

NI myRIO

XBee 無線通訊模組

使用手冊

第 1 章、產品概述

1.1 XBee 概述

XBee 為 Digi International 公司所開發之無線通訊模組，XBee 有許多不同系列產品，主要可以依據訊號傳輸距離、無線訊號頻率、支援的網路協定、資料傳輸率、耗電量和天線接線方式等作區分。

市面上常見的 XBee 一般分成兩種，XBee 和 XBee-PRO，最主要的差異是在於訊號傳輸距離和耗電量，XBee 在室內的訊號傳輸距離約為 30 公尺，XBee-PRO 則是約 100 公尺；在室外空曠處，XBee 傳輸距離可達到 100 公尺，而 XBee-PRO 可達到 1500 公尺。耗電量 XBee 為 1mW，XBee-PRO 則為 60mW。表 1-1 為基本規格表。

表 1-1

Specification	XBee	XBee-PRO
Indoor/Urban Range	Up to 100ft.(30m)	Up to 300ft.(100m)
Outdoor RF line-of-sight Range	Up to 300ft.(100m)	Up to 1mile(1500m)
Transmit Power Output	1mW(0 dBm)	60mW(18 dBm)
RF Data Rate	250,000 bps	250,000 bps
Serial Interface Data Rate	1200-115200 bps	1200-115200 bps
Receiver Sensitivity	-92 dBm	-100 dBm
Supply Voltage	2.8V-3.4V	2.8V-3.4V
Operating Frequency	ISM 2.4 GHz	ISM 2.4GHz
Operating Temperature	-40 to 85°C	-40 to 85°C

若以無線訊號頻率來區分，XBee 系列產品大多數用在 2.4GHz 的頻段，但也有少數是操作在 900MHz 的頻段，基本上可以用頻段在 900MHz 的 XBee-PRO 模組搭配高倍率的天線來傳輸超過 10 公里的距離，但是在同一個網路中，不能混合兩種不同頻段的 XBee 一起使用。

若是以支援的網路協定來區分，市面上最常見的是 XBee Series1 和 XBee Series2。Series1 基本上是支援 802.15.4 通訊協定，而 Series2 則是支援 ZigBee 通訊協定，所以在使用上 Series1 和 Series2 的 XBee 模組是不能夠互通的。

另外 XBee 和 XBee-PRO 的腳位定義是一樣的，詳細腳位定義如圖 1-1 和表 1-2 所示。詳細規格都可以在 Digi 的官網內下載查詢。

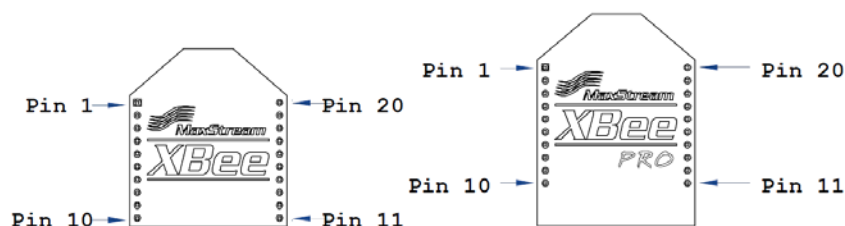


圖 1-1

表 1-2

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN/CONFIG	Input	UART Data In
4	DO8	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200ns)
6	PWM0/RSSI	Output	PWM Output 0/RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Don not connect
9	DTR/SLEEP_RQ/DI8	Input	Pin Sleep Control line or Digital Input 8
10	GND	-	Ground
11	AD4/DIO4	Either	Analog Input 4 or Digital I/O 4
12	CTS/DIO7	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON/SLEEP	Output	Module Status Indicator
14	VREF	Input	Voltage Reference for A/D Inputs
15	Associate/AD5/DIO5	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS/AD6/DIO6	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	AD3/DIO3	Either	Analog Input 3 or Digital I/O 3
18	AD2/DIO2	Either	Analog Input 2 or Digital I/O 2
19	AD1/DIO1	Either	Analog Input 1 or Digital I/O 1
20	AD0/DIO0	Either	Analog Input 0 or Digital I/O 0

1.2 ZigBee 通訊協定

XBee 基本上支援 802.15.4 或者是 ZigBee 通訊協定，802.15.4 是 IEEE(電機電子工程師學會)發布的無線傳輸標準，其特點是連線簡易、低數據傳輸率以及低功耗。

ZigBee 則是底層採用 IEEE 802.15.4 的標準規範，加入路由和其他聯網功能，主要特色有低速、低功耗、低成本、支援大量網路節點、支援多種網路拓撲、低複雜度、快速、可靠、安全。

而一般常見的無線通訊有 WiFi、Bluetooth 和 ZigBee(802.15.4)，表 1-3 比較了這幾種的差異。

表 1-3

	WiFi	Bluetooth	ZigBee(802.15.4)
應用領域	高速無線網際網路	無線設備連結	無線監測和控制
傳輸速率	100Gbps	720Kbps	250Kbps
典型傳輸距離	50~100 公尺	10~100 公尺	100 公尺
消耗功率	高	中	低
成本	高	中	低
安全性	低	高	中
普及程度	高	中	低
特色	速度、普及	使用便利	網路節點數最大可達 65000 個。

ZigBee 的網路架構主要支援點對點(Point-to-Point)、單點對多點(Point-to-Multipoint)以及網狀(Mesh)架構，而在使用 ZigBee 網路架構前，必須先設定好每一個 XBee 所代表的角色。

在 ZigBee 網路內，主要可以分成三個角色：

1. 協調器(Coordinator): ZigBee 網路內只會包含一個協調器，負責管理網路的設置。
2. 路由器(Router): 同一個 ZigBee 網路內可以包含多個路由器，負責轉發其他 XBee 的訊號。
3. 終端(End Device): 同一個 ZigBee 網路內可以包含多個終端 XBee，終端 XBee 無法直接和其他終端 XBee 通訊。

點對點通訊，在這裡指的是兩個 XBee 設備直接相連，由於在 ZigBee 網路中，終端是不能彼此相連結的，所以必須將其中一個設定成協調器(Coordinator)，另一端設定成終端(End Device)，才能互相傳輸資料。如圖 1-2 所示。

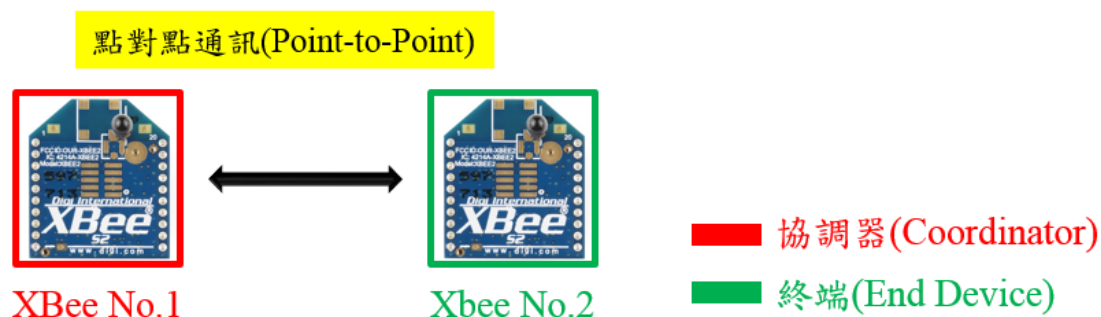


圖 1-2

單點對多點通訊指的是由一個協調器(Coordinator)分別向每一個終端(End Device)進行資料收發的動作，換句話說，如果 XBee No.2 要和 XBee No.5 互相交換資料，就必須將彼此資料先傳輸至 XBee No.1，再由 XBee No.1 分別傳到對方的位置。

一旦協調器故障，整個 ZigBee 網路就會中斷，所有終端的資料將無法再進行傳輸，如圖 1-3 所示。

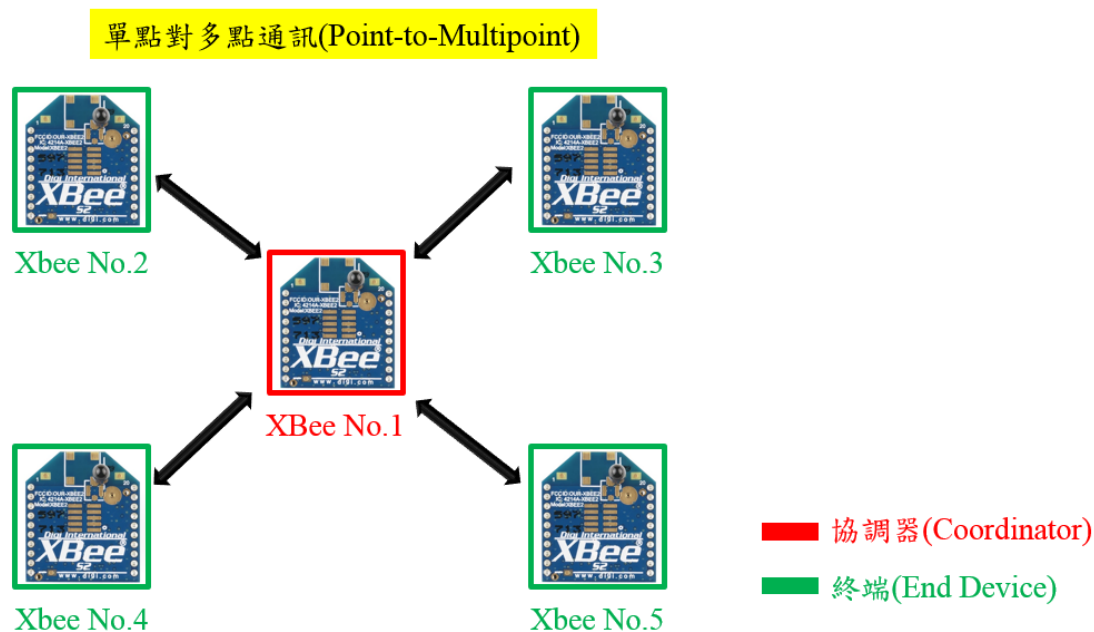
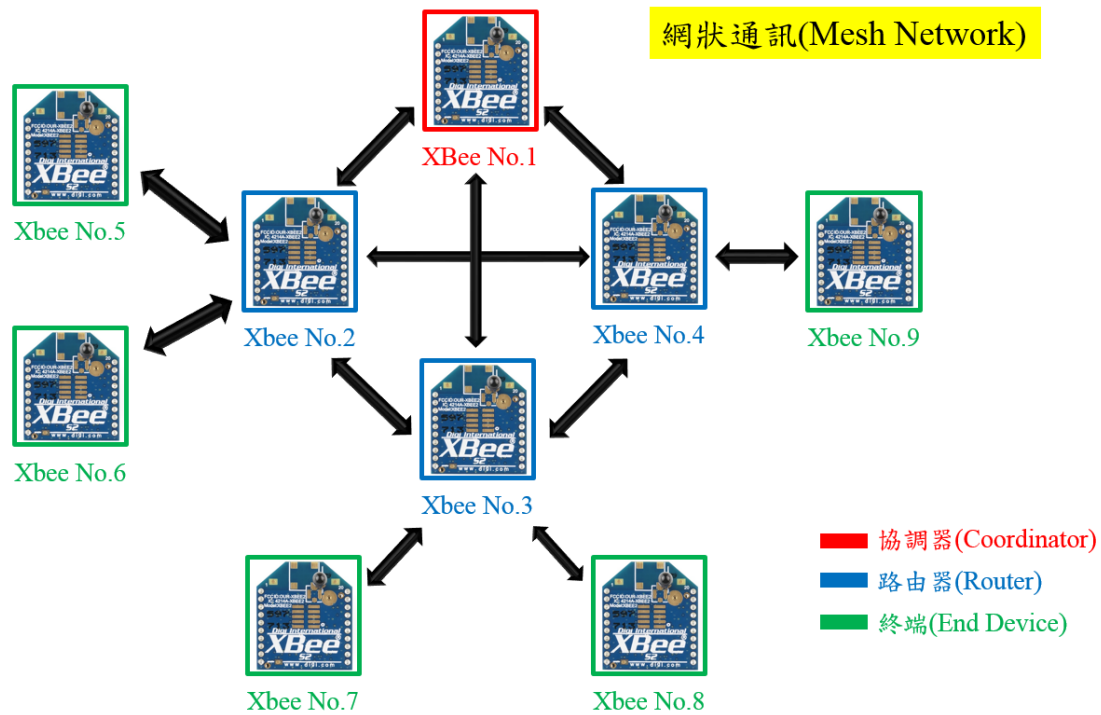


圖 1-3

網狀通訊主要架設用於長距離的通訊，而通訊距離大於單個 XBee 所能達到的範圍，就必須加入路由器(Router)的設定，在 ZigBee 網路中的路由器彼此可以互相傳輸資訊，也可以和協調器(Coordinator)交換訊息，而每個路由器可以和自己的終端 XBee 進行通訊，但是不同路由器所連結的終端 XBee 不能互相溝通，得透過路由器才能建立通訊橋梁。如圖 1-4 所示。



1.3 NI myRIO XBee 無線通訊模組

設計 XBee 無線通訊模組主要是讓 myRIO 的使用者在沒有任何無線通訊經驗下，透過使用說明書和自行開發的 API 快速建立一套無線通訊設備。

無線通訊模組可適用於 NI myRIO-1900 和 NI myRIO-1950(單機板)，在 NI myRIO 的規格表內定義了 MXP Connector 的功能，如圖 1-5(NI myRIO-1900)、圖 1-6 (NI myRIO-1950)所示。在連接 NI myRIO 時，可以將模組直接連接到 myRIO 上面，或者是自行購買 34Pin 的連接線(1 公 1 母)對接即可。

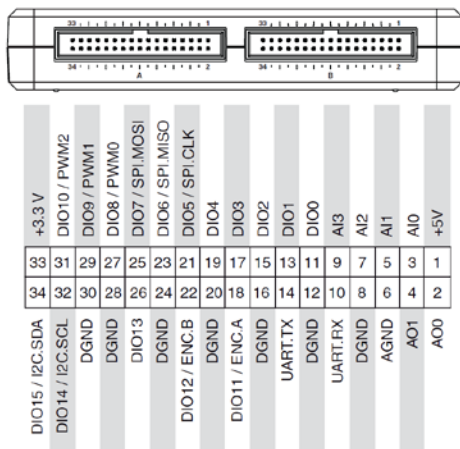


圖 1-5

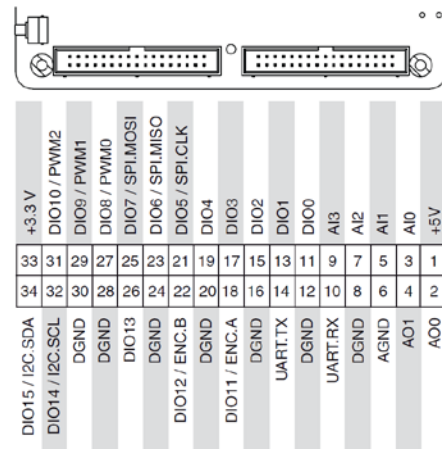


圖 1-6

無線通訊模組將取用 MXP 上面的 UART.TX、UART.RX、3.3V 和 DGND 供 XBee 模組使用，而其他沒有使用到的功能(如 DIO、AIO 等等)將透過無線通訊模組將訊號拉出來，模組詳細腳位配置如圖 1-7 所示，腳位定義以右圖為主。

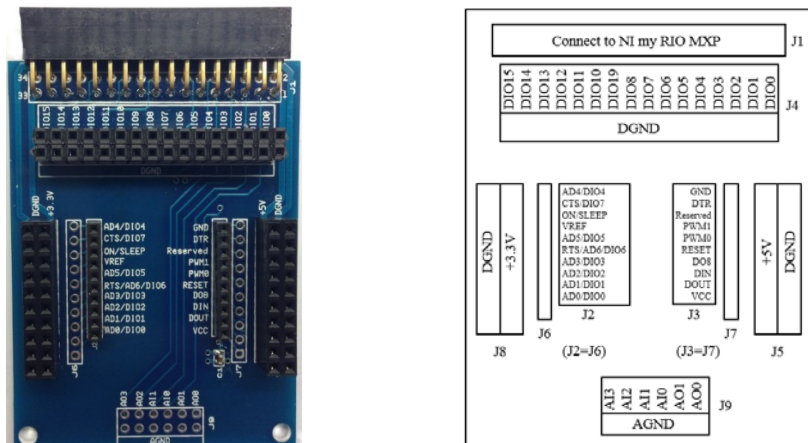


圖 1-7

在實際使用無線通訊模組時，只要將 XBee 模組連結到無線通訊模組上面，再連接 NI myRIO，透過程式設定後，便可以進行資料傳輸。如圖 1-8 所示。

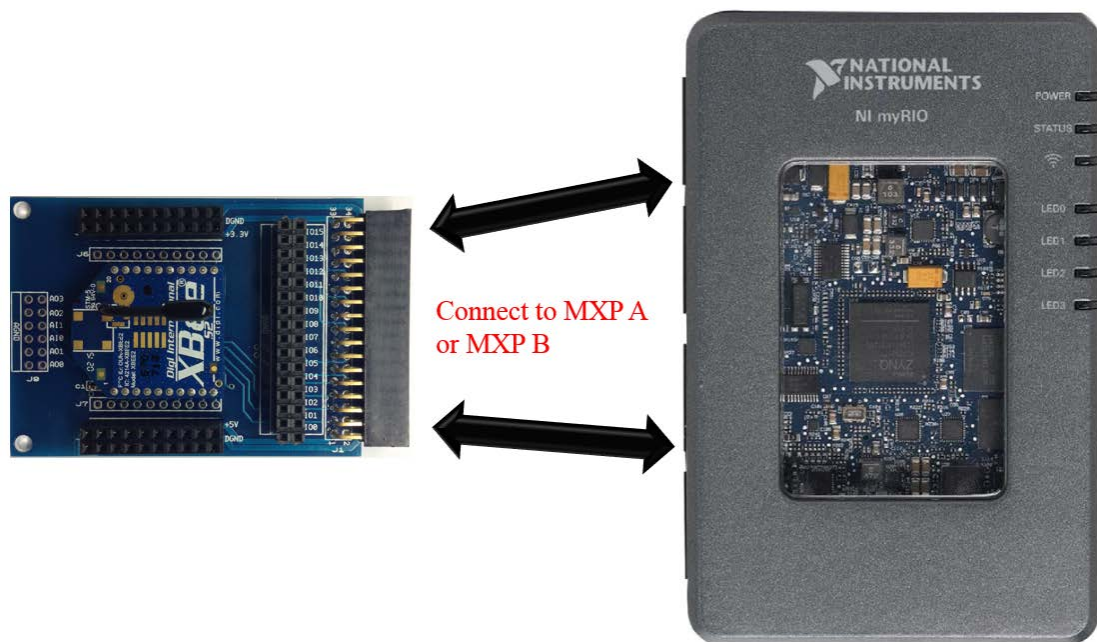


圖 1-8

第 2 章、操作說明

以下的操作說明將使用型號為 XB24-Z7WIT-004(S2 Series)的 XBee，範例中將會使用三組 XBee，一組為 Coordinator，另外兩組則是 End Device，將示範如何燒錄 XBee 內部的韌體以及撰寫 LabVIEW 程式進行 XBee 一對多的資料傳輸。

2.1 X-CTU 使用說明

X-CTU 為 Digi International 公司所開發的軟體，可以燒錄 XBee 的韌體以及設定內部參數，也可以操作一些簡單的傳輸範例，是一個相當方便的軟體，在開始進行操作前，請先至 Digi International 的官網下載 X-CTU。

進行 XBee 設定操作前，請先將 XBee 接上轉接板 XBee Explorer，透過 USB 轉接到電腦中，等待電腦抓到驅動程式後，便可以開啟 X-CTU 進行設定。如圖 2-1 所示。



圖 2-1

開啟 X-CTU 後，會在左上角看到新增 XBee 和搜尋 XBee 的選項，兩者都可以搜尋電腦中有連結到的 XBee，這裡請先點擊搜尋 XBee。如圖 2-2。

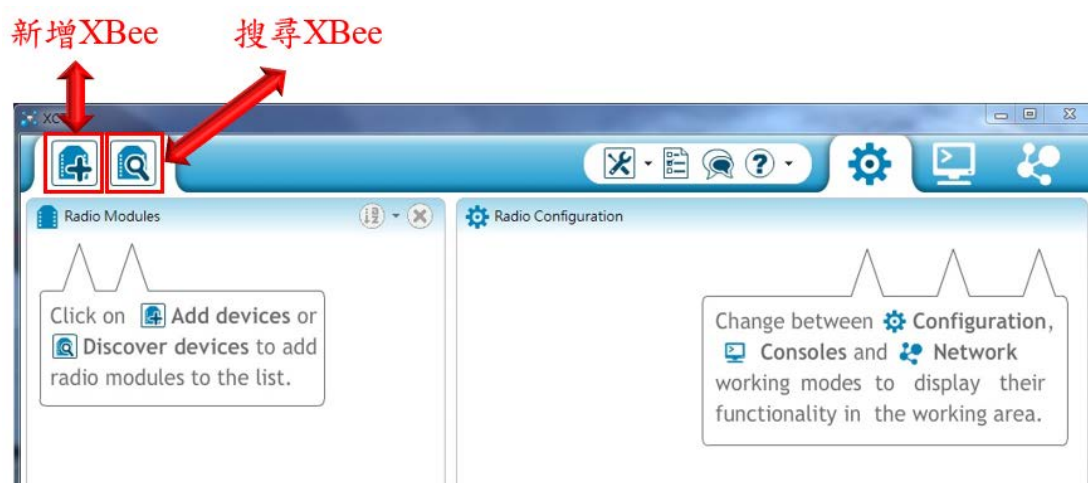


圖 2-2

請先選擇連接至電腦的序列埠，選好後點擊 Next 設定搜尋 XBee 的參數，需設定 Baud Rate、Data Bits、Parity、Stop Bits 和 Flow Control，一般以預設參數為主，若是 XBee 之前已經設定過其他參數，使用者遺忘參數的時候，可以勾選多項參數進行搜尋，當然勾選的參數越多，搜尋的時間越長。如圖 2-3。

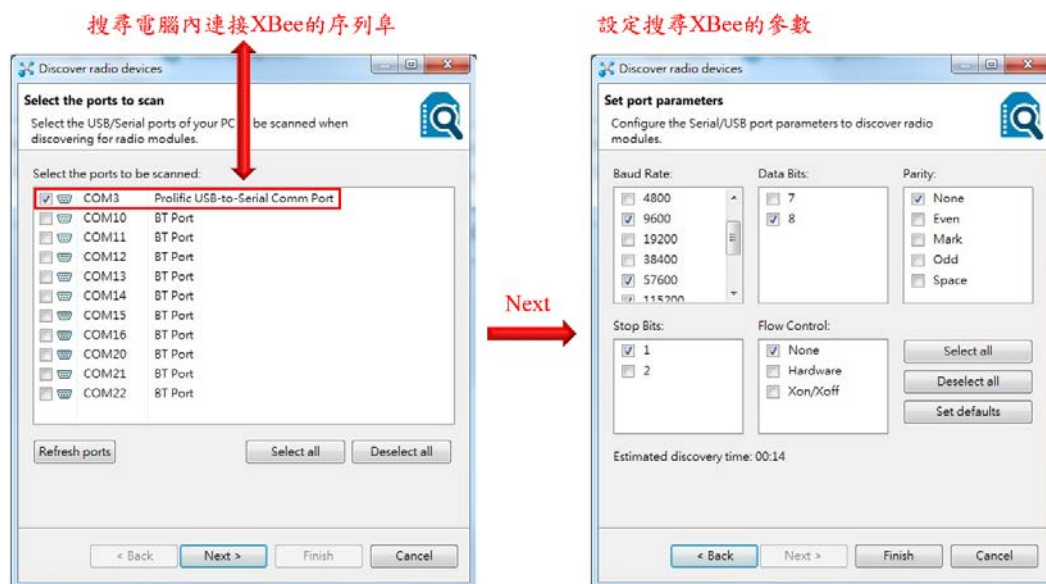


圖 2-3

搜尋完成後，會顯示目前電腦有連結到的 XBee 和內部參數，確認完成後，點擊加入裝置即可。如圖 2-4。

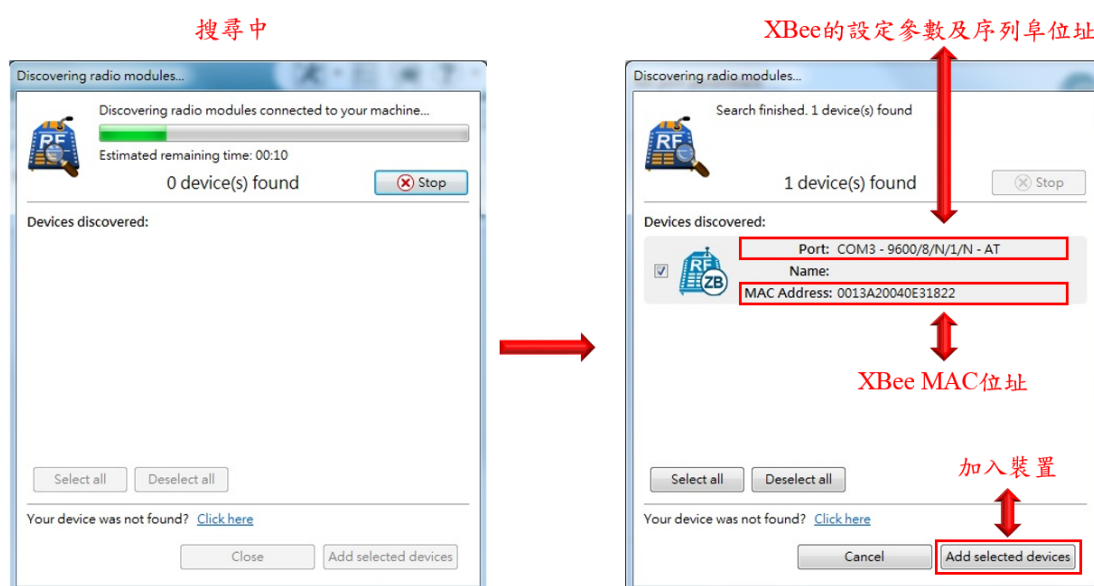


圖 2-4

當加入完裝置後，可以發現目前 XBee 的 Function 是 Router AT，而範例中需要一個 Coordinator 和兩個 End Device，所以需要更新 XBee 的韌體，因此點擊更新韌體後，在右邊可以看到需選擇 XBee 系列、XBee 執行模式和韌體版本。如圖 2-5。

1. XBee 系列一般就是看購買時候是屬於哪一系列即可。
2. XBee 執行模式則是依據使用者的應用端做改變，前面有提到範例將有三個 XBee，所以請將其中一個設定為 ZigBee Coordinator API，另外兩個設定為 ZigBee End Device API。
3. 韌體版本一般選最新的即可。



圖 2-5

更新完韌體後，請設定最基本的兩組參數，一組為 PAN ID，在相同的 PAN ID 下的 XBee 模組才能夠互相通訊，所以三個模組的 PAN ID 都要相同，範例中都設成 0，設定好後，請點選旁邊的鉛筆圖示寫入到 XBee 裡面。

另外一組則是 Baud Rate，指的是資料傳輸率，同樣的要將三組的 Baud Rate 設定成相同的，範例中將會設成 57600，設定好後一樣將參數寫入到 XBee。如圖 2-6。



圖 2-6

一般在燒錄韌體時，有時候會出現圖 2-7 左邊的畫面，此時若連接至電腦的 XBee Explorer 上面有 Reset 按鈕，請直接按下 Reset，靜候幾秒鐘便可會繼續燒錄，假如失敗的話，則需要多試幾次。

若是 XBee Explorer 上面沒有 Reset 按鈕的話，請拿一條導線連接 XBee 上面的 Pin5 和 Pin10 的腳位，一樣可以啟動 Reset XBee 的功能。

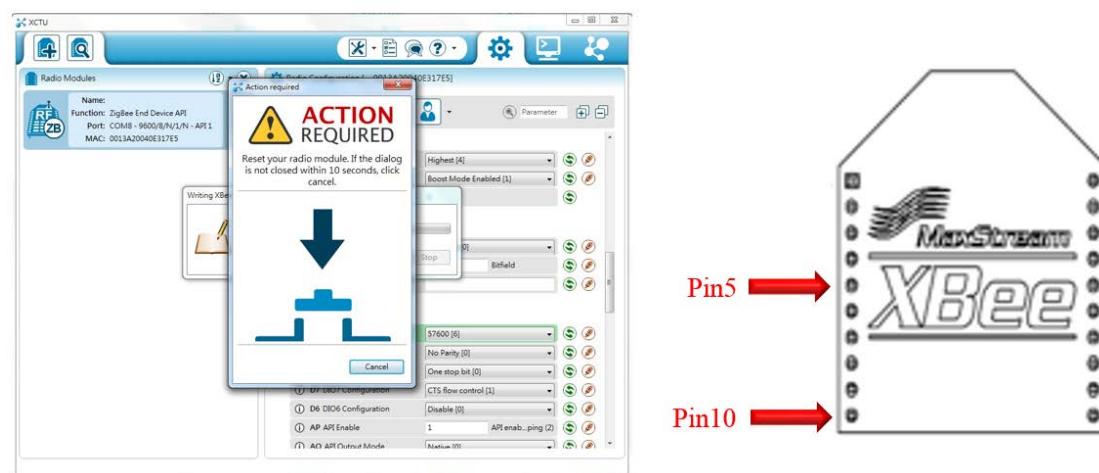


圖 2-7

最後確認所有模組都設定完成，一組 XBee 為 Coordinator，其餘兩組 XBee 為 End Device，Baud Rate 為 57600。

2.2 LabVIEW API 使用說明

此 API 是針對 XBee 模組韌體燒錄為 API 模式下才能使用，在範例的專案中，將會使用到兩個 API，分別為 Transmit Data.vi 和 Receive Data.vi，也就是傳送 XBee 的資料和接收 XBee 的資料。

在 My Computer 的目錄下可以查詢到 PC 使用的 API 程式，而在 myRIO-1900 的目錄下面可以找到 Real-Time Controller 所使用的 API。如圖 2-8。

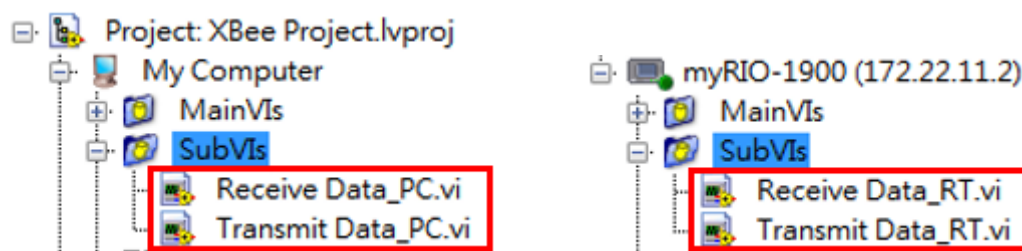


圖 2-8

Transmit Data 是用來傳輸 XBee 的資料，如圖 2-9:

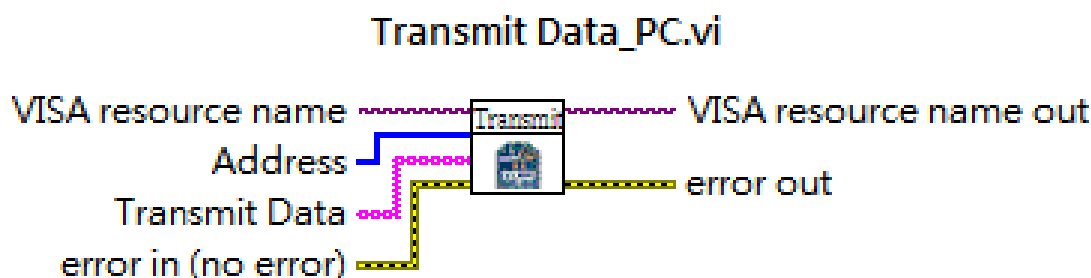


圖 2-9

1. VISA resource name 使用者在前面設定好 VISA 通訊的參數設定後，建立出來的名稱，會跟著 API 將名稱傳出。
2. Address 為目標 XBee 的 MAC Address，定義為 U64 的一維陣列，若本身為協調器(Coordinator)或是路由器(Router)，那麼目標 Address 可能會有一個或是多個；若本身為終端(End Device)，目標 Address 只會有一個，可能是協調器(Coordinator)或是路由器(Router)的 MAC Address。
3. Transmit Data 為想傳輸的資料，定義為一個 String 的一維陣列，順序會對應到 Address 的順序，舉例來說，若是要傳給 Address 陣列中的第二組 XBee，

所傳輸的資料就是填入 Transmit Data 陣列中的第二項。由於傳輸的資料型態為 String，使用者只需要在前面將其他的資料型態(例如 Boolean、Numeric、Cluster 等等)轉為 String 即可輸入進去。

4.error in 為檢視程式是否有發生錯誤，會跟著 API 將錯誤代碼傳出。

Receive Data 是用來接收 XBee 的資料，如圖 2-10:

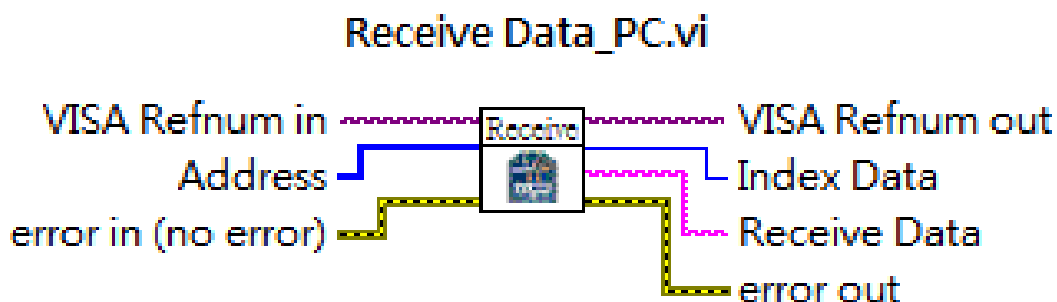


圖 2-10

- 1.VISA resource name 使用者在前面設定好 VISA 通訊的參數設定後，建立出來的名稱，會跟著 API 將名稱傳出。
2. Address 為目標 XBee 的 MAC Address，定義為 U64 的一維陣列，若本身為協調器(Coordinator)或是路由器(Router)，那麼目標 Address 可能會有一個或是多個；若本身為終端(End Device)，目標 Address 只會有一個，可能是協調器(Coordinator)或是路由器(Router)的 MAC Address。
3. error in 為檢視程式是否有發生錯誤，會跟著 API 將錯誤代碼傳出。
- 4.Index Data 為索引接收到的資料，若本身為協調器(Coordinator)或是路由器(Router)，那麼目標 Address 可能會輸入多組，而 Index Data 會根據回傳資料所對應之 Address 顯示所代表的 Index，例如收到 Index Data 為 2，代表收到 Address 陣列第三組 XBee 的資料。
- 5.Receive Data 為接收到的資料，使用者可以根據目前 Index Data 回傳的 Index，將接收的 String 資料還原成原始資料。

2.3 範例使用說明

此範例程式將使用 3 組 XBee，一組為協調器(Coordinator)，透過 XBee Explorer 連接至電腦，其餘兩組為終端(End Device)，分別裝在 NI myRIO-1900 的 MXP A 和 MXP B，myRIO 則透過 USB 線連接到電腦。如圖 2-11。

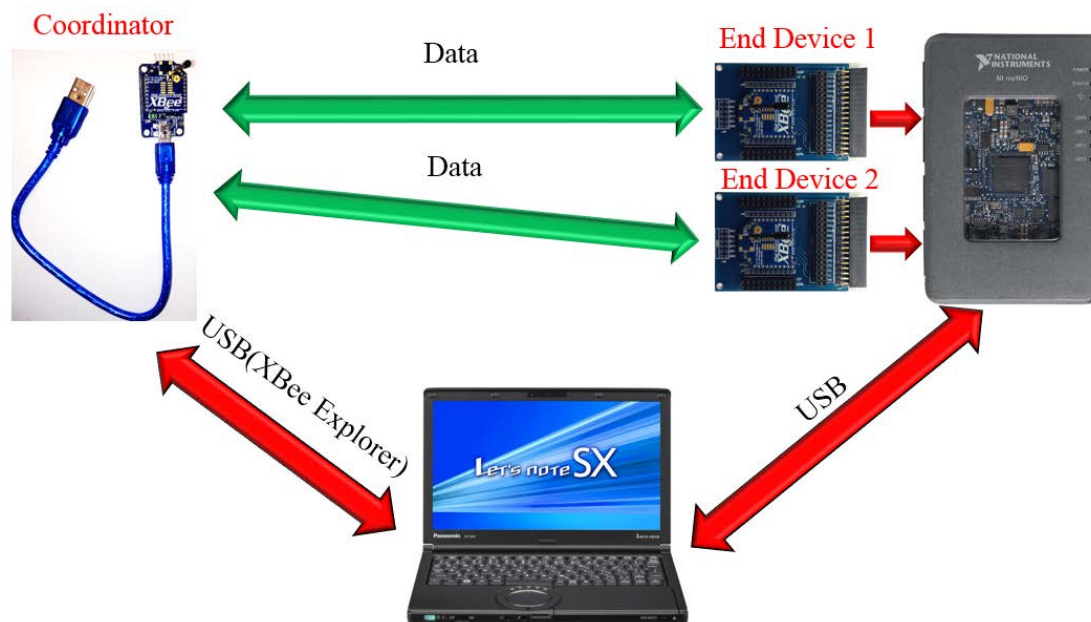


圖 2-11

當電源 myRIO 的電源接上後，便可以開啟範例專案，請先在專案中選取 myRIO-1900，右鍵點選 Connect 建立連線，待綠燈亮起即可，接下來開啟 My Computer 目錄下面的 XBee_PC.vi (代表 Coordinator)和 myRIO 目錄下面的 XBeeNO.1_RT.vi (代表 End Device1)、XBeeNO.2_RT.vi (代表 End Device2)。如圖 2-12。

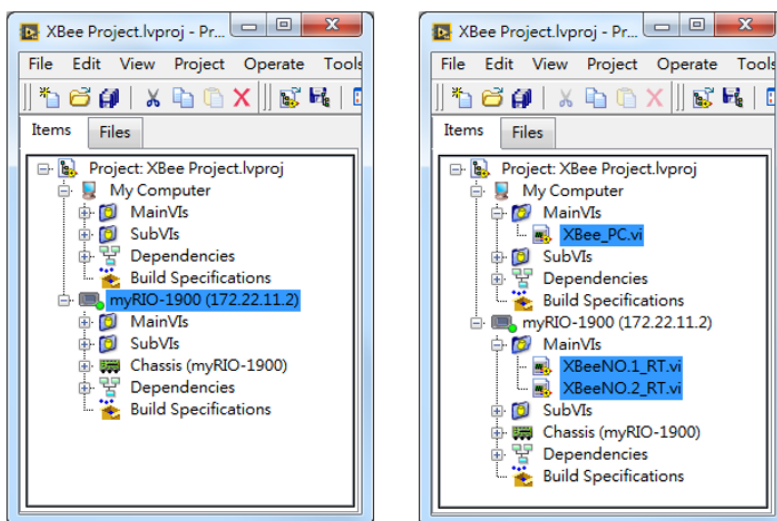


圖 2-12

XBee_PC.vi 人機介面如下圖 2-13，主要分為三個部分：

- 1.最左邊 XBee Address 為程式尚未執行前，須填入對方的 MAC Address，這裡由於 PC 為 Coordinator，所以填入的是另外兩組 End Device 的 Address，分別為 0013A20040E317E5 (XBee No.1)和 0013A20040E317D7 (XBee No.2)。
- 2.中間為 XBee No.1 所傳送和接收的資訊，在這裡會去讀取 myRIO 目前內建的加速規狀態和 myRIO 右下腳的 Button 按鈕狀態，而傳送的資料則是可以控制人機介面上四顆 LED 燈的狀態，對應到的是 myRIO 上面的 LED 燈號。
- 3.最右邊為 XBee No.2 所傳送和接收的資訊，在這裡示範的是直接以字串 String 作為傳輸及接收的資料。

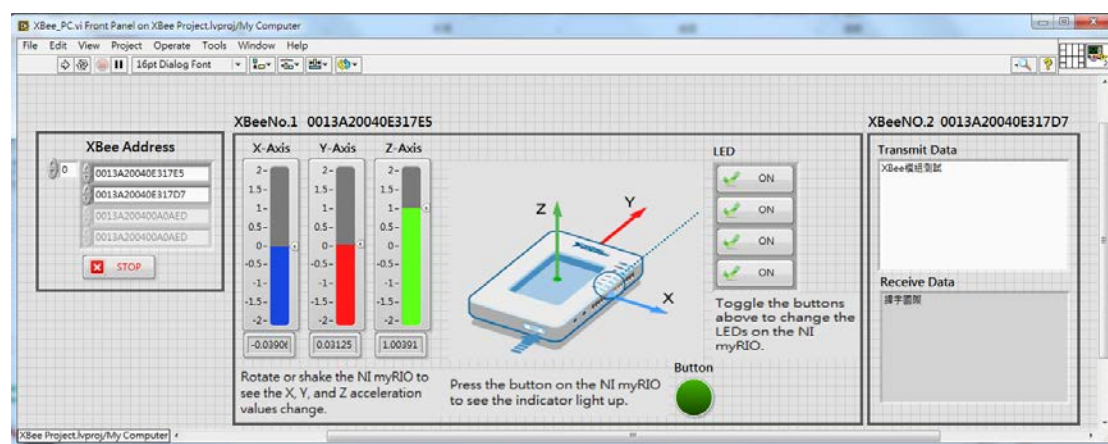


圖 2-13

XBee_PC.vi 程式區如下圖 2-14，最左邊為初始化設定，包含了最重要的 XBee 序列埠名稱、Baud Rate、Data Bits、Parity、Stop Bits、Flow Control 等等，都需要與 XBee 的設定相同才能建立通訊。

右邊迴圈裡面主要包含了傳輸和讀取資料，傳輸的資料包含了 myRIO 上面的 LED 控制(XBee No.1)和字串傳輸(XBee No.2)，將其組成一維陣列傳送。

讀取資料則是包含了 myRIO 的加速規狀態和右下腳的 Button 按鈕狀態 (XBee No.1)以及字串(XBee No.2)，會依據不同的 Index Data 進行還原。

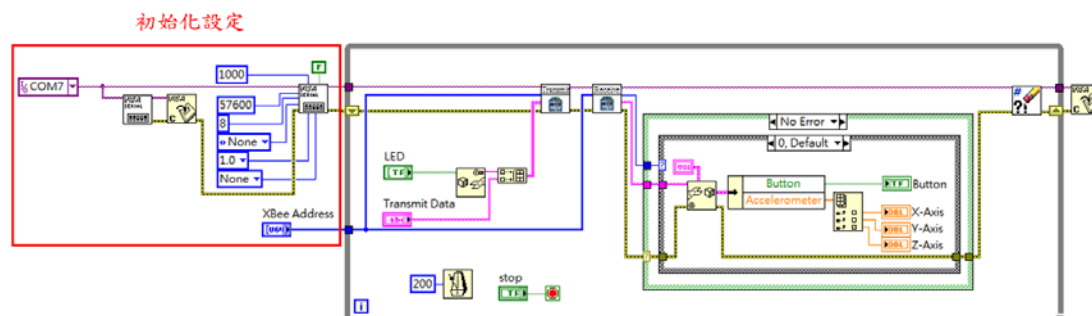


圖 2-14

XBeeNo.1_RT.vi 人機介面如下圖 2-15，由於都是讀取和控制 myRIO 上面的狀態，所以在此只需要在程式開始前輸入 Coordinator 的 Address 即可。

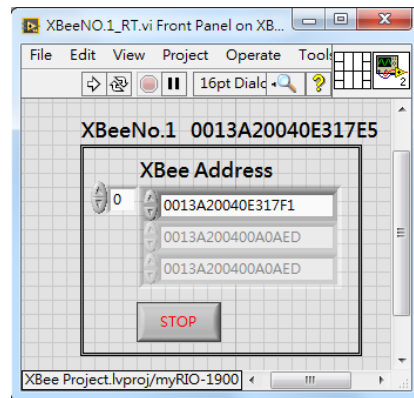


圖 2-15

XBeeNo.1_RT.vi 程式區如下圖 2-16，最左邊為同樣為初始化設定，要設定成 XBee No.1 的參數，右邊程式區則是直接透過 LabVIEW 內建的 myRIO Express VI 讀取 myRIO 的狀態，再透過 XBee 進行傳輸和接收。

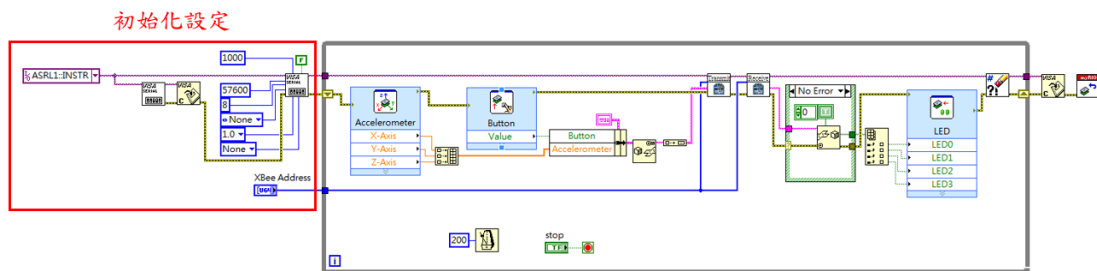


圖 2-16

XBeeNo.2_RT.vi 人機介面如下圖 2-17，由於是直接傳送和接收 String，所以提供給使用者讀取和接收的介面，另外只需要在程式開始前輸入 Coordinator 的 Address 即可。

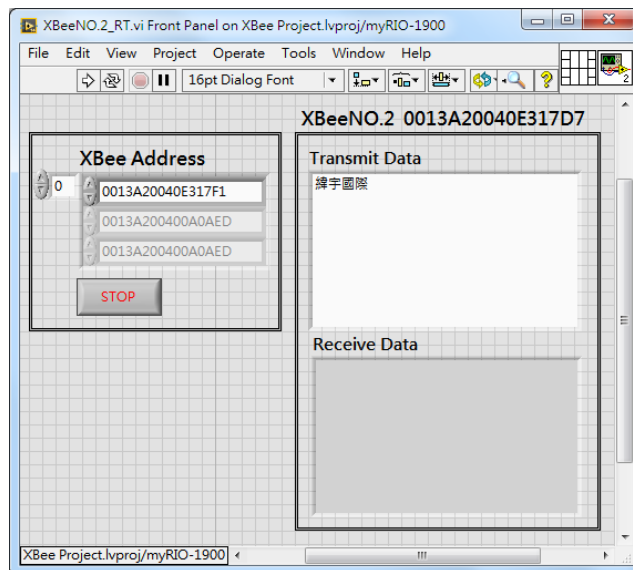


圖 2-17

XBeeNo.2_RT.vi 程式區如下圖 2-18，最左邊為同樣為初始化設定，要設定成 XBee No.2 的參數，右邊程式區則是直接將字串轉成陣列，再透過 XBee 進行傳輸和接收。

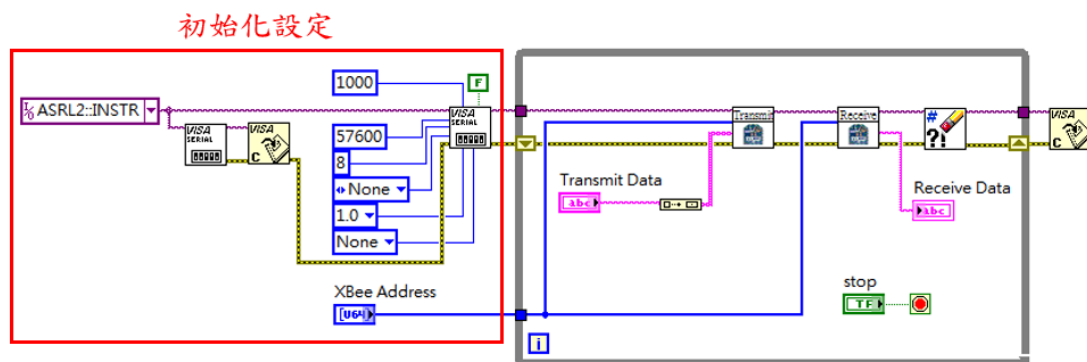


圖 2-18

完成了所有設定後，便可以將三支程式啟動，進行 XBee 單點對多點傳輸。使用者將來只需要使用 XBee 的 LabVIEW API，搭配 myRIO 的 XBee 模組便可以快速的完成一套無線通訊設備，不單是這樣，myRIO XBee 模組上面還保留了其餘的 AIO 和 DIO，方便使用者能夠進行其它量測，例如溫度、電壓、壓力等等，將 myRIO 化身成為一個遠端的量測設備，只需要透過一個主控端，就可以輕輕鬆鬆蒐集資料了。

資料參考來源:

Digi International <http://www.digi.com/lp/xbee/>

SparkFun Electronics <https://www.sparkfun.com/>

National Instruments <http://taiwan.ni.com/>

網昱多媒體 <http://swf.com.tw/>

此教材為緯宇國際有限公司開發，若有任何使用或散佈，請先告知。

若有任何侵權行為，緯宇國際有限公司保留一切法律追訴權。

若有任何疑問，請來電 02-2211-7824

Email:paul.liu@weft.com.tw

地址:台北市忠孝東路五段一號 12 樓