

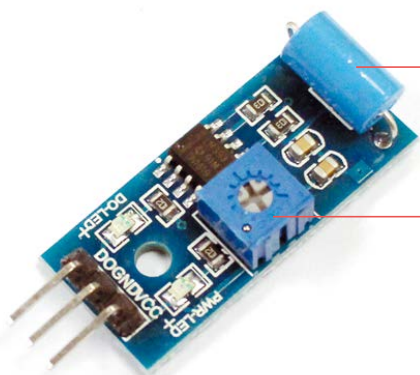
防盜即時警報器

手機簡訊、LINE 即時通知

我們將使用震動感測模組來設計一個防盜警報器，一旦震動感測模組偵測到入侵者的時候，就會立刻透過手機簡訊或 LINE 即時通知我們，這樣出門在外也可以透過手機隨時收到安全警訊。

2-1 認識震動感測模組

我們將使用以下的震動感測模組來偵測入侵者：



裡面有滾珠，模組會利用滾珠來偵測是否有震動

此旋鈕可以修改偵測靈敏度，順時針旋轉可增加靈敏度，反之則降低靈敏度

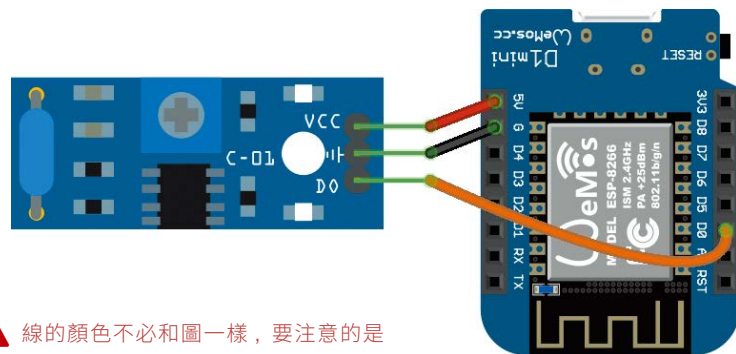
當模組感應到振動時，DO 腳位會輸出高電位。只要將這個振動感測模組放在保險箱、抽屜內，當有人翻找物品的話，模組就會感測振動輸出高電位，此時就立刻送出警報通知到我們手機。

Lab03

讀取振動感測模組的輸入值

實驗目的	用程式讀取振動感測模組的輸入值，藉以判斷是否正在振動。
材料	• D1 mini • 震動感測模組

線路圖



⚠ 線的顏色不必和圖一樣，要注意的是 V、G 兩種腳位不要接錯，其他腳位接錯都可以再重接。

fritzing

設計原理

當我們建立腳位的 Pin 物件時，可用 "Pin.IN" 作為參數，設定這個腳位為輸入腳位：

```
>>> from machine import Pin
>>> shock = Pin(16, Pin.IN)
```

上面我們建立了 16 號 (D0) 腳位的 Pin 物件，並且將其命名為 shock，因為建立物件時使用了 "Pin.IN" 參數，所以 16 號腳位就會被設定為輸入腳位。

建立好輸入腳位的 Pin 物件後，便可以使用 value() 方法來讀取外部裝置輸出的電位高低：

```
>>> shock.value()
0          ← 讀到 0 表示外部裝置輸出低電位
>>> shock.value()
1          ← 讀到 1 表示外部裝置輸出高電位
```

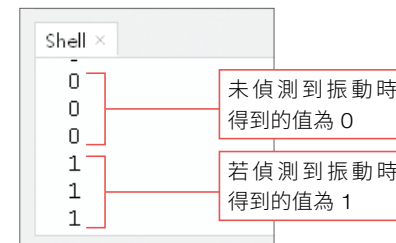
振動感測模組偵測到振動會輸出高電位，所以若讀到 1 便代表有振動了。

程式設計

```
01 from machine import Pin
02 import time
03
04 # 建立 16 號腳位的 Pin 物件，設定為輸入腳位，並命名為 shock
05 shock = Pin(16, Pin.IN)
06
07 while True:
08     # 用 value() 方法從 16 號腳位讀取按鈕輸出的高低電位
09     # 然後將讀到的值用 print() 輸出
10     print(shock.value())
11
12     # 暫停 0.05 秒
13     time.sleep(0.05)
```

實測

請按 **F5** 執行程式，然後用手搖動振動感測模組，在 Thonny 的 Shell 窗格觀察程式輸出的值：



2-2 發送手機簡訊

為了透過手機簡訊傳送感測器的資訊，我們將使用簡訊服務廠商的 API 來發送簡訊。

請連線 <http://www.message.com.tw> 如下操作加入會員：

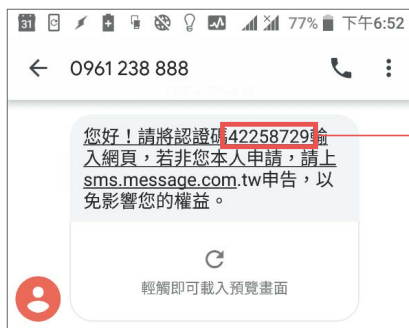


1 按此連結

2 依照網頁說明輸入資料 建議直接使用手機門號作為會員帳號

3 按此鈕建立帳戶 請注意密碼必須包含英文與數字

稍待片刻後手機簡訊會收到認證碼：

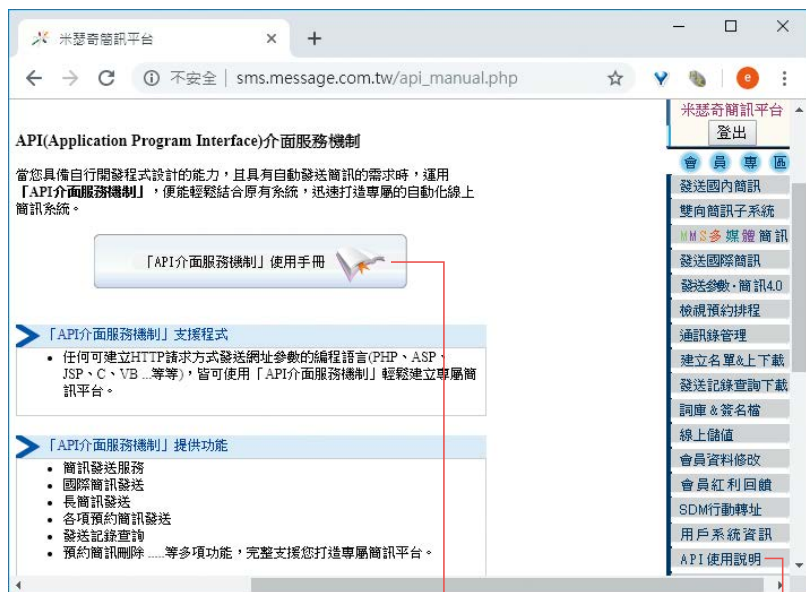


1 打開手機上的簡訊取得驗證碼



2 回到電腦瀏覽器上輸入驗證碼 3 按此鈕即可收到 100 通測試用的簡訊，若超過時需要付費購買額度

簡訊服務廠商會提供 API 介面，讓我們的程式可以透過 HTTP 連線來發送簡訊：



2 按此鈕下載文件 1 按此連結



用 HTTP 連線即可發送簡訊

⚠ 不同廠商 API 連線的 HTTP 網址格式皆不相同，若您使用其他廠商的話，請自行參閱該廠商網站提供的說明文件。

2-3 D1 mini 控制板連線 WiFi 網路

因為簡訊服務網站屬於外部網站，D1 mini 必須連上網際網路，才能連到簡訊服務網站利用其服務發送簡訊，所以接下來將說明如何用 Python 程式設定 D1 mini 連上 WiFi。

要使用網路，首先必須匯入 **network** 模組，利用其中的 **WLAN** 類別建立控制無線網路的物件：

```
>>> import network
>>> sta_if = network.WLAN(network.STA_IF)
```

在建立無線網路物件時，要注意到 D1 mini 有 2 個網路介面：

網路介面	說明
network.STA_IF	工作站 (station) 介面，專供連上現有的 Wi-Fi 無線網路基地台，以便連上網際網路
network.AP_IF	熱點 (access point) 介面，可以讓 D1 mini 變成無線基地台，建立區域網路

由於我們需要讓 D1 mini 連上網際網路擷取資訊，所以必須使用**工作站**介面。取得無線網路物件後，要先啟用網路介面：

```
>>> sta_if.active(True)
```

參數 **True** 表示要啟用網路介面；如果傳入 **False** 則會停用此介面。接著，就可以嘗試連上無線網路：

```
>>> sta_if.connect('無線網路名稱', '無線網路密碼')
```

其中的 2 個參數就是無線網路的名稱與密碼，請注意大小寫，才不會連不上指定的無線網路。例如，若我的無線網路名為 'FLAG'，密碼為 '12345678'，只要如下呼叫 **connect()** 即可連上無線網路：

```
>>> sta_if.connect('FLAGS', '12345678')
```

為了避免網路名稱或是密碼錯誤無法連網，導致後續的程式執行出錯，通常會在呼叫 **connect()** 之後使用 **isconnected()** 函式確認已經連上網路，例如：

```
>>> while not sta_if.isconnected():
    pass

>>>
```

上例中的 `pass` 是一個特別的敘述，它的實際效用是**甚麼也不做**，當你必須在迴圈中加入程式區塊才能維持語法的正確性時，就可以使用 `pass`，由於它甚麼也不會做，就不必擔心會造成任何意料外的副作用。上例就是持續檢查是否已經連上網路，如果沒有，就用 `pass` 往迴圈下一輪繼續檢查連網狀況。

⚠ `pass` 的由來就是玩撲克牌遊戲無牌可出要跳過這一輪時所喊的 `pass`。

連上 WiFi 後便可以連線簡訊服務網站使用其簡訊發送服務。

Lab04

防盜即時簡訊通知

實驗目的	利用振動感測模組感應是否有振動，若有則發送簡訊到手機。
材料	● D1 mini ● 振動感測模組

■ 線路圖

同 Lab03

■ 設計原理

在 Python 中有個 `requests` 模組可以讓我們的程式扮演瀏覽器的角色，連線網站使用各式各樣的網路服務，不過因為 D1 mini 控制板的記憶體比較少，所以在 MicroPython 中提供的是精簡版的 `urequests` 模組，名稱開頭的 "u" 是 "micro" 的意思。只要匯入此模組，即可使用該模組提供的 `get()` 連線網路服務：

```
>>> import urequests ← 匯入 urequests 模組
>>> urequests.get("https://flagtech.github.io/flag.txt") ← 連線網址
```

所以稍後我們將寫程式用 `urequests.get()` 連線 2-2 節最後看到的簡訊服務 API 網址，即可透過簡訊將感測器的資訊傳送到我們手機。

■ 程式設計

```
01 from machine import Pin
02 import time, network, urequests
03
04 # 連線 Wifi 網路
05 sta_if = network.WLAN(network.STA_IF)
06 sta_if.active(True)
07 sta_if.connect("Wifi基地台", "Wifi密碼")
08 while not sta_if.isconnected():
09     pass
10 print("Wifi已連上")
11
12 username = "簡訊服務帳號"
13 passwd = "簡訊服務密碼"
14 phone = "接收簡訊的手機號碼"
15 message = "有人打開保險箱在翻找東西,趕快去抓小偷!" #請勿輸入空格
16
17 # 建立 16 號腳位的 Pin 物件，設定為輸入腳位，並命名為 shock
18 shock = Pin(16, Pin.IN)
19
20 while True:
21     if shock.value() == 1:
22         print("感應到振動!")
23
24         # 連線簡訊服務發送簡訊通知
25         urequests.get(
26             "http://api.message.net.tw/send.php?mttype=G&id="
27             + username + "&encoding=utf8&password=" + passwd
28             + "&tel=" + phone + "&msg=" + message)
29
30         # 暫停 60 秒，避免短時間內一直收到重複的警報
31         time.sleep(60)
```


- 請依照您的環境修改程式第 7 行中的『Wifi 基地台』、『Wifi 密碼』等設定
- 請依照 2-2 節申請簡訊服務帳戶時輸入的資料，修改程式第 12、13 行的『簡訊服務帳號』、『簡訊服務密碼』等設定
- 請修改程式第 14 行的『接收簡訊的手機號碼』，輸入您的手機號碼
- 程式第 15 行的簡訊內容可以任意修改，但請注意不要輸入空格

■ 實測

請按 **F5** 執行程式，然後用手搖動振動感測模組，可以在 Thonny 的 Shell 窗格看到程式輸出 "感應到振動!"，接著稍等片刻您的手機應該就會收到簡訊通知。

軟體補給站

使用其他簡訊服務廠商

不同的簡訊服務廠商會有不同的 API，所以只要閱讀廠商的文件，了解其 API 的 HTTP 連線格式，然後修改程式內 `urequests.get()` 連線的網址即可。

範例內附一個 Lab04-2.py，這個程式改用 <https://www.twsmss.com/> 這家簡訊服務廠商，您可以到該廠商的網站申請帳號，然後修改 Lab04-2.py 試用其簡訊發送服務。

2-4 使用 IFTTT 發送 LINE 通知

LINE 已經深入我們的生活，成為每個人手機上不可或缺的通訊 App。我們將使用 IFTTT 的服務，將感測器的資訊透過 LINE 即時通知我們，LINE 通知是完全免費的，沒有任何數量限制。

IFTTT 是英文 "IF This, Then That" 的縮寫，其服務的精神就是『如果 A 然後就 B』。我們希望如果偵測到振動 (A) 就發一個 Line 通知給我們 (B)，這樣的規則稱為一個程序：



請先到 IFTTT 網站 (<https://ifttt.com>) 註冊成會員：

1 點擊 Sign up

2 可以用 Google 或 Facebook 帳號註冊，或者用其他信箱。我們選擇用其他信箱，點選下方 sign up：

3 輸入 Email 信箱作為會員帳號

4 輸入會員密碼


5 點選 Sign up

6 選右上方的 X 略過此步驟，如此即完成註冊

● D1 mini ● RFID 感測模組 ● RGB 三色 LED

▲ 線的顏色不必和圖一樣。

為了在 D1 mini 讀取 RFID 感測模組傳送來的卡號，我們需要安裝 RFID 感測模組的函式庫到 D1 mini 上。



執行選單的『File/Open』指令，或按 **Ctrl + O**

開啟

← → ↕ ↑ << RD1 > FM617A > 函式庫 > ↻ 搜尋 函式庫

組合管理 ▾ 新增資料夾

本機 3D 物件 MSN 上我的社群 下載 文件

Th BlynLib.py
Th mfr522.py

3 選擇 mfr522.py 檔案

4 按此鈕開啟函式庫檔案

檔案名稱(N): mfr522.py Python files (*.py;*.pyw)

開啟(O) 取消

Thonny - D:\RD1\FM617A\函式庫\mfrc522.py @ 230:1

File Edit View Run Device Tools Help

mfrc522.py x

```

1 from machine import Pin
2 from os import
3
4
5 class MFRC522:
6

```

Device menu options:

- Soft reboot (Ctrl+D)
- Upload current script as main script (Ctrl+U)
- Upload current script as boot script
- Upload current script with current name**
- Show device's main script
- Show device's boot script
- Close serial connection

請小心不要執行到上面這兩個指令

5 執行選單的『Device/Upload current script with current name』指令

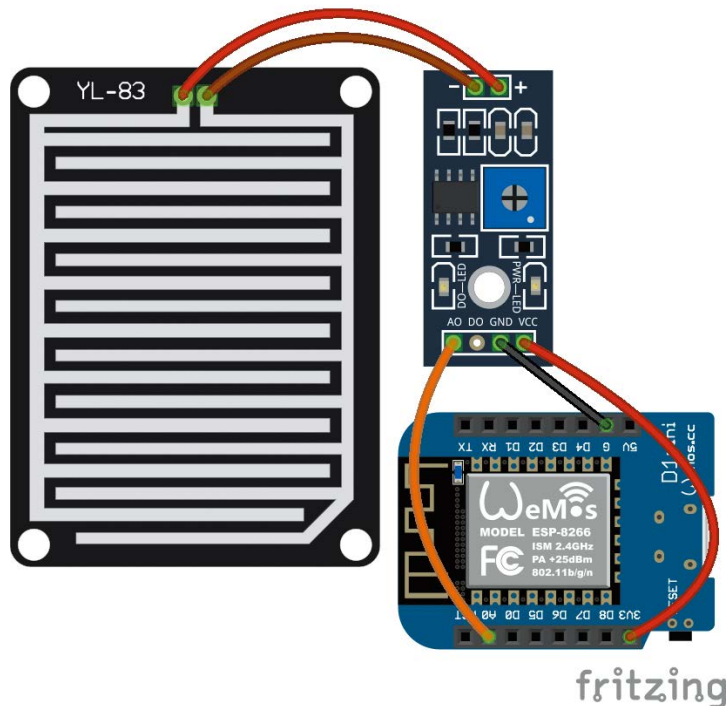
Lab09

讀取雨水感測模組的值

實驗目的 用程式讀取雨水感測器的電壓。

材料 • D1 mini • 雨水感測模組

■ 線路圖



■ 設計原理

請使用以下語法建立 A0 腳位的 ADC 物件：

```
>>> from machine import ADC
>>> adc = ADC(0)
```

然後使用 read() 方法即可讀取 ADC 轉換後的數值，數值越大表示電壓越大：

```
>>> adc.read()
168
>>> adc.read()
666
```

■ 程式設計

```
01 from machine import ADC
02 import time
03
04 # 建立 A0 腳位的 ADC 物件，並命名為 adc
05 adc = ADC(0)
06
07 while True:
08     # 用 read() 方法從 A0 號腳位讀取 ADC 轉換後的數值
09     # 然後將讀到的值用 print() 輸出
10     print(adc.read())
11
12     # 暫停 0.05 秒
13     time.sleep(0.05)
```

■ 實測

請按 **F5** 執行程式，然後以少許水滴在雨水感測模組的金屬感測面 (請小心！不要讓水接觸到感測面以外的電子零件)，在 Thonny 的 Shell 窗格觀察程式輸出的值：

