

Leaptronix®

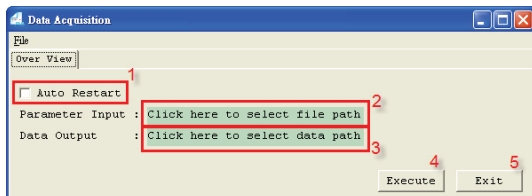
資料收集器 簡易操作手冊

〔支援Leaptronix LA及PLA系列邏輯分析儀〕

目錄

一、開啓程式.....	1
二、載入測試參數檔.....	1
三、選擇檔案輸出路徑.....	2
四、執行.....	3
五、檔案格式.....	4

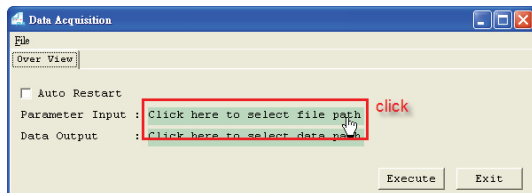
一、開啟程式 / 程式起始畫面



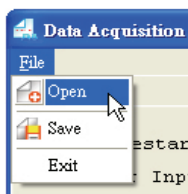
- 1.是否讓程式自動重新進入監看模式
- 2.測試參數檔的來源路徑
- 3.擷取到的資料要輸出存放的路徑
- 4.開始執行
- 5.離開本程式

二、載入測試參數檔：

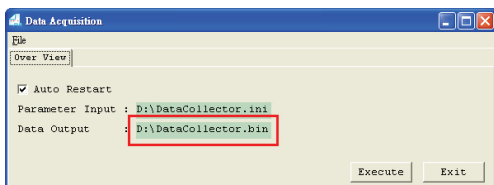
方法一、點選"測試參數檔的來源路徑"，即可選擇要開啟的測試參數檔



方法二.點選"File"→"Open"即可選擇要開啓的測試參數檔

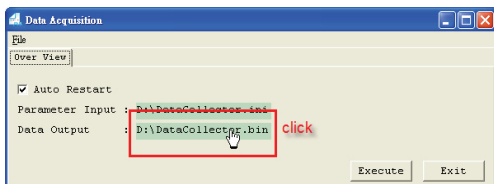


載入後會自動將測試參數檔中設定的資料檔案輸出路徑載入

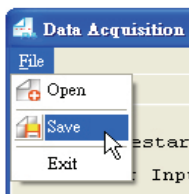


三、選擇檔案輸出路徑

方法一.如果想變更資料輸出的路徑，點選"資料要存放路徑"，即可選擇要存放檔案的路徑

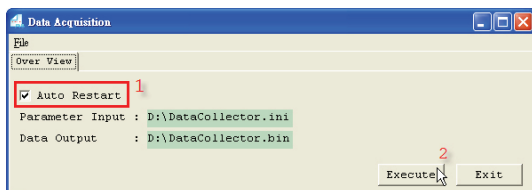


方法二.點選"File"→"Save"即可選擇 資料要存放的路徑

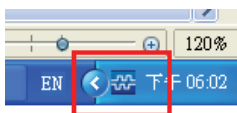


四、執行

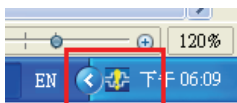
1. 按下"Execute"後，程式便會進入監看模式，如果有選擇"Auto Restart"選項，則會在擷取到資料後，自動再進入監看模式



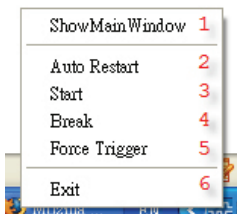
2. 啟動後，會處於背景執行，並在系統列中常駐。



3. 當被監控的參數檔內容有所變更時，便會自動套用新的參數設定，並將新擷取到的資料存放在先前設定的位置。當"Auto Restart"沒有被勾選時，會以驚嘆號提示，並停止監看模式。



4. 要重新進入監看模式，可以在常駐圖示上點選右鍵，開啟選單



1. 開啟主視窗
2. 自動再進入監看模式；當擷取到資料時，會自動的再次進入監看模式
3. 啟動監看模式；當擷取到資料且沒有啟用自動進入監看模式功能，則此選項可以使擷取器進入監看模式
4. 強制中斷資料擷取
5. 強制觸發
6. 關閉程式

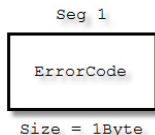
五、檔案格式：

1 Byte	23 Bytes	4 Bytes	2 MBytes or 4MBytes		
Seg 1	Seg 2	Seg 3	Seg 4	...	
ErrorCode	DateTime	Trigger Address	Data		

1.

第一個資料段(Seg 1)	錯誤檢查碼；用來表示載入的參數檔是否格式正確且資料觸發擷取動作是否正確完成。
第二個資料段(Seg 2)	檔案產生時間
第三個資料段(Seg 2)	TriggerAddress
第四個資料段(Seg 3)	由擷取器取得的資料將會依固定順序排列。

2.



錯誤檢查碼為一個Byte的資料，其內容用來標示產生檔案用的參數檔格式是否有錯誤。各錯誤碼代表的意義如下：

0x00：參數檔格式無誤，已正確完成觸發並擷取資料。

0x10：取樣頻率設定錯誤，本軟體會自動改為預設的(250Mhz)擷取。

0x20：記憶深度設定錯誤，本軟體會自動將記憶深度改為預設的自動設定模式。

0x30：延遲觸發設定錯誤，本軟體會自動改為預設的0延遲。

0x40：Trigger Pattern設定錯誤，本軟體會自動改為預設的模式(全部Don't Care)擷取。

0x50：Continue連續模式設定錯誤，本軟體會自動改為非連續觸發模式。

0x60：電壓設定錯誤，本軟體會自動改為TTL準位擷取。

0xF0：使用者使用了強制觸發。

0xFF：出現一個以上的設定錯誤。

3. 檔案產生日期

此資料段包含分隔符號共23個byte以ASCII格式存放。

內容包含：

西元年(四碼)、月(兩碼)、日(兩碼)、小時(24小時制，兩碼)、分(兩碼)、秒(兩碼)、千分秒(三碼)。

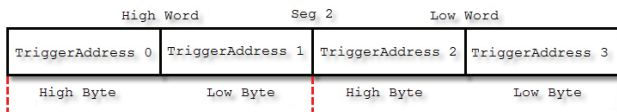
年、月、日間以"/"做為分隔符號，時、分、秒間以":"做為分隔符號，千分秒跟秒之間以"."為分隔符。

例：以2009年，1月1日，下午1時30分59秒為例：

存放的內容為2009/01/01/13:30:59.000

4. Trigger Address

Trigger Address代表觸發點在記憶體中的所在位址



Trigger Address為兩個word的資料，排列方式以High Word在前，Low Word在後的方式排列。

High Word 及 low Word 也分別以High Byte在前，Low Byte在後的方式排列。

以Trigger Address = 0x12345678為例；

Trigger Address的High Word為：0x1234、Low Word為0x5678；

0x12為High Word的High Byte，0x34為High Word的Low Byte；

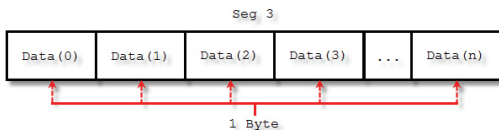
0x56為Low Word的High Byte，0x78為Low Word的Low Byte；

Trigger Address在傳回的bin中的實際位址為：

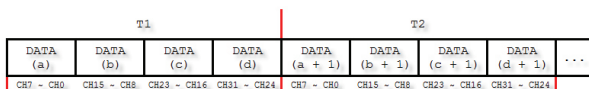
Trigger Address =

$\text{Trigger Address0} * 0x1000000$ (high word 的high byte) +
 $\text{Trigger Address1} * 0x10000$ (high word 的low byte) +
 $\text{Trigger Address 2} * 0x100$ (low word 的high byte) +
 Trigger Address3 (low word的low byte) + 5(在bin中資料起始位址的偏移值)

5. 資料段



資料段以Byte為單位，並固定以4個Byte做規則性的排列。



每四個Byte的資料代表32個channel (32bits)，因此資料的排列以四個Byte為一個循環。

透過排列的規則，同一個channel的相連兩筆資料(T1及T2)，會相隔四個Byte，即DATA(a + 1)為DATA(a)的下一筆資料。

每個Byte皆代表8個channel的資料，由MSB至LSB的順序即為較高的channel至較低的channel的順序。

以圖為例DATA(a)的b7為channel 7，b0為channel 0，而DATA(b)的b7則為channel 15、b0為channel 8，以32bit(4Byte)為循環。

如果使用的機型為16channel的機型，則代表channel 16 ~ channel 31的兩個Byte的資料內容會恆為0xFF。

圖例：

Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f	Dump
0000000000000000	00	32	30	30	39	2f	30	31	2f	30	36	2f	31	35	3a	30	.2009/01/06/15:0
0000000000000010	38	3a	31	34	2e	35	33	31	00	00	00	10	01	08	00	00	8:14.531.....
0000000000000020	01	ff	ff	ff	01	ff	ff	ff	01	ff	ff	ff	00	ff	ff	ff	.yyy.yyy.yyy.yyy
0000000000000030	00	ff	ff	ff	00	ff	ff	ff	00	ff	ff	ff	00	ff	ff	ff	.yyy.yyy.yyy.yyy
0000000000000040	00	ff	ff	ff	00	ff	ff	ff	00	ff	ff	ff	00	ff	ff	ff	.yyy.yyy.yyy.yyy

Byte +0 為錯誤檢查，代表此檔的參數設定沒有錯誤。

Byte +1 ~ Byte +17為此資料檔的存檔時間，以ASCII直接存入，共記錄西元年、月、日、時、分、秒、千分秒。

Byte +18 ~ Byte +1B為資料檔的觸發點所在位址；

Byte +18及Byte +19為High Word，Byte +1A及Byte +1B為Low Word，將此四個Byte組合後，會得出 0x00000010的結果，此結果便是Trigger address的所在位址。

如果以Byte +0為資料起始位址，則此位址需要再加上 0x1C(資料段的起始偏移量)，才是真正的觸發位址。

Byte +1C之後為資料內容。每4個byte為一組。

Byte +1C、Byte +1D、Byte +1E、Byte +1F代表Channel 31 ~ Channel 0在T1時資料，而Byte +20、Byte +21、Byte +22、Byte +23，則為Channel 31 ~ Channel 0在T2時的資料，以此循環。