



IO ports 操作及應用 for MUART0 系列 RF UART 模組

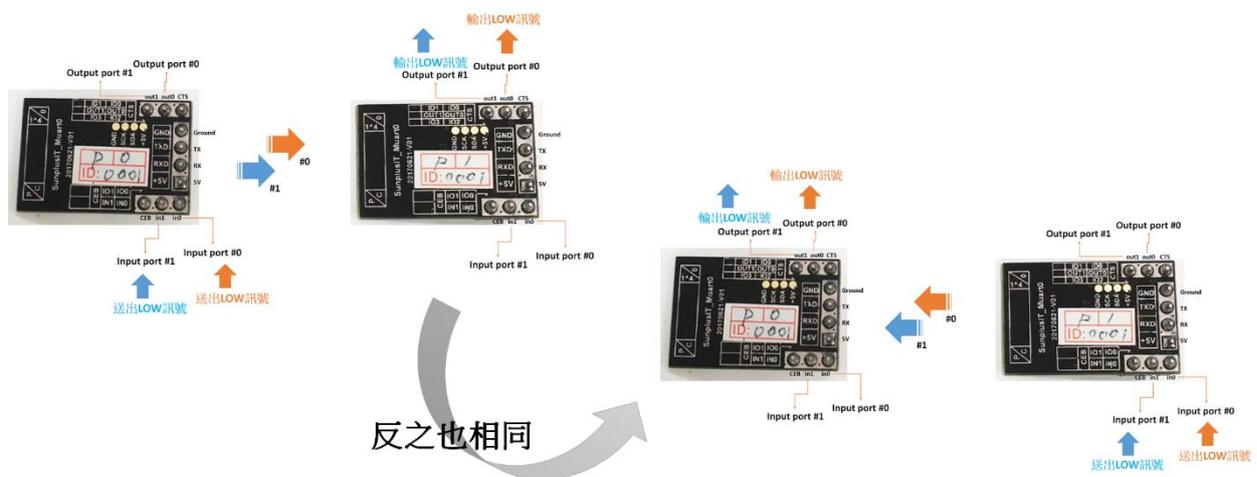
你的物聯網僅能單向的接收或傳送資料嗎？真正有價值的物聯網，應能雙向互動並依據不同的情境做出不同的反應，才能依據需求情境來客製及智慧化相關的應用。SunplusIT 所推出的 RF UART 系列模組擁以下三個特色，可讓我們馬上就能以最簡單的方式搭建起雙向的物聯網環境：

1. 穩定的 UART/TTL 無線傳輸。
2. IO port 支援：讓感測器直接具備雙向溝通能力。
3. 支援各種不同的情境需求：支援 1 對 1、1 對多...等功能。

讓感測器直接具備雙向溝通能力

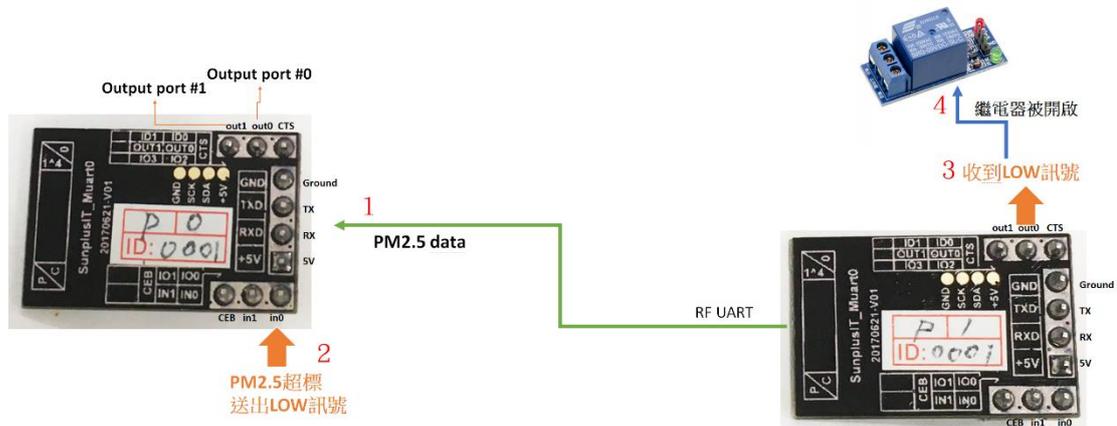
眾所周知的，SunplusIT 系列的 MUART0 能直接支援具有 UART/TTL 介面的感測模組，將模組透過 TTL/UART 發出的訊號轉為透過 RF 傳輸，升級為可獨立運作的無線模組，再加上所提供的 I/O port 功能，不需要開發板，我們就能讓模組具備雙向溝通的能力。

MUART0 系列模組上都有提供 2 組 I/O port，分別是 Input port #0, #1 以及 Output port #0, #1，當我們在 Input port #1 或 #2 輸入 LOW (GND) 訊號，則相對應一端的 Output port 也會同步收到 LOW (低電平) 訊號，反之也是相同。因此，利用這個功能我們可快速的製作出無線開關，或者傳遞 0 與 1 的訊息。

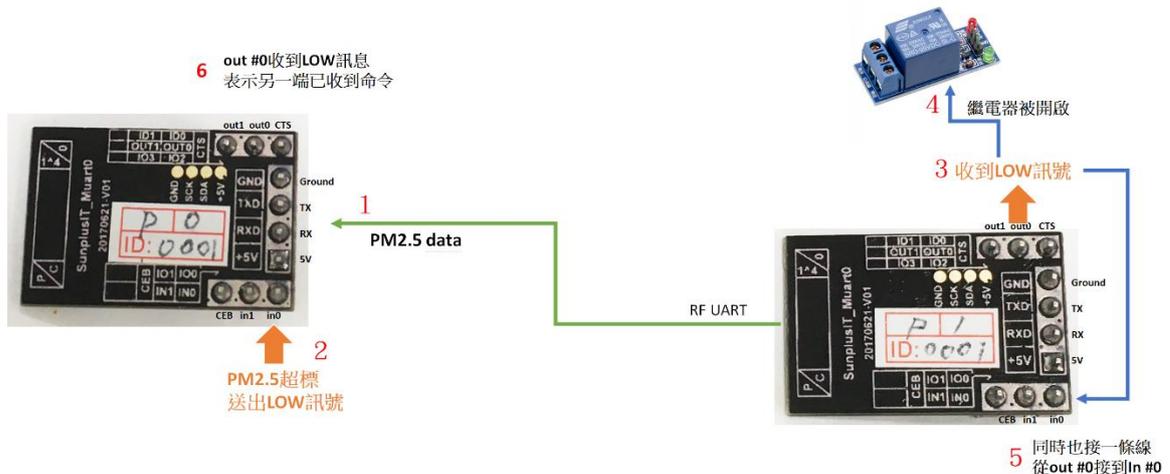


SunplusIT

舉例來說，請參考下圖，當接收端的 Arduino 收到來自傳送端 PM2.5 感測器的資訊，發現 PM2.5 指數超標，因此告知傳送端的繼電器將空氣清淨機開啟，接下來，我們要怎麼讓沒有 Arduino 的傳送端回報說已收到訊號並開啟了空氣清淨機？



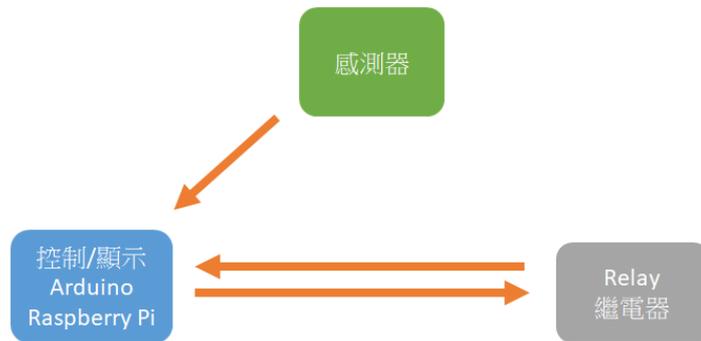
只要將乙端的 Output port #0 也接到自身的 Input port #0，這麼作的目的是，當乙端收到了 LOW 訊號，除了驅動繼電器之外，也會透過 Input port#0 送出 LOW 告知甲端已收到訊息。



是不是很簡單就能讓感測器擁有雙向的溝通確認功能呢？這種方式可應用在全系列的 MUART0 模組。

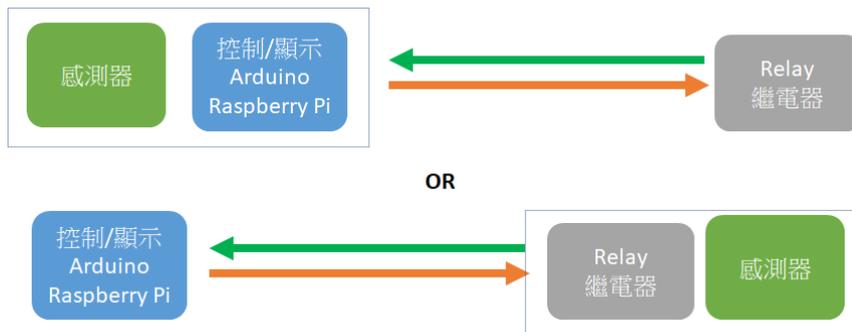
組合各種不同的情境需求應用

以最近大家在生活上最感同深受的 PM2.5 空氣指數為例，我們打算偵測並顯示目前 PM2.5 數值，若超標則透過繼電器自動開啟清淨機，並即時回報命令執行結果。類似這類依據感測器數值進行開關控制的 case 相當普遍，其流程類似下方：



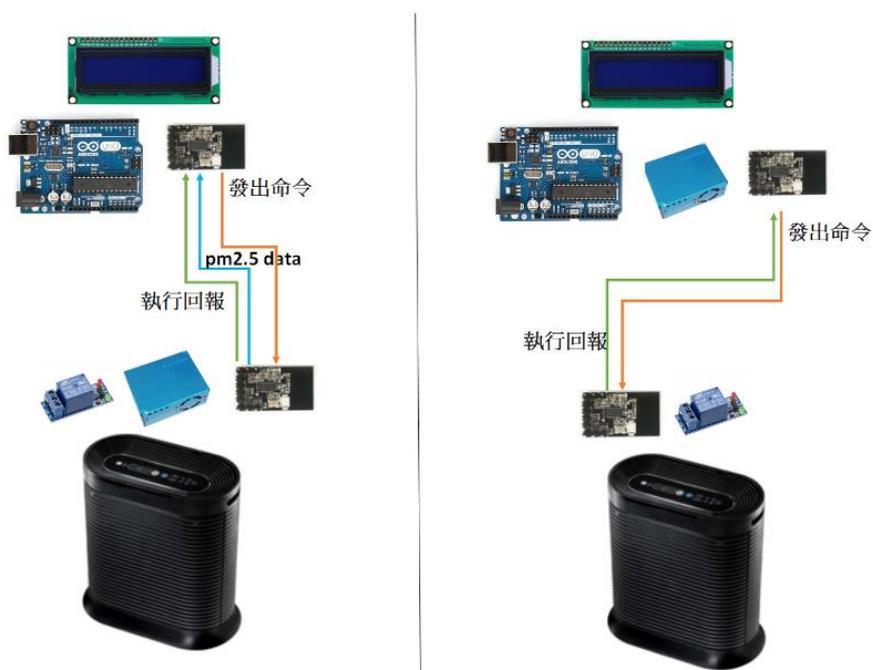
由於感測器與繼電器皆無法獨自運作，因此這三方如果要獨立分開作業就需要額外搭配的大量開發板，但透過 MUART0 無線 UART 模組及 I/O port 功能，我們就能讓感測器與繼電器獨立運作，且能組合出各種不同的應用情境，例如：

1. 感測器位於控制顯示端或繼電器端：

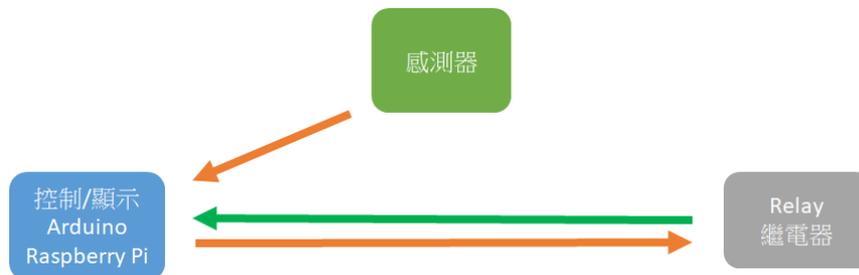


此情境可使用 MUART0-S-1-1 一對一 UART 傳輸模組。

如下圖所示，PM2.5 偵測器可視需求放置於顯示端或者後端的清淨機。

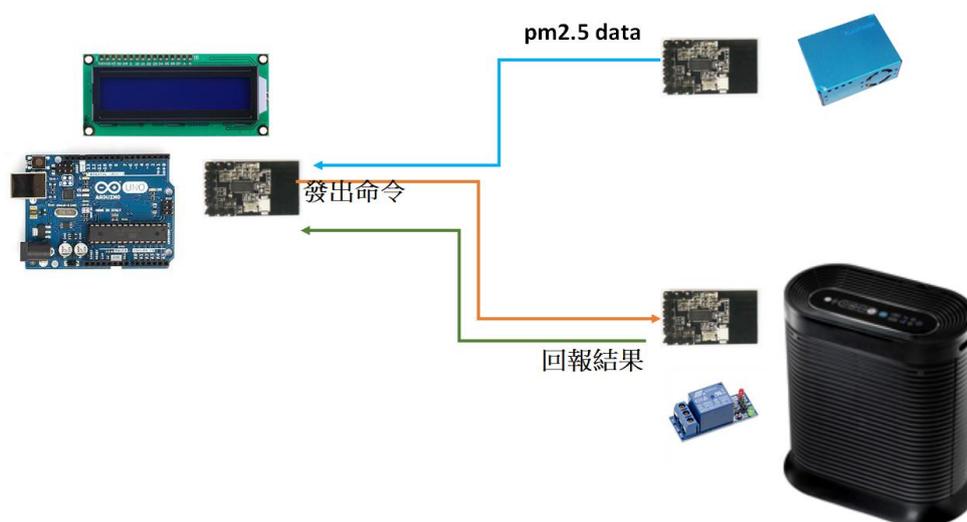


2. 顯示螢幕、PM2.5 偵測、空氣清淨機三者皆獨立：

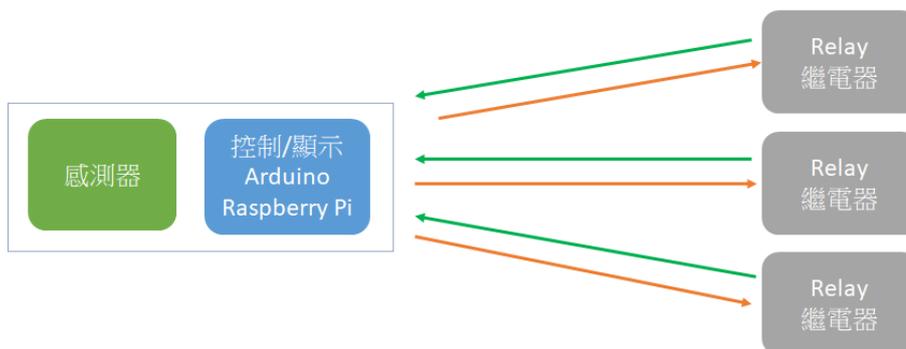


此情境可使用 MUART0-P-1-2 一對二 UART 傳輸模組。

資訊顯示螢幕、PM2.5 偵測器、空氣清淨機三者皆獨立，可分開自由擺放以因應環境的限制，如下圖所示。

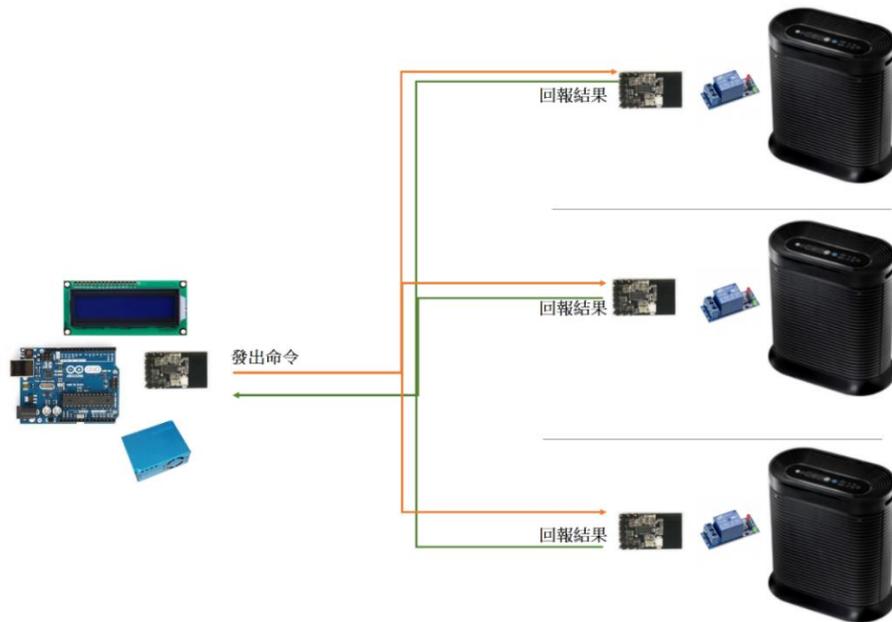


3. 一個終端顯示+PM2.5 偵測器，搭配多台空氣清淨機：

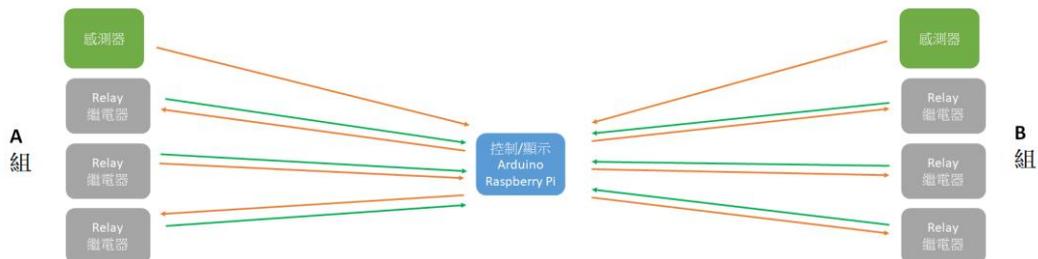


此情境可使用 MUART0-P-1-2 或 MUART0-PP 一對多 UART 傳輸模組。

把一台 PM2.5 偵測器及顯示螢幕放置於客廳（或辦公室中央），然後多台的空氣清淨機分置於不同的房間（或辦公室不同區域），如此一來，當感應到室內空品不良時，會自動開啟所有清淨機進行工作。

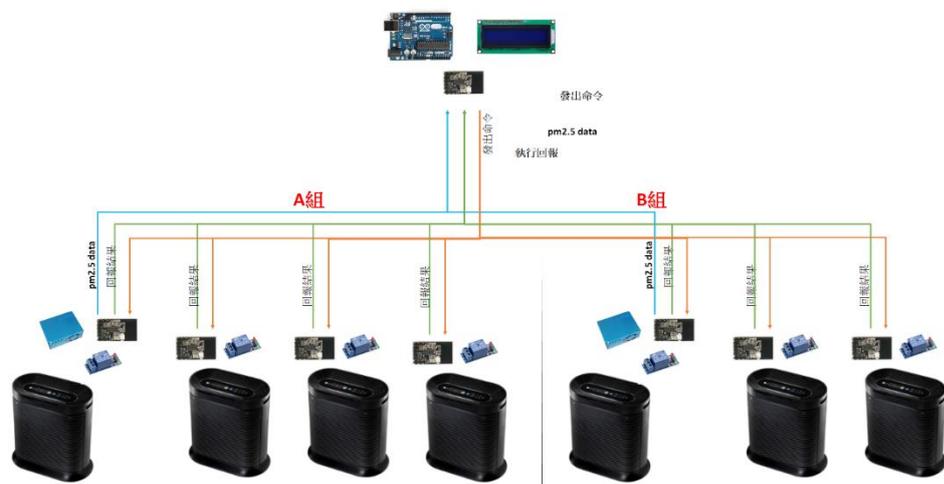


4. 一個終端顯示搭配多台 PM2.5 偵測器+空氣清淨機組成的 group :



此情境可使用 MUART0-PP 一對多 UART 傳輸模組。

假設每個樓層有數個房間，那麼每樓層我們可部署一台 PM2.5 偵測器並與同樓層其它 devices 組成一個 group。MUART0-PP 支援群組功能，您可以將不同 devices 劃規為不同的群組來控制。



5. 其它：

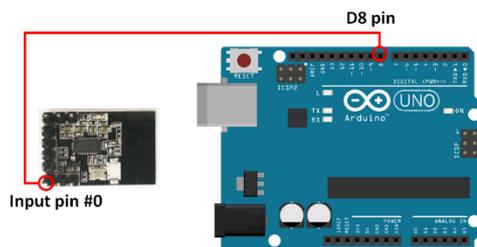
其實不止以上這些，我們還可以組合出其它更多的應用情境，您想得到更多嗎？

使用 **Arduino** 操控 **IO port**

對於 Arduino 的開發者來說，善用 **MUART0** 提供的 **IO ports** 可以讓專案更加簡捷與方便，那麼我們要如何操控 IO port 呢？請參考下方 Q & A。

A. 針對特定的 **Input ports** 腳位送出 **LOW** 訊號。

假設您將 **Input pin #0** 接到 **Arduino** 的 **D8**：

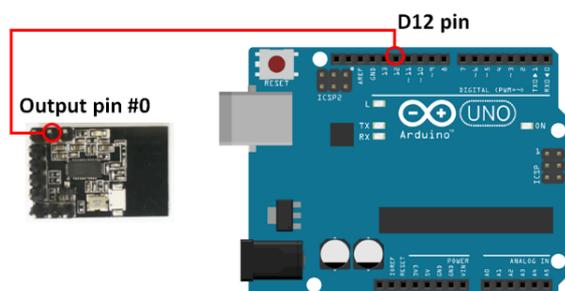


可撰寫程式如下：

```
byte pinRelay = 8; //告訴 Arduino 連到 Input #0 的是 D8 腳位
pinMode(pinRelay, OUTPUT); //設定 D8 為 OUTPUT
digitalWrite(pinRelay, HIGH); //送出 HIGH (Input #0 預設即為 HIGH)
digitalWrite(pinRelay, LOW); //送出 LOW (將觸發 Input #0 發出訊號)
```

B. 如何接收並監控 **Output ports** 的訊號？

假設您將 **Output pin #0** 接到 **Arduino** 的 **D12**：



可撰寫程式如下：

```
byte pinRelayStatus = 12; //告訴 Arduino 連到 Output #0 的是 D12 腳位
pinMode(pinRelayStatus, INPUT); //設定 D12 為 INPUT
boolean relay_status = digitalRead(pinRelayStatus); //讀取 D12 的值
if(boolean relay_status == 0) { //如果為 LOW
```

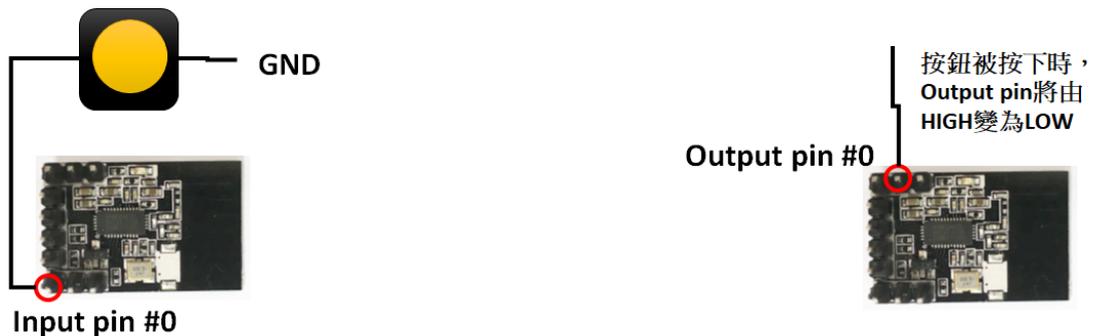
```

.....
} else { //否則(為 HIGH)
.....
}

```

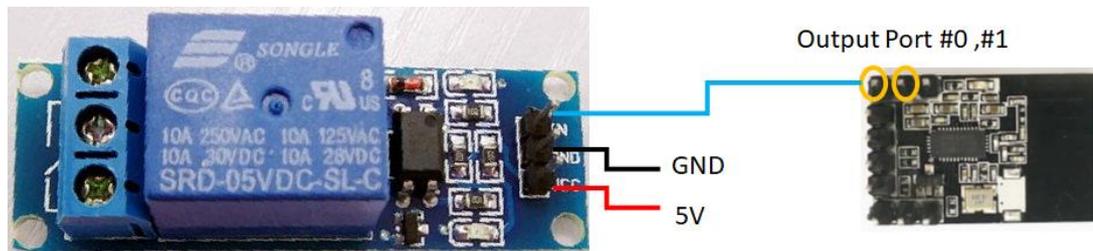
C. 如何直接使用 I/O ports 作為無線按鈕而不透過 Arduino ?

上述的 A 與 B 程式組合起來便成了透過 Arduino 所控制的無線按鈕，可進行複雜的功能操作，但我們也可以單獨使用 MUART0 來實現無線按鈕功能。



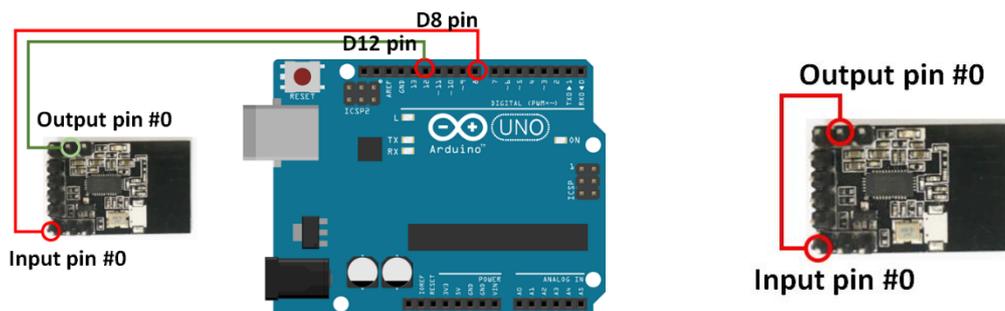
D. 如何使用 I/O ports 連接繼電器以控制家電？

由於觸發時 Output pin 是送出 LOW（低電平），因此您可以考慮使用低電平觸發的繼電器。



E. 已送出 LOW 訊號到 Input port，要如何等待並確認對方有收到回應？

假設您的 root 端與 device 端已分別進行如下的連接：



SunplusIT

接下來只要在 root 端加入如下的迴圈判斷，讓 device 端回傳已收到訊息。

```
byte pinRelay = 8; //告訴 Arduino 連到 Input #0 的是 D8 腳位
byte pinStatus = 12; //告訴 Arduino 連到 Output #0 的是 D12 腳位
void setup() {
    pinMode(pinRelay, OUTPUT); //設定 D8 為 OUTPUT
}
void loop() {
    digitalWrite(pinRelay, LOW); //送出 LOW
    unsigned long timeStart = millis();
    while (millis() - timeStart < 3000) { //在三秒內持續的 detect
        //如果收到 LOW 訊號就跳出迴圈
        if (digitalRead(pinRelay) == 0) break;
    }
}
```

F. IO ports 的速度有限制嗎？

IO ports 速度目前限制為 1Hz，也就是建議傳送的時間不要小於一秒。