

CH8 AWS 實驗

由於實驗8.3.1的內容有所更動，請同學照以下步驟進行，而其餘實驗步驟請參考過去實驗講義進行。原8.3節的實驗目的是要將「PM2.5的值傳到AWS」，而該講義則是改成將「人體紅外線感測器(PIR)之偵測結果傳到AWS」。

而該實驗作法主要是透過I2C通訊協定將RPi、人體紅外線感測器 (PIR) 和OLED之數據傳輸串通起來，其中RPi為Master，PIR和OLED則作為Slave。而RPi會聽取PIR傳送的訊息，每當接收到該訊息時，RPi會發送訊號請OLED顯示PIR的訊息結果，亦會將該結果傳送至AWS。而以下將為大家複習I2C協定，並請依循步驟完成實驗8.3.1節。

I2C 通訊協定

I2C (Inter-Integrated Circuit) 是一種同步的雙向通訊協議，用於微控制器與周邊裝置之間的數據傳輸。I2C 包含以下幾個重要概念：

- Master 與 Slave：
Master 是主控設備，負責發起通訊並控制時鐘信號 (SCL)。
Slave 是從屬設備，負責回應 Master 的請求並傳送或接收數據。
- 地址 (Address)：
每個 Slave 都有一個唯一地址，Master 根據地址識別並與特定 Slave 通訊。
- I2C 線路：
I2C 使用兩條線：資料線 SDA (Serial Data Line) 與時鐘線 SCL (Serial Clock Line)，所有設備共享這兩條線。其概念圖請參考圖1。

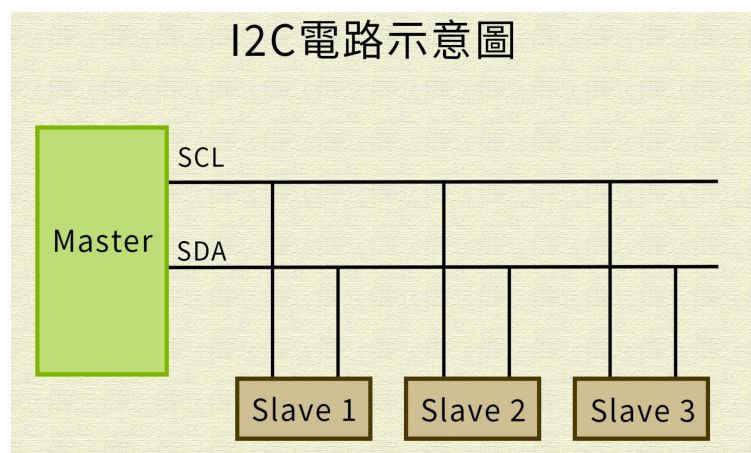


圖1

此實驗有兩個Slave，PIR（接在Arduino UNO上）及OLED，而 Master 則為RPi。

實驗說明

實驗器材

- RPi 4
- Arduino UNO
- 麵包板
- OLED SSD1315
- PIR Sensor HC-SR501

實驗步驟

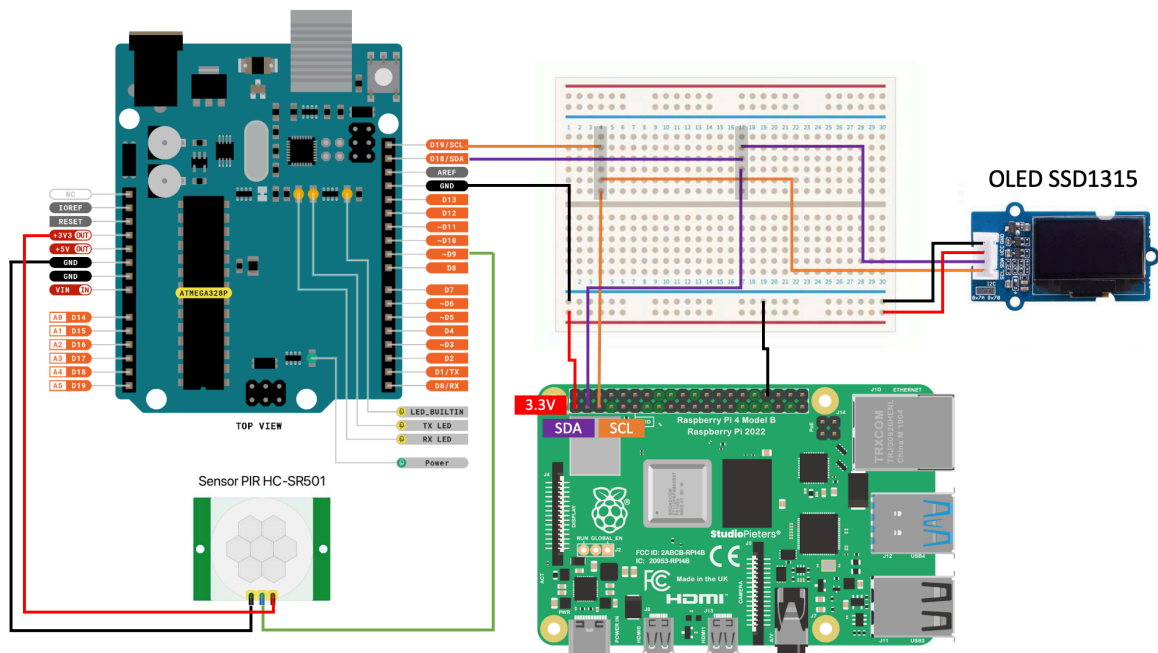


圖2、完整示意圖I2C線路示意圖

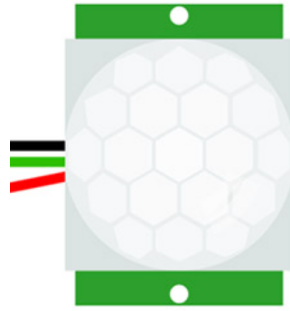


圖 3、PIR 腳位

黑色位置為地(接UNO GND)，紅色為電源(接UNO 5V)，綠色為訊號線。

Step 1：請先[下載程式碼](#)到你的電腦

Step 2：設置 PIR Sensor 與 Arduino UNO

1. 設置 Arduino 為 I2C Slave：

- 使用 Arduino UNO 開發板將 PIR Sensor (人體感應器) 連接到數位引腳9，細節請參考圖。
- 設定 Arduino 為 I2C Slave，接收 PIR Sensor 資料並透過 I2C 傳輸。請先將 Arduino UNO 接上電腦，開啟 Arduino IDE Serial Monitor(右上角)。接者，使用 Arduino IDE 上傳 **PIR_I2C.ino** 到 UNO 開發板
- 確認 PIR資料是否正確傳輸，以此初步檢查PIR 連接到 UNO 這段接線是否正確。

[Note]: Demo Code 同時通過 I2C 和序列埠(serial) 傳輸 PIR Sensor 資料。

serial 只是 for 電腦確認用，與本次系統無關

Step 3：設置OLED與RPi

1. 請將OLED和RPi的電路連接好，細節請參考圖2。
2. 檢測 I2C 地址：打開 Raspberry Pi 的終端機，輸入以下指令以檢測系統中的 Slave 地址：`i2cdetect -y 1`
 - OLED螢的預設地址通常為 **0x3C**，而 **PIR_I2C.ino**將PIR的地址設為 **0x55**。
 - 如果未檢測到兩個slave的地址，請確認接線是否正確。

[NOTE]: 接地錯誤可能導致無法偵測到上升緣 (Rising Edge) 。

Step 4 : 運行 RPi Master 程式碼

在運行 RPi Master Demo Code (`I2C_Master.py`) 之前，先確保系統安裝所需的 Python 套件來驅動 I2C 與 OLED 螢幕。請按照以下步驟進行：

下載套件：

1. 更新 Raspberry Pi 套件庫：

```
sudo apt-get update
```

2. 安裝 Python I2C 支援套件：

```
sudo apt-get install python3-smbus
```

```
sudo apt-get install i2c-tools
```

3. 安裝 I2C 與螢幕驅動套件：

```
sudo apt-get install luma.core
```

```
sudo apt-get install luma.oled
```

運行 RPi Demo Code：

- 執行程式碼 `I2C_Master.py`。
- 如果操作順利，應能在螢幕上看到由 PIR Sensor 傳輸至 RPi 的資料。

將PIR感測結果上傳至AWS

在原講義8.3.2的前半部，即在討論該如何將讀取感測器之結果上傳至AWS上，這邊一樣邀請同學們可以想想看該如何修改pubsub.py，來達到此目的。這邊助教也提供範例code (`pubsub_PIR.py`) 供同學參考，但還是鼓勵大家先自行嘗試看看～

實驗中Bug提示

在實驗中，如果使用pip install時遇到報錯 `error: externally-managed-environment` 該怎麼辦？這個錯誤常見於新版本的Python(ex: python 3.11)run在Linux、macOS、RPi上，而該錯

誤是開發者刻意為之，主要是為了防止Python的套件跟系統套件管理員衝突，詳情可以參考[PEP-688](#)。而要避免該錯誤，建議可以先開python虛擬環境，再使用pip install。

```
sudo apt install python3-venv
```

```
python3 -m venv myenv
```

```
source myenv/bin/activate
```

```
pip install [package_name]
```