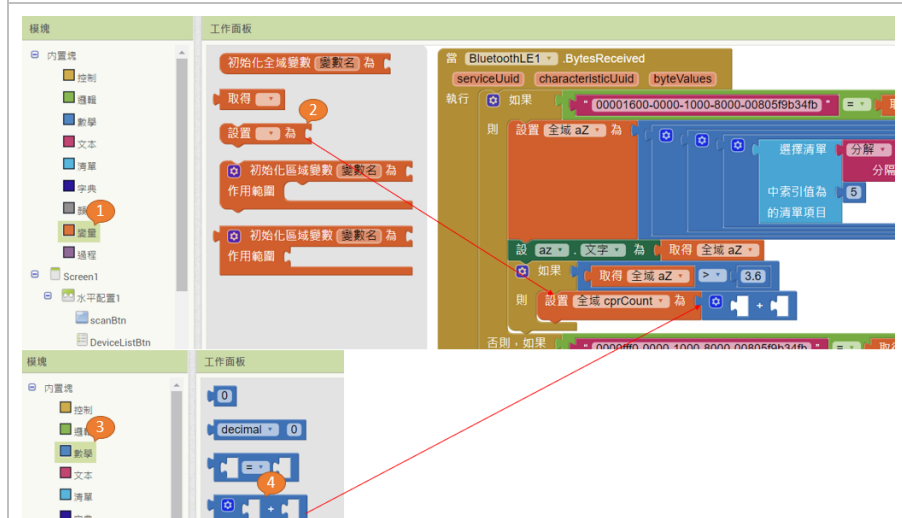


加入控制邏輯

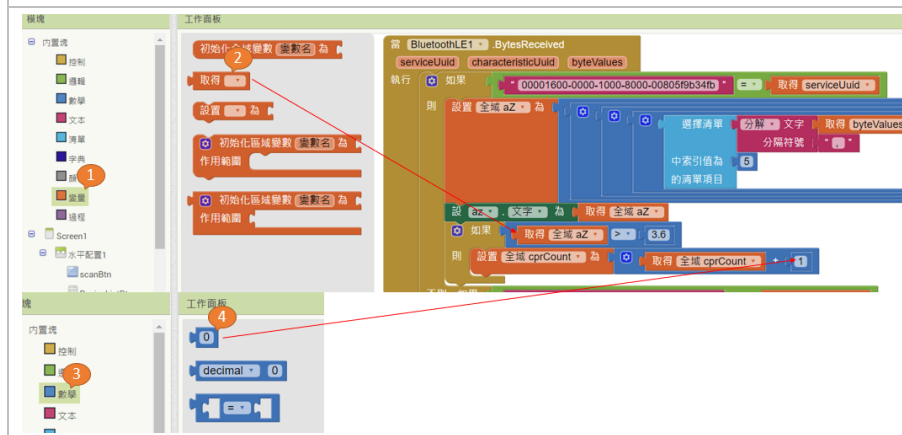
1. 點選控制。
2. 選擇 如果方塊並拖移至 BluetoothLE1 的 ByteReceived 方塊中內的 1600 執行內，並串接。
3. 點選數學。
4. 選擇 等於方塊並拖移至 如果方塊並串接，選擇下拉選單為大於。



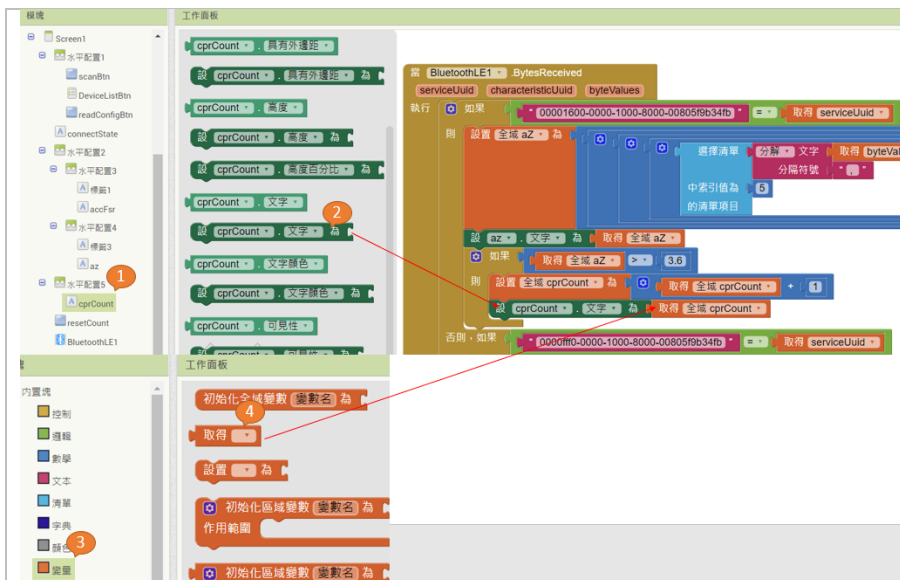
1. 點選變量。
2. 選擇 取得方塊，並拖移至大於方塊的第一格欄位，下拉選單選擇全域 aZ。
3. 點選數學。
4. 選擇 常數方塊，並拖移至大於方塊的第二格欄位，常數設定成 3.6。



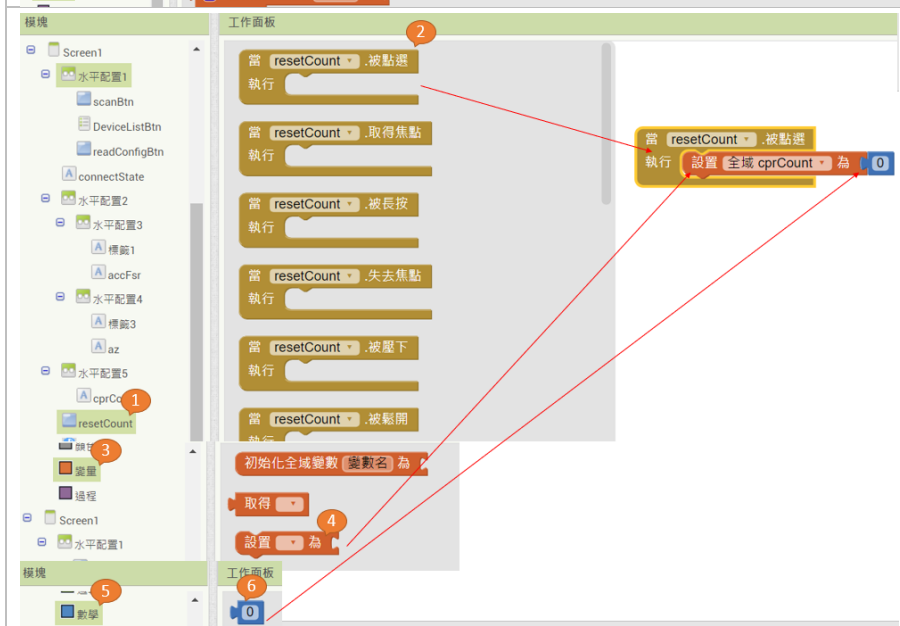
1. 點選變量。
2. 選擇 設置方塊，並拖移至如果方塊執行內並串接，下拉選單選擇全域 cprCount。
3. 點選數學。
4. 選擇 相加方塊，並拖移至設置方塊後方並串接。



1. 點選變量。
2. 選擇 取得方塊，並拖移至相加方塊的第一格欄位，下拉選單選擇 cprCount。
3. 點選數學。
4. 選擇 常數方塊，並拖移至相加選單的第二格欄位，常數設定成 1。

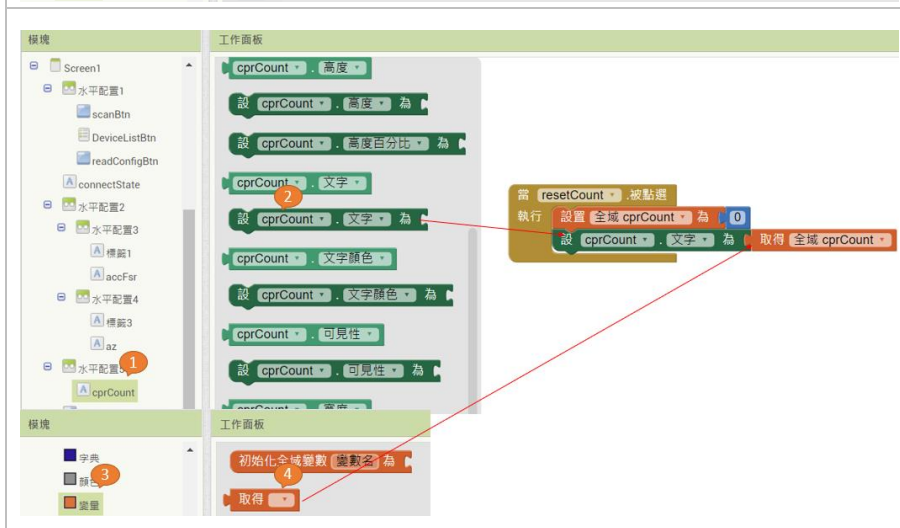


1. 點選 cprCount。
2. 選擇 文字方塊，並拖移至變數 設置方塊下方並串接。
3. 點選變量。
4. 選擇 取得方塊並拖移至文字方塊又方並串接，下拉選單選擇全域 cprCount。



點選歸零按鈕，reset 參數

1. 點選 resetCount。
2. 選擇 被點選方塊，並拖移至版面中。
3. 點選變量。
4. 選擇 設置至方塊，並拖移至被點選方塊執行內串接，下拉選單選擇全域 cprCount。
5. 點選數學。
6. 選擇 常數方塊，並拖移至設置方塊後方串接。



1. 點選 cprCount。
2. 選擇 文字方塊，並拖移至設置方塊下方串接。
3. 點選變量。
4. 選擇 取得方塊，並拖移至文字方塊後方串接，下拉選單選擇全域 cprCount。

根據以上步驟我們實際測試。

1. 裝置廣播鍵啟動。
2. 點選 APP 中的掃描裝置按鈕。
3. 點選 APP 中的藍牙裝置列表按鈕。
4. 彈出滿版裝置列表並點選 Rabboni 裝置。
5. 如果有成功看到成功連線。
6. 若連線成功，讀取設定值的按鈕就會顯示出來。
7. 點選讀取設定值後，會發現 ACC_FSR 跟 AZ 的資訊都會跑出來。
8. 開始模擬 CPR 動作。
9. 會發現 count 數開始往上加。
10. 如果數量要歸零，再點選按鈕。

