

開源的 IASS 室內環控系統

Every One Can DIY Your Own Sensor Device

IASS 是一套免費開源的室內環控系統，最早由凌陽創新科技開發並免費釋出。透過 Arduino 開發板，搭配市面上常見且價廉物美的感測器，不需要撰寫任何程式，希望讓任何人皆能在 30 分鐘內行 DIY 組裝完成符合自己需求的感測裝置。

IASS 開源程式碼下載：<https://github.com/LinkItONEDevGroup/LASS/tree/master/IASS>

IASS 目前為 LASS 的一個子計劃，也歡迎開發者們加入 IASS 予以修改精進，並持續增加新的功能，如果您有興趣加入，請加入 LASS 的 FB 社團：

<https://www.facebook.com/groups/1607718702812067/>

A) IASS 特色.....	4
B) IASS 的安裝步驟.....	4
C) 辦公室專用感測裝置的安裝步驟.....	5
材料準備	5
準備好了？開始組裝吧！	6
溫溼度感測	6
PM2.5 感測.....	7
CO2 感測	8
光照度感測	9
最後的固定工作	10
完成了！	12
D) 居家專用感測裝置的安裝步驟.....	12
材料準備	12
準備好了？開始組裝吧！	13
溫溼度感測	13
MQ7 一氧化碳感測	14
MQ4 天然氣外洩偵測	15
最後的固定工作	16
完成了！	17
E) 居家專用感測裝置的安裝步驟.....	18
材料準備	18

準備好了？開始組裝吧！	19
溫溼度感測	19
FC-37 漏水感測	20
MQ2 可燃氣體及煙霧偵測	22
最後的固定工作	23
完成了！	24
F) 替 IASS 加上 LCD 螢幕	25
準備材料：	25
開始組裝吧！	26
接上 LCD	26
將 LCD 固定好	26
程式修改	27
完成了！	27
G) 申請及設定 Thinkspeak	28
H) 把 Arduino 程式燒錄到裝置	30
I) 設定 ThingSpeak Field	32
J) 自行架設 IASS 主機	35
準備 web server，	35
建立資料庫	36
新增 MySQL account	37
設定要顯示的裝置及 sensors	37
將 IAASweb 目錄整個上傳到網站伺服器。	39

將 web 主機相關資訊更新到各感應 device	39
後續修改及調整	40

A) IASS 特色

任何人皆能 DIY 感測裝置：

不需要程式設計或相關電子工程知識，每個人皆能 DIY 自己的環控系統。

支援多種感測功能：

目前支援 12 種環境感應功能：溫度、溼度、漏水、噪音、一氧化碳、天然氣、PM2.5、PM10、PM100、二氧化碳、環境光照度，可依自身需求決定要加裝的感應器。

不需要撰寫程式：

提供現成的 Arduino 程式可直接使用，分為 sensor 與 device 兩種，使用者可依需求選擇不同的 sensors 程式搭配成 device。

提供預設搭配好的感測裝置：

除了使用者可自行搭配決定感測器種類外，亦提供預設數種 device 直接使用：

- Office 辦公室：溫溼、PM2.5、CO2、光照度
- 電腦機房：溫溼、漏水、可燃氣體
- 家居室內：溫溼、一氧化碳、天然氣

自動上傳 ThingSpeak：

您可以讓組裝後的 IASS device 將資料自動上傳 ThingSpeak 以提供遠端瀏覽觀看（需自行註冊 ThingSpeak 帳號並於程式中放入 API KEY）

支援上傳指定 server：

除上傳 ThingSpeak 之外，亦可選擇將資料傳送到指定的 server 來接收(使用 GET)，目前 IASS 也有提供一 PHP 編寫的中控台網站。

B) IASS 的安裝步驟

A.1) 選擇您需要的感測裝置類型，購買零件材料開始 IY。

IASS 目前提供下列三種組裝範例。

- 辦公室專用
- 居家安全專用
- 機房專用

您也可以從 IASS GitHub 的 `Sensors_support` 目錄中選擇需要的感應器程式來組裝自己的感應裝置。

A.2) 決定您的裝置要將資料上傳 ThingSpeak 還是傳到特定的伺服器。

- 若要上傳 ThingSpeak，請選擇 `_thingspeak` 資料夾的 Arduino 程式。
- 若要傳到指定伺服器，請選擇 `_selfserver` 資料夾的 Arduino 程式。

A.3) 決定是否要加裝 LCD？

A.4) 開始燒錄 Arduino 程式。

A.5) 如果您選擇上傳 ThingSpeak，可開始進行 ThingSpeak 設定

如果您選擇傳到 IASS 的伺服器，請開始安裝設定主機環境。

C) 辦公室專用感測裝置的安裝步驟

材料準備

本裝置範例支援溫溼、PM2.5、CO2 及光照度等四種感測器功能，所需要的材料如下：

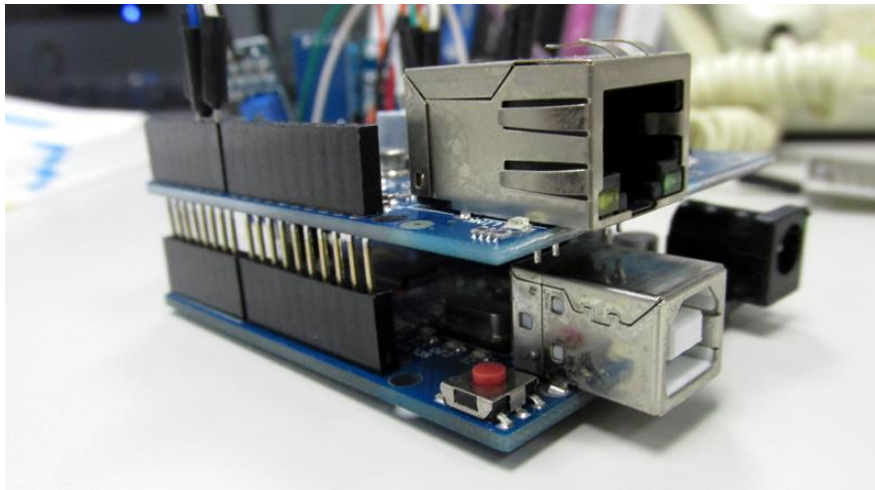
1. Arduino UNO 主板 x 1
http://www.icshop.com.tw/product_info.php/products_id/20369
2. W5100 R3 network shield x 1
<http://goods.ruten.com.tw/item/show?21434006463160>
3. DHT-22 sensor x 1 (溫溼度用)
http://www.icshop.com.tw/product_info.php/products_id/12478
4. G3 PM2.5 感測器 x 1 (PM2.5 用)
http://www.icshop.com.tw/product_info.php/products_id/20460

5. CO2 感測器 MH-Z16 x 1 (二氧化碳用)
http://www.icshop.com.tw/product_info.php/products_id/14133
6. 光照度感測器 TEMT6000 x 1 (光照度使用)
http://www.icshop.com.tw/product_info.php/products_id/11432
7. 9V 變壓器 x 1
(如果您同時裝上很多感應器，建議使用 9V 變壓器取代 USB，以避免電源不足)
http://www.icshop.com.tw/product_info.php/products_id/17470
8. 迷你麵包板 x 1
http://www.icshop.com.tw/product_info.php/products_id/11516
9. 杜邦線 數條
http://www.icshop.com.tw/product_info.php/products_id/12374
10. 小型 L 型立書夾 x 1 (\$39 元店有很多)

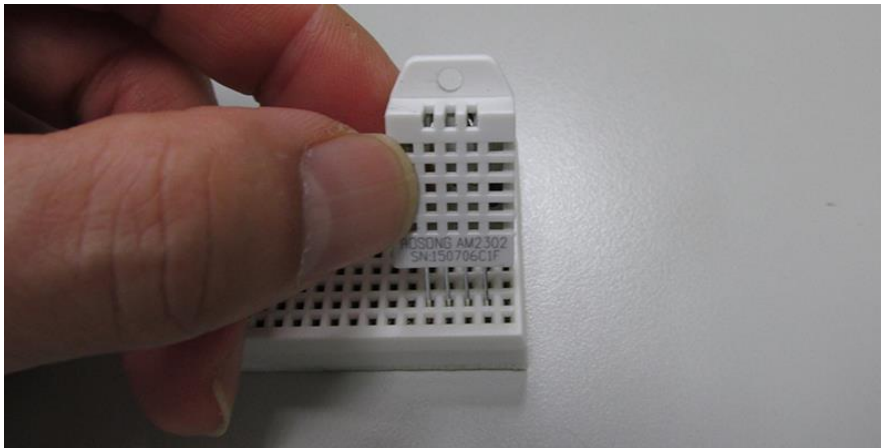
準備好了？開始組裝吧！

溫溼度感測

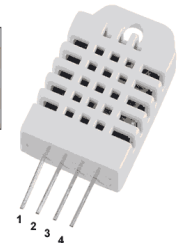
A) 首先，將材料#1 與材料#2 合體如下：



B) 把材料#3 插上，注意一下左圖它的腳位

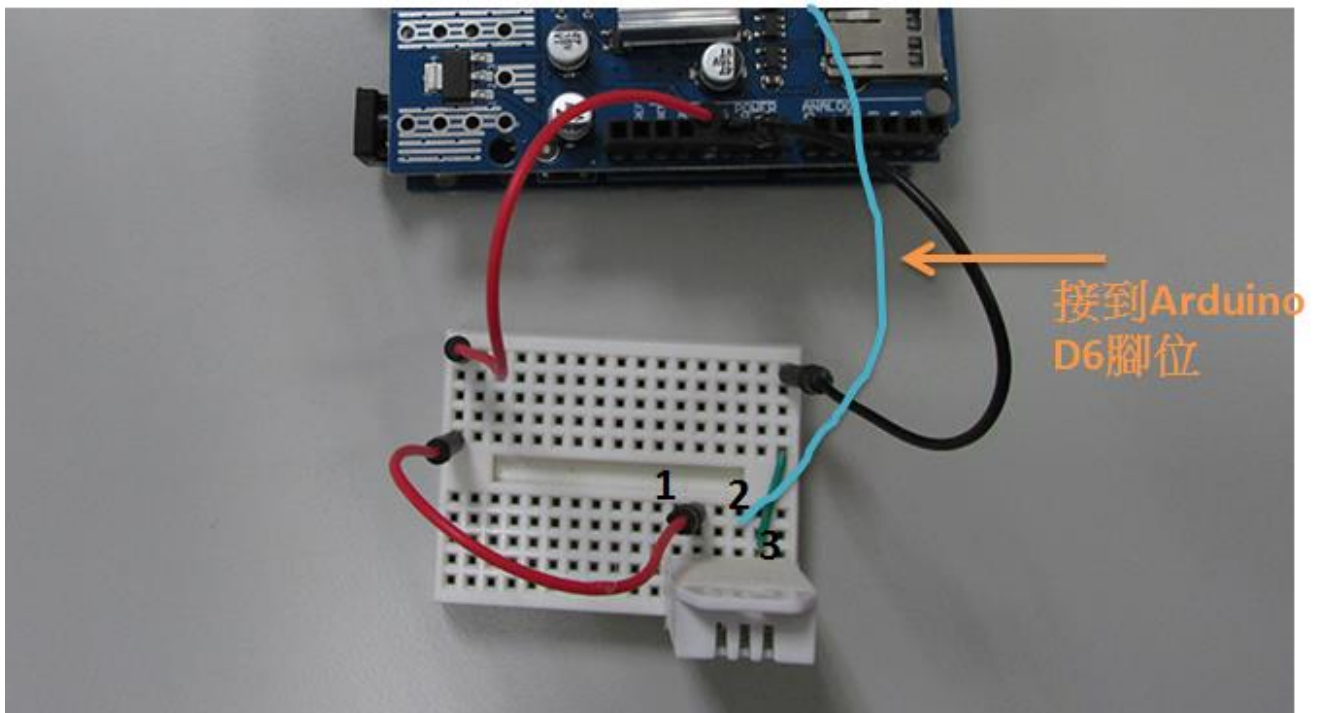


DHT22 pins	
1	VCC
2	DATA
3	NC
4	GND



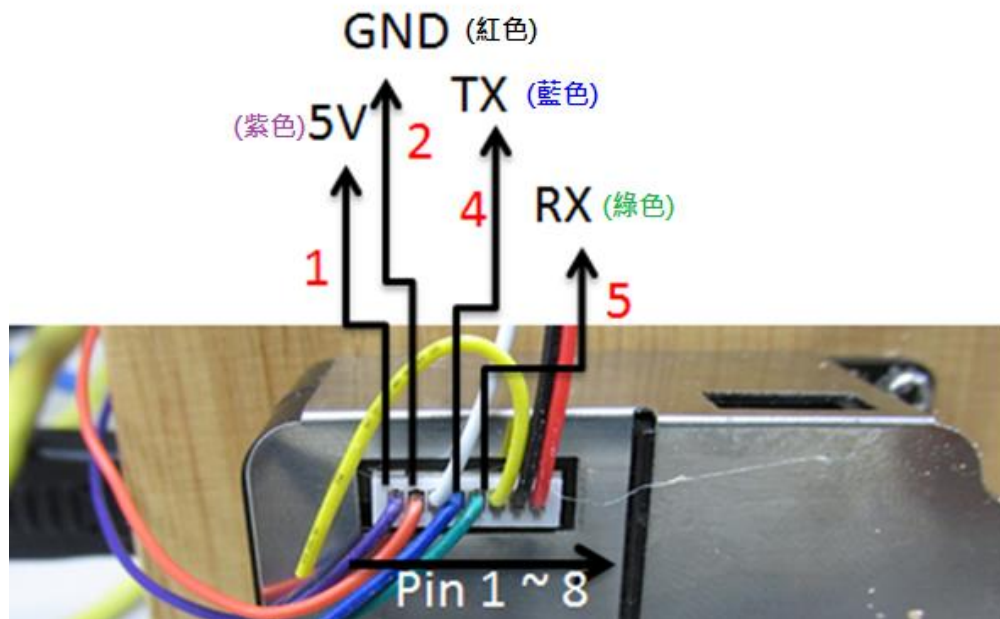
請照下圖方式來連接，Arduino 的 5V 和 GND 先接到麵包板上，再將材料#3 的第一根接腳接到 5V，第四根接到 GND，第三根則接到 Arduino D6。

如此，我們就完成了溫溼感測器的組裝。

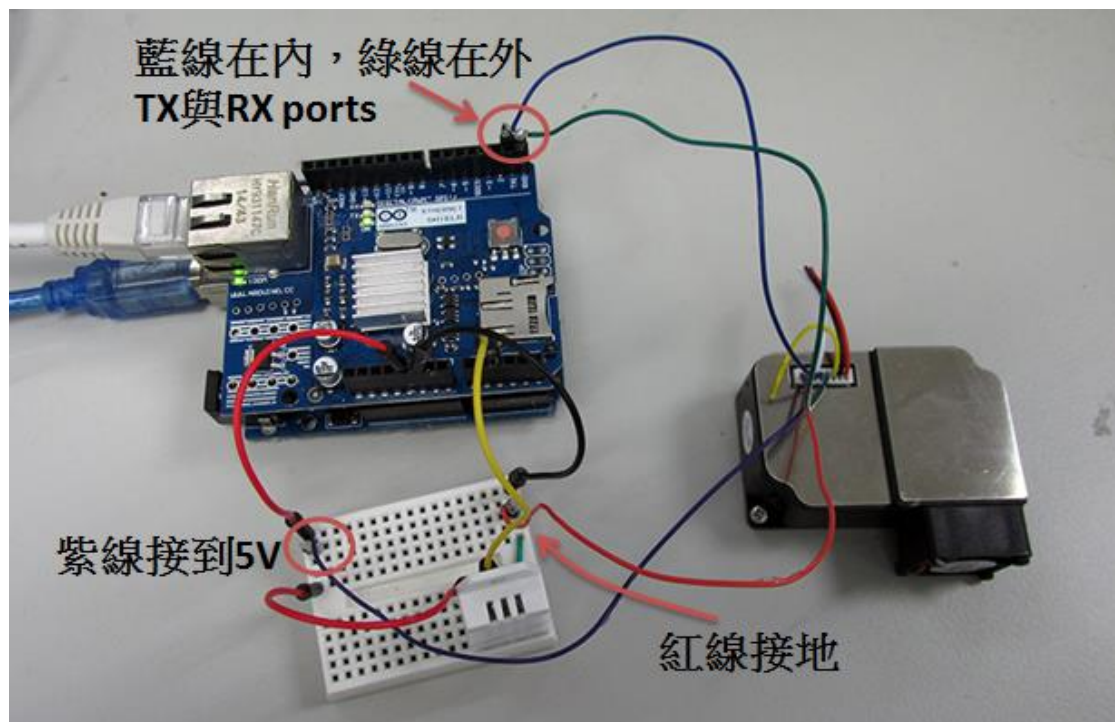


PM2.5 感測

請參考下方 G3 的腳位，我們只用到紫紅藍綠這四條。



請依照下圖使用材料#4 將其組裝即可順利完成。

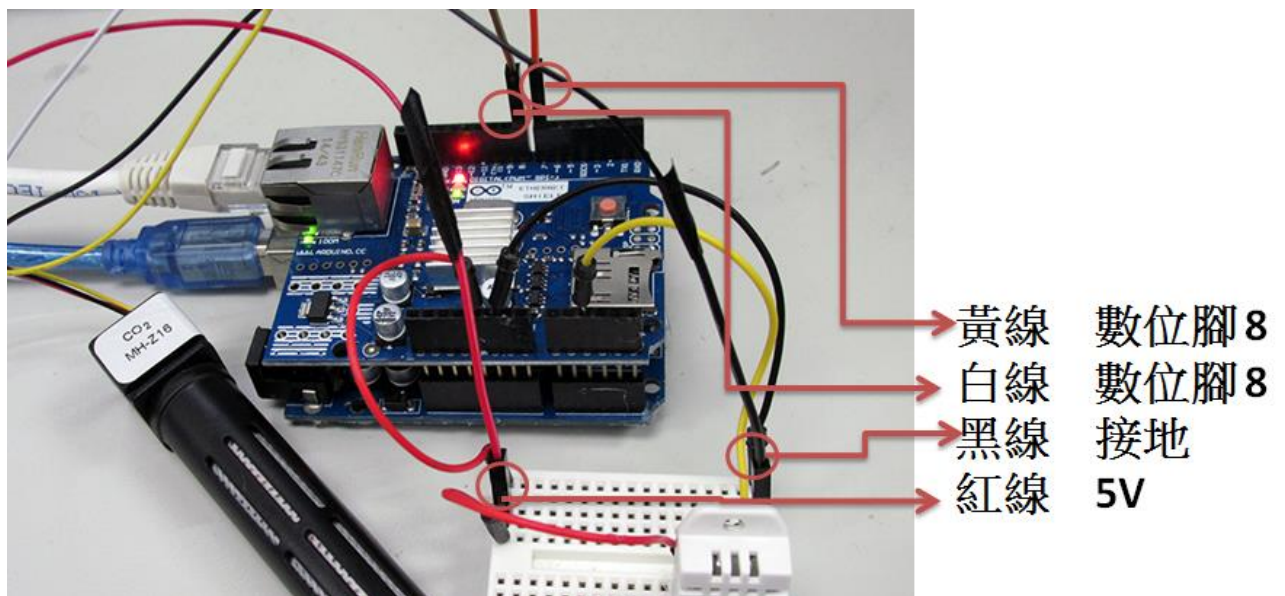


CO2 感測

MH-Z16 CO2 感測器是 Grove 接頭，其腳位功能如下：

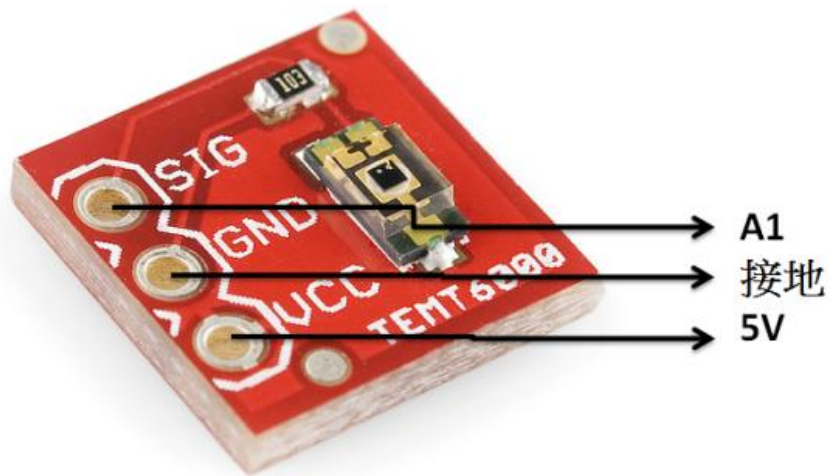


由於我們使用麵包板，因此要將此頭改裝為杜邦接頭，再分別將各接頭接到麵包板 5V, GND 及 Arduino A7 及 A8。請參考下圖：

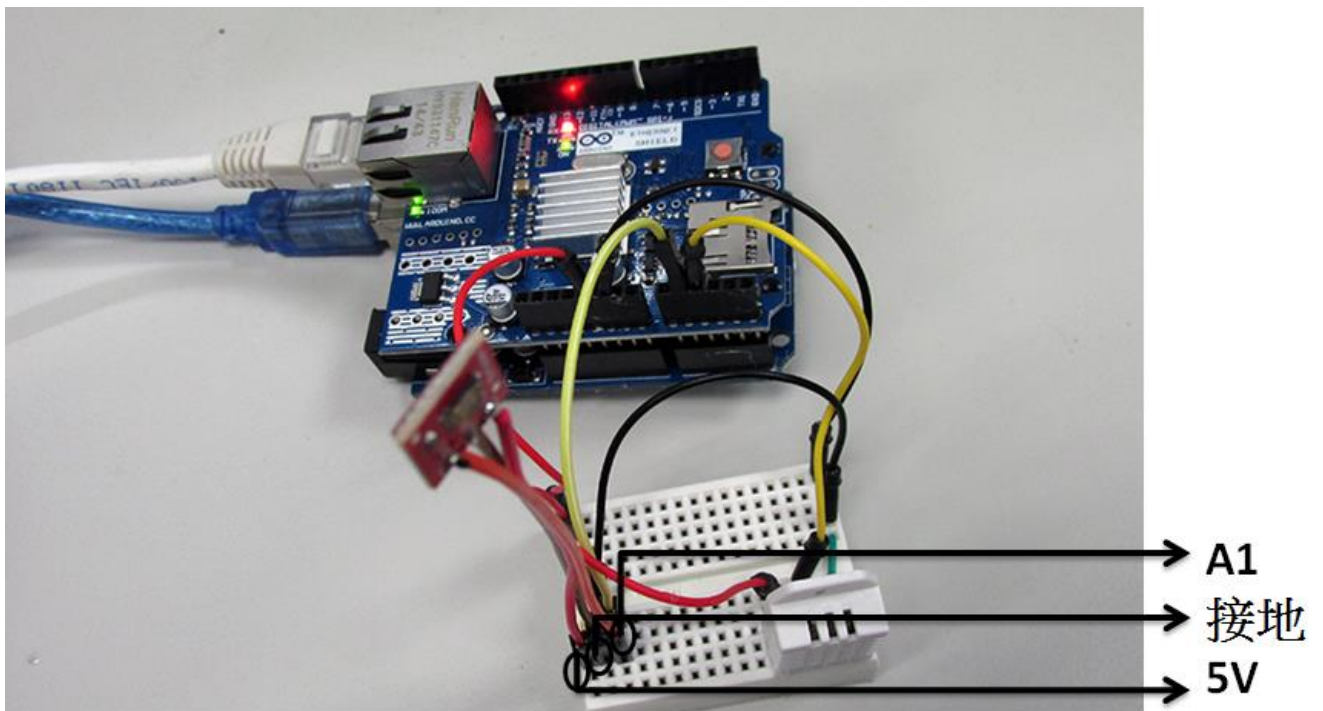


光照度感測

TEMT6000 其腳位定義如下：



由於我們使用麵包板，因此請將此模組焊上杜邦線接頭，再分別接到麵包板 5V, GND 及 Arduino A1。



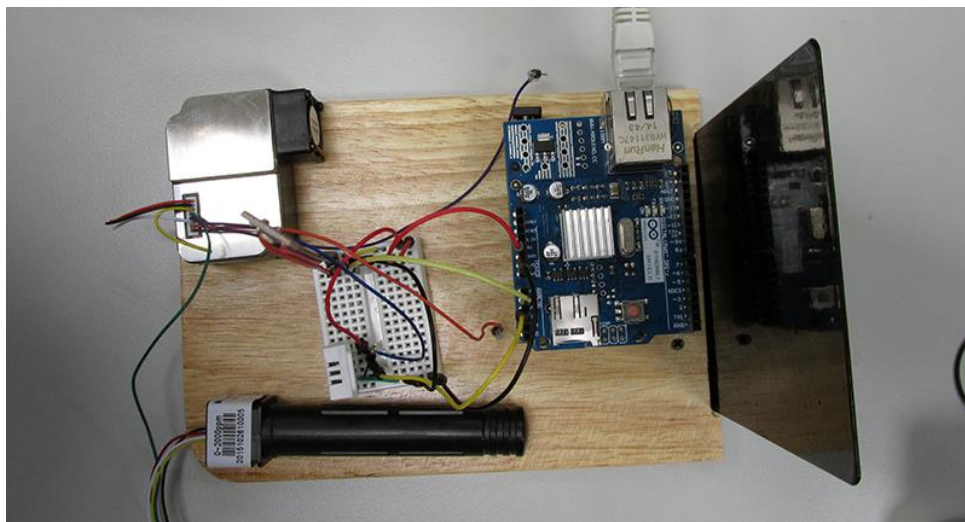
最後的固定工作

完成上方的組裝步驟後，接下來，我們將這些零件固定在 L 型書檔上，看起來會比較像是成品，簡單快速也耐看。不過您也可以自行設計外殼或其它固定的方式。

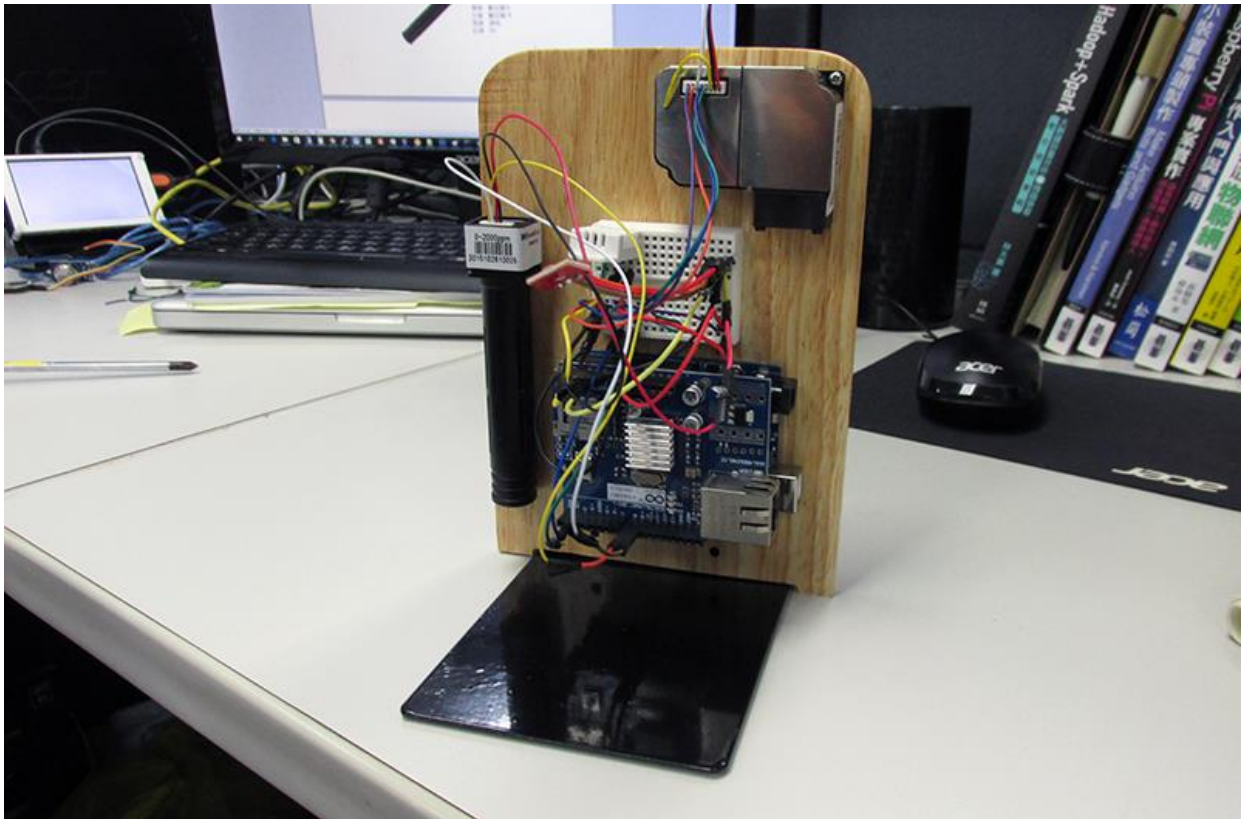
L 型書檔如果您使用的是木板材質，那麼可以直接用螺絲釘將所材料零件鎖上固定，如果使用的是鐵或塑膠材質，那麼可能需要使用熱熔膠來固定。



先將各零件要擺放位置先來預排看看。由於我們做的是一台可偵測溫溼度、PM2.5、二氧化碳、環境光照度的感測裝置，因此有總共四個零件（Arduino 主板+W5100、迷你麵包板、G3 PM2.5 感測器、CO2 感測器），把所有零件依照上述方式組裝好之後，預定擺放位置如下：



確定好位置之後就可以用螺絲將它們固定起來。好了，不用半小時，我們很快就能完成一台可偵測多種環境狀態的感測裝置。



完成了！

D) 居家專用感測裝置的安裝步驟

材料準備

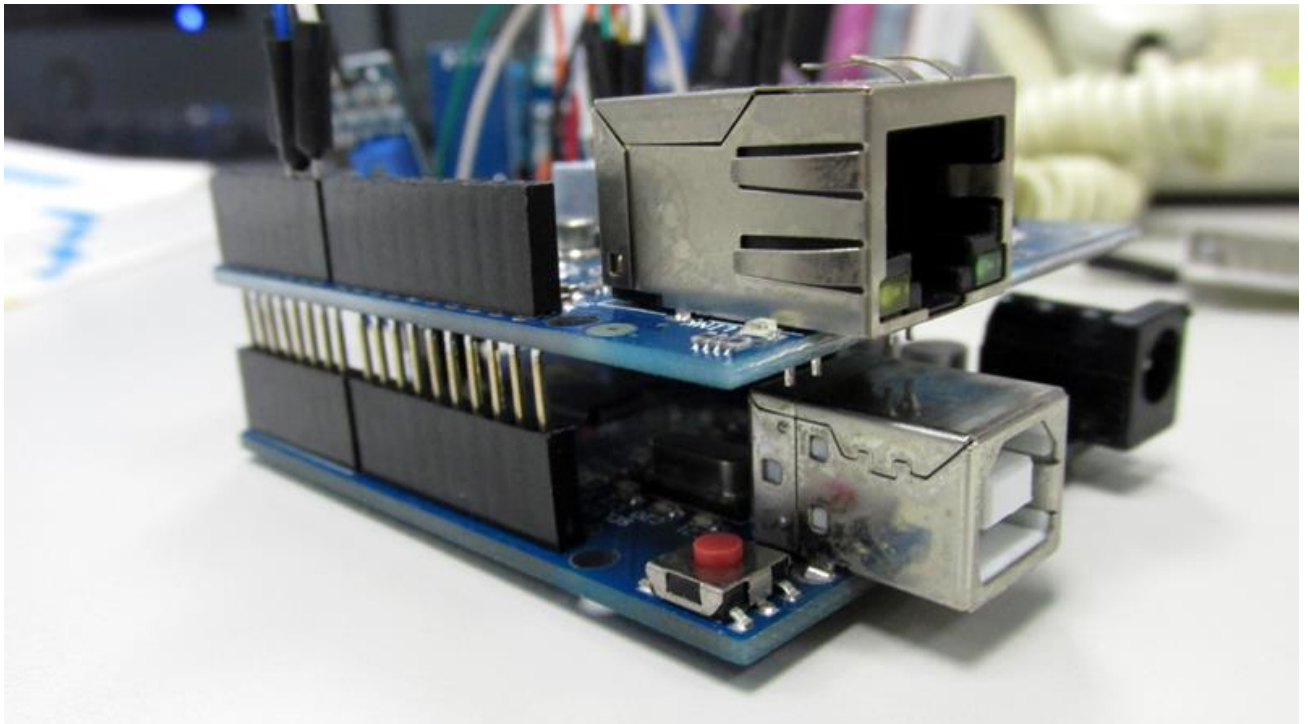
1. Arduino UNO 主板 x 1
http://www.icshop.com.tw/product_info.php/products_id/20369
2. W5100 R3 network shield x 1
<http://goods.ruten.com.tw/item/show?21434006463160>
3. DHT-22 sensor x 1 (溫溼度用)
http://www.icshop.com.tw/product_info.php/products_id/12478
4. MQ7 感測器 x 1 (偵測一氧化碳)
http://www.icshop.com.tw/product_info.php/products_id/12491
5. MQ4 感測器 x 1 (偵測瓦斯天然氣)
http://www.icshop.com.tw/product_info.php/products_id/12488

6. 9V 變壓器 x 1
(如果您同時裝上很多感應器，建議使用 9V 變壓器取代 USB，以避免電源不足)
http://www.icshop.com.tw/product_info.php/products_id/17470
7. 迷你麵包板 x 1
http://www.icshop.com.tw/product_info.php/products_id/11516
8. 杜邦線 數條
http://www.icshop.com.tw/product_info.php/products_id/12374
9. 小型 L 型立書夾 x 1 (\$39 元店有很多)

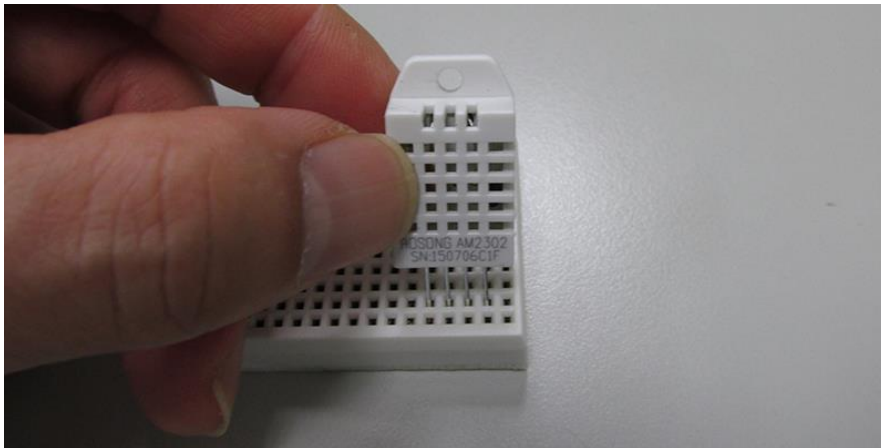
準備好了？開始組裝吧！

溫溼度感測

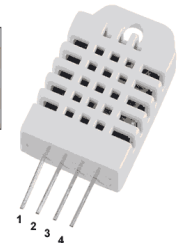
A) 首先，將材料#1 與材料#2 合體如下：



B) 把材料#3 插上，注意一下左圖它的腳位

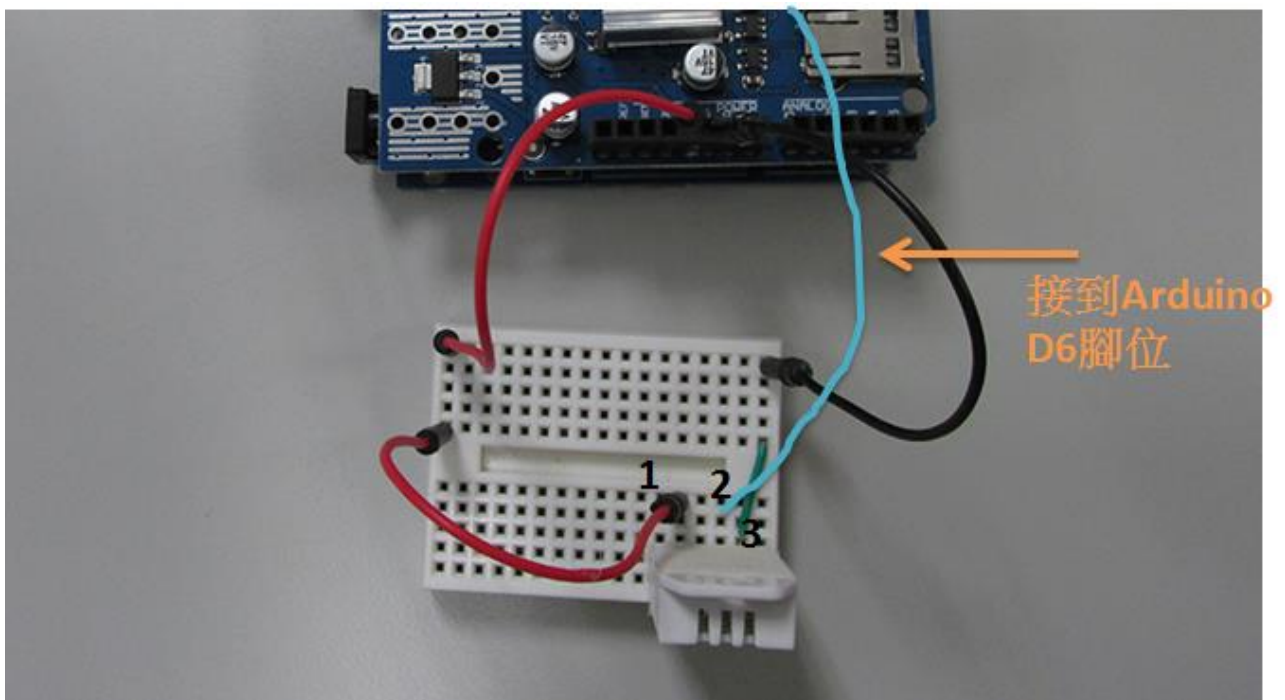


DHT22 pins	
1	VCC
2	DATA
3	NC
4	GND



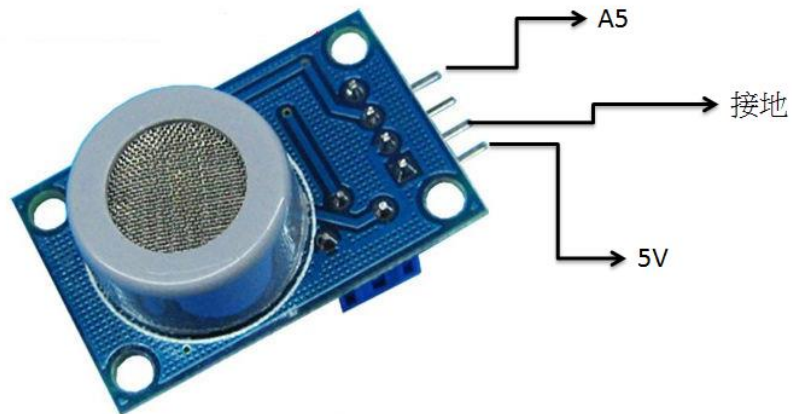
請照下圖方式來連接，Arduino 的 5V 和 GND 先接到麵包板上，再將材料#3 的第一根接腳接到 5V，第四根接到 GND，第三根則接到 Arduino D6。

如此，我們就完成了溫溼感測器的組裝。

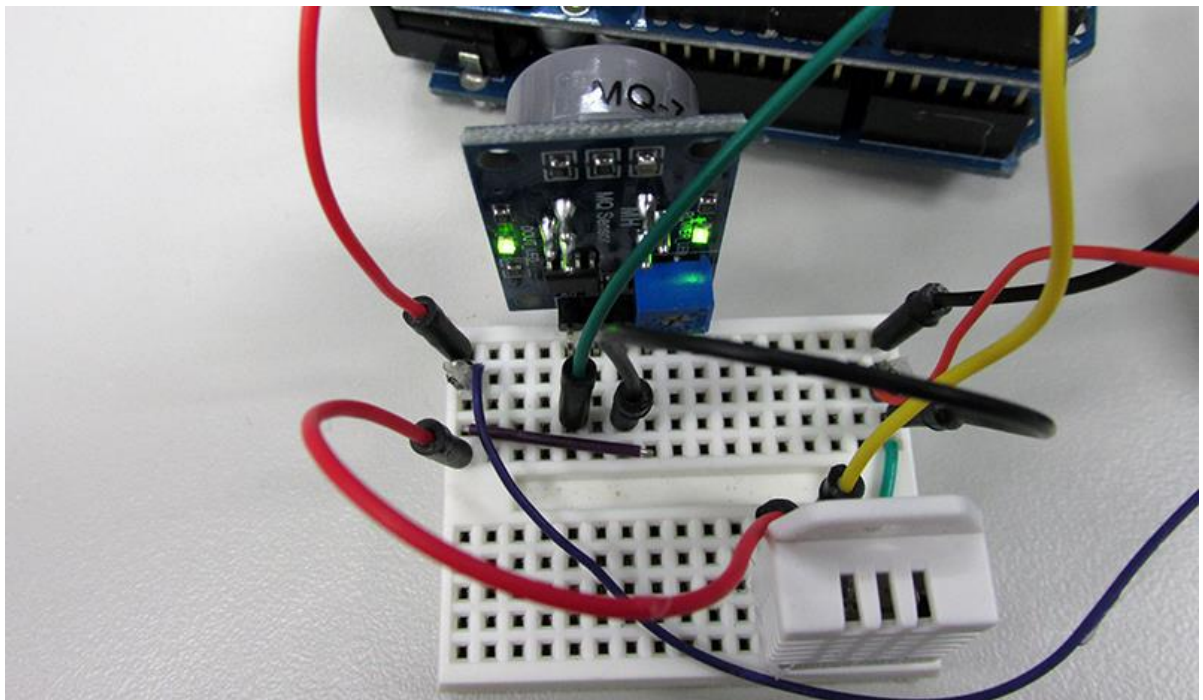


MQ7 一氧化碳感測

請參考下方 MQ7 的腳位。



在麵包板找個適當位置將 MQ7 插上，然後分別將各接腳連接到麵包板 5V, GND 及 Arduino A5 port。

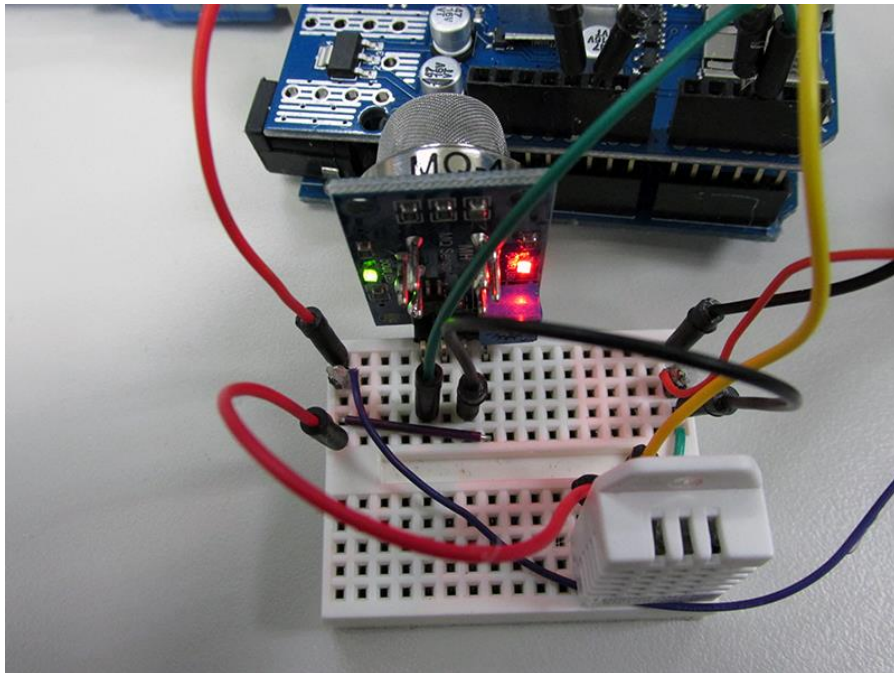


MQ4 天然氣外洩偵測

請參考下方 MQ4 的腳位。（您也可以將 MQ7 更換成其它 MQ 系列腳位相同的感測器，如 MQ2, 9 等等）



在麵包板找個適當位置將 MQ4 插上，然後分別將各接腳連接到麵包板 5V, GND 及 Arduino A4 port。



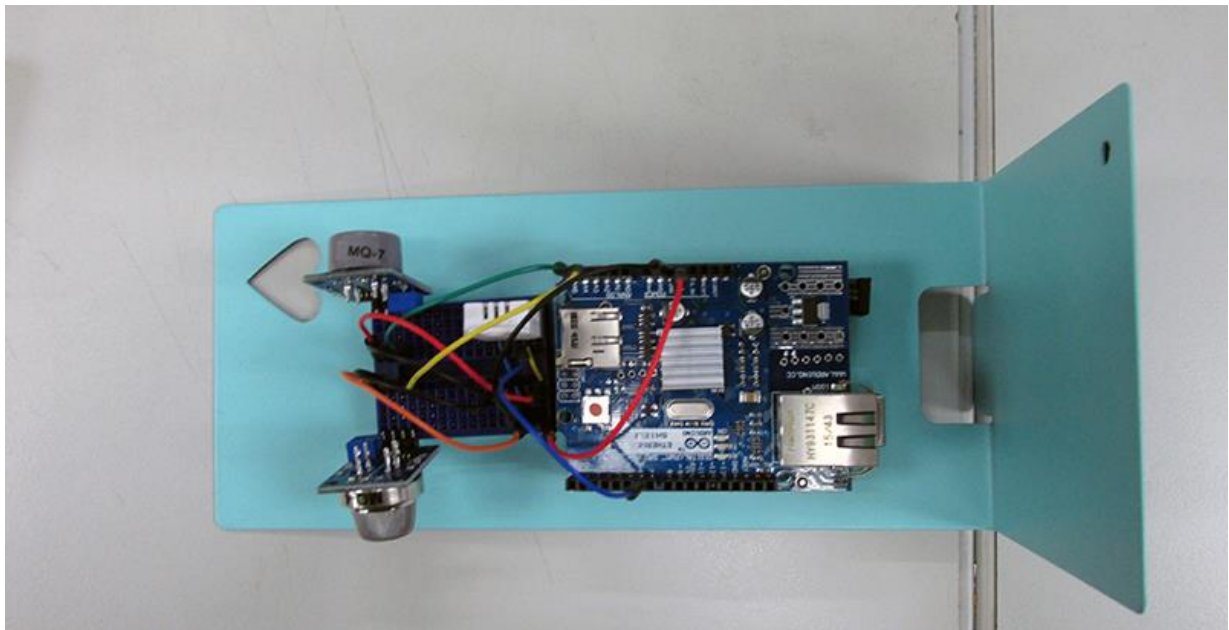
最後的固定工作

完成上方的組裝步驟後，接下來，我們將這些零件固定在 L 型書檔上，看起來會比較像是成品，簡單快速也耐看。不過您也可以自行設計外殼或其它固定的方式。

L 型書檔如果您使用的是木板材質，那麼可以直接用螺絲釘將所材料零件鎖上固定，如果使用的是鐵或塑膠材質，那麼可能需要使用熱熔膠來固定。

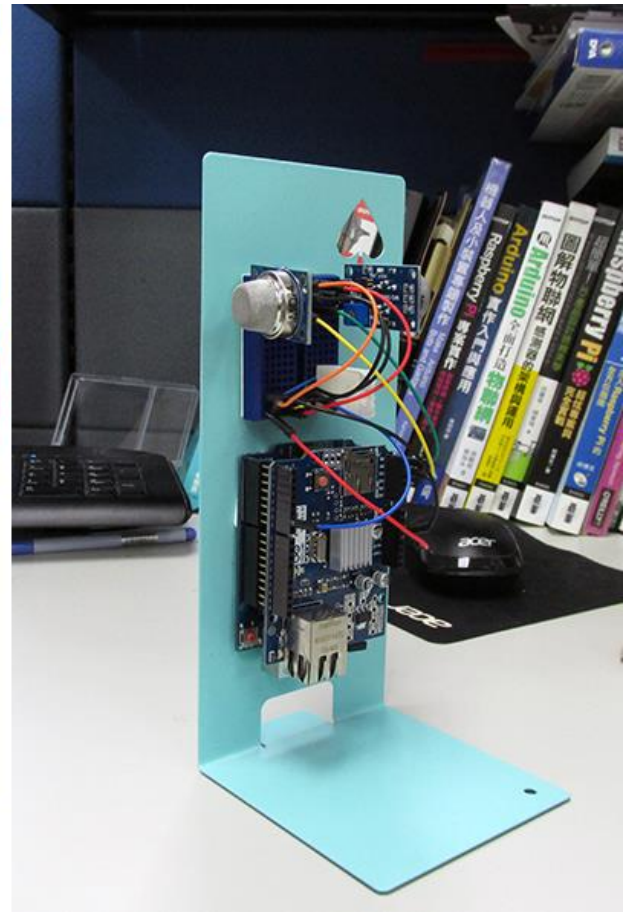


總共會有二個零件要放置並固定（Arduino 主板+W5100、迷你麵包板），先將零件放在適當位置上預排看看，確定沒問題後再用螺絲或熱融膠固定。



完成了！

好了，不用半小時，我們很快就完成一台可偵測溫溼度、一氧化碳及天然氣外洩的居家感測裝置。



E) 居家專用感測裝置的安裝步驟

材料準備

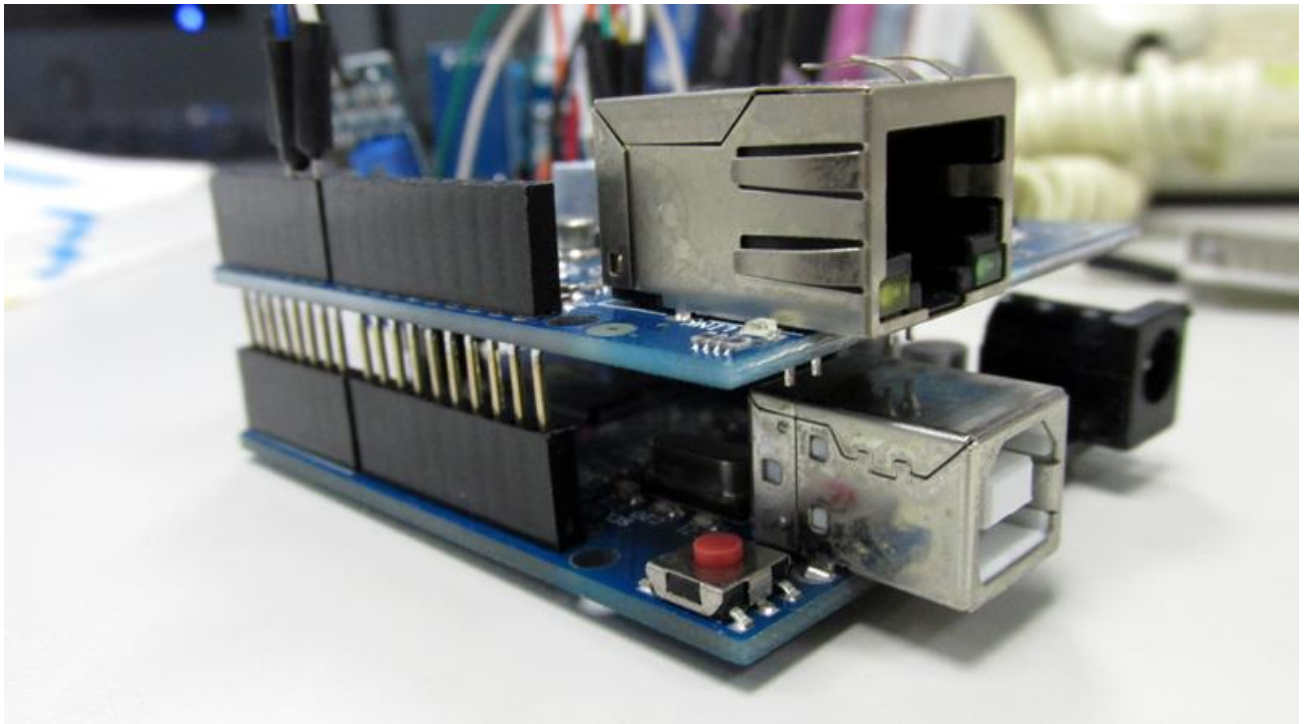
1. Arduino UNO 主板 x 1
http://www.icshop.com.tw/product_info.php/products_id/20369
2. W5100 R3 network shield x 1
<http://goods.ruten.com.tw/item/show?21434006463160>
3. DHT-22 sensor x 1 (溫溼度用)
http://www.icshop.com.tw/product_info.php/products_id/12478
4. FC-37 雨水感測器： x 1 (偵測高架地板漏水)
http://www.icshop.com.tw/product_info.php/products_id/12483
5. MQ2 感測器 x 1 (偵測易燃氣體及煙霧)
http://www.icshop.com.tw/product_info.php/products_id/14133

6. 9V 變壓器 x 1
(如果您同時裝上很多感應器，建議使用 9V 變壓器取代 USB，以避免電源不足)
http://www.icshop.com.tw/product_info.php/products_id/17470
7. 迷你麵包板 x 1
http://www.icshop.com.tw/product_info.php/products_id/11516
8. 杜邦線 數條
http://www.icshop.com.tw/product_info.php/products_id/12374
9. 小型 L 型立書夾 x 1 (\$39 元店有很多)

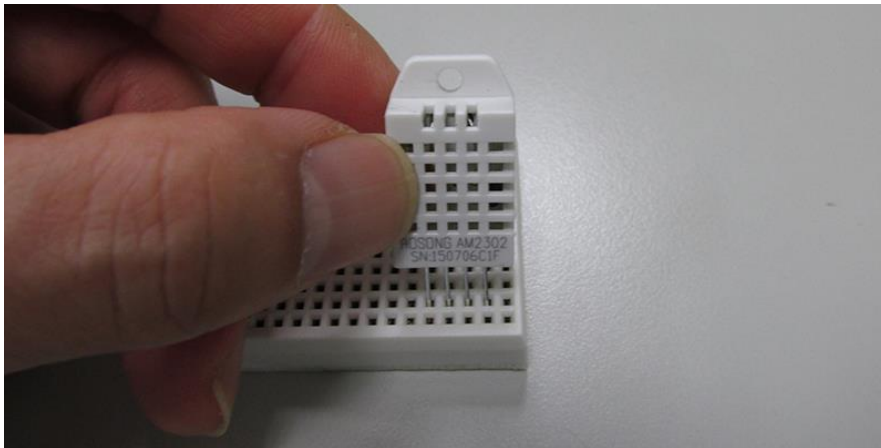
準備好了？開始組裝吧！

溫溼度感測

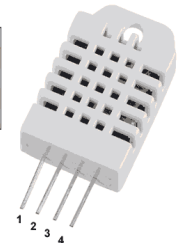
A) 首先，將材料#1 與材料#2 合體如下：



B) 把材料#3 插上，注意一下左圖它的腳位

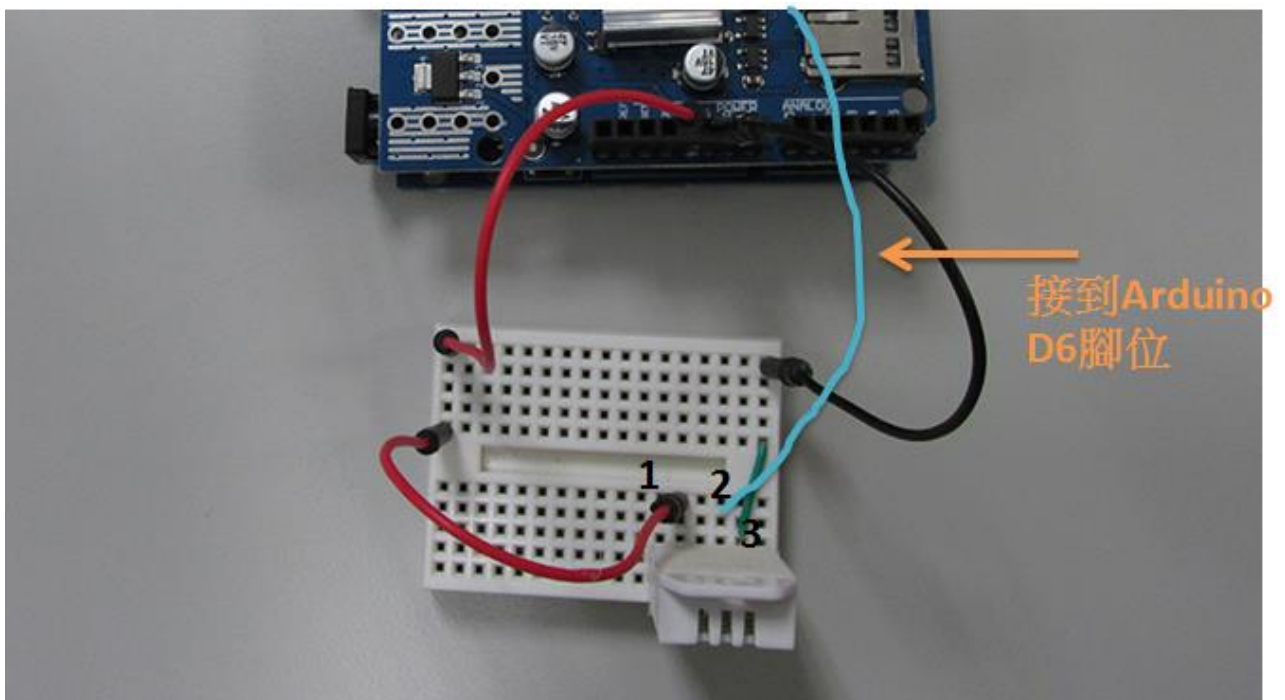


DHT22 pins	
1	VCC
2	DATA
3	NC
4	GND



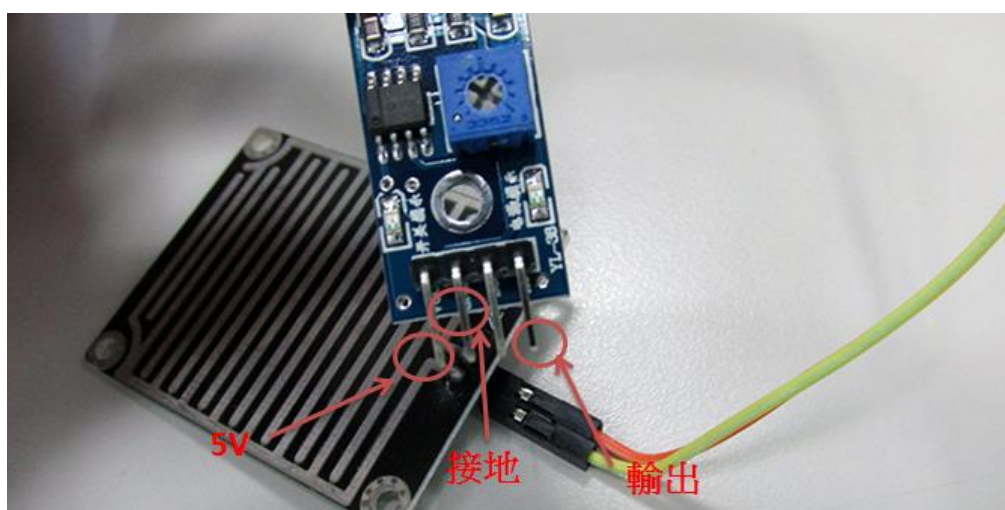
請照下圖方式來連接，Arduino 的 5V 和 GND 先接到麵包板上，再將材料#3 的第一根接腳接到 5V，第四根接到 GND，第三根則接到 Arduino D6。

如此，我們就完成了溫溼感測器的組裝。

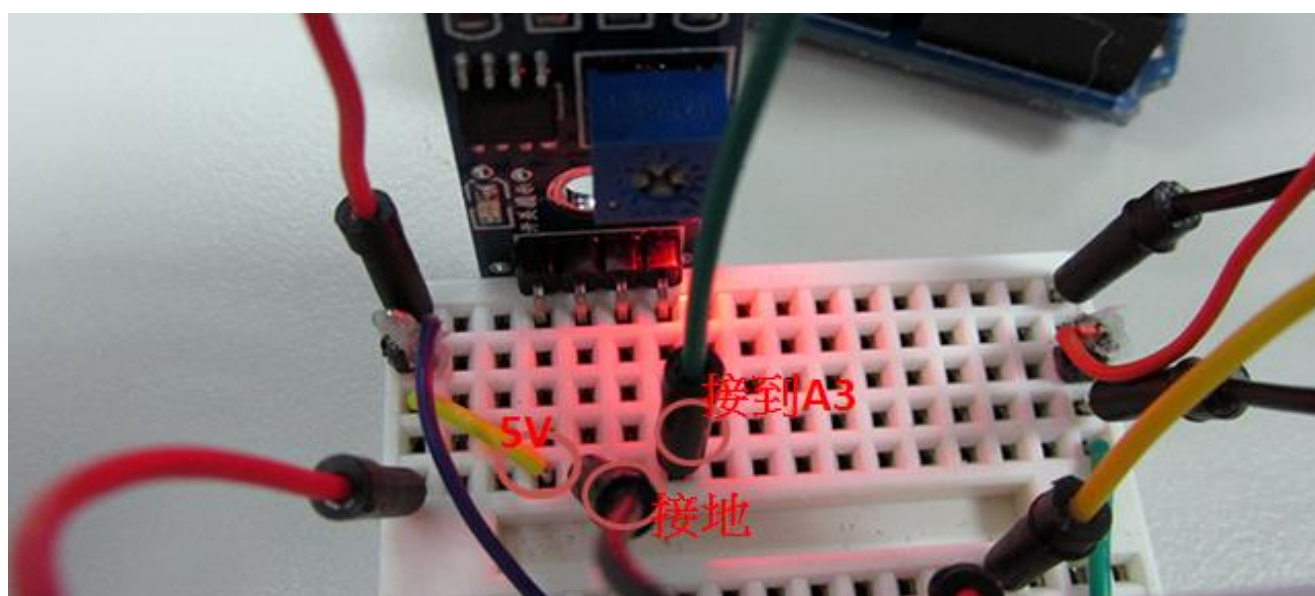


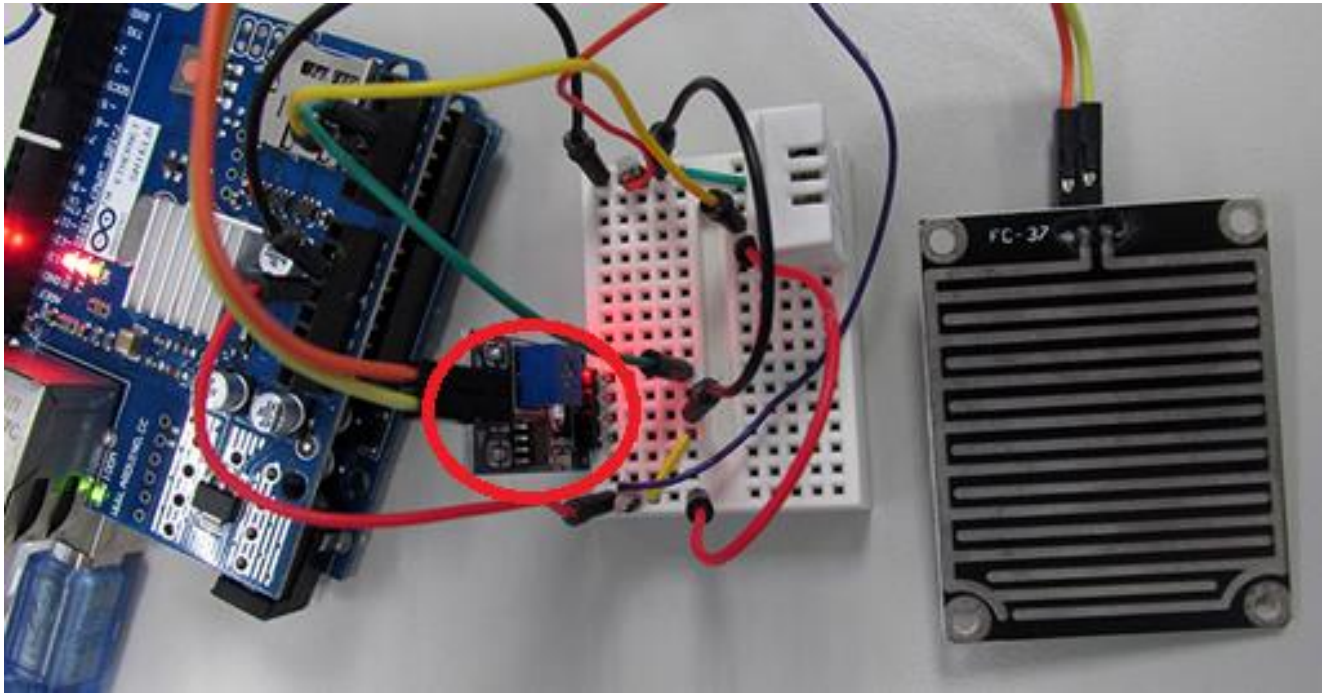
FC-37 漏水感測

請參考下方 FC-37 的腳位定義。



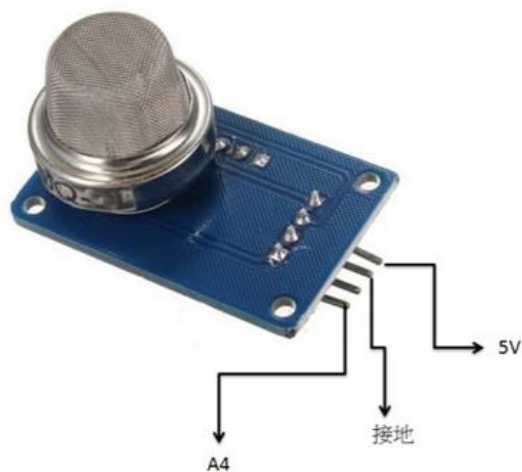
在麵包板找個適當位置將 FC-37 插上，然後分別將各接腳連接到麵包板 5V, GND 及 Arduino A3 port。



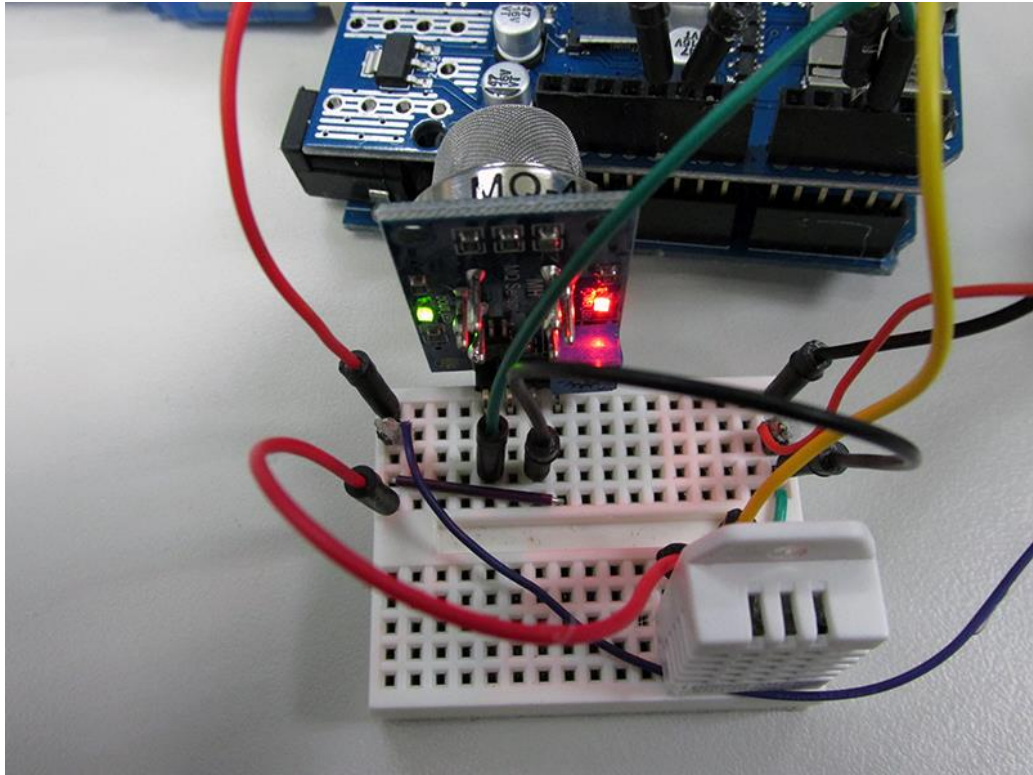


MQ2 可燃氣體及煙霧偵測

請參考下方 MQ2 的腳位。（您也可以將 MQ2 更換成其它 MQ 系列腳位相同的感測器，如 MQ4,7,9 等等）



在麵包板找個適當位置將 MQ2 插上，然後分別將各接腳連接到麵包板 5V, GND 及 Arduino A4 port。



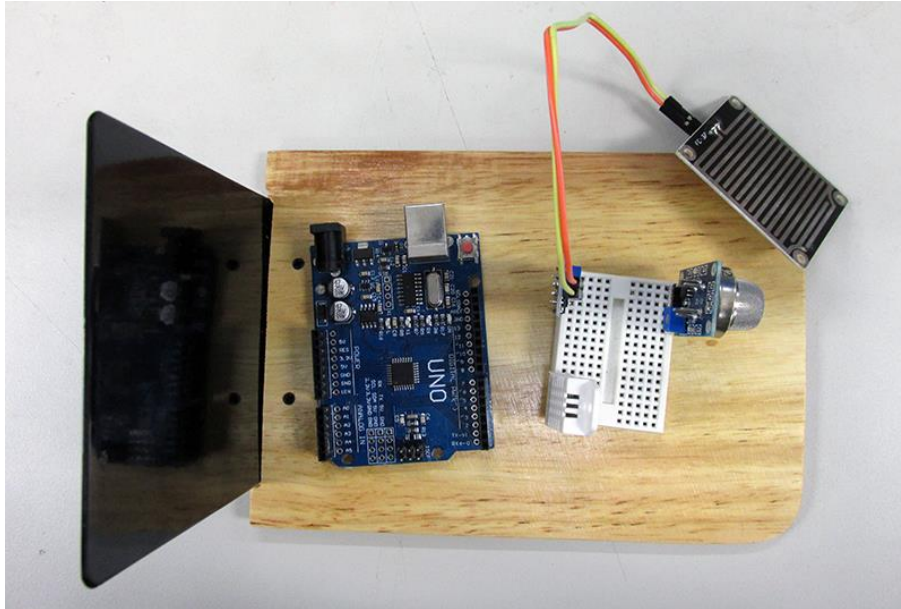
最後的固定工作

完成上方的組裝步驟後，接下來，我們將這些零件固定在 L 型書檔上，看起來會比較像是成品，簡單快速也耐看。不過您也可以自行設計外殼或其它固定的方式。

L 型書檔如果您使用的是木板材質，那麼可以直接用螺絲釘將所材料零件鎖上固定，如果使用的是鐵或塑膠材質，那麼可能需要使用熱熔膠來固定。

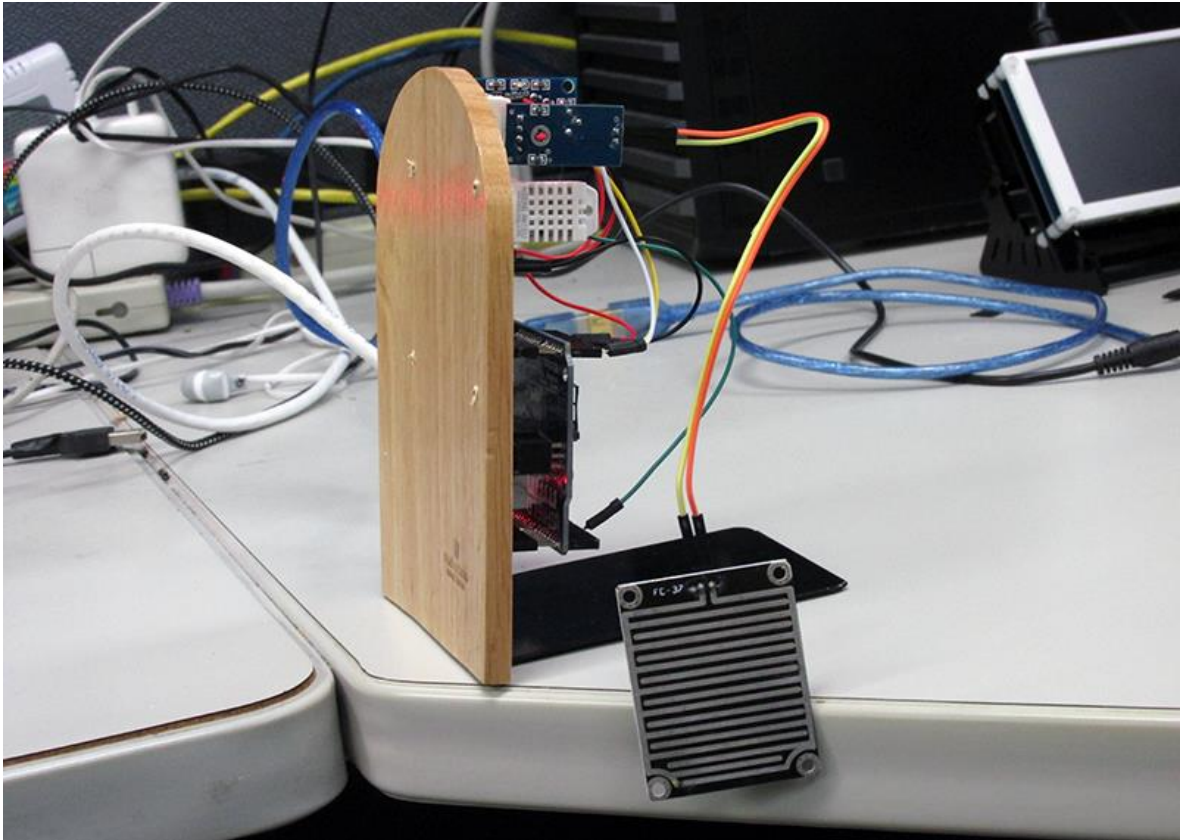


總共會有三個零件要放置並固定（Arduino 主板+W5100、迷你麵包板、FC-37 的外接雨水感測板），先將零件放在適當位置上預排看看，確定沒問題後再用螺絲或熱融膠固定。



完成了！

好了，不用半小時，我們很快就能完成一台可偵測溫溼度、漏水及火災的機房感測裝置。



F) 替 IASS 加上 LCD 螢幕

如果您想要在 IASS 感測器上立即看到目前的感測值，那麼就需要替它加上 LCD 螢幕。在 IASS 感測器加上螢幕是很容易的事，只要將 LCD 用杜邦線接到板子各對應的孔，並將 LCD_DISPLAY 變數值更改為 1，上傳到 Arduino 就可以了。

IASS 預設使用內建 I2C 模組的 16x2(1602)的 LCD，如下圖右側，而左側為編號 2004（20x4）的 LCD，如果您使用不同尺寸的 LCD，需自行修改程式。



準備材料：

1. LCD 1602 5V LCM IIC I2C 16x2 \$95 x 1

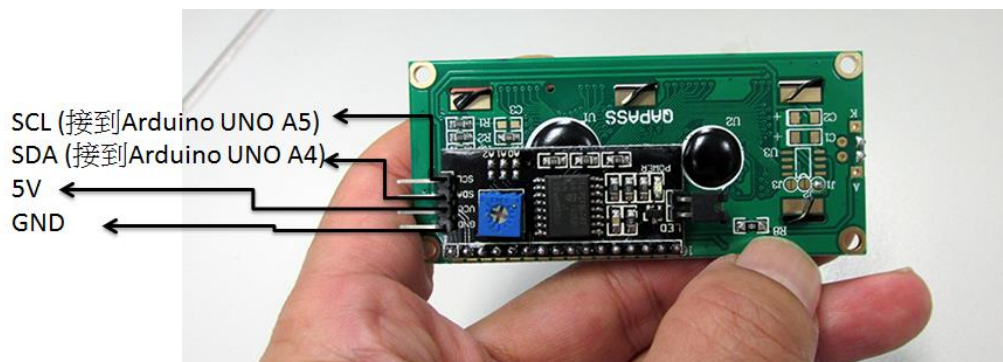
<http://goods.ruten.com.tw/item/show?21408019307836>

2. 杜邦線（母↔公） 數條

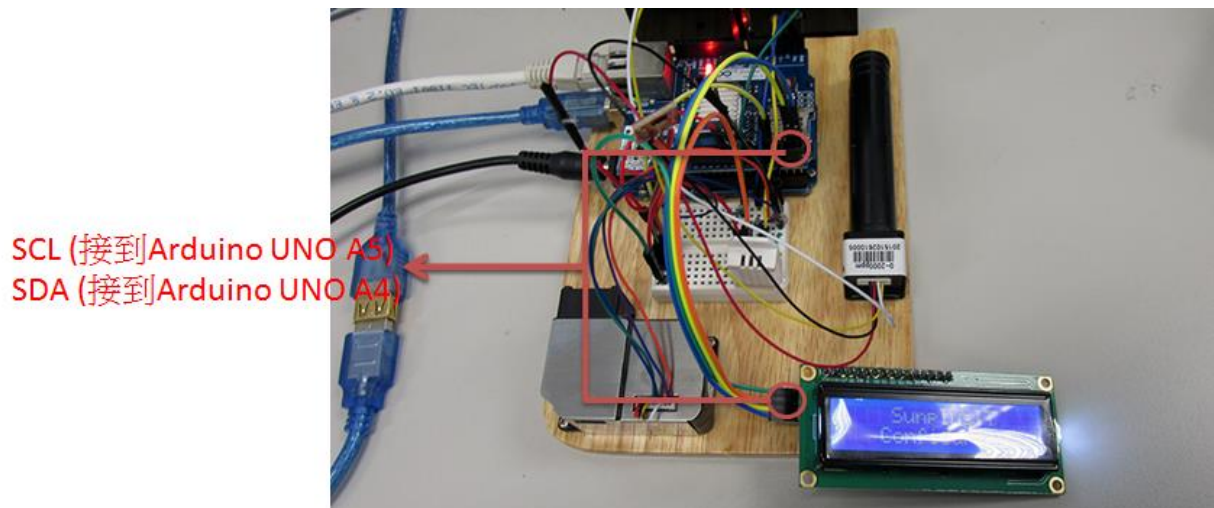
開始組裝吧！

接上 LCD

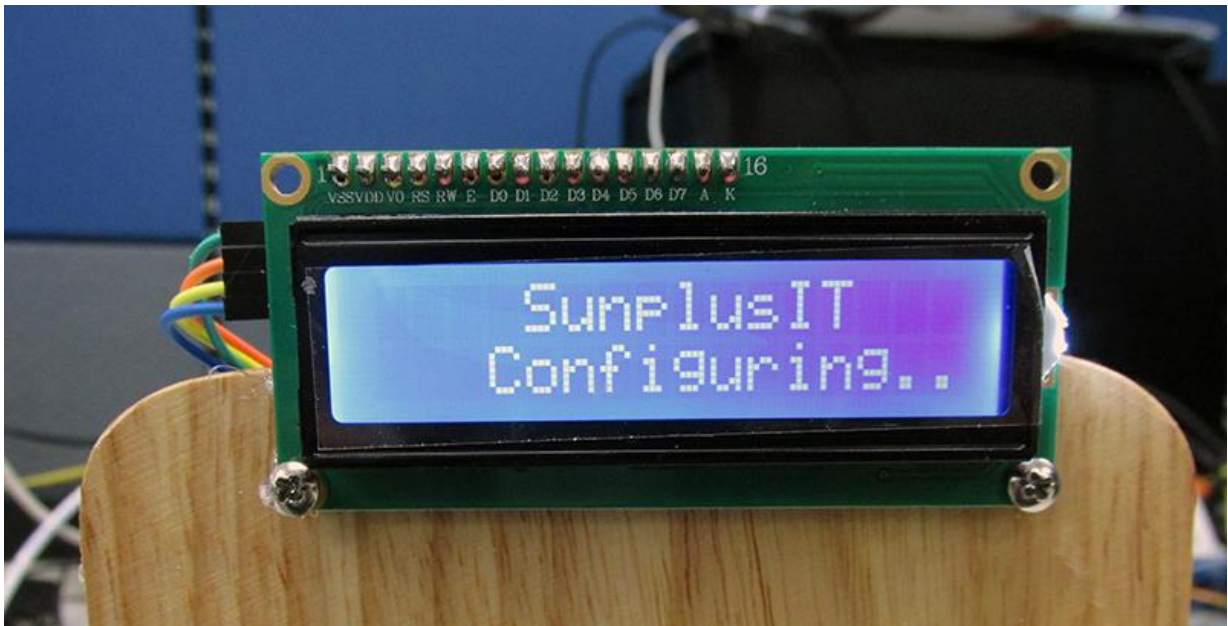
此類的 I2C LCD 模組皆有四個接腳如下：



請參考下圖使用杜邦線將各接腳接到麵包板及 Arduino 上。



將 LCD 固定好



程式修改

我們只要將 Arduino code 中的 LCD_DISPLAY 變數更改為 1 就可以，改好後上傳寫入 Arduino，完成後您便會看到 LCD 開始顯示訊息。

```
#include <SPI.h>
#include <Ethernet.h>

//===== Configuration =====

//Sensor代碼定義： 1->溫度, 2->濕度, 3->漏水, 4->煙霧可燃氣體 5->CO, 6->天然氣, 7->PM1,8->PM2.5, 9->PM10, 10->CO2, 11->環境光亮度
const unsigned sensorsList[] = {1,2,5,6}; //要啟用那些感應器? 建議上限最多5個.
const unsigned nums = 4; //啟用的感應器數量, 最多5個

#define LCD_DISPLAY 1 //是否要啟用LCD顯示?

int unsigned countsAVG = 6; //要取幾次的sensor值, 去除最大與最小值後, 作最終平均? 最少需3次

const long updateInterval = 2 * 60000; //單位ms
const String sensorID = "HR"; //給這個感測裝置取個英文ID, 注意此ID不要與其它裝置重複.
```

完成了！

接上測試，您會看到目前的感測值顯示在 LCD 上，並由右往左循序的移動顯示。

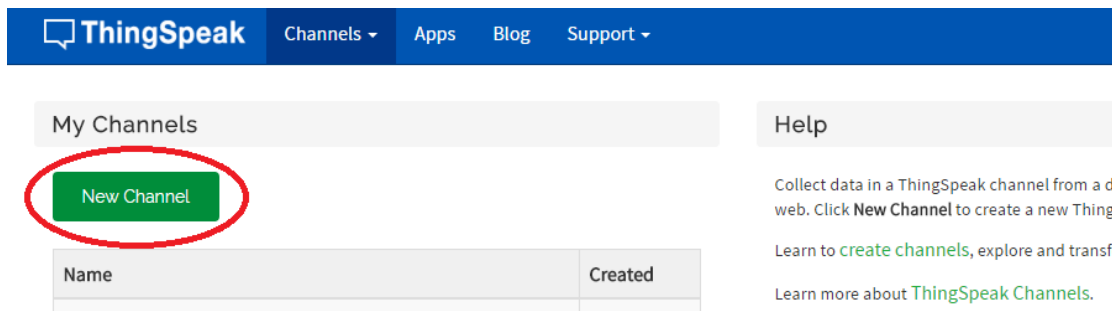


G) 申請及設定 THINKSPEAK

要馬上看到感測裝置所搜集的資訊，以及各時段的記錄和變化曲線，除了自己架一台 **server** 來處理之外，最快也最方便的方式就是使用 **IoT** 平台了，有些 **IoT** 平台是免費的，而且可以替你繪出詳細的統計曲線，例如 **FreeBoard**、**ThingSpeak** 等等。

目前 **IASS** 系統支援 **ThingSpeak**，在您註冊 **ThingSpeak** 帳號之後，便可將專屬的 **API KEY** 放在 **Arduino** 程式中，系統便會搜集到的資料直接上傳 **ThingSpeak** 繪製如下圖：

註冊帳號並登入之後，請先在 **My Channels** 當中建立一個新 **Channel**，這個 **Channel** 您可以將它視為剛剛所組裝好的感測設備資料接收顯示端。



請依下圖所示來設定此 Channel。

The screenshot shows the 'New Channel' form with several orange annotations explaining the fields:

- Name:** 給這個感測裝置取個名稱 (Give this sensor device a name).
- Description:** 一些非必要的補充說明 (Some optional supplementary information).
- Fields:** 因為這個感測裝置我們想要偵測溫度, 溼度, PM2.5, PM10, PM100, CO2, 光照度等共有七種, 因此會產出七組感測數值, 請將field1~field7勾選起來。 (Because this sensor device we want to detect temperature, humidity, PM2.5, PM10, PM100, CO2, light intensity etc. there are seven types, so it will produce seven groups of sensor values, please check field1~field7).
- Save Channel:** 其它欄位非必要, 可視需求來輸入, 按此按鈕即可產生此Channel (Other fields are optional, input as needed, click this button to create this channel).

按下 Save Channel 按鈕後，會產生如下的畫面，這是此 Channel 的主頁面。

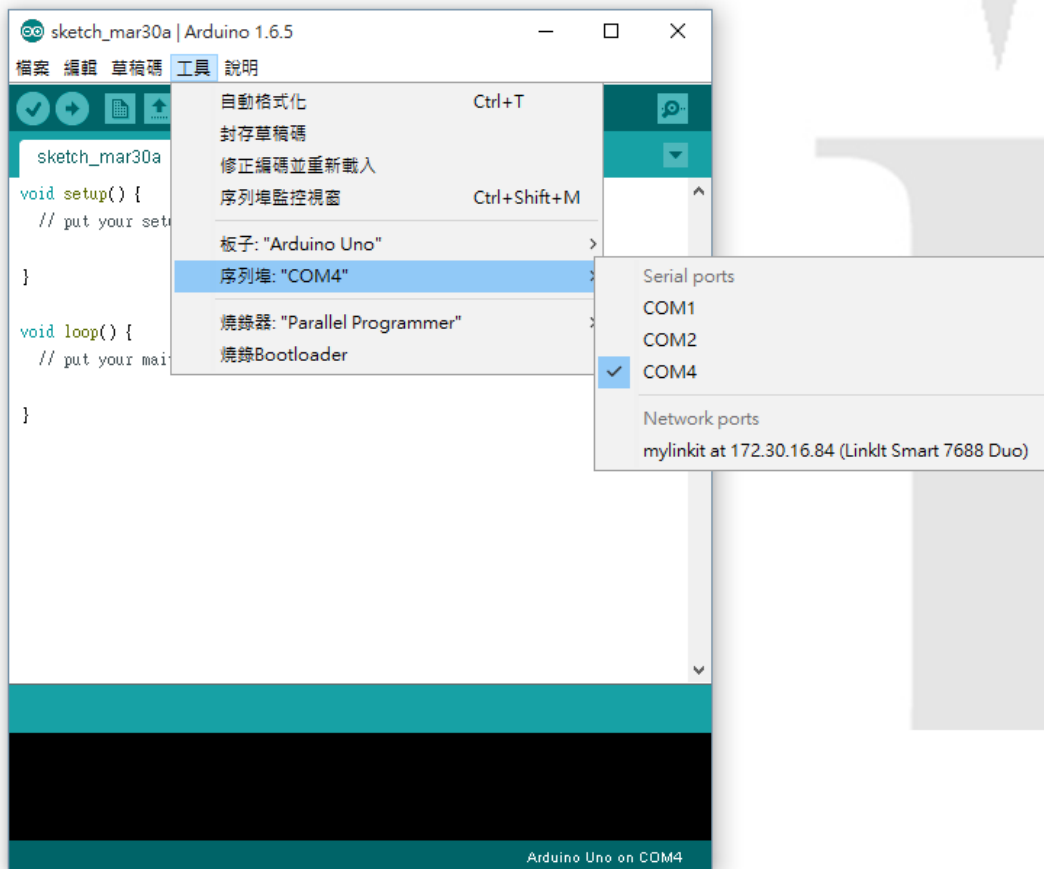
The screenshot shows the ThingSpeak channel page for '凌陽創新辦公室環控裝置'. The channel ID is 104120, the author is tony5897, and the access is Private. The location is noted as '安裝於新竹科學園區力行一路'. Navigation tabs include Private View, Public View, Channel Settings, API Keys, and Data Import / Export. Action buttons for 'Add Visualizations', 'Data Export', 'MATLAB Analysis', 'MATLAB Visualization', and 'More Apps' are present. Channel stats show it was created and updated 'less than a minute ago' with 0 entries.

請按一下 API Keys，然後記下 Write API Key 的值，稍後我們的感測裝置需要此 Key 來將資料上傳到此 Channel。

This screenshot shows the 'API Keys' section of the ThingSpeak channel. The 'API Keys' tab is selected and circled in red. Under 'Write API Key', a key 'XK76Q4EWPK5WY4K' is displayed in a text box and circled in red. Below it is a 'Generate New Write API Key' button. The 'Read API Keys' section is empty. A 'Help' section explains that API keys are used for writing and reading data. 'API Keys Settings' includes instructions for Write API Key, Read API Keys, and a Note about tracking users.

H) 把 ARDUINO 程式燒錄到裝置

接下來，請在 PC 上安裝好 Arduino IDE，並將感測裝置接上，確定 Arduino IDE 的設定及序列埠都正確。



到 https://github.com/LinkItONEDevGroup/LASS/tree/master/IASS/IAAS_CODES/devices 目錄下，進入 Office 目錄，將 office.ino 程式內容 Copy 下來 Paste 到 Arduino IDE，請視需要修改下方的參數（紅色部份是必要的修改）：

//Sensor 代碼定義： 1->溫度, 2->溼度, 3->漏水, 5->CO, 6->天然氣, 7->PM2.5, 8->PM10, 9->PM100, 10->CO2, 11->環境光照度

const unsigned sensorsList[] = {1,2,7,8,9,10,11}; //要啟用那些感應器?

const unsigned nums = 7; //啟用的感應器數量

const String sensorID = "MR"; //給這個感測裝置取個英文 ID，注意此 ID 不要與其它裝置重複.

const String deviceName = "電腦機房"; //說明此感測裝置用途或地點.

int unsigned countsAVG = 6; //要取幾次的 sensor 值，去除最大與最小值後，作最終平均? 最少需 3

次

const boolean uploadThingsSpeak = 1; //要上傳 ThingsSpeak? 是:1, 否:0

const String writeAPIKey = "{你的 ThinkSpeak APIKEY}";

const long updateThingSpeakInterval = 5 * 60000; //單位 ms

const boolean uploadDataServer = 0; //要上傳指定的 server? 是:1, 否:0

IPAddress DataServerIP(192,168,3,2); //將資料以 GET 方式傳到特定 server IP，若不需要上傳則用 ""

const short DataServerPort = 80; //將資料以 GET 方式傳到特定 server 的 port 埠號

const String DataServerPage = "/iaas/update.php"; //將資料以 GET 方式傳到特定 server 的接收網頁路

徑

```
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; //請自行設定一組 MAC Address，可隨意指定，但  
請勿跟網路上已有的網卡相同避免衝突
```

```
const boolean networkDHCP = 1; //使用 DHCP: 1, 使用固定 IP: 0
```

```
//---> 固定 IP,如果使用固定 IP 資訊請填入下方資訊
```

```
byte ip[] = { 10,0,0,35 }; //如果沒有成功得到 DHCP 的位址, 則會使用這個 IP.
```

```
byte myDns[] = { 8,8,8,8 }; //如果沒有成功得到來自 DHCP 的 DNS, 則會使用這個 DNS IP.
```

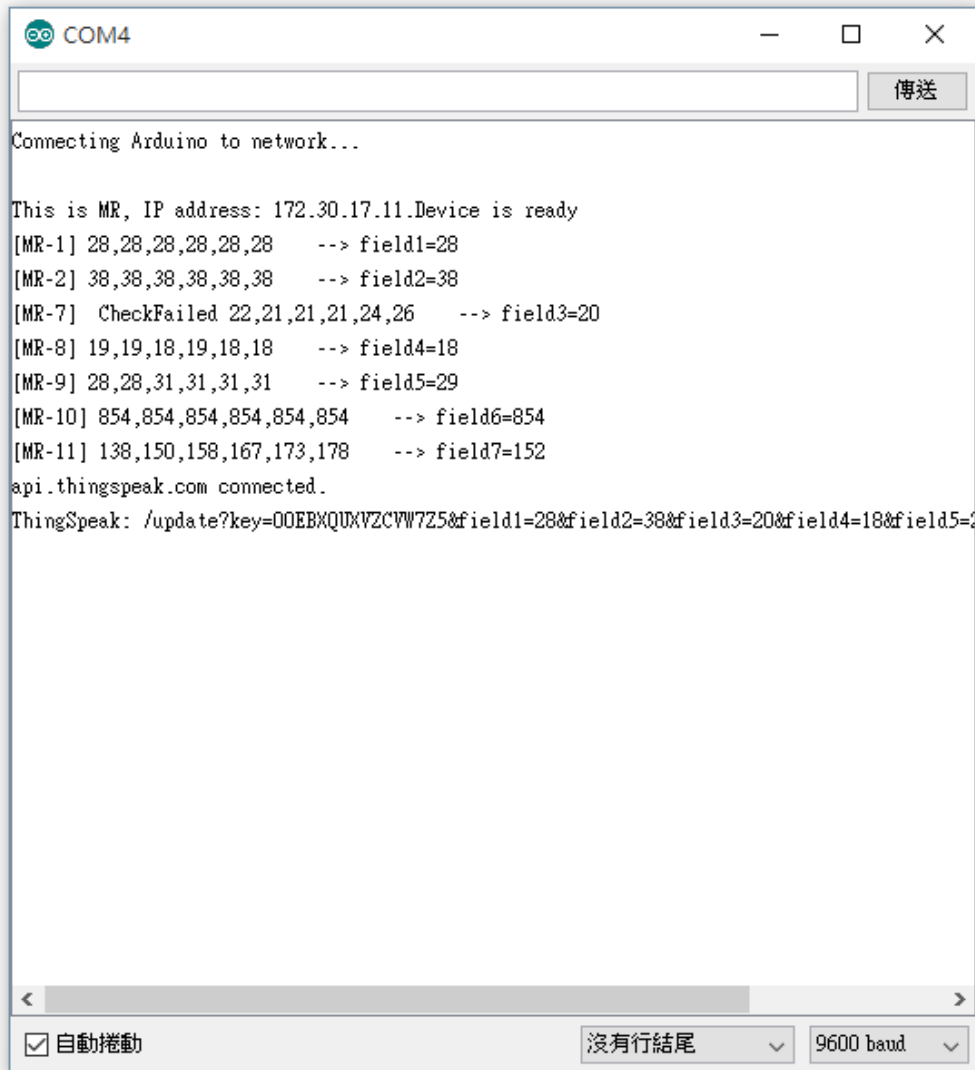
```
byte gateway[] = { 10,0,0,254 };
```

```
byte subnet[] = { 255, 255, 255, 0 };
```

修改後，便可直接燒錄至 Arduino（燒錄前請先把 G3 PM2.5 感測器的綠線不要插上，以避免 RX port 被佔用無法燒錄，燒錄成功後再插回）。

I) 設定 THINGSPEAK FIELD

在 Arduino 的序列埠監控視窗上，您會看到如下的訊息。

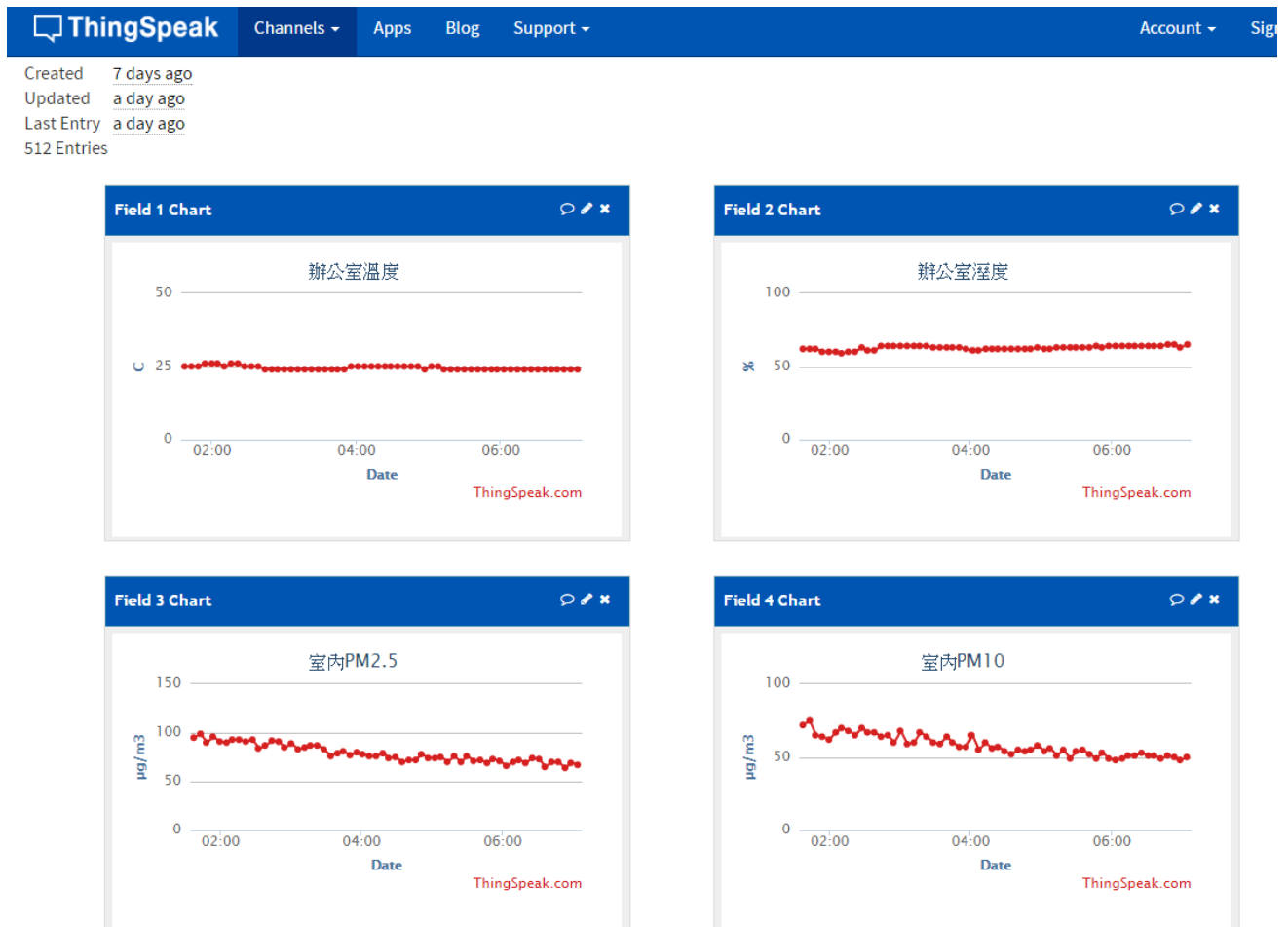


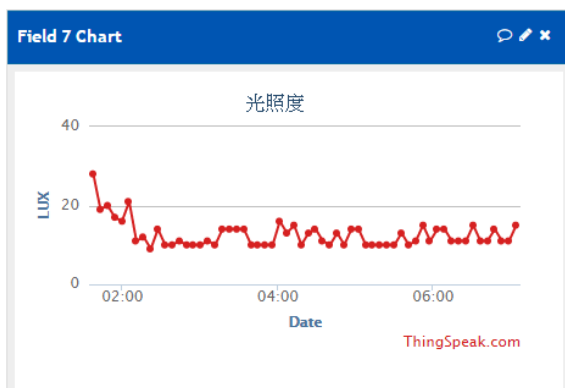
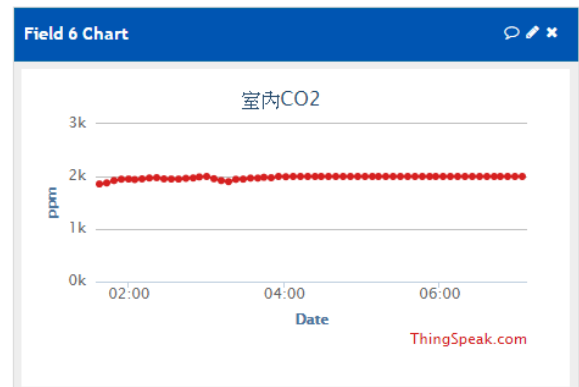
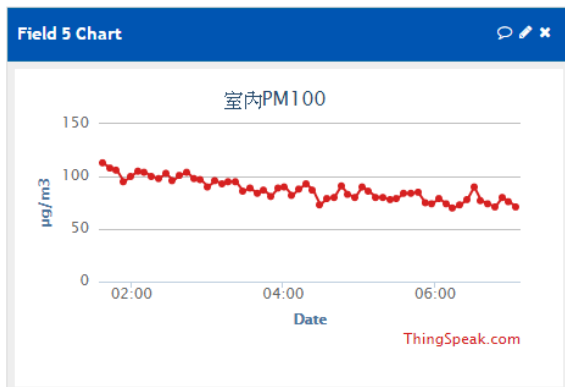
說明：

- | | | |
|--------------------------------------|----------------|-----------------------------------|
| [MR-1] 28,28,28,28,28,28 | --> field1=28 | → 溫度會丟到 ThingSpeak 的 Field1 |
| [MR-2] 38,38,38,38,38,38 | --> field2=38 | → 溼度會丟到 ThingSpeak 的 Field2 |
| [MR-7] CheckFailed 22,21,21,21,24,26 | --> field3=20 | → PM2.5 會丟到 ThingSpeak 的 Field3 |
| [MR-8] 19,19,18,19,18,18 | --> field4=18 | → PM10 會丟到 ThingSpeak 的 Field4 |
| [MR-9] 28,28,31,31,31,31 | --> field5=29 | → PM100 溼度會丟到 ThingSpeak 的 Field5 |
| [MR-10] 854,854,854,854,854,854 | --> field6=854 | → CO2 會丟到 ThingSpeak 的 Field6 |
| [MR-11] 138,150,158,167,173,178 | --> field7=152 | → 光照度會丟到 ThingSpeak 的 Field7 |

所以，我們可以準備回到 ThingSpeak 的 Channel 設定頁面，將各 Field 的內容補充進去。

再次進入 ThingSpeak，點選剛剛所建立的 Channel，您應該會發現各個 Field 都有各自獨立的統計圖，且已經開始有資料進來了，您可以點選各 Field 圖示的修改按鈕，進行細部的修正，如 Title、Y 軸文字、上下限等等。例如下圖：





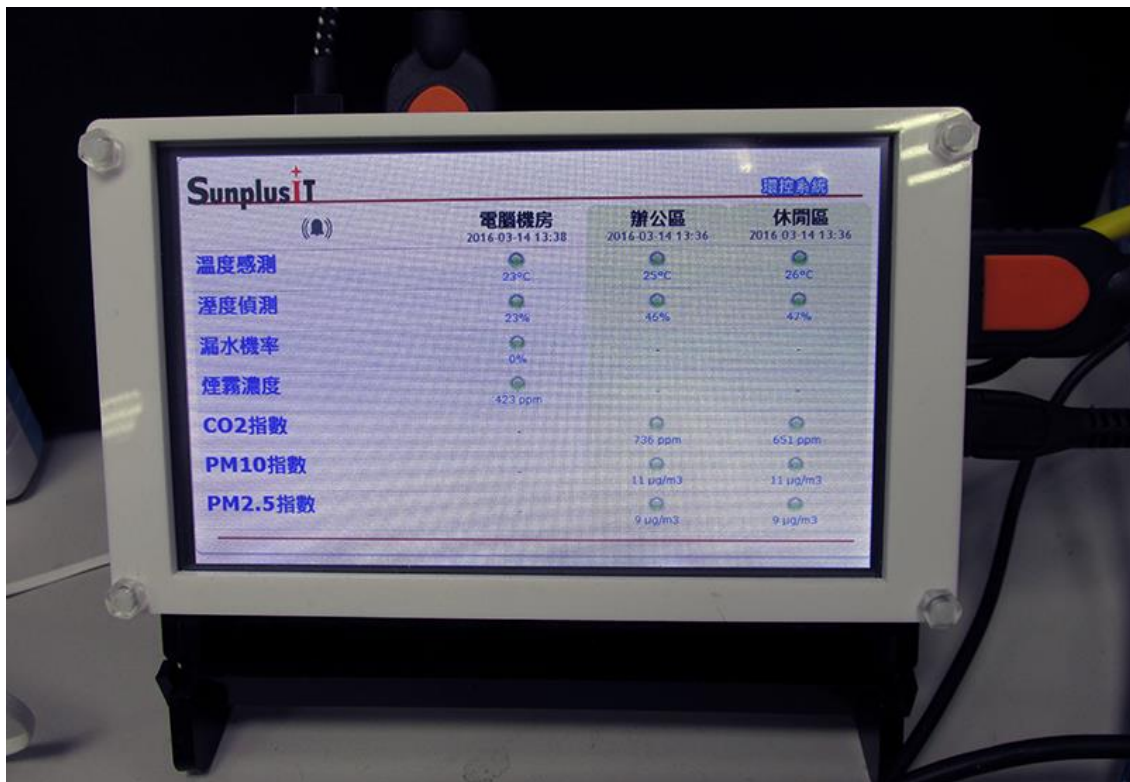
J) 自行架設 IAAS 主機

IAAS 感測裝置除了可上傳資訊到 thingspeak.com 之外，也可傳送到指定的 IP，讓我們可將這些資訊自行接收儲存下來，並自行開發程式來處理及運用這些資訊。IAAS 本身也提供了一套基本的 IAAS 主機程式，提供使用者可快速的安裝使用。該範例為使用 PHP+MySQL 所編寫，安裝上相當容易，步驟如下：

- 準備一個可執行 PHP + MySQL 的 web server，下載並解壓 IAASweb.zip。
- 建立資料庫：下載 `iaas_db.sql`，並透過 phpMyAdmin 匯入 MySQL。
- 透過 phpMyAdmin 新增一 MySQL account，將此帳號資訊填入 `pdo.php`。
- 將各感應 device 的 ID 及 sensors 代號列表填入 `pdo.php`。
- 將整個 IAASweb 目錄上傳 web server。
- 將 web 主機相關資訊更新到各感應 device。
- 後續修改及調整。

準備 WEB SERVER，

您可以使用一般的 PC，也可以用樹莓派 RPI 或香蕉派 BPI 搭配一台小型 LCD 螢幕來架設。



主機的程式請到 IAAS_Server\html 目錄下載。

建立資料庫

1. 下載 IAAS_Server 資料夾下的 iaas_db.sql，使用 phpMyAdmin 匯入執行，完成後會建立一個 sensor_data 資料庫，裏面有四個 tables。

phpMyAdmin

最近使用 我的最愛

- 新增
- information_schema
- mysql
- performance_schema
- phpmyadmin
- sensor_data**
 - Procedure
 - 資料表
 - 新增
 - data_storage
 - data_storage2
 - sensor_list
 - sensor_type

伺服器: localhost

資料庫 SQL 狀態 使用者 匯出 匯入 設定 備援 變數 字元編碼 引擎

✓ 匯入成功完成，共執行了 32 個查詢指令。(iaas_db.sql)

– phpMyAdmin SQL Dump
– version 4.2.12deb2+deb8u1
– http://www.phpmyadmin.net
–
– 主機: localhost
– 產生時間: 2016 年 04 月 12 日 15:17
– 伺服器版本: 5.5.44-0+deb8u1
– PHP 版本: 5.6.19-0+deb8u1

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";# MySQL 回傳空的查詢結果 (即零資料列)。

SET time_zone = "+00:00";# MySQL 回傳空的查詢結果 (即零資料列)。

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;# MySQL 回傳空的查詢結果 (即零資料列)。

/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;# MySQL 回傳空的查詢結果 (即零資料列)。

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;# MySQL 回傳空的查詢結果 (即零資料列)。

/*!40101 SET NAMES utf8 */;# MySQL 回傳空的查詢結果 (即零資料列)。

新增 MYSQL ACCOUNT

1. 透過 phpMyAdmin 新增一 mySQL 帳號，給予它存取 sensor_data 資料庫的權限。



2. 編輯 conn/pdo.php 檔案，將剛剛新增的帳號及密碼填入。

```

1 <?php
2 $servername = "localhost";
3 $username = "pi";
4 $password = "bananapi";
5 $dbname = "sensor_data";
6
7 $jsonSensors = '
8 {
9   "sensors": [
10     {
11       "sensorIndex": 0,
12       "sensorID": "MR",
13       "sensorList": [ 1, 2, 3, 4 ] },
14     {
15       "sensorIndex": 1,
16       "sensorID": "OR",
17       "sensorList": [ 1, 2, 8, 10, 11 ] },
18     {
19       "sensorIndex": 2,
20       "sensorID": "HR",
21       "sensorList": [ 1, 2, 5, 6 ]
22     }
23   ]
24 }
25 '

```

設定要顯示的裝置及 SENSORS

1. 接下來，我們要設定主機上要顯示那些感測器的資訊。請回頭檢視一下您目前使用中的 IAAS 感應裝置，假設您目前有三台，請看一下它們的 Arduino Code，分別記下 sensorsList[] 以及 sensorID 的值。（這兩個值是可自行設定的）

第一台：

```
#include <SPI.h>
#include <Ethernet.h>

//===== Configuration =====

//Sensor代碼定義：1->溫度，2->濕度，3->漏水，4->煙霧可燃氣體，5->CO，6->天然氣，7->PM1，8->PM2.5，9->PM10，10->CO2，11->環境光亮度
const unsigned sensorsList[] = {1,2,8,10,11}; //要啟用那些感應器？建議上限最多5個。
const unsigned nums = 5; //啟用的感應器數量，最多5個

#define LCD_DISPLAY 1 //是否要啟用LCD顯示？

int unsigned countsAVG = 6; //要取幾次的sensor值，去除最大與最小值後，作最終平均？最少需3次

const long updateInterval = 2 * 60000; //單位ms
const String sensorID = "OR"; //給這個感測裝置取個英文ID，注意此ID不要與其它裝置重複。

#define uploadDataServer 1 //要上傳指定的server？是:1，否:0
const String deviceName = "新竹Office"; //說明此感測裝置用途或地點。
```

第二台：

```
#include <SPI.h>
#include <Ethernet.h>

//===== Configuration =====

//Sensor代碼定義：1->溫度，2->濕度，3->漏水，4->煙霧可燃氣體，5->CO，6->天然氣，7->PM1，8->PM2.5，9->PM10，10->CO2，11->環境光亮度
const unsigned sensorsList[] = {1,2,5,6}; //要啟用那些感應器？建議上限最多5個。
const unsigned nums = 4; //啟用的感應器數量，最多5個

#define LCD_DISPLAY 1 //是否要啟用LCD顯示？

int unsigned countsAVG = 6; //要取幾次的sensor值，去除最大與最小值後，作最終平均？最少需3次

const long updateInterval = 2 * 60000; //單位ms
const String sensorID = "HR"; //給這個感測裝置取個英文ID，注意此ID不要與其它裝置重複。

#define uploadDataServer 1 //要上傳指定的server？是:1，否:0
const String deviceName = "台中家"; //說明此感測裝置用途或地點。
```

第三台：

```
#include <SPI.h>
#include <Ethernet.h>

//===== Configuration =====

//Sensor代碼定義：1->溫度，2->濕度，3->漏水，4->煙霧可燃氣體，5->CO，6->天然氣，7->PM1，8->PM2.5，9->PM10，10->CO2，11->環境光亮度
const unsigned sensorsList[] = {1,2,3,4}; //要啟用那些感應器？
const unsigned nums = 4; //啟用的感應器數量

#define LCD_DISPLAY 1 //是否要啟用LCD顯示？

int unsigned countsAVG = 6; //要取幾次的sensor值，去除最大與最小值後，作最終平均？最少需3次

const long updateInterval = 2 * 60000; //單位ms
const String sensorID = "MR"; //給這個感測裝置取個英文ID，注意此ID不要與其它裝置重複。

#define uploadDataServer 1 //要上傳指定的server？是:1，否:0
```

2. 編輯 conn/pdo.php 檔案，將各台的 sensorsList[] 及 sensorID 資訊填入。

```

1  <?php
2  $servername = "localhost";
3  $username = "pi";
4  $password = "bananapi";
5  $dbname = "sensor英ata";
6
7  $jsonSensors = '
8      { "sensors":[
9          { "sensorIndex": 0,
10             "sensorID": "MR",
11             "sensorList": [ 1, 2, 3, 4] },
12          { "sensorIndex": 1,
13             "sensorID": "OR",
14             "sensorList": [ 1, 2, 8, 10, 11] },
15          { "sensorIndex": 2,
16             "sensorID": "HR",
17             "sensorList": [ 1, 2, 5, 6]
18          }
19      ]
20  }';

```

將 IAASWEB 目錄整個上傳到網站伺服器。

上傳後，請記下該網站的存取路徑，下一步驟中需要使用。

將 WEB 主機相關資訊更新到各感應 DEVICE

接下來我們要更新 IAAS 各 device 的 Arduino code，讓感測資料能正確傳送到指定的伺服器，請修改下列的紅圈處。

```

const unsigned sensorsList[] = {1,2,5,6}; //要啟用那些感應器? 建議上限最多5個.
const unsigned nums = 4; //啟用的感應器數量, 最多5個

#define LCD_DISPLAY 1 //是否要啟用LCD顯示?

int unsigned countsAVG = 6; //要取幾次的sensor值, 去除最大與最小值後, 作最終平均? 最少需3次

const long updateInterval = 2 * 60000; //單位ms
const String sensorID = "HR"; //給這個感測裝置取個英文ID, 注意此ID不要與其它裝置重複.

#define uploadDataServer 1 //要上傳指定的server? 是:1, 否:0
const String deviceName = "台中家"; //說明此感測裝置用途或地點.
IPAddress DataServerIP(172,30,16,135); //將資料以GET方式傳到特定server IP, 注意IP間的小數點要改用逗號
const short DataServerPort = 80; //將資料以GET方式傳到特定 server 的 port 埠號
const String DataServerPage = "/sensors/newdata.php"; //傳到特定server的接收網頁

byte mac[] = { 0xB8, 0x27, 0xEB, 0xEF, 0x36, 0x31 }; //請自行設定一組MAC Address, 可隨意指定, 但請勿跟網路上已有的網卡相同避免衝突
const boolean networkDHCP = 1; //使用DHCP: 1, 使用固定IP: 0

//---> 固定IP,如果使用固定IP資訊請填入下方資訊
byte ip[] = { 10,0,0,35 }; //如果沒有成功得到DHCP的位址, 則會使用這個IP.
byte myDns[] = { 8,8,8,8 }; //如果沒有成功得到來自DHCP的DNS, 則會使用這個DNS IP.
byte gateway[] = { 10,0,0,254 };
byte subnet[] = { 255, 255, 255, 0 };

```

後續修改及調整

當網站開始有資料進來之後，進入網頁後您會發現所有燈號是紅色的，這是因為還未設定上限警示值的緣故。



	電腦機房 2016-04-13 14:49	新竹Office 2016-04-13 14:48	台中家 2016-04-13 14:48
溫度	28°C	30°C	29°C
溼度	46%	48%	42%
漏水	6%		
煙霧	263ppm		
一氧化碳			162ppm
瓦斯			181ppm
PM2.5		10 ug/m3	
CO2		984ppm	
光照度		83Lux	

請進入 phpMyAdmin，修改 sensor_list 表格中 alarm_max 及 alarm_min 此兩項資訊，即可讓頁面正確的顯示綠燈或紅燈。

效能分析 [本頁編輯] [編輯] [SQL 語句分析]

資料列數: 25 搜尋資料列: 搜尋此資料表

依主鍵排序: 無

+ 選項

	id	idname	name	moredesc	type_id	create_date	alarm_max	alarm_min	email_time
<input type="checkbox"/> 編輯 複製 刪除	14	HR1	HR	台中家	1	1460528386	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	15	HR2	HR	台中家	2	1460528386	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	16	HR5	HR	台中家	5	1460528386	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	17	HR6	HR	台中家	6	1460528386	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	18	MR1	MR	電腦機房	1	1460528444	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	19	MR2	MR	電腦機房	2	1460528444	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	20	MR3	MR	電腦機房	3	1460528444	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	21	MR4	MR	電腦機房	4	1460528444	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	22	OR1	OR	新竹Office	1	1460528500	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	23	OR2	OR	新竹Office	2	1460528500	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	24	OR8	OR	新竹Office	8	1460528520	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	25	OR10	OR	新竹Office	10	1460528520	0	0	0
<input type="checkbox"/> 編輯 複製 刪除	26	OR11	OR	新竹Office	11	1460528520	0	0	0

全選 已選擇項目: 修改 刪除 匯出

我們依據不同感應器並針對不同場所，輸入上下區間值如下：

+ 選項

	id	idname	name	moredesc	type_id	create_date	alarm_max	alarm_min	email_time
<input type="checkbox"/> 編輯 複製 刪除	14	HR1	HR	台中家	1	1460528386	35	10	0
<input type="checkbox"/> 編輯 複製 刪除	15	HR2	HR	台中家	2	1460528386	85	30	0
<input type="checkbox"/> 編輯 複製 刪除	16	HR5	HR	台中家	5	1460528386	300	0	0
<input type="checkbox"/> 編輯 複製 刪除	17	HR6	HR	台中家	6	1460528386	300	0	0
<input type="checkbox"/> 編輯 複製 刪除	18	MR1	MR	電腦機房	1	1460528444	26	20	0
<input type="checkbox"/> 編輯 複製 刪除	19	MR2	MR	電腦機房	2	1460528444	60	15	0
<input type="checkbox"/> 編輯 複製 刪除	20	MR3	MR	電腦機房	3	1460528444	30	0	0
<input type="checkbox"/> 編輯 複製 刪除	21	MR4	MR	電腦機房	4	1460528444	450	0	0
<input type="checkbox"/> 編輯 複製 刪除	22	OR1	OR	新竹Office	1	1460528500	30	20	0
<input type="checkbox"/> 編輯 複製 刪除	23	OR2	OR	新竹Office	2	1460528500	70	20	0
<input type="checkbox"/> 編輯 複製 刪除	24	OR8	OR	新竹Office	8	1460528520	20	0	0
<input type="checkbox"/> 編輯 複製 刪除	25	OR10	OR	新竹Office	10	1460528520	1000	0	0
<input type="checkbox"/> 編輯 複製 刪除	26	OR11	OR	新竹Office	11	1460528520	90	20	0

全選 已選擇項目: 修改 刪除 匯出

OK，此時網頁已能正確的顯示燈號了。

SunplusIT

環控系統

	電腦機房 2016-04-13 15:13	新竹Office 2016-04-13 15:14	台中家 2016-04-13 15:14
溫度	28°C	29°C	29°C
溼度	50%	50%	44%
漏水	7%		
煙霧	262ppm		
一氧化碳			167ppm
瓦斯			181ppm
PM2.5		11ug/m3	
CO2		1011ppm	
光照度		74Lux	

另外，sensor_type 表格中的 maxV 及 minV 則是用來設定曲線圖 Y 軸的上下限，您可依據不同感應的特性來調整這些值。

sensor_data

Procedure

資料表

新增

data_storage

data_storage2

sensor_list

sensor_type

+ 選項

← T →

	id	name	unit	maxv	minv
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	1	溫度	°C	60	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	2	溼度	%	100	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	3	漏水	%	100	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	4	煙霧	ppm	1000	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	5	一氧化碳	ppm	1000	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	6	瓦斯	ppm	1000	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	7	PM1.0	ug/m3	30	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	8	PM2.5	ug/m3	50	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	9	PM10	ug/m3	100	0
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	10	CO2	ppm	1600	400
<input type="checkbox"/> 編輯 <input type="checkbox"/> 複製 <input type="checkbox"/> 刪除	11	光照度	Lux	300	0

☐ 全選
 已選擇項目: ☐ 修改 ☐ 刪除 ☐ 匯出

各感測值的曲線圖顯示結果。

