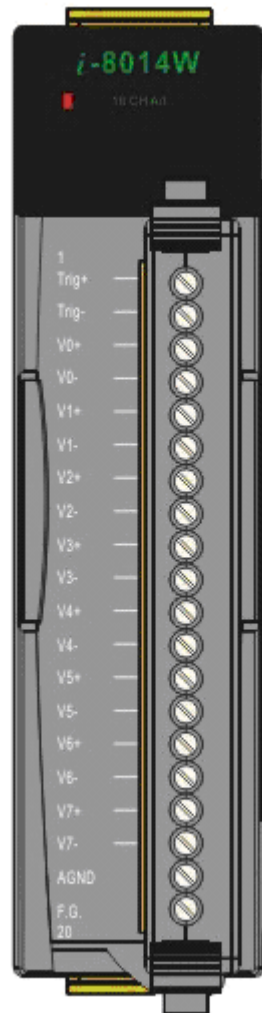


I-8014W 類比輸入模組

250 kS/s, 16 位元, 單點輸入/8 差動訊號輸入

使用手冊

發行版本 1.0.1/ July 2011



產品保固

ICP DAS (泓格科技股份有限公司) 所生產製造的產品自交貨日起提供一年的產品保固，此產品保固僅限於材料和製造上的瑕疵。

免責聲明

ICP DAS 有權在不通知的情況下隨時對本手冊進行變更，且相信本手冊的資訊是正確及可靠的，對於任何因產品使用或手冊使用上所造成的損壞及第三者損失賠償，ICP DAS 概不承擔責任。

版權宣告

版權所有 © 2011 泓格科技股份有限公司 保留所有權利。

註冊商標

本手冊中所提到之所有商標，均屬於其合法註冊公司所有。

聯絡我們

若您對產品有任何問題與建議，都歡迎您隨時與我們聯繫，我們將會盡快的回覆您。

E-mail: service@icpdas.com

目錄

前言	5
產品介紹	6
應用領域	11
規格說明	12
基本規格	12
硬體規格	13
腳位定義	13
內部架構	14
Jumper 設定	15
範例及快速入門	16
I-8014W 範例快速入門	18
其他範例	20
Magic.exe	22
Mag_ISR.exe	27
軟體開發及配置	29
I-8014W API	30
i8014W_Init	32
i8014W_GetFirmwareVer_L1	33
i8014W_GetFirmwareVer_L2	34
i8014W_GetLibVersion	35
i8014W_GetLibDate	36
i8014W_GetSingleEndJumper	37
i8014W_ReadGainOffset	38
i8014W_ReadAI	39

i8014W_ReadAIHex.....	40
i8014W_ConfigMagicScan	42
i8014W_StartMagicScan.....	45
i8014W_StopMagicScan	46
i8014W_ReadFIFO.....	47
i8014W_UnLockFIFO	49
i8014W_ClearFIFO	50
i8014W_InstallMagicScanISR.....	51
i8014W_UnInstallMagicScanISR.....	54
i8014W_ClearInt	55
疑難排解	57
手動校正	62

前言

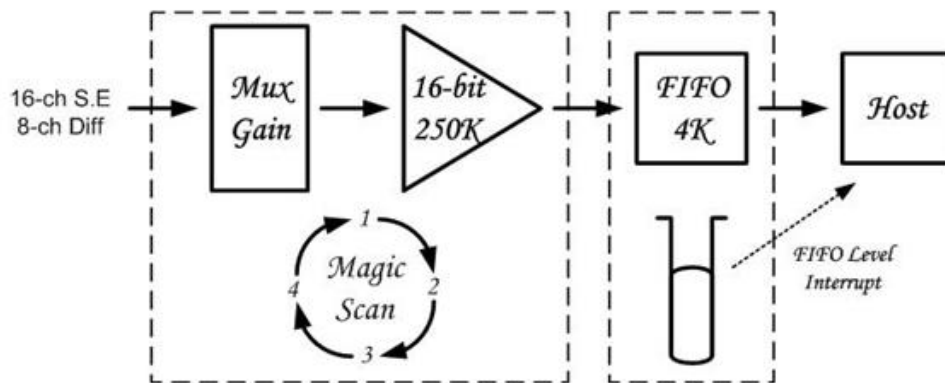
本使用手冊除了介紹一般基本硬體規格及 I-8014W 特有的 Magic Scan 的使用方式及應用外,在快速入門及範例這個章節希望透過實際範例程式的細節解說,讓使用者了解設定 Magic Scan 參數的流程與程式如何在 Magic Scan 過程中從硬體 FIFO 讀取 AD 資料.

另外疑難排解這個章節中會提到如何透過提供的工具程式及電池當作校正電壓源驗證 I-8014W 的精度是否有符合規格,也介紹如何調整接線方式或是透過調整組抗匹配的方式來解決實際訊號的跳動誤差.

I-8014W 出貨時都有經過詳細的驗證與校正程序,當出廠的校正參數與現場訊號沒辦法匹配時,通常是阻抗匹配不符,這時需要進行手動校正程序,手動校正這個章節介紹不同平台的校正式式及校正的操作相關細節介紹.

產品介紹

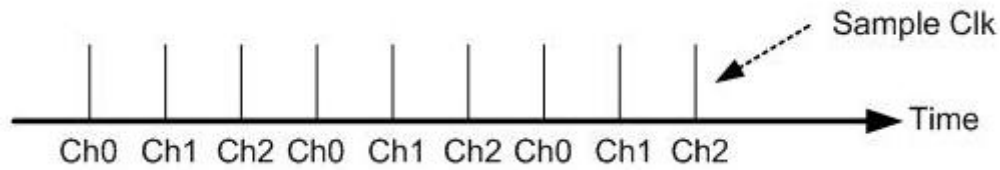
I-8014W 是一個具備 16-位元解析度及高速隔離型的類比輸入模組，它提供了 16 單端點訊號輸入或是 8 個差動訊號輸入通道，I-8014W 最重要特色是本身硬體能將 AD 資料掃描取樣後直接儲存於 4k 整數資料型態的 FIFO 空間，這種掃描方式我們稱之為 “Magic Scan” 。



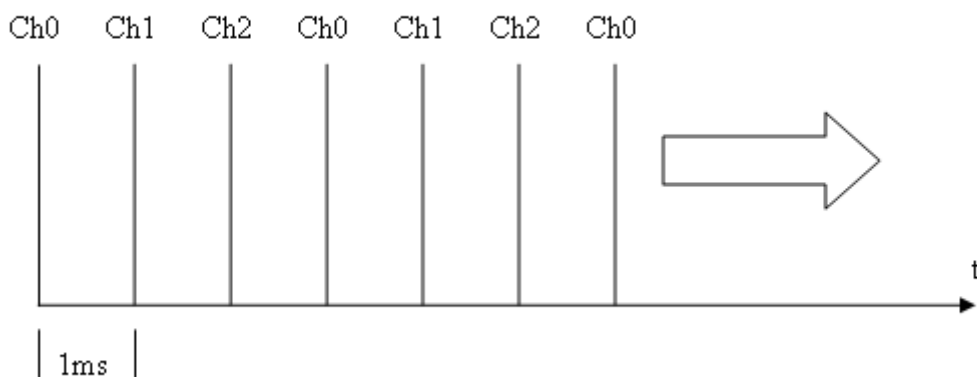
為什麼需要 Magic Scan:

I-8014W Magic Scan 有兩種掃描取樣方式，一種我們稱之為標準方式、另一種我們稱之為虛擬多通道同步取樣(Virtual Sample and hold)方式。

標準方式，每一個取樣時間只取樣一筆訊號，依序取樣。



如上圖所示，Magic Scan 裡面有規劃 Ch0, Ch1, Ch2 作為取樣三個通道，假設取樣頻率為 1k KHz 代表 Ch0 到 Ch1 需要 1 ms 的時間間距。三個通道掃描完需要 3 ms 的時間。



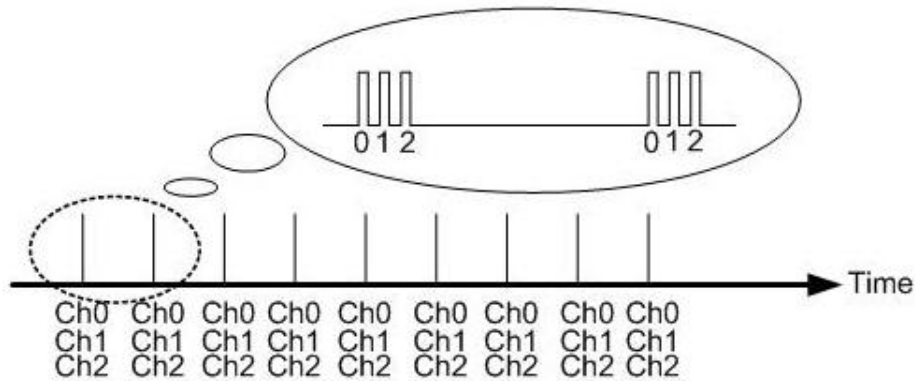
高速精確時間中斷 AD 取樣需求:

一般類比輸入模組的應用經常需要在精確的時間間隔對模組做 AD 訊號採集，要達到這個需要作業系統必須能夠提供精確的時間中斷的，讓程式能夠再很精準的時間間隔從模組的將 AD 訊號一個通道接著一個通道讀出來，但是多工系統因為是多工模式，很多沒有辦法提供穩定的時間中斷或是時間精度有所限制，例如時間中斷只能達到 1 ms 等級的精度或是更差的精度。要高達 1 kHz 以上高速的時間中斷服務更是困難。

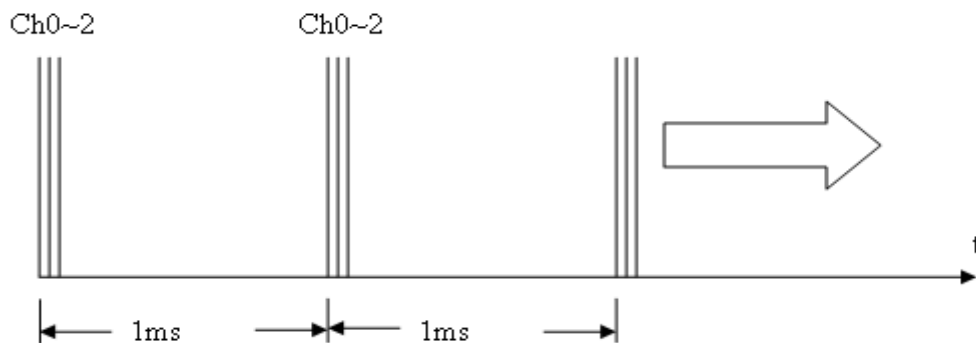
I-8014W Magic Scan 提供的標準掃描方式，可提供精確的取樣頻率逐一對每一個通道做資料採集，作業系統無須具備任何精確的時間中斷呼叫服務程式，直接由 I-8014W 硬體內部的時脈，提供 2 Hz~250 kHz 精確的取樣頻率。程式只需透過 API 將資料從 4k 16-位元 FIFO 儲存資料空間讀出即可。

多通道同步取樣需求:

當需要在固定採集頻率狀況下同時對好幾個通道做資料採集時,I-8014W 提供
虛擬多通道同步取樣 (Virtual Sample and hold)方式,每一個取樣時間內會將設
定好要取樣的通道全部取樣一次

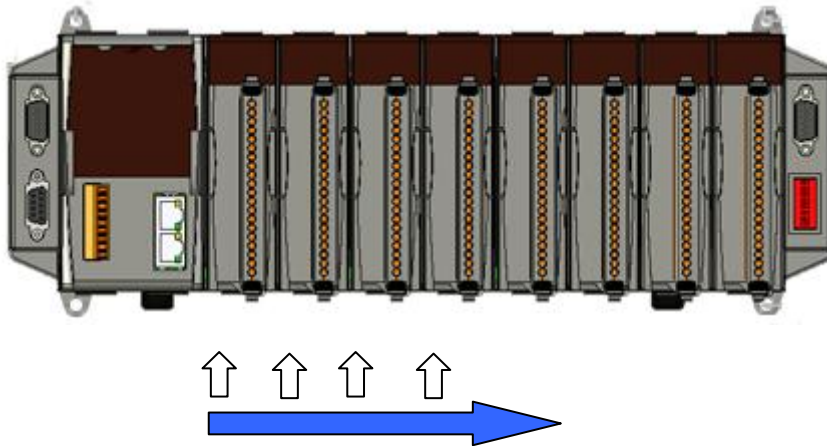


如上圖所示，Magic Scan 裡面有規劃 Ch0, Ch1, Ch2 作為取樣三個通道，假設
取樣頻率為 1 kHz，I-8014W 每 1 ms 會將 Ch0, Ch1, Ch2 取樣完畢，等下一個
1 ms 再進行下一次取樣。其中每通道之間的取樣速度為 250 kHz，把取樣後的
數據作成圖表會像下圖一般，每三點會幾乎的並排一起為聚落，而聚落的間距
都是 250 kHz。



內部命令觸發啟動 Magic Scan:

一般透過螢幕操作 I-8014W 都是使用 API 呼叫的方式啟動 Magic Scan 去做資料採集動作,如下圖所示,



程式按照順序每槽逐一下達 Start Magic Scan 指令

內部硬體中斷訊號觸發啟動 Magic Scan:

如果有好幾片要同時啟動 Magic Scan,透過 API 呼叫必須如上圖一樣,逐一對每一槽 I-8014W 下達命令.透過內部硬體中斷方式可以在幾乎同一個時間點啟動 Magic Scan 的動作.



程式透過硬體中斷訊號通知每一槽啟動 Magic Scan,.

外部硬體電氣訊號觸發啟動 Magic Scan:

有時候量測動作想透過外部硬體開關控制來啟動 Magic Scan,這時候 I-8014W 最前面兩個端子接點可用來作為觸發啟動 Magic Scan 的訊號源.當設定 Magic Scan 的參數時,如果觸發源選擇外部硬體觸發方式,這時候可以選擇電氣觸發的方式為上升緣訊號或下降緣訊號設定完成後,接著啟動 Magic Scan,此時 I-8014W 並不會馬上開啟做資料採集,而必須等到外部觸發訊號產生後,I-8014W 會開始作資料採集.



系統效能考量

當資料採集所需的頻率越高及通道數越多,系統越繁忙,如果應用程式還有資料庫應用或是其他複雜的資料運算要進行,系統的效能考量將越顯重要,如果按照傳統利用無窮迴圈一直從 FIFO 將資料讀出,這樣的作法不但沒有效率,對 CPU 負載更是一大負擔,當 CPU 負載過高時,往往會造成系統的不穩定.

而 I-8014W 內含有 AD 使用的 FIFO 達到 4 K samples 的容量。更在 FIFO 上發展一個新的技術。您可以設定 FIFO 的觸發準位、當取樣的筆數大於設定的 FIFO 準位。即刻會產生中斷訊號通知主機。這個技術讓資料採集程式能更有效率,也提供 CPU 更多時間做其他資料運算及資料儲存動作.

應用領域

- 高速型多通道資料擷取系統
- 振動分析

規格說明

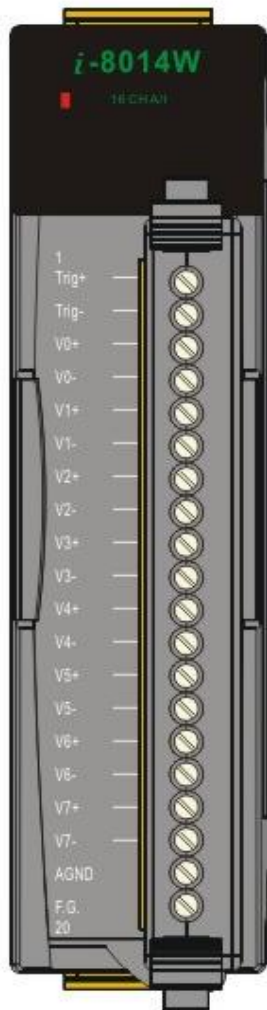
基本規格

輸入範圍	+/- 10 V, +/- 5 V, +/- 2.5 V, +/- 1.25 V -20 mA ~ +20 mA(需外接 125 歐姆精密 電阻)
解析度	16 位元
取樣頻率	單通道 Magic Scan 輪詢模式 : 250 KS/s
FIFO 緩衝區	8 k bytes (4k 整數資料)
精度	0.05 % of FSR +/- 1 LSB
資料讀取方式	Polling, Pacer (Magic Scan)
Magic Scan 取樣模式	模式 I:標準模式, 模式 II:虛擬同步取樣
過電壓保護範圍	+60 V ~ -45 V
輸入阻抗匹配選項	20 k, 200 k, 20 M (使用跨接器調整)
Intra-module Isolation, Field to Logic	2500 Vrms
LED 燈號顯示	1 電源顯示 LED 燈
耗電功率	2.5 W Max
操作溫度範圍	-25 ~ +75 °C
儲存溫度範圍	-30 ~ +85 °C
溼度	5 to 95 % RH, 未凝結狀態
尺寸大小(寬 x 長 x 高)	30 mm x 102 mm x 115 mm (W x L x H)
輸入範圍	+/- 10 V, +/- 5 V, +/- 2.5 V, +/- 1.25 V -20 mA ~ +20 mA(需外接 125 歐姆精密 電阻)

硬體規格

腳位定義

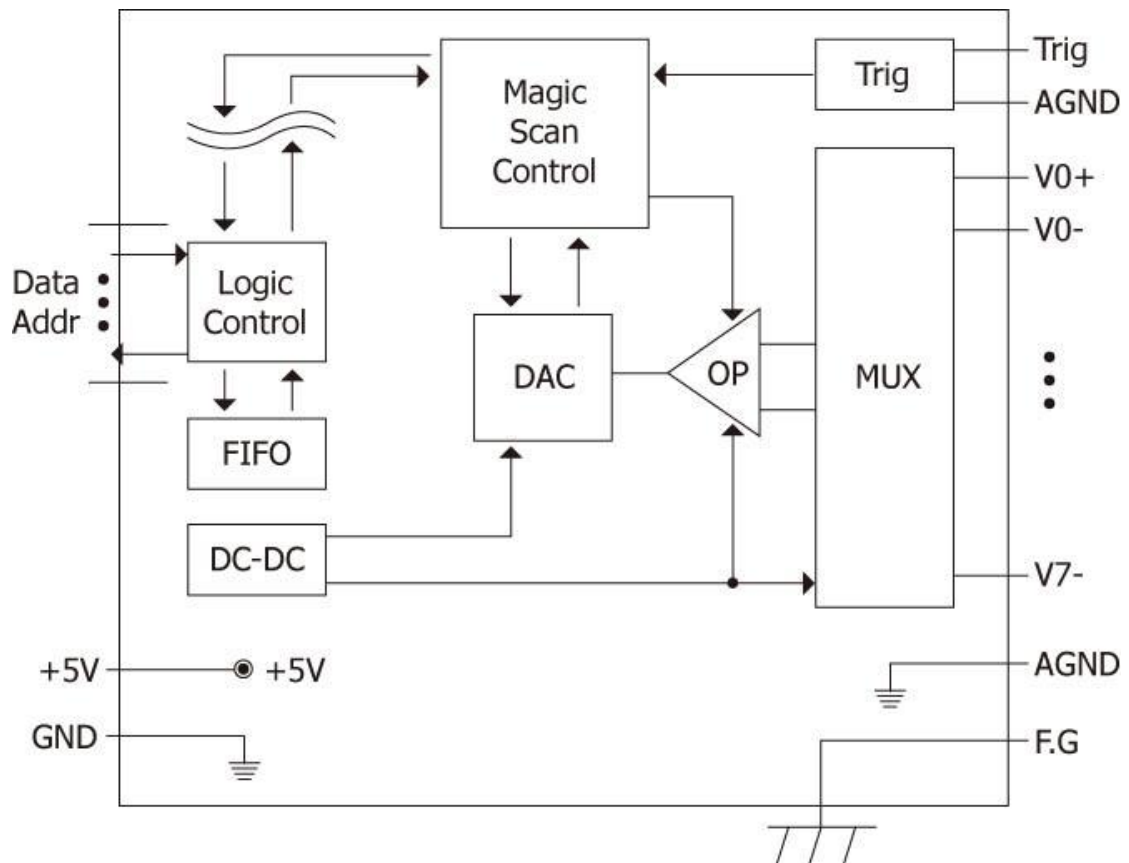
I-8014W 類比模組輸出入訊號腳位定義如下圖：



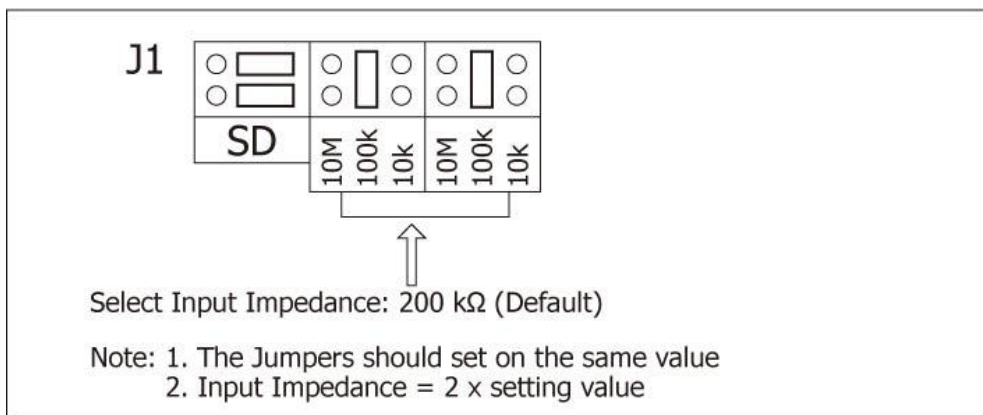
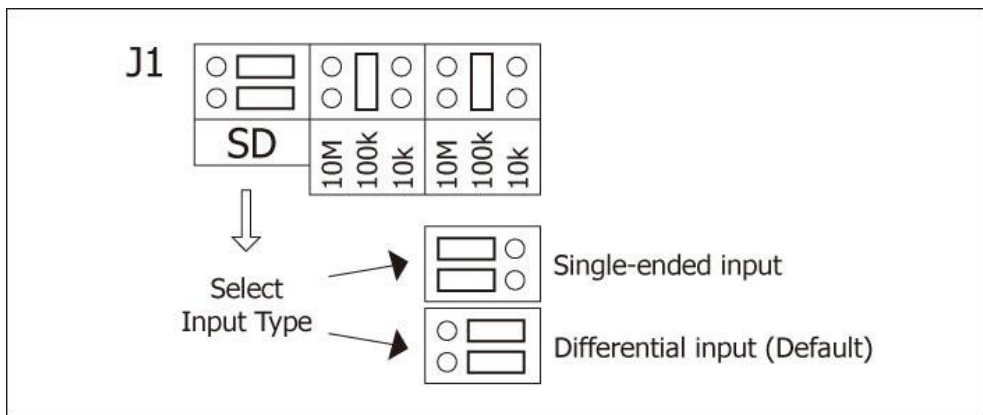
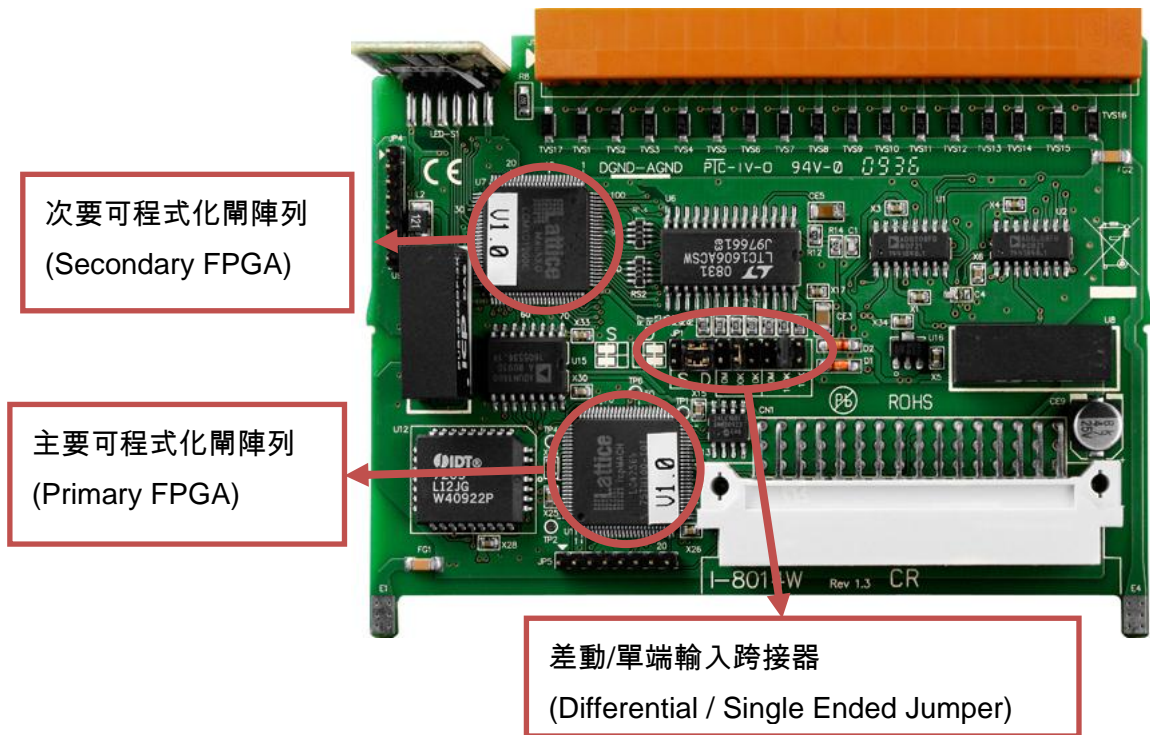
Terminal No.	Pin Assignment	
	Differential	Single-ended
01	Trig+	Trig
02	Trig-	AGND
03	V0+	V0
04	V0-	V8
05	V1+	V1
06	V1-	V9
07	V2+	V2
08	V2-	V10
09	V3+	V3
10	V3-	V11
11	V4+	V4
12	V4-	V12
13	V5+	V5
14	V5-	V13
15	V6+	V6
16	V6-	V14
17	V7+	V7
18	V7-	V15
19	AGND	AGND
20	F.G.	AGND

內部架構

I-8014W 類比模組輸出入訊號腳位定義如下圖：



Jumper 設定



範例及快速入門

I-8014W 範例包含快速入門的範例，以及其他可示範 Magic Scan 功能的範例。

Library 及範例程式

主機平台	主要型號分類	API 使用差異
I-8000 Lib/Demo 網址		ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/8000/841x881x/demo/lib/ ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/8000/841x881x/demo/io_in_slot/
I-8000 Lib/Demo 光碟路徑	CD:\Napdos\8000\841x881x\demo\Lib CD:\Napdos\8000\841x881x\demo\IO_in_Slot	
iPAC-8000 Lib/Demo 網址		ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/ipac8000/demo/basic/ip-84x1_ip-88x1/lib/ ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/ipac8000/demo/basic/ip-84x1_ip-88x1/io_in_slot/
iPAC-8000 Lib/Demo 光碟路徑	CD:\Napdos\iPAC8000\Demo\Basic\iP-84x1_iP-88x1\Lib CD:\Napdos\iPAC8000\Demo\Basic\iP-84x1_iP-88x1\IO_in_Slot	
Windows CE5 網址		ftp://ftp.icpdas.com/pub/cd/winpac/napdos/wp-8x4x_ce50/sdk/io_modules/ ftp://ftp.icpdas.com/pub/cd/winpac/napdos/wp-8x4x_ce50/demo/winpac/evc/pac_io/local/ ftp://ftp.icpdas.com/pub/cd/winpac/napdos/wp-8x4x_ce50/demo/winpac/dotnet/c%23.net/pac_io/local/
Windows CE5 光碟路徑	CD:\napdos\wp-8x4x_ce50\SDK\IO_Modules CD:\napdos\wp-8x4x_ce50\Demo\WinPAC\evc\PAC_IO\Local CD:\napdos\wp-8x4x_ce50\Demo\WinPAC\DOTNET\C#.NET\PAC_IO\Local	
Windows CE6 網址		ftp://ftp.icpdas.com/pub/cd/xp-8000-ce6/sdk/special_io/ ftp://ftp.icpdas.com/pub/cd/xp-8000-ce6/demo/xpac/vc2008/io/local/ ftp://ftp.icpdas.com/pub/cd/xp-8000-ce6/demo/xpac/c%23io/local/

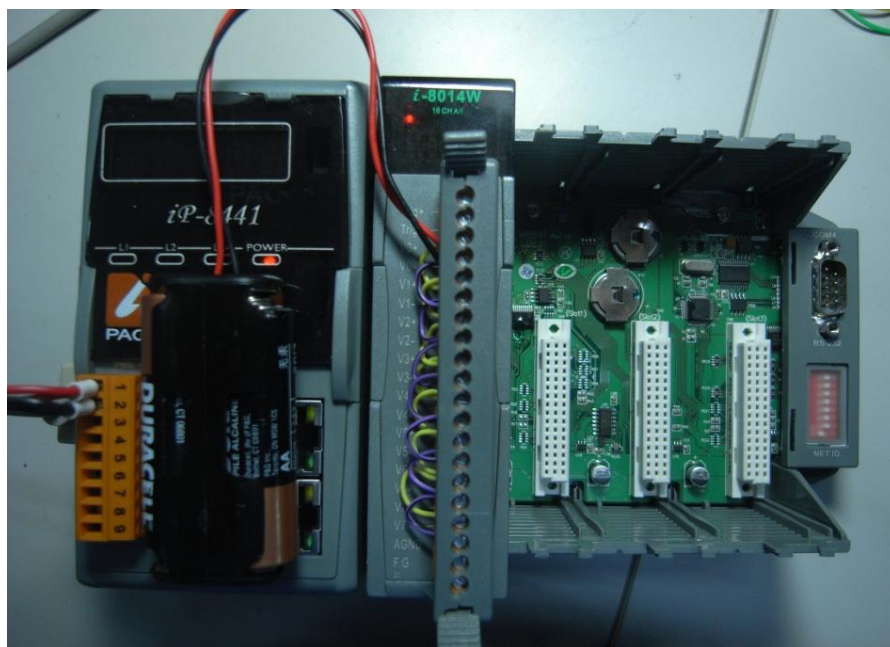
	ftp://ftp.icpdas.com/pub/cd/xpac-atom-ce6/sdk/special_io/ ftp://ftp.icpdas.com/pub/cd/xpac-atom-ce6/demo/xpac/vc2008/io/local/ ftp://ftp.icpdas.com/pub/cd/xpac-atom-ce6/demo/xpac/c%23/io/local/
Windows CE6 光碟路徑	CD:\XP-8000-CE6\SDK\Special_IO CD:\XP-8000-CE6\Demo\XPAC\VC2008\IO\Local CD:\XP-8000-CE6\Demo\XPAC\C#\IO\Local CD:\XPAC-ATOM-CE6\SDK\Special_IO CD:\XPAC-ATOM-CE6\Demo\XPAC\VC2008\IO\Local CD:\XPAC-ATOM-CE6\Demo\XPAC\C#\IO\Local
Windows Embedded Standard 網址	ftp://ftp.icpdas.com/pub/cd/xp-8000/sdk/io/ ftp://ftp.icpdas.com/pub/cd/xpac-atom/sdk/io/ ftp://ftp.icpdas.com/pub/cd/xp-8000/demo/specialized_io/ ftp://ftp.icpdas.com/pub/cd/xpac-atom/demo/specialized_io/
Windows Embedded Standard 光碟路徑	CD:\XP-8000\SDK\IO CD:\XP-8000\Demo\Specialized_IO CD:\XPAC-ATOM\SDK\IO CD:\XPAC-ATOM\Demo\Specialized_IO

I-8014W 範例快速入門

這裡將利用所提供的 I-8014W 範例 - AI_Info 檢測一般常用的 3/4 號電池當作入門範例，示範如何讀取 I-8014W 模組資訊及檢測 AI 讀取數據是否正確。

當 I-8014W 模組使用過程中發現數據有問題時，亦可使用此範例對 I-8014W 模組做基本檢測。

在使用此範例前，請先連接測試用的電池



AI_Info.exe 這個程式可以顯示出一些資訊，這些資訊可以幫助使用者來為他們軟體和硬體除錯：

1. Library 資訊 (版本和建立日期)
2. FPGA 版本
3. 獨立的 FPGA 版本
4. 檢查輸入模式的 Jumper 位置 → Single-Ended 或 Differential 模式
5. "+/-10 V", "+/-5 V", "+/-2.5 V", "+/-1.25 V", "+/-20 mA" 等輸入範圍的 Gain 和 Offset 校正值

i8014W_ReadAI 會用工程 (浮點) 格式來顯示輸入的信號，用 i8014W_ReadAI 可以使 AD 讀到工程格式已經校正過的資料，i8014W_ReadAIHex 則是讀到已校正的 16 bit 的 Hex(16 進制)資料。

這個範例主要讓剛開始入門的使用者確認模組的相關資訊及 AI 數據是否準確。當使用過程發現數據有問題，請使用這個範例對模組做基本測試檢查。

下圖為 MiniOS7 主機平台的測試結果：

This demo show how to use i8014W_ReadAI to read hex and float format analog input data. There is an i8014 at slot 0

```

*****
Primary FPGA Version =: 0001
Secondary FPGA Version =: 0002
Library Version =: 1005
Build Date =: Jul 20 2010
*****
i8014W Input Mode=Differential

Select 0 : +/-10U
Select 1 : +/-5U
Select 2 : +/-2.5U
Select 3 : +/-1.25U
Select 4 : +/-20mA
Select Gain (<0~4):0
Select Gain[0]=+/-10U ,the Calibrated Gain= 32833, Calibrated Offset= -39

[00]=[2.6645]
[01]=[2.6642]
[02]=[2.6639]
[03]=[2.6639]
[04]=[2.6642]
[05]=[2.6639]
[06]=[2.6642]
[07]=[2.6642]

```

Annotations:

- Primary FPGA Version =: 0001, Secondary FPGA Version =: 0002, Library Version =: 1005, Build Date =: Jul 20 2010: 比對相關版本訊息
- i8014W Input Mode=Differential: 偵測差動/單端輸入跨接器位置
- Select Gain (<0~4):0, Select Gain[0]=+/-10U ,the Calibrated Gain= 32833, Calibrated Offset= -39: Gain 值在30000 上下,差距太大或顯示 65535 表示校正參數有問題
- [00]=[2.6645] to [07]=[2.6642]: AI 數據檢測是否準確
空的通道請用電線短接,可避免數值會跟隨其他通道任意跳動

Form1

I-8014W slot Index: Slot 0, Firmware 1: 1, Firmware 2: 2

Library Version: 1007, Single-Ended/ Differential: Differential

Gain Value: 32833, Offset Value: -39

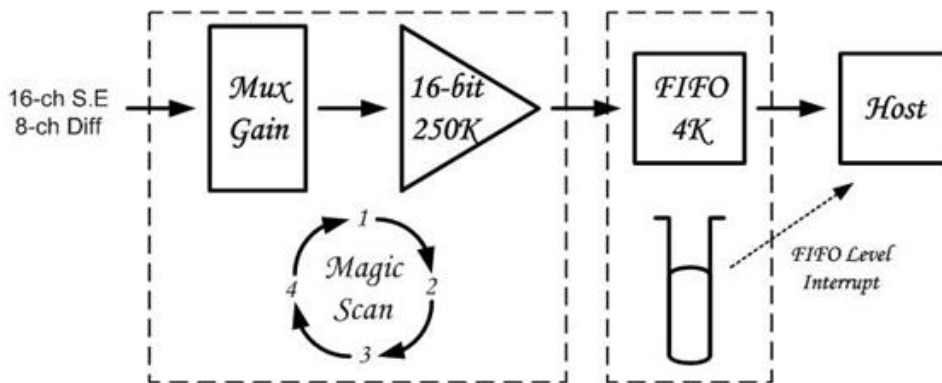
CH:0 Gain[1] 2.664, CH:1 Gain[1] 2.664, CH:2 Gain[1] 2.664, CH:3 Gain[1] 2.664, CH:4 Gain[1] 2.663, CH:5 Gain[1] 2.664, CH:6 Gain[1] 2.664, CH:7 Gain[1] 2.664

Annotations:

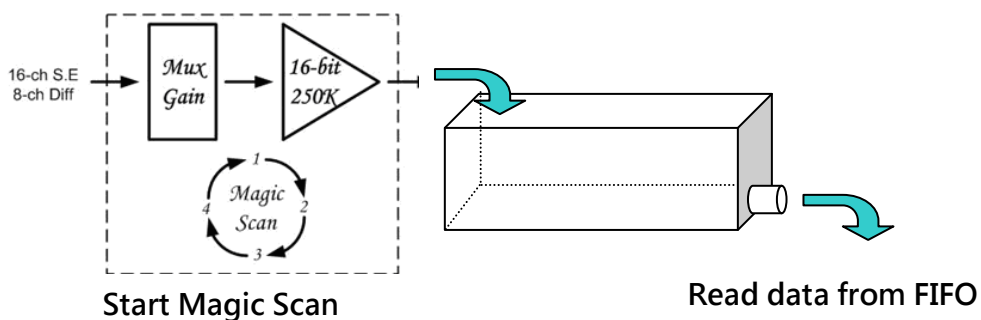
- I-8014W slot Index: Slot 0, Firmware 1: 1, Firmware 2: 2: 比對相關版本訊息
- Library Version: 1007, Single-Ended/ Differential: Differential: 偵測差動/單端輸入跨接器位置
- Gain Value: 32833, Offset Value: -39: Gain 值在30000 上下,差距太大或顯示 65535 表示校正參數有問題
- CH:0 Gain[1] 2.664 to CH:7 Gain[1] 2.664: AI 數據檢測是否準確
空的通道請用電線短接,可避免數值會跟隨其他通道任意跳動

其他範例

這裡將介紹兩個 I-8014W 模組主要功能 Magic Scan 的基本範例，示範如何設定 Magic Scan 去讀取 FIFO 的資料。在介紹 Magic Scan 相關範例前，這裡先介紹 I-8014W FIFO 的運作方式及注意事項。



上圖表示 I-8014W 具備 4 k 16-位元 FIFO，可存放 4096 筆整數資料，當使用 Magic Scan 取樣時，資料會按照設定的取樣頻率將設定取樣通道的數值不斷的往 FIFO 放置，如果任由資料一直往 FIFO 堆積到達極限 4096 筆 16-位元資料，FIFO 將被鎖死，資料就無法再更新到 FIFO 內，直到程式下命令將 FIFO 清空，所以過程中必須不斷的將資料從 FIFO 讀出，讓 FIFO 保持適當的空間。FIFO 的控制類似下圖水位控制一般，如果進水大於出水則水箱將溢出水箱。

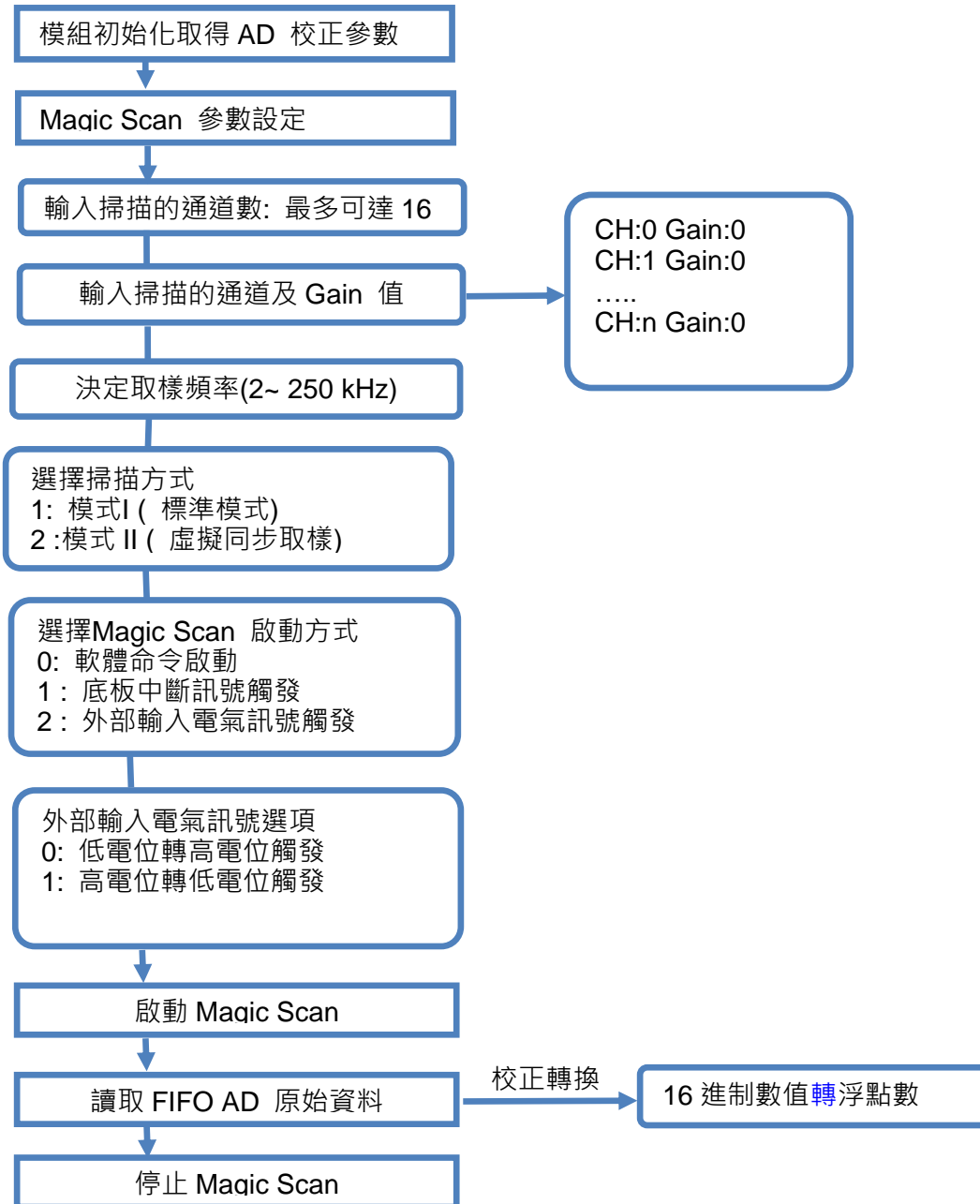


所以使用 Magic Scan 做資料採集時必須考量所設定的頻率與程式從 FIFO 將資料讀出來處理的速度之間取得平衡。

介紹完下面兩個 Magic Scan 基本範例，後面會介紹一個常用的應用範例。

這兩個範例大致相同，最主要的差異在於一個是不斷透過 i8014W_ReadFIFO 去將 I-8014W 的 FIFO 將資料讀出來。另一個則是透過 I-8014W 內部控制 FIFO 資料量的中斷機制，透過底板中斷通知 CPU 直接呼叫安裝的中斷副程式。

Magic Scan 設定流程



Magic.exe

Magic.exe 範例介紹 I-8014W 主要功能 – Magic Scan，此範例會一步一步的引導您輸入 Magic Scan 相關設定，最後還會說明如何從 Magic Scan 的過程中將資料從 FIFO 讀出來。

下圖為 Magic.exe 範例在 MiniOS7 主機平台的操作程序和顯示結果

This Demo will show how to use magic scan function to read analog input

```
Search I-8014W ....
There is an i8014 at slot 0
i8014W Input Mode=Differential and can have maximum 8 analog input

Input all i8014W_ConfigMagicScan parameters :
Step 1: Define scanned channel counts for magic scan:
Input scanned channel counts (1~16) :4
Now we have scanned channel counts = 4

Step 2: Define 4 elements for channel and gain array
The Gain definition of I-8014W
Select 0 : +/-10V
Select 1 : +/-5V
Select 2 : +/-2.5V
Select 3 : +/-1.25V
Select 4 : +/-20mA

Differential Mode range : channel 0 ~ 7
Select which Channel of Arr[0] (0~7) :0
Select which Gain of Arr[0] (0~4):0
Select which Channel of Arr[1] (0~7) :1
Select which Gain of Arr[1] (0~4):0
Select which Channel of Arr[2] (0~7) :2
Select which Gain of Arr[2] (0~4):0
Select which Channel of Arr[3] (0~7) :3
Select which Gain of Arr[3] (0~4):0

Step 3: Define Sample Rate of I-8014W
Input Sample rate of 8014W (1~2500000) :200
Note: the real sample rate may not the same as user input
the function i8014W_ConfigMagicScan return code is the
real sample rate accepted by I-8014W

Step 4: Select Scan Mode of I-8014W:
Scan Mode 1= M1 Standart Mode
Scan Mode 2= M2 Sample and Hold Mode
Input Scan Mode of 8014W (1 or 2) :1

Step 5: Select Trigger Source of I-8014W,
I-8014W can have 3 types of trigger source
trigger source 0= Software Command
trigger source 1= Internal Interrupt Signal
trigger source 2= External Trigger Signal
Input trigger source of 8014W (0~2) :0

Step6: Select Trigger State of I-8014W if select external
Not external trigger source, trigger state =0

The Magic Scan Configurations of I-8014W are:
Scan channel count = 4
CH[0]= 0      Gain[0]= 0 ( +/-10V )
CH[1]= 1      Gain[1]= 0 ( +/-10V )
CH[2]= 2      Gain[2]= 0 ( +/-10V )
CH[3]= 3      Gain[3]= 0 ( +/-10V )
Scan Mode = 1 ( Standard Mode )
Trigger Source = 0 ( Software Command )
Trigger State = 0 ( Normal for External Trigger Signal )
Set Sample Rate = 200.000 Real Sample Rate = 200.000
Press any Key to Start magic scan
```

步驟一
輸入掃描通道數
最小 1，最多 16

步驟二
選擇通道號碼及Gain 值
通道可重複也可不按照順序
每個的通道的Gain 也可以不相同

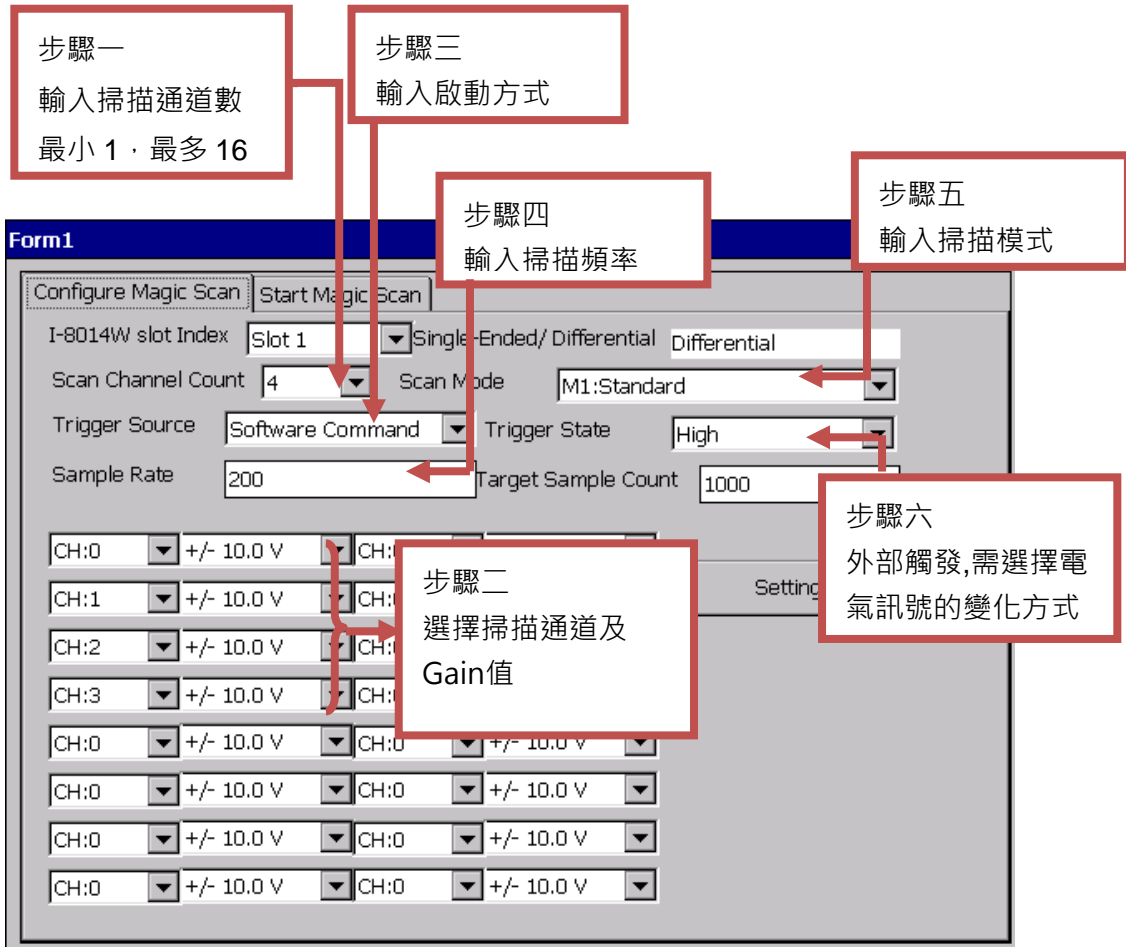
步驟三
輸入掃描頻率

步驟四
輸入掃描模式

步驟五
輸入啟動方式

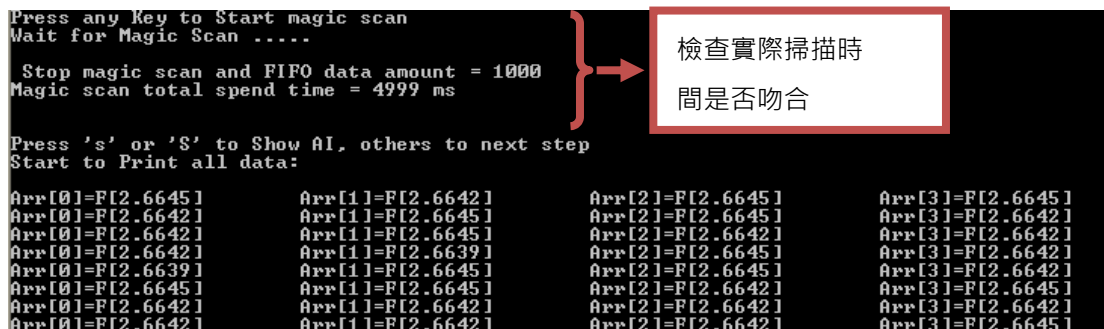
步驟六
外部觸發,需選擇電氣訊號的變化方式

下圖為 Magic.exe 範例在 WinCE 和 XPE 主機平台的操作程序和顯示結果

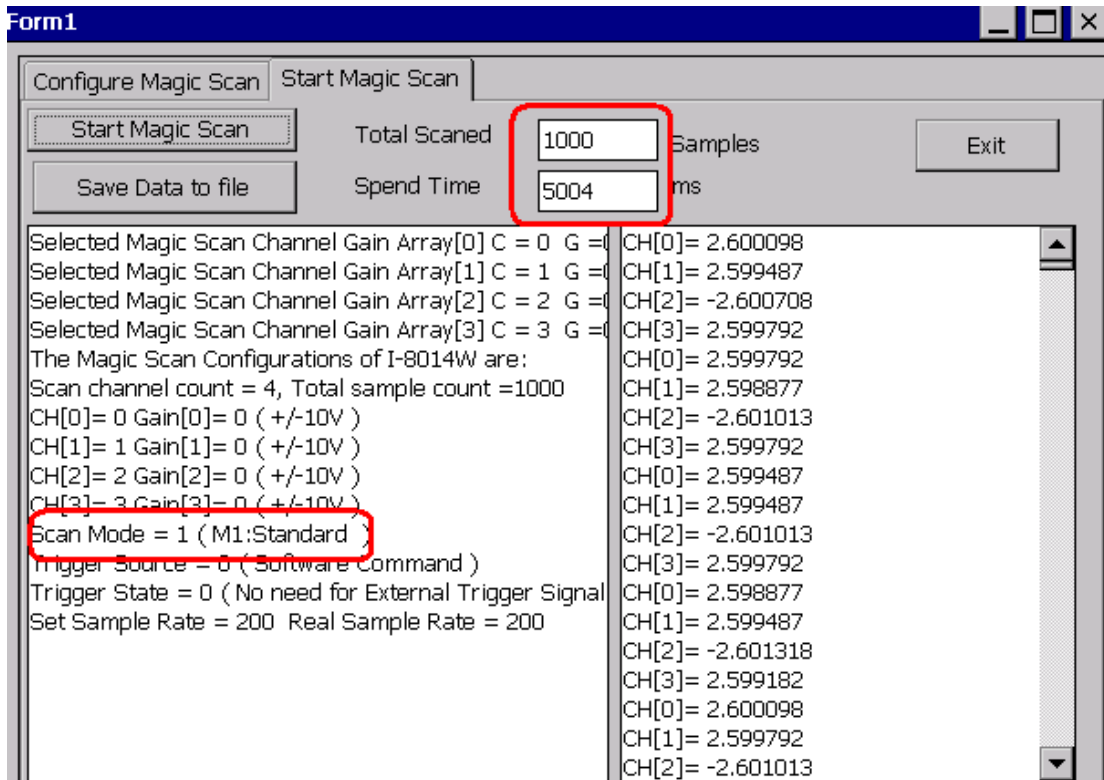


完成 Magic Scan 的參數設定，會整理出設定清單，其中掃描頻率這個項目須特別說明，I-8014W Magic Scan 的掃描頻率為並非連續性也就是說步驟三輸入的頻率跟實際 I-8014W 內部取樣的頻率可能會不相同，例如上圖的範例 200 Hz，經過實際運算 I-8014W 可以產生 200 Hz 的掃描頻率，如果輸入頻率為 2000 Hz 則實際掃描頻率是 1984.127 Hz，這點在實際應用上需注意。

下圖為 Magic.exe 範例在 MiniOS7 主機平台的顯示結果



下圖為 Magic.exe 範例在 WP-8000/ XP-8000/ XP-8000-CE6 平台的顯示結果

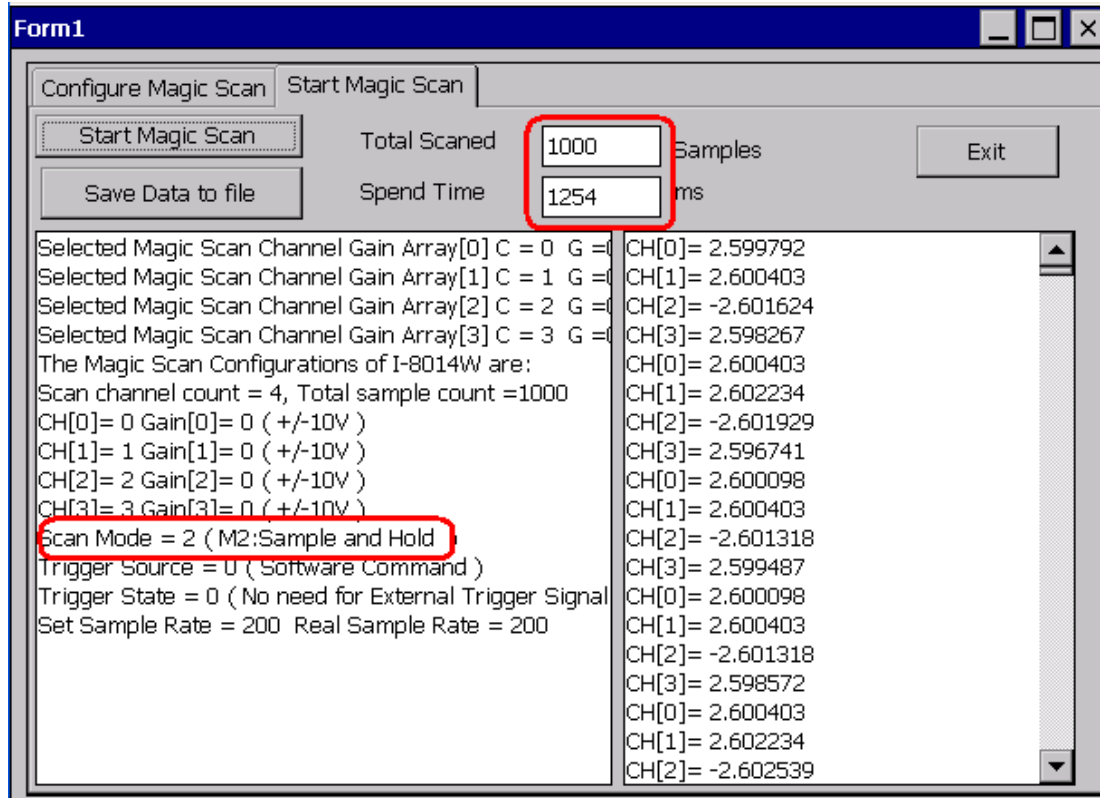


上述範例設定 200 Hz，掃描方式為標準模式就是 1 個通道需要 5 ms，掃描 1000 筆需要 5000 ms，所以時間上與實際設定的完全吻合，最後驗證數據是否正確。

假設取樣頻率為 200 Hz，取樣通道為 4 個，每通道取樣 250 筆，總取樣數為 1000 筆，採用標準模式 (模式 I)。

i8014W_ReadFIFO 的動作，函式進去後直接問 I-8014W FIFO 裡面目前有幾筆資料，按照上述條件每 5 ms 才會產生 1 筆資料，5 ms 中間有充裕的時間可以去處理其他事情。

如果上述範例掃描方式改為虛擬多通道同步取樣(模式 II), 假設取樣頻率為 200 Hz, 取樣通道為 4 個, 每通道取樣 250 筆, 總取樣數為 1000 筆, 這時候每 5 ms 時間一到, 就會採集到 4 個通道的數值, 1000 筆資料所需要的採集時間約為 $5 \times 250 = 1250 \text{ ms}$ 。



因為啟動 Magic Scan 後資料會一直往 FIFO 放置，所以讀完規劃的筆數後必須及時將 Magic Scan 停下來，這個是避免 Magic Scan FIFO 讀滿被鎖住。

若掃描頻率過快，則資料進入 FIFO 的速度將快於讀出的速度，最後 FIFO 會滿到極限，此時 FIFO 將被鎖死，資料無法再更新到 FIFO 內，但是 FIFO 內仍然會有之前留存下來的資料。

當發生 FIFO 被鎖死的狀態要如何處置？

1. 將 Magic Scan 停下來
2. 可以繼續將 FIFO 內的資料讀完或是清空 FIFO
3. 接著要解除 FIFO 鎖定的狀態
4. 繼續 Magic Scan 程序

Mag_ISR.exe

Mag_ISR.exe 這個範例為範例 Magic.exe 的進階應用，在 Magic.exe 範例是採用輪詢的方式不斷的去讀取 FIFO 的資料，當掃描頻率設定 200 Hz，掃描方式為標準模式就是 1 個通道需要 5 ms,如果用輪詢的方式讀取 FIFO,則 CPU 會浪費很多時間及資源在這個迴圈等待資料從 FIFO 出來。

此範例則教導如何使用 I-8014W 提供的 FIFO 觸發中斷，程式不需一直去詢問 FIFO 裡面是否有資料，而是經由 FIFO 的資料觸發準位產生中斷，由 CPU 主動通知安裝的副程式將 FIFO 內的資料讀出。這樣程式就不用把時間浪費在一直詢問 FIFO 是否有資料,而能更有效率的去處理其他資料轉換儲存或傳送的工作。

首先按照上一個範例先設定好 Magic Scan 所需要的參數，然後在啟動 Magic Scan 之前必須多做一道安裝中斷副程式的動作，如下面兩行程式碼：

```
i8014W_InstallMagicScanISR(slotIndex,Slot_ISR,triggerLevel);  
i8014W_StartMagicScan(slotIndex);
```

triggerLevel 代表 I-8014W FIFO 裡面資料到達哪一個準位要對底板發出中斷訊號由 CPU 直接通知下面的中斷副程式將 FIFO 資料讀出來，請參閱下頁表例。

中斷副程式裡面需要特別注意的是每次進來一定要記得呼叫 i8014W_ClearInt 解除中斷，讓下次中斷發生時能夠再一次進來。

下面表列表示 triggerLevel 的數字與相對觸發中斷的 AD 資料量：

triggerLevel	觸發中斷的資料量
0	8
1	16
2	32
3	64
4	128
5	256
6	512
7	2048

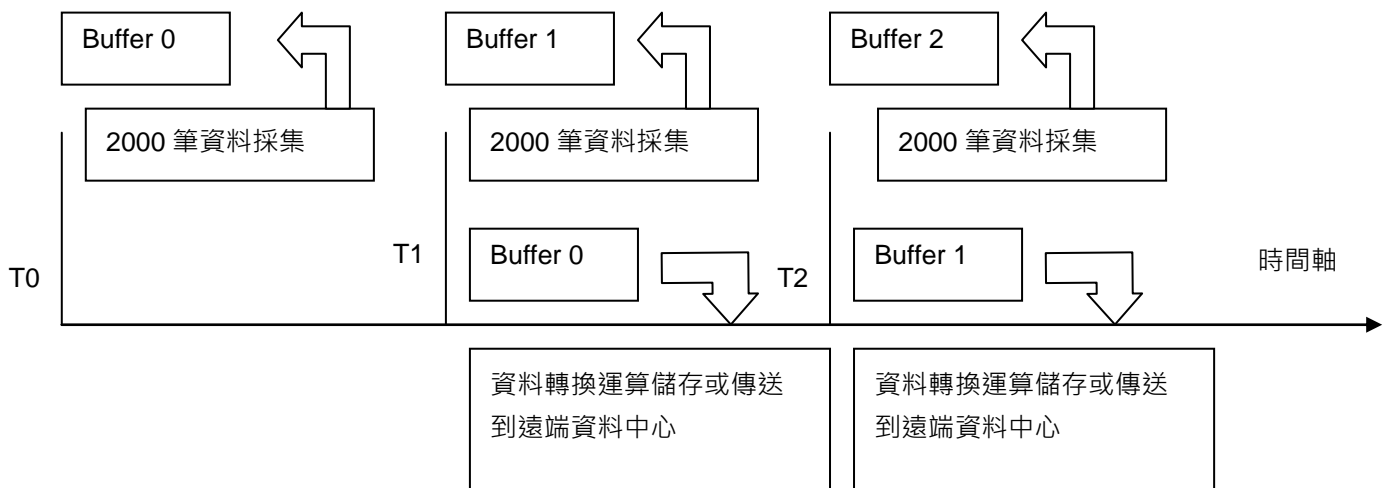
應用案例設計

假設有個案例

1. 需要量測 4 個通道 $\pm 10V$ 之間的差動電壓訊號
2. 每 5 ms 需要量測 4 個通道,每通道之間的時間差距能在 10us 以內
3. 每收集到 2000 筆的資料需透過 Ethernet 不斷的往資料中心傳送或存放到資料儲存裝置.

根據上述的條件

1. I-8014W 輸入訊號端需要調整為差動訊號模式(Differential Mode)
2. 輸入通道指定為 Ch0 ~ Ch3, Gain 值均為 0 ($\pm 10V$)
3. 取樣頻率需要定為 200Hz,同時必須採用 Magic Scan 模式 II 虛擬同步資料採集方式,這樣每個通道之間取樣時間差距為 4 us,符合 10 us 以內的需求.
4. 5 ms 需要量測 4 個通道,每收集 2000 筆資料大約需要 2500 ms
5. 如果是採用 MiniOS 系統的主機,浮點資料對系統運算或網路資料傳送都是很大的負擔,建議直接將採集到的原始資料直接傳送到 PC 儲存處理,如果是採用 WinCE 或 XPE 系統,浮點運算速度不會造成系統負擔,可先轉換成浮點數直接透過網路傳回 PC.
6. 從 FIFO 讀取資料時,建議多準備幾份 Buffer 陣列,交替循環使用,這樣系統有較多的緩衝時間處理資料傳送及儲存.



軟體開發及配置

I-8014W 軟體開發軟件提供開發 I-8014W 應用程式所需的元件和程式庫，以下為 I-8014W 開發軟件所支援各主機平台的型態：

I-8014W 可以使用在以下各種平台，不同平台 API 名稱的命名方式雖有些差異，從 API 完整名稱及參數說明可以了解功能他們的規則。所以本手冊僅針對從 MiniOS7 平台的使用方式介紹 API 的功能。

下表列出 I-8014W 模組在各不同平台之間的命名規則。

主機平台	主要型號分類	API 使用差異
MiniOS7	I-8000 系列 iPAC-8000 系列 VP-2000 系列	i8014W_API 名稱
Windows CE5 /Windows CE6 /Windows Embedded Standard (WES)	WP-8000 系列 WP-2000 系列 XP-8000-CE6 系列 XP-8000 系列	pac_i8014W_API 名稱
Linux		i8014W_API 名稱

I-8014W API

I-8014W 包含下列函數

函數	說明
i8014W_Init	這個函數可以初始化 I-8014W 模組以及檢查硬體的 ID
i8014W_GetFirmwareVer_L1	這個函數可以讀取 I-8014W 主要 FPGA 的韌體 (firmware) 版本
i8014W_GetFirmwareVer_L2	這個函數可以讀取 I-8014W 第 2 個 FPGA 的韌體 (firmware) 版本
i8014W_GetLibVersion	這個函數可以讀取 8014W.lib 的 library 版本
i8014W_GetLibDate	這個函數可以讀取 8014W.lib 的建立日期
i8014W_GetSingleEndJumper	可以獲得 I-8014W 的輸入模式。I-8014W 可以透過選擇 jumper 來選則輸入模式
i8014W_ReadGainOffset	獲得每個輸入範圍的已校正 Gain 和 Offset 參數
i8014W_ReadAI	從 I-8014W 獲得已校正完浮點格式 (十進制) 的輸入值
i8014W_ReadAIHex	從 I-8014W 獲得已校正完 Hex 格式 (十六進制) 的輸入值
i8014W_ConfigMagicScan	這個函數是 I-8014W 運作 Magic Scan 最重要的功能，在使用 Magic Scan 之前，程式需要用他來設定好 Magic Scan 所需的參數
i8014W_StartMagicScan	開始 Magic Scan 的功能，模組會開始將資料儲存到 FIFO 裡 (假如 i8014W_ConfigMagicScan 是使用外部觸發信號模式，模組不會馬上開始儲存資料，要等到準位信號已經觸發才會開始儲存資料)
i8014W_StopMagicScan	停止 Magic Scan 的功能以及停止儲存資料到 FIFO
i8014W_ReadFIFO	這個函數可以從 FIFO 來讀取資料以及讀取目前 FIFO 已有多少的資料量
i8014W_UnLockFIFO	這個函數可以打開鎖住的 FIFO (當 FIFO overflow 之後，FIFO 會被鎖住以及存滿資料，再下次使用 start Magic Scan 之前，需要打開鎖住的 FIFO 以及清除 FIFO)

i8014W_ClearFIFO	這個函數可以清除 FIFO 裡面全部的資料 (當 FIFO overflow 之後, FIFO 會被鎖住以及存滿資料, 再下次使用 start Magic Scan 之前, 需要打開鎖住的 FIFO 以及清除 FIFO)
i8014W_InstallMagicScanISR	這個函數可以使用 Magic Scan 去安裝一個 ISR。當 FIFO 的數量是高於觸發準位時, CPU 將會去執行 ISR
i8014W_UnInstallMagicScanISR	這個函數可以移除 Magic Scan ISR
i8014W_ClearInt	這個函數可以去清除 Magic Scan 的中斷信號

i8014W_Init

這個函數可以初始化 I-8014W 模組以及檢查硬體的 ID 。

假如輸入插槽的回應是 0 表示這個插槽的 ID 是 I-8014W 模組,假如回應是 -1, 表示這個插槽沒有 I8014W 模組。

語法

```
short i814W_Init ( int slot );
```

參數

slot : 0 ~ 7

回應值

請參考錯誤碼列表

註釋

在讀類比輸入資料之前，程式需要先使用 i8014W_Init(slot) 這個函數，使用這個參數時會將校正參數從 EEPROM 讀出來使用。

範例

[C/C++]

```
int slotIndex,err;
err=i8014W_Init(slotIndex);
if(err==0)
{
    Print("There is an I-8014W at slot %d\n",slotIndex);
}
else
{
    Print("There is no I-8014W at slot %d\n",slotIndex);
}
```


i8014W_GetFirmwareVer_L1

這個函數可以讀取 I-8014W 主要 FPGA 的韌體 (firmware) 版本。

語法

```
short i8014W_GetFirmwareVer_L1(int slot);
```

參數

slot : 0 ~ 7

回應值

I-8014W 主要 FPGA 的韌體版本

範例

[C++]

```
short ver_L1=0, slot=0;  
  
ver_L1= i8014W_GetFirmwareVer_L1 (slot);  
  
Print( "\nPrimary FPGA Version =:  
%04X",i8014W_GetFirmwareVer_L1(slot) );
```

i8014W_GetFirmwareVer_L2

這個函數可以讀取 I-8014W 第 2 個 FPGA 的韌體 (firmware) 版本。

語法

```
short i8014W_GetFirmwareVer_L2(int slot);
```

參數

slot : 0 ~ 7

回應值

I-8014W 第 2 個 FPGA 的韌體版本

範例

[C++]

```
short ver_L2=0, slot=0;  
  
ver_L2= i8014W_GetFirmwareVer_L2 (slot);  
  
Print( "\nSecondary FPGA Version =:  
%04X",i8014W_GetFirmwareVer_L2(slot) );
```

i8014W_GetLibVersion

這個函數可以讀取 8014W.lib 的 library 版本。

語法

```
short i8014W_GetLibVersion();
```

參數

無

回應值

讀取 8014W.lib 的 library 版本

範例

[C++]

```
short version;  
version = i8014W_GetLibVersion();  
Print("\nLibrary Version =: %04X",i8014W_GetLibVersion());
```

i8014W_GetLibDate

這個函數可以讀取 8014W.lib 的建立日期。

語法

```
void i8014W_GetLibDate(char *LibDate);
```

參數

LibDate : library 建立日期的字串 buffer

回應值

8014W.lib 的建立日期

範例

[C++]

```
char libDate [32];  
  
i8014W_GetLibDate(libDate);  
Print("\nBuild Date =: %s",libDate);
```

i8014W_GetSingleEndJumper

可以獲得 I-8014W 的輸入模式。I-8014W 可以透過選擇 jumper 來選則輸入模式。

假如 jumper 是選擇 “Differential” 模式，最多可以使用 8 個輸入的通道。

假如 jumper 是選擇 “Single Ended” 模式，最多可以使用 16 個輸入的通道。

語法

```
short i8014W_GetSingleEndJumper(int slot);
```

參數

slot : 0 ~ 7

回應值

0 : Differential 模式

1 : Single Ended 模式

範例

[C++]

```
short jumper=0, maxCh=0;

jumper = i8014W_GetSingleEndJumper(slot);
if(jumper)
{
    maxCh=16;
    Print("i8014W Input Mode=Single-End\n\r");
}
else
{
    maxCh=8;
    Print("i8014W Input Mode=Differential\n\r");
}
```

i8014W_ReadGainOffset

獲得每個輸入範圍的已校正 Gain 和 Offset 參數。

I-8014W 有 5 種不同的輸入範圍：

0 : +/-10 V, 1 : +/-5 V, 2 : +/-2.5 V, 3 : +/-1.25 V, 4 : +/-20 mA

語法

```
void i8014W_ReadGainOffset
(
    int slot,
    int gain,
    unsigned short* gainValue,
    short* offsetValue
);
```

參數

slot : 0 ~ 7

gain : 0 ~ 4

*gainValue : 從 EERPOM 讀出/載入已校正完的 gain 值

*offsetValue : 從 EERPOM 讀出/載入已校正完的 offset 值

回應值

無

範例

[C++]

```
unsigned short gVal=0;
short oVal=0;
i8014W_ReadGainOffset(slot,gain,&gVal,&oVal);
Print("\nThe Gain and Offset for Calibration is
Gain=%u; Offset=%d",gVal,oVal);
```

i8014W_ReadAI

從 I-8014W 獲得已校正完浮點格式 (十進制) 的輸入值。

語法

```
short i8014W_ReadAI(int slot,int ch,int gain, float* fVal);
```

參數

slot : 0 ~ 7

ch : 0 ~ 2

gain : 0 ~ 4

*fVal : 已校正完浮點格式的輸入值

回應值

請參考錯誤碼列表。

範例

[C++]

```
int slot,ch,gain;
float fVal=0.0;

slot = 0;
gain = 0; // "+/-10V"
for(ch=0;ch<8;ch++)
{
    i8014W_ReadAI( slot, ch, gain, & fVal);
    Print("\n[%02d]= [ %05.4f ]",ch,,fVal);
}
```

i8014W_ReadAIHex

從 I-8014W 獲得已校正完 Hex 格式 (十六進制) 的輸入值。

語法

```
short i8014W_ReadAIHex(int slot,int ch,int gain, short* hVal);
```

參數

slot : 0 ~ 7

ch : 0 ~ 2

gain : 0 ~ 4

*hVal : 已校正完 Hex 格式的輸入值

回應值

請參考錯誤碼列表

範例

[C++]

```
int slot,ch,gain;  
short hVal=0.0;  
  
slot = 0;  
gain = 0; // "+/-10V"  
for(ch=0;ch<8;ch++)
```



```
{  
    i8014W_ReadAIHex( slot, ch, gain, & hVal);  
    Print("\n[%02d]= [ %04X ] ",ch,,hVal);  
}
```

i8014W_ConfigMagicScan

這個函數是 I-8014W 運作 Magic Scan 最重要的功能。在使用 Magic Scan 之前，程式需要用他來設定好 Magic Scan 所需的參數。

語法

```
float i8014W_ConfigMagicScan
(
    int slot,
    int chArr[],
    int gainArr[],
    int scanChCount,
    float sampleRate,
    int scanMode,
    int triggerSource,
    int triggerState
);
```

參數

slot : 0 ~ 7

chArr[] : 需取樣通道的陣列，最多可以有 16 個通道數量

gainArr[] : 設定每個通道的輸入範圍 (0 ~ 4)

scanChCount : Magic Scan 要執行的總共數量

sampleRate : 取樣率，1 ~ 250 kHz

scanMode :

1 → 標準模式 Mode

2 → Sample and Hold 模式

triggerSource :

- 0 → 軟體觸發
- 1 → 內部中斷訊號觸發
- 2 → 外部輸入訊號觸發

triggerState :

- 0 → 當外部輸入訊號是上升緣訊號
- 1 → 當外部輸入訊號是下降緣訊號

回應值

I-8014W 可接受的真實取樣率

範例

[C++]

```
int slot, chArr[16], gainArr[16], scanChCount;
float sampleRate, realsampleRate;
int scanMode, triggerSource, triggerState;
slot = 0;
chArr[0]=0; // element 0 assigned channel 0
chArr[1]=1;
...
chArr[15]=15; // element 15 assigned channel 15
gainArr[0]=0; // element 0 assigned Input range 0
gainArr[1]=1; // element 1 assigned Input range 1
...
gainArr[15]=4; // element 15 assigned Input range 4
scanChCount=1; //only sample chArr[0] (channel 0)
sampleRate=25000.0; //set Sample rate 25 KHz
scanMode=1; // use M1 standard mode
triggerSource=1; // use internal interrupt signal Mode
triggerState=0;

realsampleRate=i8014W_ConfigMagicScan(slotIndex,chArr,gainArr,s
canChCount, sampleRate, scanMode,triggerSource,triggerState);
Print ("Set Sample Rate = %6.3f Real Sample Rate = %6.3f
\n",sampleRate, realsampleRate);
```

```
i804W_StartMagicScan(slot);  
...  
i8014W_ReadFIFO();  
// Detail reviews i8014W_ReadFIFO section
```

i8014W_StartMagicScan

開始 Magic Scan 的功能，模組會開始將資料儲存到 FIFO 裡 (假如 i8014W_ConfigMagicScan 是使用外部觸發信號模式，模組不會馬上開始儲存資料，要等到準位信號已經觸發才會開始儲存資料)。

語法

```
short i804W_StartMagicScan (int slot);
```

參數

slot : 0 ~ 7

回應值

請參考錯誤碼列表

範例

[C++]

```
int slot;  
slot=0;  
i804W_StartMagicScan(slot);
```

i8014W_StopMagicScan

停止 Magic Scan 的功能以及停止儲存資料到 FIFO。

語法

```
short i804W_StopMagicScan (int slot);
```

參數

slot : 0 ~ 7

回應值

請參考錯誤碼列表

範例

[C++]

```
int slot;  
slot = 0;  
i804W_StopMagicScan (slot);  
//Detail reviews i804W\_ReadFifo section
```

i8014W_ReadFIFO

這個函數可以從 FIFO 來讀取資料以及讀取目前 FIFO 已有多少的資料量。

語法

```
short i804W_ReadFIFO
(
    int slot,
    short hexData[],
    short readCount,
    short* dataCountFromFIFO
);
```

參數

slot : 0 ~ 7

hexData [] : 從 FIFO 讀取 Hex (十六進制) 格式的類比輸入資料陣列

readCount : 設定要從 FIFO 讀取資料的數量

* dataCountFromFIFO : 讀取 FIFO 裡面有多少資料量可以讀取

回應值

請參考錯誤碼列表。

範例

[C++]

```
int slot;
short hexData[8192];
long readCnt=0;
short totalScaned=0;
short TargetCnt=1000;
slot = 0;
i8014W_ReadFIFO(slot,hexData+totalScaned,
TargetCnt-totalScaned,&readCnt);
if(readCnt>0)
    totalScaned+=readCnt;
if(readCnt==MAX_FIFO || totalScaned>=TargetCnt)
{
    i8014W_StopMagicScan(slot);
    i8014W_UnLockFIFO(slot);
    i8014W_ClearFIFO(slot);
}
```


i8014W_UnLockFIFO

這個函數可以打開鎖住的 FIFO (當 FIFO overflow 之後，FIFO 會被鎖住以及存滿資料，再下次使用 start Magic Scan 之前，需要打開鎖住的 FIFO 以及清除 FIFO)。

語法

```
void i804W_UnLockFIFO (int slot);
```

參數

slot : 0 ~ 7

ch : 0 ~ 2

回應值

請參考錯誤碼列表

範例

[C++]

```
int slot;  
slot = 0;  
i804W_UnLockFIFO (slot);  
//Detail reviews i804W\_ReadFIFO section
```

i8014W_ClearFIFO

這個函數可以清除 FIFO 裡面全部的資料 (當 FIFO overflow 之後, FIFO 會被鎖住以及存滿資料, 再下次使用 start Magic Scan 之前, 需要打開鎖住的 FIFO 以及清除 FIFO)。

語法

```
void i804W_ClearFIFO (int slot);
```

參數

slot : 0 ~ 7

回應值

請參考錯誤碼列表

範例

[C++]

```
int slot;  
slot = 0;  
i804W_ClearFIFO (slot);  
//Detail reviews i804W\_ReadFIFO section
```

i8014W_InstallMagicScanISR

這個函數可以使用 Magic Scan 去安裝一個 ISR。當 FIFO 的數量是高於觸發準位時，CPU 將會去執行 ISR。

語法

```
void i804W_ InstallMagicScanISR  
(  
    int slot,  
    void (*isr)(int slot),  
    int triggerLevel  
);
```

參數

slot : 0 ~ 7

*isr (int slot) : 一個指向 ISR 中斷副程式的指標

triggerLevel : 0 ~ 7 (當值大於 7 時這個數值會等於 7)

下面表列表示 triggerLevel 的數字與相對觸發中斷的 AI 資料量：

triggerLevel	觸發中斷的資料量
0	8
1	16
2	32

3	64
4	128
5	256
6	512
7	2048

回應值

請參考錯誤碼列表

範例

[C++]

```
void main()
{
    int slot,TrgLevel;
    slot = 0;
    TrgLevel=100;
    i8014W_Install_MagicScanISR(slot,ISRFUN, TrgLevel);
    i8014W_ConfigMagicScan(...);
    // Detail reviews i8014W_ConfigMagicScan section
    i8014W_StartMagicScan(slot);
    ...
    while(1)
    {
        if(IntCnt>1)
        {
            i8014W_UnInstall_MagicScanISR(slot);
            break;
        }
    }
    ...
}
```

```
}  
  
void ISRFUN(int slot);  
{  
    Int IntCnt=0;  
    IntCnt++;  
    ret=i8014W_ReadFIFO(slot, hexData+totalScaned,  
    TargetCnt-totalScaned,&readCnt);  
    if(readCnt >0)  
        {  
            totalScaned+=readCnt;  
            printCom1("TotalScaned= %d\n\r",totalScaned);  
            totalRead+=readCnt;  
        }  
    i8014W_ClearInt(slot);  
}
```

i8014W_UnInstallMagicScanISR

這個函數可以移除 Magic Scan ISR。

語法

```
void i804W_UnInstallMagicScanISR(int slot);
```

參數

slot : 0 ~ 7

回應值

請參考錯誤碼列表

範例

[C++]

```
int slot;  
slot = 0;  
i804W_UnInstallMagicScanISR (slot);  
// Detail reviews i8014W\_Install\_MagicScanISR section
```

i8014W_ClearInt

這個函數可以去清除 Magic Scan 的中斷信號。

語法

```
void i804W_ClearInt (int slot);
```

參數

slot : 0 ~ 7

回應值

請參考錯誤碼列表

範例

[C++]

```
int slot;  
slot = 0;i804W_StopMagicScan (slot);  
// Detail reviews i8014W\_Install\_MagicScanISR section
```

錯誤碼列表:

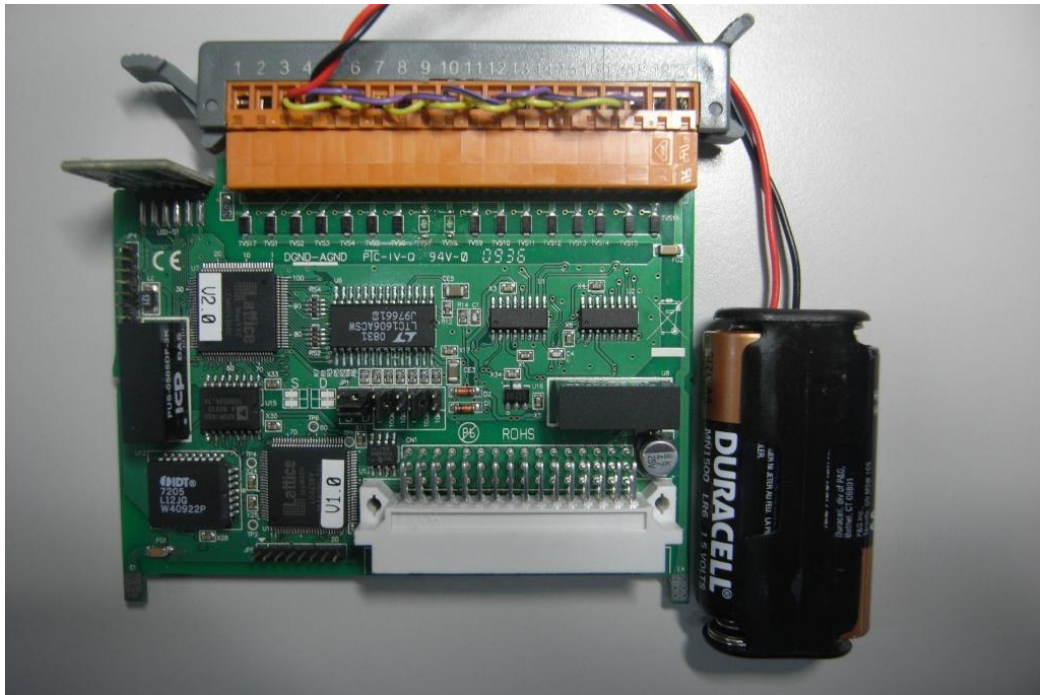
錯誤碼	錯誤碼定義	描述
0	NoError	No error.
-1	ID_ERROR	Check module's ID
-2	SLOT_ERROR	Slot index (0 ~ 7)
-3	CHANNEL_ERROR	Channel index (0 ~ 15)
-4	GAIN_ERROR	Gain (0 ~ 4)
-5	FIFO_EMPTY	No data in FIFO.
-6	FIFO_LATCHED	FIFO is full and be latched
-7	FIFO_OVERFLOW	FIFO is full
-8	TX_NOTREADY	Error between primary FPGA and secondary FPGA

疑難排解

本節 pac_i8014W_Utility.exe 程式可於各 Windows 平台 C#範例路徑下載

當懷疑 I-8014W 量測類比訊號是否準確可以按照下面步驟進行檢測。

1. 以下測試採用預設的差動訊號輸入方式 (Differential)，如果實際應用採用單點輸入方式，請按照單點輸入的接線方式操作。
2. 量測範圍使用預設的 $\pm 10V$ 。
3. 電壓源採用電池，接線方式如下：



註一：在使用差動訊號輸入方式 (Differential) 量測電壓時，有時候需要將 V-接腳接到 AGND，讓訊號更穩定。量測電流時則不能使用這個方式。

註二：量測電流時，需外 125 歐姆接精密電阻。

執行 pac_i8014W_Utility.exe, 驗證基本參數

步驟 1：選擇 I-8014W 所在的槽位

步驟 2：選擇 Basic Information 頁籤

步驟 3：儲存 Save I-8014W 版本及校正相關訊息

檔案名稱：Slot1_8014W_Info.txt

The screenshot shows a software window titled "Form1" with a dropdown menu for "I-8014W slot Index" set to "Slot 1". There are two tabs: "Basic Information" (selected) and "AI Test".

Fields in the "Basic Information" tab:

- Library Version: 1007
- Firmware 1: 1
- Firmware 2: 2

Buttons: "Refresh" and "Save".

Calibration parameters table:

Single-Ended/	Differential		
+/- 10V	Gain	32833	Offset -39
+/- 5V	Gain	32831	Offset -43
+/- 2.5V	Gain	32826	Offset -52
+/- 1.25V	Gain	32665	Offset -51
+/- 20mA	Gain	32826	Offset -52

開始進行量測前先檢查模組的校正參數是否有問題

正常的 Gain 值應該在 33000 上下，如果 Gain 值差太多或是出現 65535 的數字就是表示模組校正有問題，請按照下面步驟檢測：

步驟 1：按 Refresh 重新將校正參數讀出來

步驟 2：將 I-8014W 換到其他插槽試試看

驗證訊號量測數據：

I-8014W 訊號量測規格如下表：

量測範圍	0.05 % of FSR +/- 1 LSB
+/- 10 V	10mV ± 0.30518mV
+/- 5 V	5mV ± 0.15259mV
+/- 2.5 V	2.5mV ± 0.0763mV
+/- 1.25 V	1.25mV ± 0.03815mV
+/- 20 mA	0.02mA ± 0.0006 mA

切換到 AI Test 頁籤

選擇使用的量測範圍

輸入取樣次數

選擇顯示數據資料格式

按下 **Start** 開始測試

	First Data	Min Data	Max Data	Delta		First Data	Min Data	Max Data	Delta
C0	02.6645	02.6636	02.6651	00.0015	C8				
C1	02.6642	02.6636	02.6651	00.0015	C9				
C2	02.6642	02.6639	02.6648	00.0009	C10				
C3	02.6642	02.6639	02.6651	00.0012	C11				
C4	02.6642	02.6636	02.6651	00.0015	C12				
C5	02.6642	02.6639	02.6648	00.0009	C13				
C6	02.6642	02.6636	02.6651	00.0015	C14				
C7	02.6642	02.6639	02.6651	00.0012	C15				

取樣結束後，會將每的數據及採樣時間顯示出來。按下 **Save** 可將取樣的所有資料存成 `SampleData_Hex_mm_dd_hh_mim_sec.csv` 的資料格式。

若實際使用電池作為電壓源，量測結果還有問題，請將下面三點物件寄到 service@icpdas.com 信箱，由專人負責處理

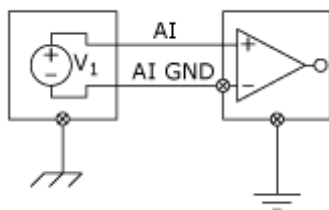
1. 實際的接線方式以數位相機拍照下來
2. Basic Information 頁籤所存的檔案
3. AI Test 所存的檔案

若量測電池訊號沒問題，但是接上實際訊號後仍然會有大的數據跳動，有可能是

1. 實際訊號上有雜訊干擾
2. 訊號來源本身的跳動
3. 浮接的訊號源 (訊號源與大地是浮接)

I-8014W 為一個高速量測的類比輸入模組，若訊號來源上有小於 250k Hz 的雜訊干擾，若是來源訊號本身即會跳動，I-8014W 都有可能真實呈現實際的數值，這時候應要改善訊號本身的跳動若是隔離雜訊的干擾。

當量測的訊號源為浮接的訊號源，建議將量測訊號的 V-的訊號與 I-8014W 的 AGND 連接在一起，如下圖所示 V1 為量測訊號。



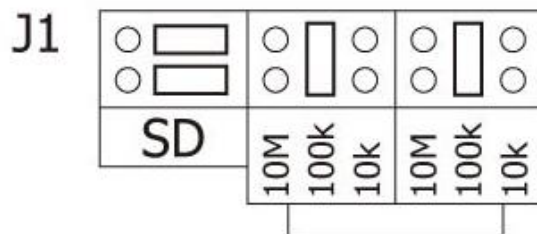
輸入阻抗的調整

I-8014W 本身提供了 20k、200k (預設值)、20M Ohm 的輸入阻抗設定

大部份的使用者使用預設值即可以正常的量測數值。

若使用預設值的跳動太大，有可能是因為量測的訊號與 I-8014W 的阻抗匹配的問題，可以嘗試將輸入阻抗調整成 20k Ohm，若是 20M Ohm。

變更輸入阻抗設定後，必須使用手動校正程式重新校正。



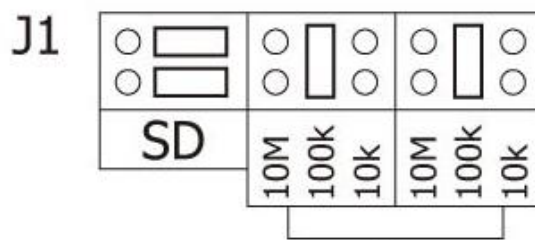
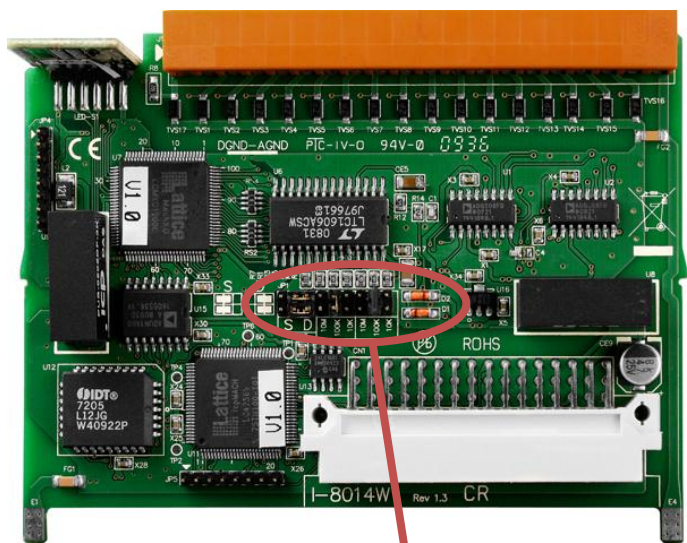
Select Input Impedance: 200 k Ω (Default)

Note: 1. The Jumpers should set on the same value
2. Input Impedance = 2 x setting value

手動校正

I-8014W 出貨時都有經過詳細的驗證與校正程序,當出廠的校正參數與現場訊號沒辦法匹配時才需要進行手動校正程序。

I-8014W 模組有輸入阻抗的跨接器 (Jumper), 可以選擇 3 種輸入阻抗(10k、100k 和 10M)的其中一種,預設值是使用 100k 這段,並且在出廠之前模組都已經校正完.當模組不準確或是要使用其他的輸入阻抗時,需要做重新校正,假如沒有重新校正,模組將會失去準確性。



Select Input Impedance: 200 k Ω (Default)

Note: 1. The Jumpers should set on the same value
2. Input Impedance = 2 x setting value

校正時所須要的配備：

1. 模組：I-8014W
2. 控制器：i-8000/iPAC-8000/ WP-8000/ XP-8000/ XP-8000-CE6
3. 穩定的電源。例如，電源供應器、電池等，但是不要使用不穩定的電源。
4. 量測電表：4 又 1/2 位半 (15 bit 解析度) 或是更好的量測電表。

註：較好的解析度的量測電表和穩定的電源可以獲得較好的校正結果。



硬體配備

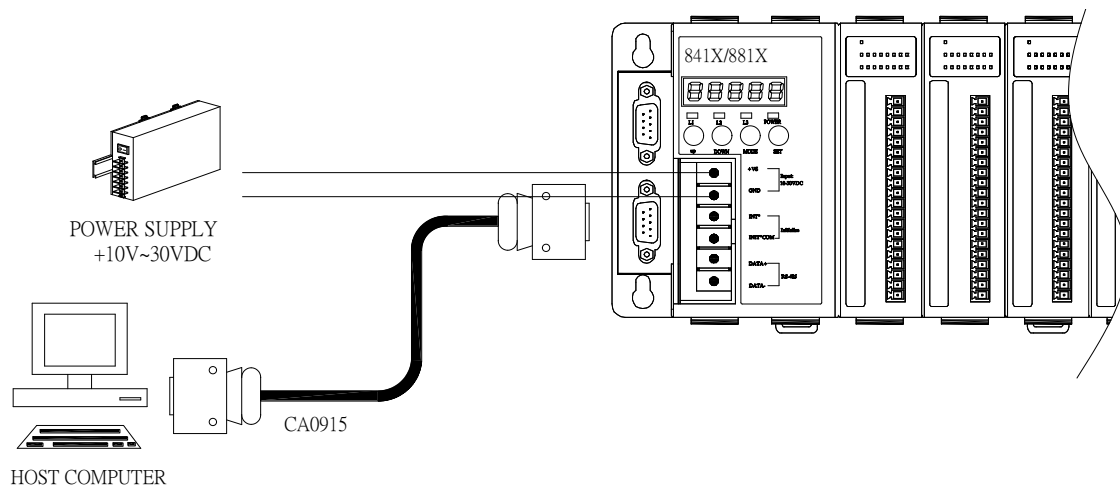
5. 校正式式：

主機	名稱	檔案位置
I-8000	8014cal.exe,	與範例程式同目錄
iP-8000	iP_8014cal.exe	與範例程式同目錄
WP-8000/ XP-8000/ XP-8000-CE6	pac_i8014W_Calibration.exe	與 C# 範例程式同目錄

6. utility：MiniOS7 Utility (i-8000/IPAC-8000 更新檔案用)
(<http://www.icpdas.com/download/minios7.htm>)

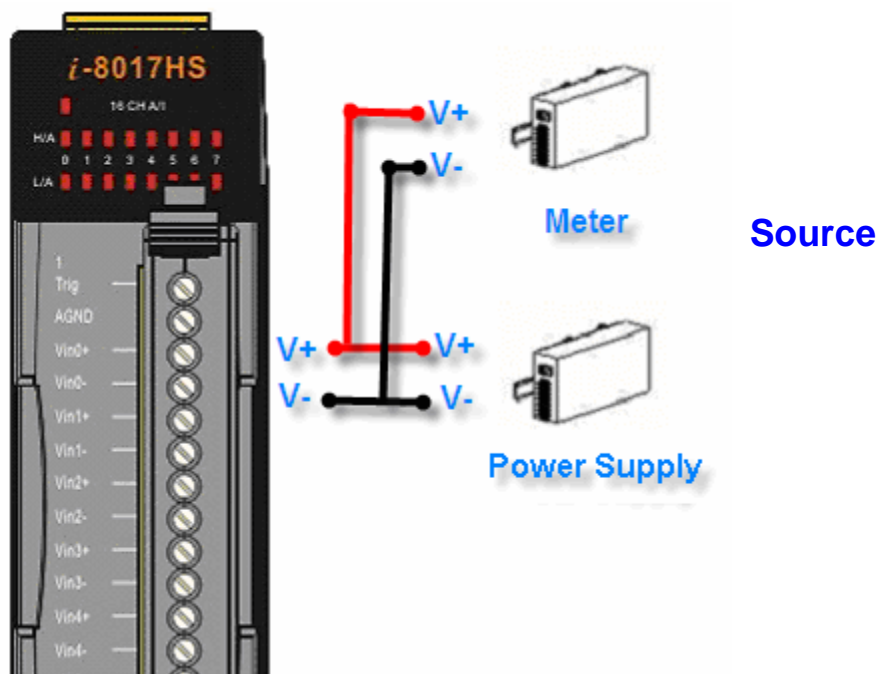
操作步驟：

步驟 1. 透過 RS-232 介面安裝好硬體 (包含接線，電源供應器和量測電錶)



控制器接線圖：

校正時需要在 I-8014W 模組, 電源供應器和量測電錶之間接上 4 條線, 2 條線給電源供應器使用, 2 條線給量測電錶使用



I-8014W 模組接線圖

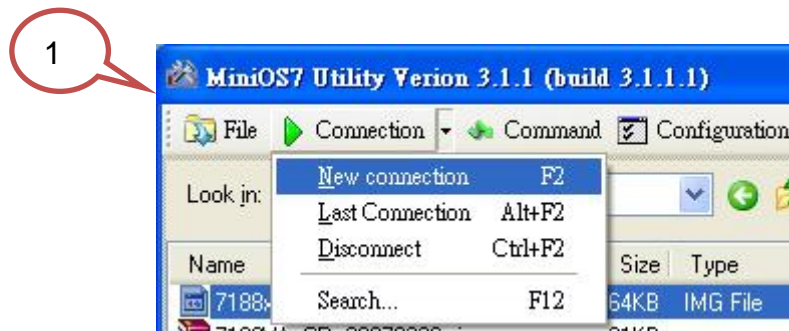
Note：電源供應器和量測電錶需要連接到模組的通道 0

步驟 2. 將校正程式放入控制器並且執行

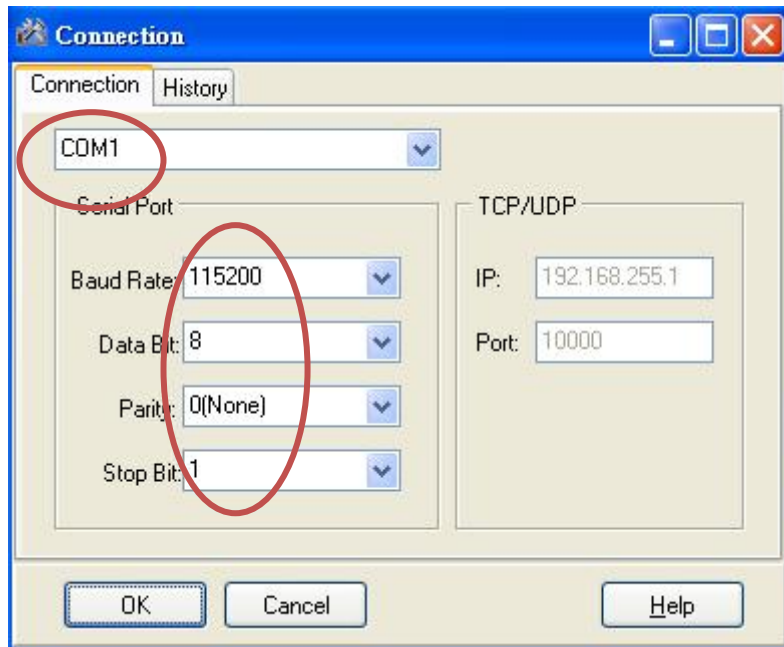
註：i-8000/IPAC-8000 部份請參考 MiniOS7 Utility 使用手冊

WP-8000/ XP-8000/ XP-8000-CE6 部份請參考各自手冊將校正程式放入控制器並且執行。

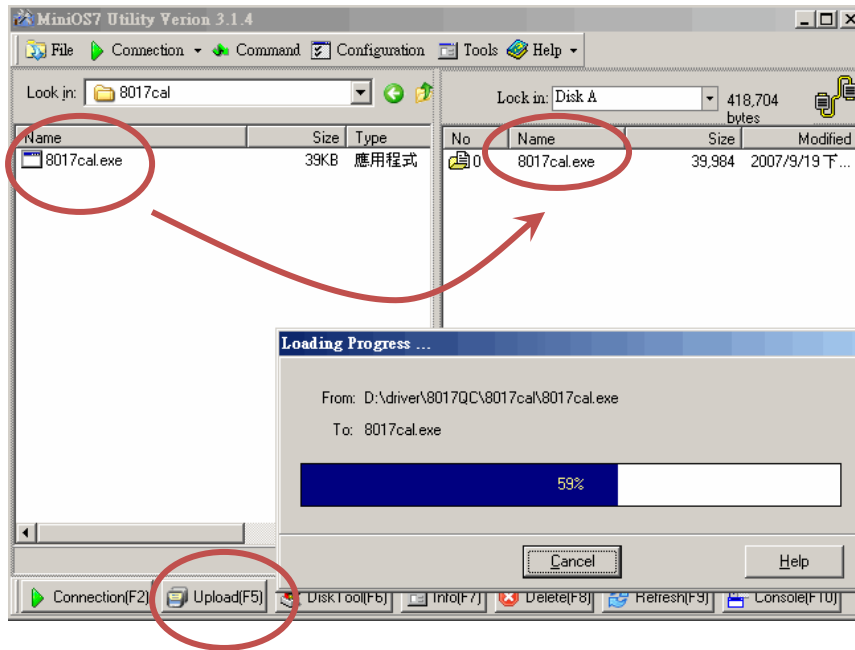
I. 打開 MiniOS7 Utility 選擇 Connection → New connection



II. 選擇正確的通訊埠和其他的參數然後按 OK 按鈕，可以連接到控制器

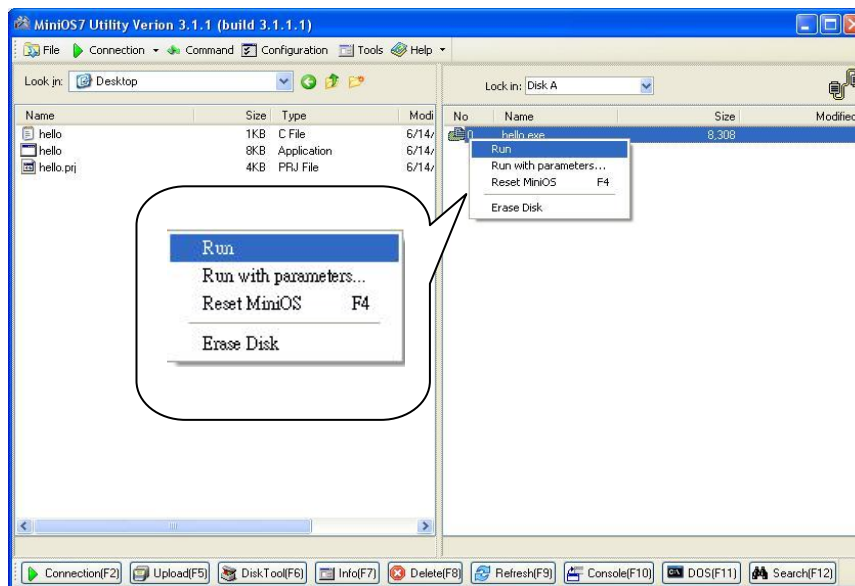


III. 選擇執行檔然後按 Upload 按鈕，可以將檔案上傳到控制器裡面



上傳 8014cal.exe 到控制器

IV. 按滑鼠右鍵去“ Run “執行程式



執行程式

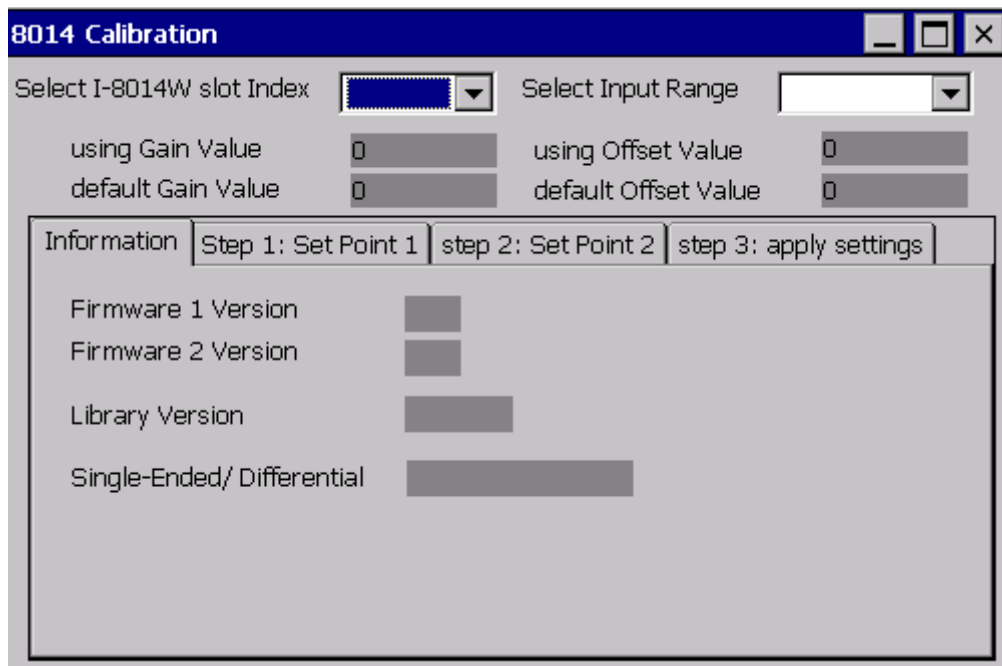
步驟 3. 校正式式：

- 程式開始時會如下圖所示

i-8000/IPAC-8000 部份：

```
7188XW 1.31 [COM1:115200,N,8,1],FC=0,CTS=1, DIR=D:\temp
now baudrate = 115200!
i8K_UDP>run #1
8017 Found in slot2
*****
* Calibration program for 8017H/8017HS *
* *
* Lattice Firmware Version = 4 *
* Please connect a voltage signal *
* to ch0 of the 8017H/8017HS first. *
* ver 1.0.1 _ Oct 08 2007 by Martin *
*****
* <0>Calibrate Gain_0 -10.00U to +10.00U *
* <1>Calibrate Gain_1 - 5.00U to + 5.00U *
* <2>Calibrate Gain_2 - 2.50U to + 2.50U *
* <3>Calibrate Gain_3 - 1.25U to + 1.25U *
* <r>Recover default calibration settings *
* <t>Read calibrated AI value of Ch0 *
* <s>Show calibrated Gain/Offset parameters *
* <q>quit *
*****
Please choose <0~3,r,t,s,q>:
```

WP-8000/ XP-8000/ XP-8000-CE6 部份：



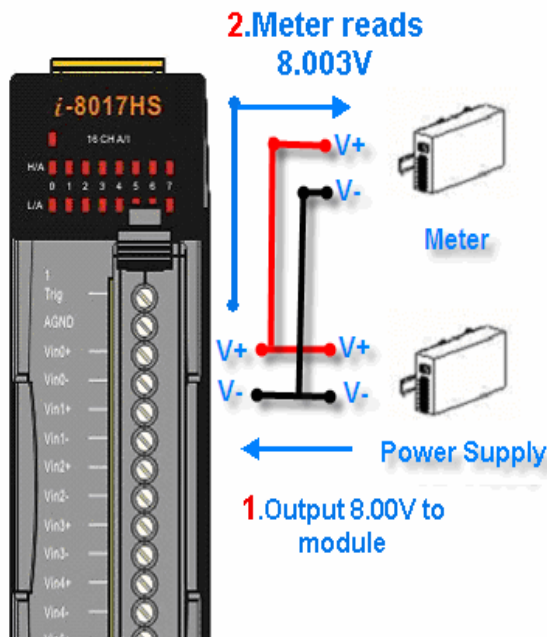
註 1：這個程式只能讀模組通道 0 的 AI 值，請參考硬體的 Pin Assignment 來做接線

- 選擇輸入範圍和給 2 個值來校正

例如：輸入範圍 -10V ~ +10V，以及給 8.0V 和 -8.0V 來校正

- I. 當我們使用這個校正式，校正時需要使用較準確的電壓源和量測電錶，例如可以用 3 又 1/2 位半 (或者更好) 的電源供應器來輸出 8.00 V 電壓給 8014 的通道 0，然後使用量測電錶來量測準確的電壓值。

II.



電源供應器輸出 8.00V 電壓給模組



量測儀器量測到準確的電壓值為：8.003V

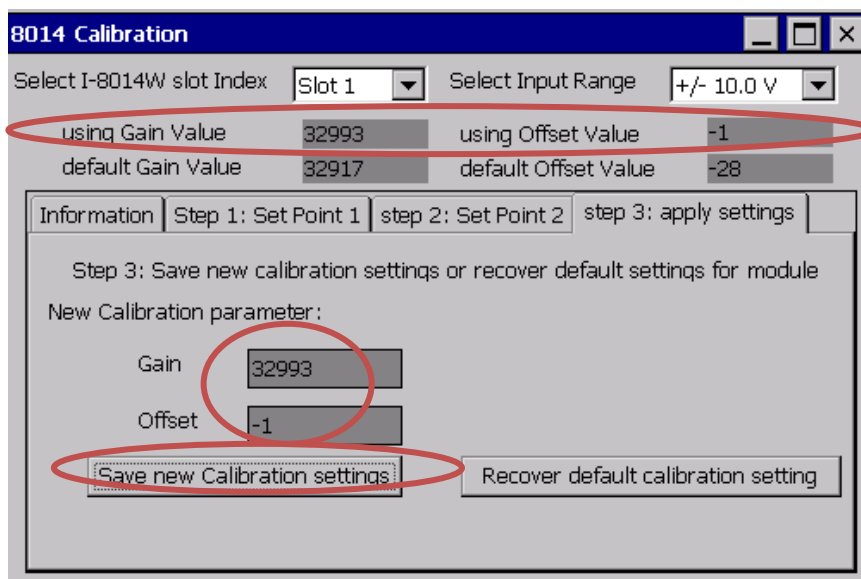
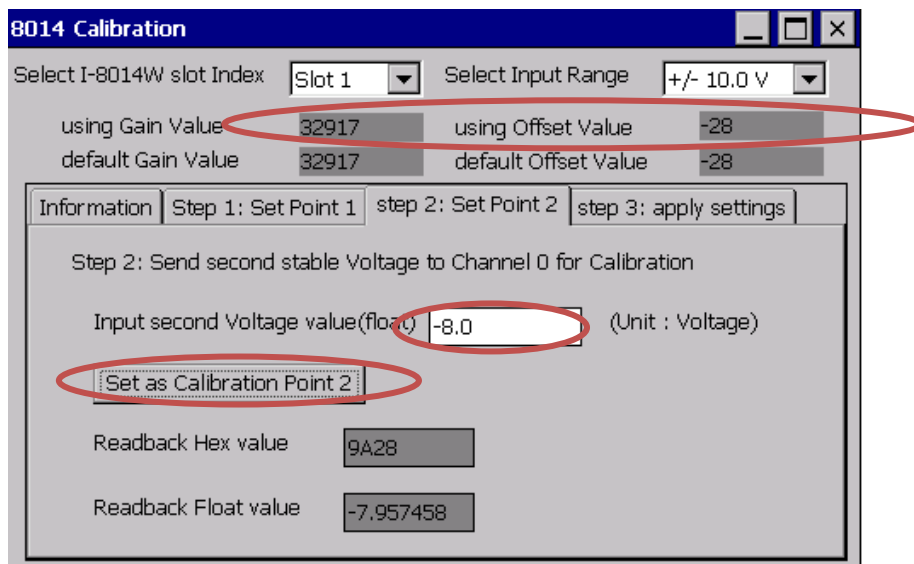
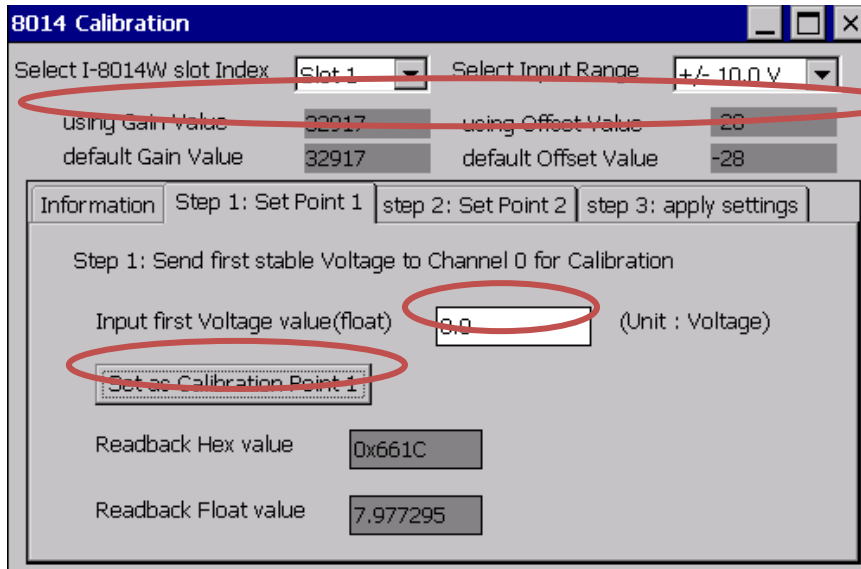
- III. 輸入量測儀器量測到的電壓值 8.003V 到校正程式，如下圖，重複上述步驟在輸出另一各數值 (例如-8V) 給 8014 模組，再輸入量測儀器量測到的電壓值，完成後程式會自動計算出新的 Gain 和 offset 的值，請將他們儲存到 EEPROM，如下圖：

i-8000/IPAC-8000 部份：

```

7188XW 1.31 [COM1:115200,N,8,1],FC=0,CTS=1, DIR=D:\temp
* (<0>)Calibrate Gain_0 -10.000 to +10.000 *
* (<1>)Calibrate Gain_1 - 5.000 to + 5.000 *
* (<2>)Calibrate Gain_2 - 2.500 to + 2.500 *
* (<3>)Calibrate Gain_3 - 1.250 to + 1.250 *
* (<r>)Recover default calibration settings *
* (<t>)Read calibrated AI value of Ch0 *
* (<s>)Show calibrated Gain/Offset parameters *
* (<q>)quit *
*****
Please choose (<0~3,r,t,s,q>):0
Original Gain_0=34074 Offset_0=-74
Please input 1st voltage (<0.0~+10.0>):8.003
Point 1=(0517 Hex)
Please input 2nd voltage (<0.0~-10.0>):-8.003
Point 2=(FB00 Hex)
New Gain= 36110 ,Offset=-366 Save to EEPROM? (<y/n>):y
Gain0 is calibrated.
  
```

WP-8000/ XP-8000/ XP-8000-CE6 部份：



Note：不可以使用不穩定的電源來校正，不然 I-8014W 模組將會不夠準確，假如沒有較準確的電壓源，可以使用一些穩定的電壓源 (例如電池) 來進行校正。

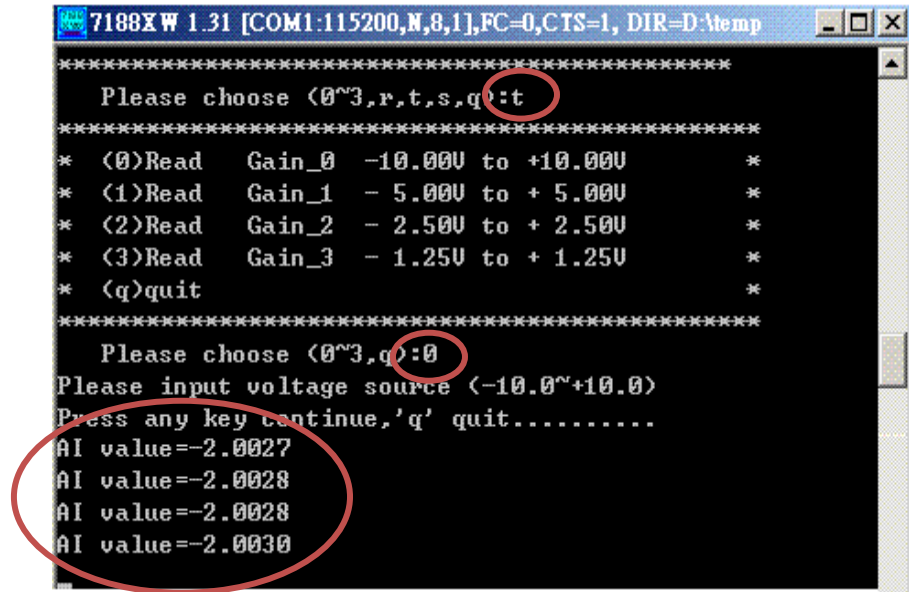
步驟 4. 檢驗校正結果：使用電源供應器輸出另一個電壓給 I-8014W 模組，用程式來量測模組讀到的數值並且和量測儀器量的數值作比較，我們就可以知道校正是否正確。

例如：使用電源供應器輸出-2V 電壓給 I-8014W，比較程式和量測儀器所讀到的數值。



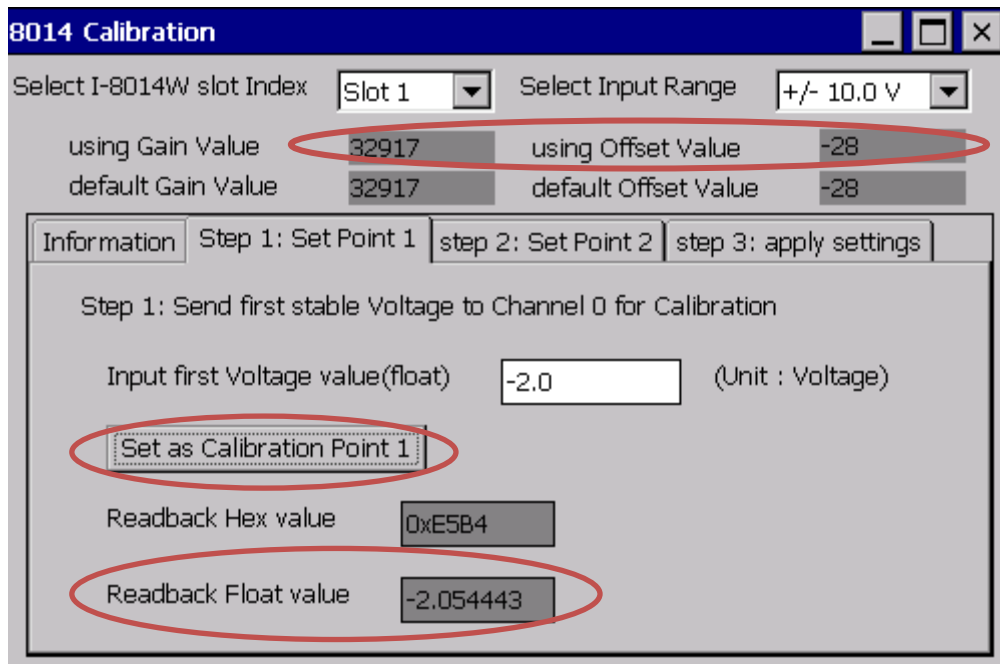
量測儀器量測到 -2.0111 V

i-8000/IPAC-8000 部份



從程式讀到的數值

WP-8000/ XP-8000/ XP-8000-CE6 部份 :



從程式讀到的數值

I-8014W 模組的 Accuracy 是 $\pm 0.1\%$ of FSR，再輸入範圍 $\pm 10V$ 下，相當於 $\pm 0.0V$ 。程式讀回的 AI 值就要在 -2.0311 to -1.9911 的範圍之內 (-2.0111 ± 0.02)，假若是如此則校正式正確的，假如程式讀回的 AI 值不在這個範圍內，這個校正就不準確，需要再重新校正。

步驟 5. 假如需要使用其他的輸入範圍 ($\pm 5V$ 、 $\pm 2.5V$ and $\pm 1.5V$ 、 $\pm 20 mA$)，重複步驟 3 和步驟 4 來校正以及檢驗校正結果，當再這個輸入阻抗的全部輸入範圍都校正完畢之後，模組用來讀 AI 的值將會更加的準確。

註 1：在 8014 模組上，輸入範圍 $\pm 20mA$ 和 $\pm 2.5V$ 是使用相同的校正值，假如需要校正 $\pm 20 mA$ 的輸入範圍，可以用校正 ± 2.5 輸入範圍的方式來完成。

註 2：在 100k 輸入阻抗下，假如不知道如何校正或是需要回復成預設值，可以在程式使用選項 r 來回復成出廠預設值 (只有 100k 的輸入阻抗可以回復成預設值，其他的就沒辦法)。

i-8000/IPAC-8000 部份 :

```
7188XW 1.31 [COM1:115200,N.8,1,FC=0,CTS=1, DIR=C:\Program...
+/- 10U      Gain =34074 Offset =-74
+/- 5U       Gain =34072 Offset =-76
+/- 2.5U     Gain =34069 Offset =-84
+/- 1.25U    Gain =34054 Offset =-79
+/- 20mA     Gain =34069 Offset =-84

Gain/Offset parameters which in using
+/- 10U      Gain =31383 Offset =-64
+/- 5U       Gain =31359 Offset =-68
+/- 2.5U     Gain =34069 Offset =-84
+/- 1.25U    Gain =34054 Offset =-79
+/- 20mA     Gain =34069 Offset =-84

*****
* <0>Calibrate Gain_0 -10.00U to +10.00U *
* <1>Calibrate Gain_1 - 5.00U to + 5.00U *
* <2>Calibrate Gain_2 - 2.50U to + 2.50U *
* <3>Calibrate Gain_3 - 1.25U to + 1.25U *
* <r>Recover default calibration settings *
* <t>Read calibrated AI value of Ch0 *
* <s>Show calibrated Gain/Offset parameters *
* <q>quit *
*****
Please choose <0~3,r,t,s,q>:r

Backup default Gain/Offset parameters settings for 100K
+/- 10U      Gain =34074 Offset =-74
+/- 5U       Gain =34072 Offset =-76
+/- 2.5U     Gain =34069 Offset =-84
+/- 1.25U    Gain =34054 Offset =-79
+/- 20mA     Gain =34069 Offset =-84

Gain/Offset parameters which in using
+/- 10U      Gain =34074 Offset =-74
+/- 5U       Gain =34072 Offset =-76
+/- 2.5U     Gain =34069 Offset =-84
+/- 1.25U    Gain =34054 Offset =-79
```

回復成預設的校正值

WP-8000/ XP-8000/ XP-8000-CE6 部份 :

